

On Convergence of Multi-objective Particle Swarm Optimizers

Prithwish Chakraborty, Swagatam Das, Ajith Abraham, Václav Snasel and Gourab Ghosh Roy

Abstract— Several variants of the Particle Swarm Optimization (PSO) algorithm have been proposed in recent past to tackle the multi-objective optimization problems based on the concept of Pareto optimality. Although a plethora of significant research articles have so far been published on analysis of the stability and convergence properties of PSO as a single-objective optimizer, till date, to the best of our knowledge, no such analysis exists for the multi-objective PSO (MOPSO) algorithms. This paper presents a first, simple analysis of the general Pareto-based MOPSO and finds conditions on its most important control parameters (the inertia factor and acceleration coefficients) that control the convergence behavior of the algorithm to the Pareto front in the objective function space. Limited simulation supports have also been provided to substantiate the theoretical derivations.

Keywords: *multi-objective optimization, particle swarm optimization, Pareto optimality, Pareto dominance, convergence*

I. INTRODUCTION

The concept of particle swarms originated from the simulation of the social behavior commonly observed in animal kingdom and evolved into a very simple but efficient technique for global numerical optimization in recent past. Since its advent in 1995, the Particle Swarm Optimization (PSO) [1, 2] algorithm has attracted the attention of a lot of researchers all over the world resulting into a huge number of variants of the basic algorithm as well as many parameter selection/control strategies, comprehensive surveys of which can be found in [3 – 7]. In a D -dimensional search space, the position vector of the i -th particle is given by

$X_i = (x_{i,1}, x_{i,2}, \dots, x_{i,D})$ and velocity of the i -th particle is given by $V_i = (v_{i,1}, v_{i,2}, \dots, v_{i,D})$.

Prithwish Chakraborty is with Department. of Electronics and Telecommunication Engineering, Jadavpur University, India

Swagatam Das is with Department. of Electronics and Telecommunication Engineering, Jadavpur University, India (swagatamdas19@yahoo.co.in).

Ajith Abraham is with Machine Intelligence Research Labs (MIR Labs), USA and Faculty of Electrical Engineering and Computer Science, VSB Technical University of Ostrava, Czech Republic (ajith.abraham@ieee.org)

Václav Snasel is with the Faculty of Electrical Engineering and Computer Science, VSB Technical University of Ostrava, Czech Republic (vaclav.snasel@vsb.cz)

Gourab Ghosh Roy is with Department. of Electronics and Telecommunication Engineering, Jadavpur University, India

Positions and velocities are adjusted and the objective function to be optimized i.e. $f(X_i)$ is evaluated with the new positional coordinates at each time-step. Expressions for velocity and position of the i -th individual at t -th iteration in a geographical best PSO may be given as:

$$V_i(t+1) = \omega \cdot V_i(t) + \varphi_1 \cdot R_1 \cdot (P_i^l - X_i(t)) \quad (1a)$$

$$+ \varphi_2 \cdot R_2 \cdot (P^g - X_i(t)),$$
$$X_i(t+1) = X_i(t) + V_i(t+1), \quad (1b)$$

where, P_i^l is the personal best position found so far by an individual particle and P^g represents the best position found so far by the entire swarm, for *gbest* PSO model. R_1 and R_2 are random positive numbers uniformly distributed in (0,1) and are drawn anew for each dimension of each particle. Constants φ_1 and φ_2 are called acceleration coefficients and they determine the relative influences of the cognitive and social parts on the velocity of the particle. The particle's velocity may be optionally clamped to a maximum value $V_{\max} = [v_{\max,1}, v_{\max,2}, \dots, v_{\max,D}]^T$. If in d -th dimension, $|v_{i,d}|$ exceeds $v_{\max,d}$ specified by the user, then the velocity of that dimension is assigned to $\text{sign}(v_{i,d}) * v_{\max,d}$, where $\text{sign}(x)$ is the triple-valued signum function [2].

The first stability analysis of the particle dynamics was due to Clerc and Kennedy in 2002 [8]. F van den Bergh [9] undertook an independent theoretical analysis of the particle swarm dynamics. Clerc and Kennedy [8] considered a deterministic approximation of the swarm dynamics by treating the random coefficients as constants, and studied stable and limit cyclic behavior of the dynamics for the settings of appropriate values to its parameters. A more generalized stability analysis of particle dynamics based on Lyapunov stability theorems was undertaken by Kadirkamanathan *et al.* [10]. Recently Poli [11] analyzed the characteristics of a PSO sampling distribution and explained how it changes over any number of generations, in the presence of stochasticity, during stagnation. Some other significant works towards the theoretical understanding of PSO can be found in [12 - 15]. However, to the best of our knowledge, all the theoretical research works undertaken so far, are centered on the single-objective PSO algorithm, although, during the past few years, several efficient multi-objective variants of PSO have been proposed.

The field of Multi-objective Optimization (MO) deals

with the simultaneous optimization of multiple, possibly competing, objective functions, without combining them in a weighted sum. The MO problems tend to be characterized by a family of alternatives, which must be considered equivalent in the absence of information concerning the relevance of each objective relative to the others. The family of solutions of an MO problem is composed of the parameter vectors, which cannot be improved in any objective without causing degradation in at least one of the other objectives, and this set is said to be the *Pareto optimal set* and its image in the objective function space is usually called the *Pareto front*. In case of several MO problems, knowledge about this set helps the decision maker in choosing the best compromise solution [16].

