Multiple Particle Swarm Optimizers with Inertia Weight for Multi-objective Optimization

Hong Zhang, Member, IAENG

Abstract—An improved particle swarm optimizer with inertia weight (PSOIW) was applied to multi-objective optimization (MOO). In this paper we present a method of multiple particle swarm optimizers with inertia weight (MPSOIW), which belongs to a kind of the methods of cooperative particle swarm optimization. The crucial idea of the MPSOIW, here, is to reinforce the search ability of the PSOIW by the union's power of plural swarms. To demonstrate its effectiveness and search performance, computer experiments on a suite of 2-objective optimization problems are carried out by a weighted sum method. The resulting Pareto-optimal solution distributions corresponding to each given problem indicate that the linear weighted aggregation among the adopted three kinds of dynamically weighted aggregations is the most suitable for acquiring better search results. Throughout quantitative analysis to experimental data, we clarify the search characteristics and performance effect of the MPSOIW contrast with that of the PSOIW and MPSOIW.

Index Terms—particle swarm optimization, swarm intelligence, hybrid search, multi-objective optimization, weighted sum method.

I. INTRODUCTION

MULTI-objective optimization (MOO) is the processing of optimizing simultaneously two and more conflicting objectives subject to certain constraints [4], [6]. Since many practical problems are involved in MOO, which can be mainly found in different domains of science, technology, industry, finance, automobile design, aeronautical engineering and so on [8], [11], [23], how to efficiently deal with MOO becomes a live issue, and is centered on the development of the treatment technique.

Particle swarm optimization (PSO), which was created by Kennedy and Eberhart in 1995, is an adaptive, stochastic, and population-based optimization technique [15]. Based on the special features, i.e., information exchange, intrinsic memory, and directional search, the technique has higher latent search ability in optimization compared to some methods of evolutionary computation (EC) such as genetic algorithms and genetic programming [19], [20], [26], [27]. Especially, in recent years, a large number of studies, and investigations on cooperative PSO in relation to symbiosis, group behavior, and synergy are in the researcher’s spotlight. Various kinds of the methods of cooperative PSO, for example, hybrid PSO, multi-layer PSO, multiple PSO with decision-making strategy etc. were published [2], [10], [17], [27]. In contrast to those methods running a single particle swarm, many attempts and strategies can be perfected with operating multiple particle swarms for more efficiently finding an optimal solution or near-optimal solutions [3], [14], [17], [28]. Owing to the plain advantage, utilizing the techniques of group searching, parallel and intelligent processing has become one of extremely important approaches to optimization, and a lot of publications and reports have been shown that the methods of cooperative PSO have better adaptability and higher search performance than ones of uncooperative PSO in dealing with various optimization and practical problems [18].

An improved particle swarm optimizer with inertia weight (PSOIW) was published [30]. For further upgrading its search performance to MOO, in this paper we propose to use a method of cooperative PSO, called multiple particle swarm optimizers with inertia weight (MPSOIW). The crucial idea of the MPSOIW, here, is to reinforce the search ability of the PSOIW by the union’s power of plural swarms. Although the search feature and performance of some PSO methods in MOO with fitness assignment manners such as criterion-based manner or dominance-based manner were studied and investigated [24], [25], there are insufficient results for systematically solving MOO problems by an aggregation-based manner, and analyzing the potential characteristics in details from the obtained experimental results [6], [16].

To demonstrate the effectiveness and performance effect of the MPSOIW, computer experiments on a suite of 2-objective optimization problems are carried out by a weighted sum method. For interpreting the information treatment and search effect of the method, we show the distributions of the obtained Pareto-optimal solutions corresponding to each given problem by respectively using three kinds of dynamically weighted aggregations (i.e. linear weighted aggregation, bang-bang weighted aggregation, and sinusoidal weighted aggregation6), point out that which one of them is the most suitable for acquiring good search results to the given MOO problems, and clarify the search characteristics and performance of the MPSOIW contrast with that of the PSOIW and MPSOIW.