Recently, several MOPSO algorithms have been developed based on the Pareto optimality concept. One fundamental issue is the selection of the cognitive and social leaders (P_i and P_g) such that they can provide an effective guidance to reach the most promising Pareto front region but at the same time maintain the population diversity. For the selection procedure researchers have suggested two typical approaches: selection based on quantitative standards and random selection [17 – 21].

Coello and Lechuga [22] made an attempt to incorporate the Pareto dominance into PSO. In this case, the non-dominated solutions are stored in a secondary population and the primary population uses a randomly selected neighborhood best from this secondary population to update their velocities. The authors proposed an adaptive grid to generate well-distributed Pareto fronts and mutation operators to enhance the exploratory capabilities of the swarm [23]. Keeping the same two goals (obtaining a set of non-dominated solutions as close as possible to the Pareto front and maintaining a well-distributed solution set along the Pareto front), Li [24] proposed sorting the entire population into various non-domination levels such that the individuals from better fronts can be selected. In this way, the selection process pushes towards the true Pareto front. Other authors have developed different approaches such as combining canonical PSO with auto fitness sharing concepts [25], dynamic neighborhood PSO [26], or vector evaluated PSO [27].

In this article, we present a simple theoretical analysis of the general continuous multi-objective PSO algorithm. Conditions for the convergence of MOPSO to some solutions (at least one) in the Pareto optimal set have been deduced based on the non-dominated selection scheme for updating the personal best and the global best positions. The analysis provides suitable ranges of the control parameters like ω, φ_1 and φ_2 that ensures the convergence of MOPSO. Limited experimental results on two well-known MO benchmarks have been provided to support the analytical results derived in the article.

II. ANALYTICAL TREATMENT

For MOPSO, suppose n particles are randomly scattered in the search space and following (1). Expectedly, decisions like the updating of local best or global best are determined using the concept of Pareto-optimality. We assume a Pareto-based approach to be taken for implementing the selection of the globally best particle of the swarm in every iteration. The algorithm is expected to identify the set of non-dominated solutions of the population at each iteration and store the best non-dominated solutions found throughout the search process in an external archive (e.g. see the MOPSO described in [23]). The global best particle P_g may be chosen from this archive. The use of global attraction mechanisms combined with a historical archive of previously found non-dominated vectors can motivate convergence toward globally non-dominated solutions.

We attempt to investigate the convergence characteristics of the MOPSO algorithm by examining the evolution of the probability distribution of the population, based on which the search algorithm is run. Our method is inspired by the work reported in [28] for multi-objective evolutionary algorithms. Let the Pareto-optimal set be represented by Λ^* . Then to show that the expected population mean converges to the centre of the Pareto-optimal set amounts to verifying the convergence properties of PSO in multi-objective domain. Resultant deduction may be summarized in form of a theorem as given below:

Theorem 1: *If X be the initial swarm, then subject to restraints outlined in (10), in a multi-objective search space MOPSO algorithm will cause the swarm mean to converge to the center of the Pareto-optimal set Λ^* i.e. if μ_n be the expected population mean after n iterations, then $\lim_{n \rightarrow \infty} \mu_n = \hat{X}^*$; where \hat{X}^* denotes the center of the front.*

Proof: Let X^* be a randomly chosen solution from the Pareto-optimal set Λ^* . Then without any loss of generality, local best position of any particle and the global best position of the swarm, in a particular iteration may be expressed as:

$$\left. \begin{aligned} P_i^l(t) &= X^* + D_{Li}(t) \\ P^g(t) &= X^* + D_{Gi}(t) \end{aligned} \right\} \quad (2)$$

Where, $D_{Li}(t)$ and $D_{Gi}(t)$ are difference vectors.

Then, equation 1(a) may be written as:

$$\begin{aligned}
V_i(t+1) &= \omega \cdot V_i(t) + \varphi_1 \cdot R_1 \cdot (P_i^l(t) - X_i(t)) \\
&\quad + \varphi_2 \cdot R_2 \cdot (P^s - X_i(t)) \\
&= \omega \cdot V_i(t) + \varphi_1 \cdot R_1 \cdot (X_i^* + D_{Li}(t) - X_i(t)) \\
&\quad + \varphi_2 \cdot R_2 \cdot (X_i^* + D_{Gi}(t) - X_i(t)) \\
&= \omega \cdot V_i(t) + (\varphi_1 R_1 + \varphi_2 R_2) \cdot (X_i^* - X_i(t)) \\
&\quad + (\varphi_1 \cdot R_1 \cdot D_{Li}(t) + \varphi_2 \cdot R_2 \cdot D_{Gi}(t))
\end{aligned} \tag{3}$$

In discrete domain $V_i(t) = (X_i(t) - X_i(t-1))$ and using equation 1(b), equation (3) may be modified as shown below:

$$\begin{aligned}
X_i(t+1) - X_i(t) &= \omega \cdot (X_i(t) - X_i(t-1)) \\
&\quad + (\varphi_1 R_1 + \varphi_2 R_2) \cdot (X_i^* - X_i(t)) \\
&\quad + (\varphi_1 \cdot R_1 \cdot D_{Li}(t) + \varphi_2 \cdot R_2 \cdot D_{Gi}(t))
\end{aligned}$$

Or,

$$\begin{aligned}
X_i(t+1) &= \{1 + \omega - (\varphi_1 R_1 + \varphi_2 R_2)\} X_i(t) - \omega \cdot X_i(t-1) \\
&\quad + (\varphi_1 R_1 + \varphi_2 R_2) \cdot X_i^* \\
&\quad + (\varphi_1 \cdot R_1 \cdot D_{Li}(t) + \varphi_2 \cdot R_2 \cdot D_{Gi}(t))
\end{aligned} \tag{4}$$

Then, taking expectations on both sides of equation (4) and considering the above-mentioned terminology we get,

$$\begin{aligned}
E(X_i(t+1)) &= \{1 + \omega - (\varphi_1 E(R_1) + \varphi_2 E(R_2))\} \cdot E(X_i(t)) \\
&\quad - \omega \cdot E(X_i(t-1)) + (\varphi_1 E(R_1) + \varphi_2 E(R_2)) \cdot E(X_i^*) \\
&\quad + (\varphi_1 \cdot E(R_1) \cdot E(D_{Li}(t)) + \varphi_2 \cdot E(R_2) \cdot E(D_{Gi}(t)))
\end{aligned} \tag{5}$$

Let us denote the mean value of the position of the “ i -th” particle at iteration number “ t ” as $\mu_i(t)$. Also, let us define, $E(D_{Li}(t)) \equiv \hat{D}_{Li}(t)$, $E(D_{Gi}(t)) \equiv \hat{D}_{Gi}(t)$ and $E(X_i^*) \equiv \hat{X}^*$.

Now, for uniform random numbers, $E(R_1) = E(R_2) = \frac{1}{2}$.

From these statements equation (5) may be rearranged and given as

$$\begin{aligned}
\mu_i(t+1) &= \left[(1 + \omega) - \frac{\varphi_1 + \varphi_2}{2} \right] \cdot \mu_i(t) - \omega \cdot \mu_i(t-1) \\
&\quad + \frac{\varphi_1 + \varphi_2}{2} (\hat{X}^*) + \left[\frac{\varphi_1}{2} \hat{D}_{Li}(t) + \frac{\varphi_2}{2} \hat{D}_{Gi}(t) \right] \\
\mu_i(t) &= \left[(1 + \omega) - \frac{\varphi_1 + \varphi_2}{2} \right] \cdot \mu_i(t-1) - \omega \cdot \mu_i(t-2)
\end{aligned}$$

Or,

$$\begin{aligned}
&\quad + \frac{\varphi_1 + \varphi_2}{2} (\hat{X}^*) + \left[\frac{\varphi_1}{2} \hat{D}_{Li}(t-1) + \frac{\varphi_2}{2} \hat{D}_{Gi}(t-1) \right]
\end{aligned} \tag{6}$$

To examine convergence, we take Z-transform [29] of both sides of (6). Dropping the subscript for each such particle ‘ i ’, we get

$$\begin{aligned}
Z(\mu(t)) &= \left[(1 + \omega) - \frac{\varphi_1 + \varphi_2}{2} \right] \cdot Z(\mu(t-1)) - \omega \cdot Z(\mu(t-2)) \\
&\quad + Z\left(\frac{\varphi_1 + \varphi_2}{2} (\hat{X}^*)\right) + \left[\frac{\varphi_1}{2} Z(\hat{D}_L(t-1)) + \frac{\varphi_2}{2} Z(\hat{D}_G(t-1)) \right]
\end{aligned}$$

One can safely assume the system to be a causal one and

hence we can take the one sided Z-transform. Thus the equation may be simplified as:

$$\begin{aligned}
\mu(z) \cdot \left(1 - \alpha^* z^{-1} + \omega \cdot z^{-2} \right) &= \frac{\varphi_1 + \varphi_2}{2} \hat{X}^* Z(U(t)) \\
&\quad + \left[\frac{\varphi_1}{2} z^{-1} \hat{D}_L(z) + \frac{\varphi_2}{2} z^{-1} \hat{D}_G(z) \right],
\end{aligned}$$

where, $\alpha = \left[(1 + \omega) - \frac{\varphi_1 + \varphi_2}{2} \right]$ and $U(t)$ denotes the unit step function.

$$\begin{aligned}
\mu(z) &= \frac{\varphi_1 + \varphi_2}{2} \hat{X}^* \frac{1}{(1 - \alpha^* z^{-1} + \omega \cdot z^{-2})} \cdot \frac{z}{z-1} \\
\text{Or,} &\quad + \frac{\varphi_1}{2} \frac{z^{-1}}{(1 - \alpha^* z^{-1} + \omega \cdot z^{-2})} \cdot \hat{D}_L(z) + \frac{\varphi_2}{2} \frac{z^{-1}}{(1 - \alpha^* z^{-1} + \omega \cdot z^{-2})} \cdot \hat{D}_G(z) \\
\mu(z) &= \frac{\varphi_1 + \varphi_2}{2} \hat{X}^* \frac{z^3}{(z^2 - \alpha^* z + \omega)} \cdot \frac{1}{z-1} \\
\text{Or} &\quad + \frac{\varphi_1}{2} \frac{z}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_L(z) + \frac{\varphi_2}{2} \frac{z}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_G(z)
\end{aligned} \tag{7}$$

Now final value theorem states that

$\lim_{t \rightarrow \infty} \mu(t) = \lim_{z \rightarrow 1} (z-1) \mu(z)$, provided the limits exist.