II. BASIC CONCEPTS

For explaining how to treat with MOO by a fitness assignment manner, some basic concepts and definitions on a general MOO problem, Pareto-optimal solution, front distance, cover rate, a weighted sum method, and three kinds of dynamically weighted aggregations are briefly described.
A. MOO Problem

In general, the formulation of a MOO problem can be defined as follows.

\[
\text{Minimize } \quad (f_1(\bar{x}), f_2(\bar{x}), \ldots, f_I(\bar{x}))^T \\
\text{s.t. } \quad g_j(\bar{x}) \geq 0, \quad j = 1, 2, \ldots, J \\
\quad \quad \quad \quad \quad \quad \quad h_m(\bar{x}) = 0, \quad m = 1, 2, \ldots, M \\
\quad \quad \quad \quad \quad \quad \quad x_n \in [x_{nl}, x_{nu}], \quad n \in (1, 2, \ldots, N)
\]

where \( f_i(\bar{x}) \) is the \( i \)-th objective, \( g_j(\bar{x}) \) is the \( j \)-th inequality constraint, \( h_m(\bar{x}) \) is the \( m \)-th equality constraint, \( \bar{x} = (x_1, x_2, \ldots, x_N)^T \in \mathbb{R}^N = \Omega \) (search space) is the vector of decision variable, \( x_{nl} \) and \( x_{nu} \) are the superior boundary value and the inferior boundary value of each component \( x_n \) of the vector \( \bar{x} \), respectively.

Due to the given condition of \( I \geq 2 \), the \( I \)-objectives may be conflicting with each other. Under this circumstance, it is difficult to obtain the global optimum corresponding to each objective by traditional optimization methods at the same time. Consequently, the goal of handling the MOO problem is effectively to achieve a set of solutions that satisfy Pareto optimality for improvement of mental capacity.

B. Pareto-optimal Solution

A solution \( \bar{x}^* \in \Omega \) is said to be Pareto-optimal solution if and only if there does not exist another solution \( \bar{x} \in \Omega \) so that \( f_i(\bar{x}) \) is dominated by \( f_i(\bar{x}^*) \). The formula of the above relationship is expressed as

\[
f_i(\bar{x}) \not\preceq f_i(\bar{x}^*) \quad \forall i \in I \quad \text{iff} \quad f_i(\bar{x}) \not\preceq f_i(\bar{x}) \quad \exists i \in I
\]

In other words, this definition says that \( \bar{x}^* \) is a Pareto-optimal solution if there exists no feasible solution (vector) \( \bar{x} \) which would decrease some criteria without causing a simultaneous increase in at least one other criterion.

Furthermore, all the Pareto-optimal solutions for a given MOO problem are composed of the Pareto-optimal solution set (\( P^* \)), or the Pareto front (\( PF \)).

C. Weighted Sum Method

There are some fitness assignment manners such as aggregation-based one, criterion-based one, and dominance-based one, which are used for MOO [7], [11]. As to be generally known, a conventional weighted sum (CWS) method is a straightforward approach applied to deal with MOO problems. In this case, the different objectives are summed up to a single scalar \( F_s \) (criterion) with some prescribed weights as follows.

\[
F_s(\bar{x}) = \sum_{i=1}^{I} c_i f_i(\bar{x})
\]

where \( c_i (i = 1, 2, \ldots, I) \) is the non-negative weight. During the optimization, usually, these weights are fixed by the constraint of \( \sum_{i=1}^{I} c_i = 1 \), and prior knowledge is also needed to specify these weights for obtaining good solutions.

To thoroughly conquer the weakness of the CWS method run, the following dynamically weighted sum (DWS) method is often used to MOO in practice. The criterion \( F_d \) of the method can be expressed as follows.

\[
F_d(t, \bar{x}) = \sum_{i=1}^{I} c_i(t) f_i(\bar{x})
\]

where \( t \) is time-step to search, and \( c_i(t) \geq 0 \) is the dynamic weight. In order to present the method, a 2-objective optimization problem is considered as an example. Hence, the definitions of three kinds of the adopted dynamically weighted aggregations are expressed below.