Then,

$$\begin{aligned}
\lim_{t \rightarrow \infty} \mu(t) &= \frac{\varphi_1 + \varphi_2}{2} \hat{X}^* \lim_{z \rightarrow 1} \left[\frac{z^3}{(z^2 - \alpha^* z + \omega)} \right] \\
&\quad + \frac{\varphi_1}{2} \lim_{z \rightarrow 1} \left[\frac{z(z-1)}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_L(z) \right] \\
&\quad + \frac{\varphi_2}{2} \lim_{z \rightarrow 1} \left[\frac{z(z-1)}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_G(z) \right]
\end{aligned} \tag{8}$$

Considering 1st term of the expansion,

$$\begin{aligned}
\lim_{z \rightarrow 1} \left[\frac{z^3}{(z^2 - \alpha^* z + \omega)} \right] &= \frac{1}{1 - \alpha + \omega} \\
&= \frac{1}{1 + \omega - (1 + \omega) + \frac{\varphi_1 + \varphi_2}{2}} = \frac{1}{\frac{\varphi_1 + \varphi_2}{2}},
\end{aligned} \tag{9}$$

provided roots of the characteristic equation, $Q(z) = z^2 - \alpha^* z + \omega$; lie within unit circle in the z-domain. The required condition may be investigated using Jury’s test as shown below:

Conditions for pole placement within unit circle from Jury’s Test

1. $Q(1) > 0$:

$$1 - \alpha + \omega > 0 \quad \text{i.e.} \quad \frac{\varphi_1 + \varphi_2}{2} > 0 \quad (10.a)$$

2. $\underline{(-1)^n Q(-1) > 0}$:

$$\text{i.e. } 1 + 1 + \omega - \frac{\varphi_1 + \varphi_2}{2} + \omega > 0$$

$$\text{Or, } \frac{\varphi_1 + \varphi_2}{2} < 2(1 + \omega) \quad (10.b)$$

3. $\underline{a_n > |a_0|}$:

$$\omega < 1 \quad (10.c)$$

For other two components, limiting value depends upon the nature of $\hat{D}_L(z)$ and $\hat{D}_G(z)$ i.e. the function to be optimized. If the poles of these terms are within unit circle i.e. the difference vectors are bounded and not oscillatory in nature, with constant magnitude; then these difference vectors may be expanded as:

$$\hat{D}(z) = \sum_{z_j \in \text{poles of } \hat{D}(z)} \frac{C_j z}{z - z_j}, \text{ such that } |z_j| < 1.$$

Therefore,

$$\begin{aligned} \lim_{z \rightarrow 1} \left[\frac{z(z-1)}{(z^2 - \alpha * z + \omega)} \cdot \hat{D}(z) \right] \\ = \lim_{z \rightarrow 1} \left[\frac{z(z-1)}{(z^2 - \alpha * z + \omega)} \cdot \sum \frac{C_j z}{z - z_j} \right] \\ = \lim_{z \rightarrow 1} \sum \left[\frac{z^2(z-1) \cdot C_j}{(z^2 - \alpha * z + \omega)(z - z_j)} \right] \end{aligned}$$

When, $|z_j| < 1$ and condition set (9) is satisfied, then this limit exists and the limiting value is given as:

$$\lim_{z \rightarrow 1} \left[\frac{z^2(z-1)}{(z^2 - \alpha * z + \omega)} \cdot \hat{D}(z) \right] = 0 \quad (11)$$

Another viewpoint to analyze the effect of these difference vectors is presented in the Appendix from where it can be said that our assumption holds true provided

$$|z_j| \cdot \max \left| \frac{D_{i-1}}{D_i} \right| < 1$$

Using results obtained in (9) and (11), (8) simplifies as:

$$\lim_{t \rightarrow \infty} \mu(t) = \frac{\varphi_1 + \varphi_2}{2} \hat{X}^* \cdot \frac{1}{\frac{\varphi_1 + \varphi_2}{2}} + 0 + 0$$

$$\text{i.e. } \lim_{t \rightarrow \infty} \mu(t) = \hat{X}^*$$

Proved

Lemma 1: For any practical problems, with $\omega = 0.4$, the conditions when MOPSO will never converge to true Pareto

front reduces to $\varphi > 2.8$, where $\varphi = \frac{\varphi_1 + \varphi_2}{2}$ is the average acceleration coefficient of the swarm. For lower values of average acceleration co-efficient, convergence characteristics of the swarm will, in general, be dependant on the nature of the optimization function.

Proof: From conditions outlined in equation set (10) the swarm will never converge if

$$\frac{\varphi_1 + \varphi_2}{2} > 2(1 + \omega)$$

Then putting $\omega = 0.4$, the condition reduces to $\varphi > 2.8$.

Convergence characteristics of MOPSO will depend on position of poles $\hat{D}_L(z)$ and $\hat{D}_G(z)$. The process is to a certain extent stochastic in nature. However, without proof it can be said that in a non-dominated sorting method probability of oscillation of error vectors is not very high. This can be used as a guideline in parametric selection. For example one can expect the swarm to diverge while using a parametric set $(\varphi_1, \varphi_2, \omega) = (3.5, 3.5, 0.4)$, and to converge in most of the runs while using a set $(\varphi_1, \varphi_2, \omega) = (0.5, 0.5, 0.4)$, a limited experimental verification of which is presented in section 3.

Observation 1: The parametric set used in [23] is given as $(\varphi_1, \varphi_2, \omega) = (1, 1, 0.4)$. From Lemma 1, with $\omega = 0.4$, the swarm would not converge in general if $\varphi = \frac{\varphi_1 + \varphi_2}{2} > 2.8$.