- **Linear weighted aggregation (LWA):**

\[
c_1(t) = \text{mod}(\left\lfloor \frac{t}{T} \right\rfloor, 1), \quad c_2(t) = 1 - c_1(t)
\]

- **Bang-bang weighted aggregation (BWA):**

\[
c_1(t) = \frac{\text{sign}((2\pi t/T) + 1)}{2}, \quad c_2(t) = 1 - c_1(t)
\]

- **Sinusoidal weighted aggregation (SWA):**

\[
c_1(t) = \left| \frac{\pi t}{T} \right|, \quad c_2(t) = 1 - c_1(t)
\]

where \( T \) is a period of the variable weights in the above equations.

D. Front Distance

Front distance is expressed as a metric of checking how far the elements are in the set of non-dominated solutions found from those in the true Pareto-optimal solution set. It directly reflects the estimation accuracy of the optimizer used. Concretely, the definition of front distance (FD) is expressed as

\[
FD = \frac{1}{Q} \sum_{q=1}^{Q} d_q, \quad d_q = f_i(\bar{x}_q^*) - f_i(\bar{x}_q), \quad \forall i \in I
\]

where \( Q \) is the number of the elements in the set of non-dominated solutions found, and \( d_q \) is the Euclidean distance (measured in objective space) between each of these obtained optimal solutions, \( \bar{x}_q^* \), and the nearest member, \( \bar{x}_q \), of the true Pareto-optimal solution set.

E. Cover Rate

Cover rate (CR) is an other metric for checking the coverage of the elements being in the set of non-dominated solutions found to the Pareto front. This is because the estimation accuracy is insufficiency to reveal the distribution status of the obtained Perato-optimal solutions and their possibility for dealing with the given problem.

Here, the formulation of CR is mathematically expressed by

\[
CR = \frac{1}{T} \sum_{i=1}^{I} CR_i
\]

where \( CR_i \) is the partial cover rate corresponding to the \( i \)-th objective, which is defined as

\[
CR_i = \frac{\sum_{l=1}^{\Gamma} \gamma_l}{\Gamma}
\]

where \( \Gamma \) is the number of dividing the \( i \)-th objective space which is from the minimum to the maximum of the fitness value, i.e. \([f_i(\bar{x})^{\text{min}}, f_i(\bar{x})^{\text{max}}]\), and \( \gamma_l \in (0, 1) \) indicates the existence status of the obtained optimal solutions in the \( l \)-th subdivision for the \( i \)-th objective.
### III. ALGORITHMS

For the convenience of the following description to the used every optimizer, let the search space be $N$-dimensional, the number of particles of a swarm be $P$, the position of the $i$-th particle be $\vec{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iN})^T \in \Omega$, and its velocity be $\vec{v}_i = (v_{i1}, v_{i2}, \ldots, v_{iN})^T \in \Omega$, respectively.

#### A. The PSOIW

To overcome the weak convergence of the original PSO [1], [5], Shi et al. modified the update rule of the $i$-th particle’s velocity by constant reduction of the inertia coefficient over time-step [9], [21]. Concretely, the formulation of the particle swarm optimizer with inertia weight (PSOIW) is defined as

$$
\begin{align*}
\vec{x}_{k+1} &= \vec{x}_k + \vec{v}_{k+1} \\
\vec{v}_{k+1} &= w(k) \vec{v}_k + w_1 \vec{r}_1 \otimes (\vec{p}_k - \vec{x}_k) + w_2 \vec{r}_2 \otimes (\vec{q}_k - \vec{x}_k)
\end{align*}
$$

where $w_1$ and $w_2$ are coefficients for individual confidence and swarm confidence, respectively, $\vec{r}_1, \vec{r}_2 \in \mathbb{R}^N$ are two random vectors, each element of which is uniformly distributed on the interval $[0, 1]$, and the symbol $\otimes$ is an element-wise operator for vector multiplication. $\vec{p}_k = \{ \arg \max_{i=1,2,\ldots,N} g(\vec{x}_i) \}$, where $g(\vec{z})$ is the criterion value of the $i$-th particle at time-step $k$ is the local best position of the $i$-th particle up to now, $\vec{q}_k = \{ \arg \max_{i=1,2,\ldots,N} g(\vec{p}_i) \}$ is the global best position among the whole particles at time-step $k$. $w(k)$ is the following variable inertia weight which is linearly reduced from a starting value $w_s$ to a terminal value $w_e$ with the increment of time-step $k$.

$$
w(k) = w_s + \frac{w_e - w_s}{K} \times k
$$

where $K$ is the number of iteration for the PSOIW run. In the original PSOIW, two terminal values, $w_s$ and $w_e$, are set to 0.9 and 0.4, respectively, and $w_1 = w_2 = 2.0$ are used as same as the original PSO.

Owing to the bigger difference between the two boundary values of the variable inertia weight, it is obvious that the search behavior of the PSOIW achieves a search shift which smoothly changes from exploratory mode to exploitative one in the whole optimization process. Hence, this way is very simple and useful for conquering the weakness of the PSO in convergence and enhancing the solution accuracy. On the other hand, the shortcoming of the PSOIW is easily to fall into a local minimum and hardly to escape from that place in dealing with multimodal problems because the terminal value $w_e$ is set to small.

#### B. The PSOIWα

For alleviating the weakness of the PSOIW search, we introduce the LRS [22], [29] into the PSOIW to form a hybrid search optimizer (called PSOIWα). Implementing the PSOIWα, here, is to enable a particle swarm search escapes from local minimum sooner for efficiently obtaining an optimal solution or near-optimal solutions.

The PSOIWα’s procedure is implemented as follows.

1. **Initialize the Swarm:** Set the initial position of each particle $\vec{x}_i$ and the initial velocity $\vec{v}_i$.

2. **Update the Inertia Weight:** Compute the inertia weight $w(k)$ using the equation $w(k) = w_s + \frac{w_e - w_s}{K} \times k$.

3. **Update the Positions and Velocities:** Update the position and velocity of each particle according to the following equations:

   $$
   \begin{align*}
   \vec{x}_{k+1}^{new} &= \vec{x}_k + \vec{v}_{k+1} \\
   \vec{v}_{k+1} &= w(k) \vec{v}_k + w_1 \vec{r}_1 \otimes (\vec{p}_k - \vec{x}_k) + w_2 \vec{r}_2 \otimes (\vec{q}_k - \vec{x}_k)
   \end{align*}
   $$

4. **Evaluate the Fitness:** Evaluate the fitness of each particle using the objective function.

5. **Update the Personal Best and Global Best:** Update the personal best position $\vec{p}_i$ and the global best position $\vec{q}_k$ for each particle.

6. **Check the Stopping Criteria:** If the stopping criteria are met, stop the algorithm and return the best solution found. Otherwise, go back to step 3.

C. **The MPSOIWα**

For improving the search ability of the existent PSOIWα to MOO, we propose to use multiple particle swarm optimizers with inertial weight, MPSOIWα. Figure 1 illustrates a flowchart of the MPSOIWα.

1. **Initialize the Swarms:** Set the initial position of each particle $\vec{x}_i$ and the initial velocity $\vec{v}_i$.

2. **Update the Inertia Weight:** Compute the inertia weight $w(k)$ using the equation $w(k) = w_s + \frac{w_e - w_s}{K} \times k$.

3. **Update the Positions and Velocities:** Update the position and velocity of each particle according to the following equations:

   $$
   \begin{align*}
   \vec{x}_{k+1}^{new} &= \vec{x}_k + \vec{v}_{k+1} \\
   \vec{v}_{k+1} &= w(k) \vec{v}_k + w_1 \vec{r}_1 \otimes (\vec{p}_k - \vec{x}_k) + w_2 \vec{r}_2 \otimes (\vec{q}_k - \vec{x}_k)
   \end{align*}
   $$

4. **Evaluate the Fitness:** Evaluate the fitness of each particle using the objective function.

5. **Update the Personal Best and Global Best:** Update the personal best position $\vec{p}_i$ and the global best position $\vec{q}_k$ for each particle.

6. **Check the Stopping Criteria:** If the stopping criteria are met, stop the algorithm and return the best solution found. Otherwise, go back to step 3.