Putting the corresponding values of acceleration coefficients used in (23), we get $\varphi = 1$. Thus the parameters used in [23], seems to be in agreement with the theoretically suitable region and as reported in [23] produced near optimal pareto-fronts for a variety of optimization functions.

III. EXPERIMENTAL RESULTS

In order to validate the theoretical results derived in the last section, we tested the MOPSO algorithm proposed in [23] on a set of well-known benchmark MO problems. Here for the sake of space economy we are providing the results of the following two functions [30 – 32] only:

Test Function 1 (Schaffer's Function [29]): Minimize $F = (f_1(x), f_2(x))$, where

$$\left. \begin{aligned} f_1(x) &= x^2 \\ f_2(x) &= (x-2)^2 \end{aligned} \right\} -100 \leq x \leq 100$$

Test Function 2 (Fonseca's function [30]): Minimize $F = (f_1(\vec{x}), f_2(\vec{x}))$, where

$$\left. \begin{aligned} f_1(\vec{x}) &= 1 - \exp\left(-\sum_{i=1}^3 \left(x_i - \frac{1}{\sqrt{3}}\right)^2\right) \\ f_2(\vec{x}) &= 1 - \exp\left(-\sum_{i=1}^3 \left(x_i + \frac{1}{\sqrt{3}}\right)^2\right) \end{aligned} \right\} -4 \leq x_i \leq 4$$

On both the test problems we run MOPSO with two different parametric set up: $(\varphi_1, \varphi_2, \omega) = (0.5, 0.5, 0.4)$ and $(\varphi_1, \varphi_2, \omega) = (3.5, 3.5, 0.4)$. Rest of the parameters has been kept similar to [4] and the algorithm is run up to 4000

Function Evaluations (FEs). In Figures 1 and 4 we show the initial solutions in objective function space, at the start of MOPSO iterations w. r. t. the true Pareto front for the two test functions. Figures 2 and 3 show the convergence behavior of MOPSO with two different parametric settings, in terms of the solutions plotted in the objective function space after 2000 and 4000 FEs for Test Function 1. Note that MOPSO is run with two different parameter sets but from the same initial population for each problem. Figures 5 and 6 bear an exact correspondence with Figures 2 and 3, but the formers are now plotted for Test Function 2. Figures 1 to 6 represent mean results over 20 independent runs of the algorithm over both the test problems.

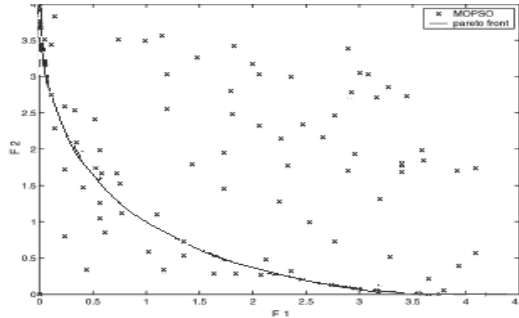
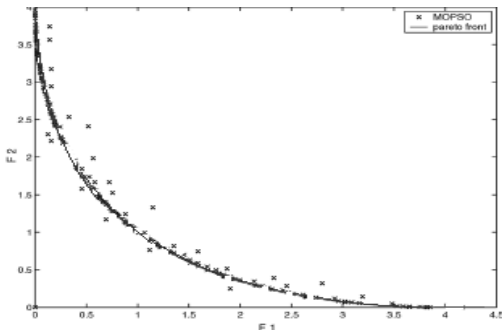
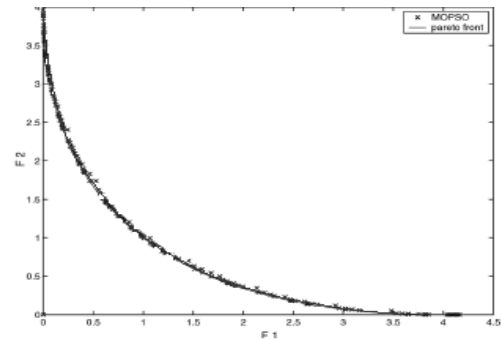


FIGURE 1: INITIAL SOLUTIONS GENERATED BY MOPSO POPULATION FOR TEST PROBLEM 1

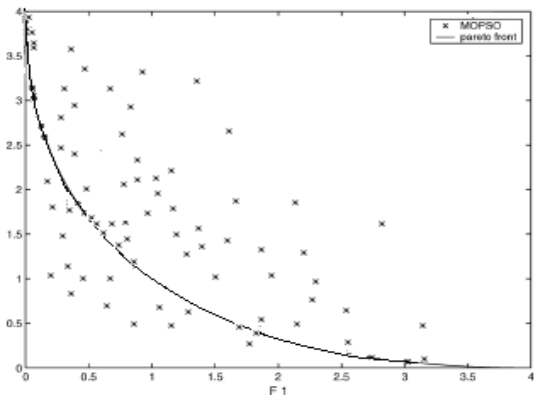


(A) SOLUTIONS GENERATED BY MOPSO AFTER 2000 FES.