IV. **COMPUTER EXPERIMENTS**

To facilitate comparison and analysis of the search performance of the proposed MPSOIWα, the suite of 2-objective problems is used to test the performance of the proposed algorithm.
TABLE I
A SUITE OF 2-OBJECTIVE OPTIMIZATION PROBLEMS

<table>
<thead>
<tr>
<th>problem</th>
<th>objective</th>
<th>search range</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZDT1</td>
<td>(f_{11}(\vec{x}) = x_1), (g(\vec{x}) = 1 + \frac{9}{N-1} \sum_{n=2}^{N} x_n), (f_{12}(\vec{x}) = g(\vec{x}) \left(1 - \sqrt{\frac{f_{11}(\vec{x})}{g(\vec{x})}}\right)) (\Omega \in [0, 1]^N)</td>
<td></td>
</tr>
<tr>
<td>ZDT2</td>
<td>(f_{21}(\vec{x}) = x_1), (f_{22}(\vec{x}) = g(\vec{x}) \left(1 - \left(\frac{f_{21}(\vec{x})}{g(\vec{x})}\right)^2\right)) (\Omega \in [0, 1]^N)</td>
<td></td>
</tr>
<tr>
<td>ZDT3</td>
<td>(f_{31}(\vec{x}) = x_1), (f_{32}(\vec{x}) = g(\vec{x}) \left(1 - \sqrt{\frac{f_{31}(\vec{x})}{g(\vec{x})}} - \left(\frac{f_{31}(\vec{x})}{g(\vec{x})}\right) \sin\left(10\pi f_{31}(\vec{x})\right)\right)) (\Omega \in [0, 1]^N)</td>
<td></td>
</tr>
</tbody>
</table>

Table II gives the major parameters of the MPSOIW\(\alpha\) for solving the given problems in Table I. The choice of their values is referred to the results of some preliminary experiments.

TABLE II
MAJOR PARAMETERS OF THE MPSOIW\(\alpha\) RUN

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>the number of particles, (P)</td>
<td>10</td>
</tr>
<tr>
<td>the number of iterations, (K)</td>
<td>25</td>
</tr>
<tr>
<td>the number of period, (T)</td>
<td>2500</td>
</tr>
<tr>
<td>the number of random points, (U)</td>
<td>10</td>
</tr>
<tr>
<td>the search range of the LRS, (\sigma)</td>
<td>0.1</td>
</tr>
<tr>
<td>the number of multiple particle swarms, (S)</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig. 2. Solution distributions of the MPSOIW\(\alpha\) and MPSOIW by using the LWA (red-point), BWA (blue-point) and SWA (green-point), respectively. Notice: the distance between the experimental data sets for each subgraph is 0.05 (shift only in horizontal direction).

optimization problems [31] in Table I is used in the next computer experiments. The characteristics of the Pareto fronts of the given problems include the convex (ZDT1), concave (ZDT2), and discontinuous multimodal (ZDT3), respectively.

A. Performance Comparison

For the sake of observation, Figure 2 shows the resulting solution distributions of the MPSOIW\(\alpha\) and MPSOIW by using the LWA, BWA, and SWA, respectively. According to the distinction of each solution distribution corresponding to these given problems, the analytical judgment can be described as follows.

1) Regardless of the used methods either the MPSOIW\(\alpha\) or MPSOIW, and the characteristic of each given problems, the resulting features and solution distributions are nearly same.
2) Regardless of the used methods and the characteristics of the given problems, the conditions of solution distributions by using the BWA are worse than that by using the LWA or SWA special for the ZDT1 and ZDT3 problems.

3) In comparison with the solution distributions of using the LWA for both the ZDT1 (convex) and ZDT2 (concave) problems, the former is relatively in the higher density.

For quantitative analysis to the experimental results of the MPSOIWα and MPSOIW in Figure 2, Table III gives the statistical data, i.e. the number of the obtained optimal solutions \( \bar{x} \), and the corresponding \( FD \) and \( CR \) for each given problem.

The following features can be observed from Table III. Firstly, there is the most number of solutions obtained by using the LWA for the given problems even for the ZDT2 one in where a large number of Pareto-optimal solutions are in unstable position [13]. Secondly, the solution accuracy of the MPSOIWα is superior to that of the MPSOIW for each given problem. Thirdly, the obtained results of using the LWA in \( CR \) index are the best than that of using BWA and SWA, respectively. Fourthly, the search performance of using the LWA is not only much better than that of using the BWA, but also is relatively better than that of using the SWA as a whole.

Therefore, the effectiveness and search ability of the MPSOIWα are roughly confirmed by the above analytical results. Furthermore, better solution distribution and higher solution accuracy can be observed as well by using either the LWA or SWA. Our experimental results indicate that smooth change of their criteria with the growth of time-step \( t \) can make that the probability finding good solutions greatly goes up in the same period, \( T=2500 \), as evidence.

Based on the above mentioned comparison and observation, the relationship of domination reflecting the search performance (SP) of the MPSOIWα by using each dynamically weighted aggregation can be expressed as follows.

\[
SP_{LWA} \succ SP_{SWA} \succ SP_{BWA}
\]

The relationship of the above domination indicates that the uniform change of the weights can make the moving process of variable criterion to be equalization which raises the probability finding the Pareto-optimal solution to the maximum under the condition of implementing the same optimizer. Due to this reason, more good solutions can be easily obtained during the short search cycle, \( K = 25 \).

B. Effect of Multi-swarm Search

For equal treatment in search, the number of particles used in a swarm is the same to the total number of particles used in the 3-swarms. As an example, Figure 3 shows the resulting solution distributions of both the MPSOIWα and PSOIWα (i.e. \( P = 30 \)) by using the LWA. We can see that the density of solution distributions of the MPSOIWα are higher than that of the PSOIWα for each given problem.

Fig. 3. The solution distributions of the MPSOIWα (red-point) and PSOIWα (blue-point) by using the LWA.

Table IV gives the performance indexes, i.e. the number of the optimal solutions \( \bar{x} \) obtained by using the LWA, and the corresponding \( FD \) and \( CR \) for the given problems. By directly comparing the performance indexes with the MPSOIWα and PSOIWα, the big difference between the both experimental results clearly indicate the strong points of the multi-swarm search in dealing with the given MOO problems under the condition of the same number of particles.

![Table III](image-url)
used. It is demonstrated that the MPSOIWO is a powerful method of cooperative PSO to MOO.

V. CONCLUSIONS

In this paper, multiple particle swarm optimizers with inertia weight, MPSOIWO, has been presented to MOO. Based on the composition of the MPSOIWO, it is the simplest expansion of the existent PSOIWO, which has the advantages of a hybrid search with easy-to-operation as a method of cooperative PSO.

Applications of the MPSOIWO to the given suite of multiobjective optimization problems well demonstrated its effectiveness by the aggregation-based manner. Owing to the resulting experimental data by respectively using three kinds of dynamically weighted aggregations, it is observed that the search performance of the MPSOIWO is superior to that of the PSOIWO and MPSOIW, and the comparative analysis of the MPSOIWO shows that the search performance of using the LWA is better than that of using the BWA or SWA for the given MOO problems. Therefore, it is no exaggeration to say that our experimental results could offer an important evidence, i.e. choosing the dynamically weighted sum method with the LWA for efficiently dealing with complex MOO problems.

It is left for further study to apply the MPSOIWO to MOO problems in the real-world. Furthermore, in order to enhance the adaptability, efficiency, and solution accuracy of the MPSOIWO, the search strategies and attempts on prediction, intelligent and powerful cooperative PSO algorithms [2], [10], [30] will be discussed for MOO in near future.

REFERENCES


TABLE IV

<table>
<thead>
<tr>
<th>Problem</th>
<th>Method</th>
<th>Solution</th>
<th>FD (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZDT1</td>
<td>MPSOIWO</td>
<td>1254</td>
<td>2.234×10⁻⁸</td>
</tr>
<tr>
<td></td>
<td>PSOIW</td>
<td>522</td>
<td>6.661×10⁻⁸</td>
</tr>
<tr>
<td>ZDT2</td>
<td>MPSOIWO</td>
<td>272</td>
<td>1.198×10⁻⁸</td>
</tr>
<tr>
<td></td>
<td>PSOIW</td>
<td>231</td>
<td>9.938×10⁻⁸</td>
</tr>
<tr>
<td>ZDT3</td>
<td>MPSOIWO</td>
<td>1231</td>
<td>8.964×10⁻⁸</td>
</tr>
<tr>
<td></td>
<td>PSOIW</td>
<td>432</td>
<td>4.496×10⁻⁶</td>
</tr>
</tbody>
</table>

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)