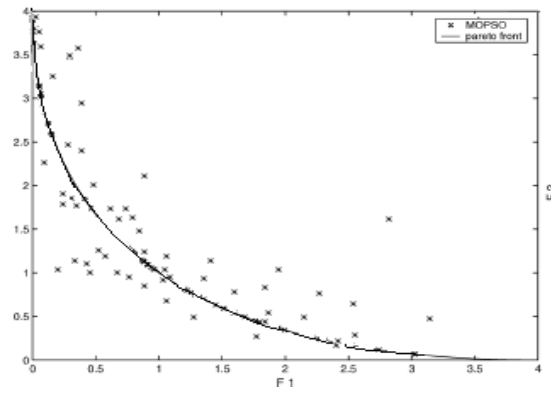


(B) SOLUTIONS GENERATED BY MOPSO AFTER 4000 FES

FIGURE 2: CONVERGENCE BEHAVIOR OF THE SOLUTIONS BY MOPSO POPULATION WITH PARAMETER-SET $(\varphi_1, \varphi_2, \omega) = (0.5, 0.5, 0.4)$ FOR TEST PROBLEM 1



(A) SOLUTIONS GENERATED BY MOPSO AFTER 2000 FES.



(B) SOLUTIONS GENERATED BY MOPSO AFTER 4000 FES

FIGURE 3: CONVERGENCE BEHAVIOR OF THE SOLUTIONS BY MOPSO POPULATION WITH PARAMETER-SET $(\varphi_1, \varphi_2, \omega) = (3.5, 3.5, 0.4)$ FOR TEST PROBLEM 1

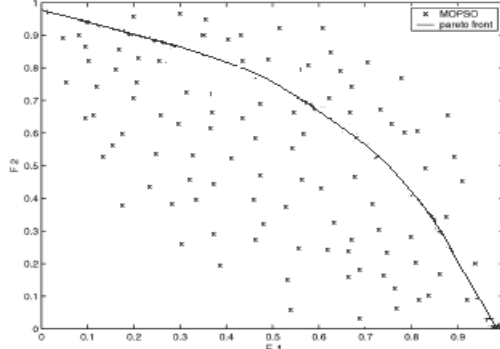
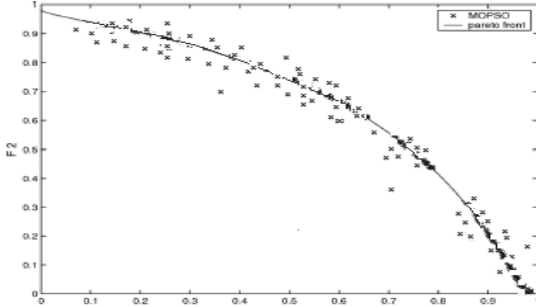
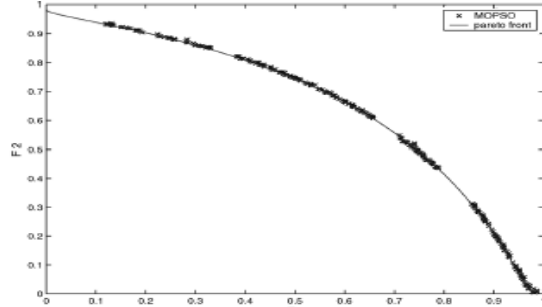


FIGURE 4: INITIAL SOLUTIONS BY MOPSO POPULATION FOR TEST PROBLEM 2

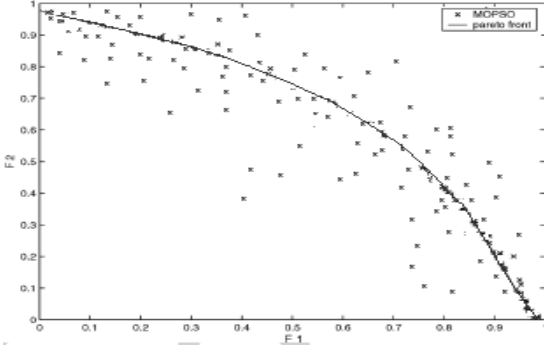


(A) SOLUTIONS GENERATED BY MOPSO AFTER 2000 FES.

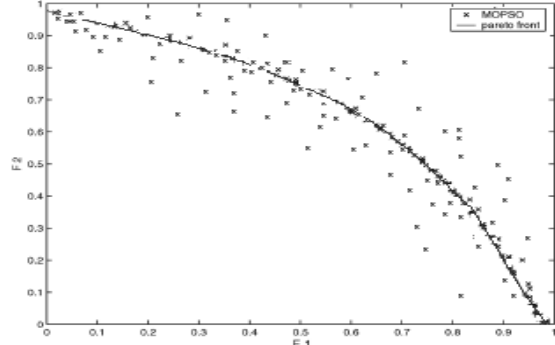


(B) SOLUTIONS GENERATED BY MOPSO AFTER 4000 FES

FIGURE 5: CONVERGENCE BEHAVIOR OF THE SOLUTIONS BY MOPSO POPULATION WITH PARAMETER-SET $(\varphi_1, \varphi_2, \omega) = (0.5, 0.5, 0.4)$ FOR TEST PROBLEM 2



(A) SOLUTIONS GENERATED BY MOPSO AFTER 2000 FES.



(B) SOLUTIONS GENERATED BY MOPSO AFTER 4000 FES

FIGURE 6: CONVERGENCE BEHAVIOR OF THE SOLUTIONS BY MOPSO POPULATION WITH PARAMETER-SET $(\varphi_1, \varphi_2, \omega) = (3.5, 3.5, 0.4)$ FOR TEST PROBLEM 2

2

The plots clearly indicate that as was predicted theoretically, the MOPSO algorithm fails to make its solutions converge to the Pareto optimal front in objective function space for the parametric set-up of $(\varphi_1, \varphi_2, \omega) = (3.5, 3.5, 0.4)$. Since on each problem, the MOPSO with different parameters were let run from the same initial population in each case; so any difference in performance of the algorithm may be well attributed to the difference in their parametric settings only.

IV. CONCLUSIONS

This article is the first of its kind to focus light on the convergence behavior of Pareto-based MOPSO algorithms

from an analytical point of view. Theorem 1 deduces the conditions imposed on three important control parameters: the inertia weight ω and acceleration coefficients φ_1, φ_2 that ensure the convergence of the algorithm to the center of the Pareto front, under a few simplifying assumptions. In lemma 1 the range of values of these control parameters were discussed that directs either convergence or non-convergence of the algorithm. Lemma 1 is intended to provide some initial guidelines for selecting these three control parameters of MOPSO. In Section 3, we show how two different settings of parameters really affects the convergence behavior of MOPSO practically over two benchmark MO problems. We restrain from giving the full

set of results of numerical solutions over many different benchmarks for the sake of space economy.

The mathematical analysis on the convergence of the MOPSO, undertaken in this work proves that the population converges to the Pareto optimal solution set. However, how the swarm is distributed in the decision variable space is still unknown and should be analyzed rigorously in future. Also the nature of stochastic variation of error vectors needs to be investigated further to derive tighter bounds for parameter selection.

ACKNOWLEDGEMENT

This work was supported by the Czech Science Foundation under the grant no.102/09/1494.

APPENDIX

Further Discussions on \hat{D}_L and \hat{D}_G :

In order to further analyze the effect of error vectors \hat{D}_L and \hat{D}_G , equation (7) is analyzed in time domain again as shown below:

From equation no (7)

$$\mu(z) = \frac{\varphi_1 + \varphi_2}{2} \hat{X}^* \frac{z^3}{(z^2 - \alpha^* z + \omega)} \cdot \frac{1}{z-1} + \left[\frac{\varphi_1}{2} \frac{z}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_L(z) + \frac{\varphi_2}{2} \frac{z}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_G(z) \right]$$

Let us consider the term

$$p(z) = \frac{\varphi_1}{2} \frac{z}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_L(z) \text{ in the above expression.}$$

If conditions as outlined in equation set (10) are satisfied, then the following equality hold true:

$$z^2 - \alpha^* z + \omega = \prod_k (z - z_k); \text{ where } |z_k| < 1$$

Then,

$$\begin{aligned} p(z) &= \frac{\varphi_1}{2} \frac{z}{(z^2 - \alpha^* z + \omega)} \cdot \hat{D}_L(z) \\ &= \hat{D}_L(z) \sum_k \left(C_k \frac{z}{z - z_k} \right) \\ &= \sum_k \left(C_k \frac{z}{z - z_k} \hat{D}_L(z) \right) \end{aligned} \quad (12)$$

; where each such C_k is the constant associated with the root z_k in the above expansion.

Now we know that if

$$AB(z) = A(z) \cdot B(z), \text{ then}$$

$$ab(n) = Z^{-1}[A(z) \cdot B(z)] = \sum_{i=0}^n (b(i) \cdot a(n-i)).$$

As such time domain representation of equation (12) may be given as:

$$\begin{aligned} p(n) &= \sum_k \left(C_k \sum_{i=0}^n (\hat{D}_i \cdot z_k^{n-i}) \right) \\ &= \sum_k (C_k Err_k), \end{aligned}$$

$$\text{where we have defined } Err_k = \sum_{i=0}^n (\hat{D}_i \cdot z_k^{n-i})$$

$$\text{Now, let us define } q_i = \frac{E(D_{i-1})}{E(D_i)} = \frac{\hat{D}_{i-1}}{\hat{D}_i}, \quad \forall i = 1(1)n$$

Also let us define $q \equiv \max q_i$

$$\text{Therefore, } \hat{D}_{n-1} \leq q \cdot \hat{D}_n, \quad \forall i = 1(1)n$$

Therefore the following expression holds:

$$\hat{D}_i \leq q^{n-i} \cdot \hat{D}_n, \quad \forall i = 1(1)n$$

So for large n,

$$\begin{aligned} |Err_k| &= \left| \sum_{i=0}^n (z_k^{n-i} \cdot \hat{D}_i) \right| \\ &\leq \left| \hat{D}_n \right| \cdot \left| \sum_{i=0}^n (q z_k)^{n-i} \right| \\ &\approx \left| \hat{D}_n \right| \cdot \left| \frac{1}{1 - z_k \cdot q} \right|, \text{ when } z_k \cdot q < 1 \end{aligned} \quad (13)$$

It is expected that by selection procedures at least the local best of any individual will be in the Pareto-optimal set after infinite no. of iterations i.e. $|\hat{D}_n| \rightarrow 0$. So taking limits on both sides of (13)

$$|Err_k| \rightarrow 0 \text{ i.e. } \lim_{n \rightarrow \infty} p(n) = 0$$

REFERENCES

- [1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw. (ICNN)*, Nov. 1995, vol. 4, pp. 1942–1948.
- [2] J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*, Morgan Kaufmann, San Francisco, CA, 2001.
- [3] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, 2006.
- [4] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. G. Harley, "Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems", *IEEE Transactions on Evolutionary Computation*, Vol. 12, Issue 2, pp. 171 – 195, 2008.
- [5] M. R. AlRashidi and M. E. El-Hawary, "A Survey of particle swarm optimization applications in electric power systems", *IEEE Transactions on Evolutionary Computation*, Vol. 14, No. 4, Page(s) 913 – 918, Aug. 2009.
- [6] A. Banks, J. Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part I: background and development", *Natural Computing: an international journal*, Vol. 6, 4, pp. 467-484, 2007.
- [7] A. Banks, J. Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative

- applications”, *Natural Computing: an international journal*, Vol. 7, 2, pp.109-124, 2008.
- [8] 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.2, pp. 58-73, Feb. 2002.
- [9] F. Van den Bergh, *An analysis of particle swarm optimizers*, Ph.D. dissertation, Univ. Pretoria, South Africa, 2002.
- [10] V. Kadiramanathan, K. Selvarajah, and P. J. Fleming, “Stability analysis of the particle dynamics in particle swarm optimizer,” *IEEE Transactions on Evolutionary Computation*, vol.10, no.3, pp.245-255, Jun. 2006.
- [11] R. Poli, “Mean and variance of the sampling distribution of particle swarm optimizers during stagnation”, *IEEE Transactions on Evolutionary Computation*, Vol. 13, Issue. 4, On page(s): 712-721, 2009.
- [12] N. R. Samal, A. Konar, S. Das and A. Abraham, “A closed loop stability analysis and parameter selection of the particle swarm optimization dynamics for faster convergence,” in *Proc. Congr. Evolu. Compu.(CEC 2007)*, Singapore, pp. 1769-1776, 2007.
- [13] S. Helwig and R. Wanka, “Theoretical analysis of initial particle swarm behavior”, In *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature*, G. Rudolph et al. (Eds.), LNCS 5199, pages 889–898, Dortmund, Germany, September 2008.
- [14] I. C. Trelea, “The particle swarm optimization algorithm: convergence analysis and parameter selection,” *Inf. Process. Letters*. vol. 85, pp. 317-325, Mar. 2003.
- [15] E.F.Campana, G.Fasano, A.Pinto, “Dynamic system analysis and initial particles position in Particle Swarm Optimization”, *IEEE Swarm Intelligence Symposium 2006*, Indianapolis 12-14 May 2006.
- [16] C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer, 2007.
- [17] T. Ray, “Constrained robust optimal design using a multiobjective evolutionary algorithm,” in *Proc. IEEE Congr. Evol. Comput.*, Vol. 1, pp. 419–424, May 2002.
- [18] S. Mostaghim and J. Teich, “Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO),” in *Proc. IEEE Swarm Intell. Symp.* pp. 26–33, Apr. 2003.
- [19] J. E. Fieldsend, R. M. Everson, and S. Singh, “Using unconstrained elite archives for multiobjective optimization,” *IEEE Transactions on Evolutionary Computation*, Vol. 7, No. 3, pp. 305–323, Jun. 2003.
- [20] X. Hu, “Particle swarm optimization,” in *Tutorial of the IEEE Swarm Intell. Symp.* 2006.
- [21] T. Ray and K. Liew, “A swarm metaphor for multi-objective design optimization,” *Engineering Optimization*, vol. 34, no. 2, pp. 141–153, Mar. 2002.
- [22] C. Coello and M. Lechuga, “MOPSO: A proposal for multiple objective particle swarm optimization,” in *Proc. IEEE Congr. Evol. Comput.*, May 2002, vol. 2, pp. 1051–1056.
- [23] C. Coello, G. Pulido, and M. Lechuga, “Handling multiple objectives with particle swarm optimization,” *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 256–279, Jun. 2004.
- [24] X. Li, “A non-dominated sorting particle swarm optimizer for multi-objective optimization,” in *Proc. Genetic and Evol. Comput. Conf.*, pp. 37–48, Jul. 2003.
- [25] M. Salazar-Lechuga and J. E. Rowe, “Particle swarm optimization and auto-fitness sharing to solve multi-objective optimization problems,” in *Proc. IEEE Swarm Intell. Symp.*, May 2006, pp. 90–97.
- [26] X. Hu and R. Eberhart, “Multiobjective optimization using dynamic neighborhood particle swarm optimization,” in *Proc. IEEE Congr. Evol. Comput.*, May 2002, vol. 2, pp. 1677–1681.
- [27] K. Parsopoulos and M. Vrahatis, “Recent approaches to global optimization problems through particle swarm optimization,” *Natural Computing*, vol. 1, pp. 235–306, May 2002.
- [28] T. Hanne, “On the convergence of multiobjective evolutionary algorithms”, *European Journal of Operational Research*, Vol. 117, No. 3, Page(s) 553–564, 1999.
- [29] B. C. Kuo, *Digital Control System*, Oxford, NY, 1992.
- [30] J. D. Schaffer, “Multiple Objective Optimization with Vector Evaluated Genetic Algorithms” In *Proceedings of the 1st international Conference on Genetic Algorithms* J. J. Grefenstette, Ed. L. Erlbaum Associates, Hillsdale, NJ, 93-100, 1985.
- [31] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *IEEE Transactions on Evolutionary Computation*, vol. 7, pp. 117-132, Apr. 2003.
- [32] S. Agrawal, Y. Dashora, M. K. Tiwari and Y-J. Son, “Interactive particle swarm: a pareto-adaptive metaheuristic to multiobjective optimization”, *IEEE Transactions on Systems, Man and Cybernetics Part A--Systems and Humans*, Vol. 38, No. 2, pp. 258--277, March 2008. W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.