Experimental Parameter Investigations on Particle Swarm Optimization Acceleration Coefficients

Jian Zhang

School of Information, Guangdong Ocean University, Zhanjiang, Guangdong, P. R. China, 524088, E-mail: jianzhang689@126.com

Abstract

Particle swarm optimization (PSO) is one of the most successful optimization techniques of swarm intelligence and has been fast developed in recent years. However, the performance of PSO is significantly depended on the acceleration coefficients $c_1$ and $c_2$ which control the exploration and convergence abilities. Parameters $c_1$ and $c_2$ are the “self-cognitive” coefficient and “social-influence” coefficient respectively and are both set to 2.0 in traditional studies. Even though some studies have been conducted and argued that the $c_1$ and $c_2$ are unnecessary to be 2.0 for good performance, few literatures that based on the experimental study of the two parameters can be found. This paper gives a comprehensive investigation on the acceleration coefficients $c_1$ and $c_2$ through a set of 13 unimodal and multimodal benchmark functions, in order to study how to set these two parameters for different functions in order to obtain better performance. The experimental results indicate a conclusion that the sum of $c_1$ and $c_2$ should be clamped in the interval of [3.5, 4.5]. This conclusion would be the guidelines and rule for adapting $c_1$ and $c_2$ during the running phases of PSO.

Keywords: Particle Swarm Optimization (Pso), Parameter Investigation, Acceleration Coefficients

1. Introduction

Particle swarm optimization (PSO) is one of the most important swarm intelligence algorithms for global optimization [1][2]. The PSO algorithm is inspired by the swarm behavior of bird flocking and has developed fast in recently years, mainly due to its simple concept and easy implementation [3]. PSO is a population-based, generation iterative algorithms like genetic algorithm (GA) [4]. However, it is novel that PSO uses only two formulations to update each particle’s velocity and position generation by generation instead of the selection, crossover, and mutation operations which are essential in GA [5].

Since its introduction in 1995, PSO has been fast developed, with lots of work on performance improvement and real-world application. In the aspect of performance improvement, many researches can be found that conducted on the acceleration coefficients. Suganthan [6] and Ratnaweera et al. [7] suggested the time varying acceleration coefficients. Zhan et al. [8] proposed an adaptive PSO (APSO) that controls both the inertia weight and the acceleration coefficients. Apart from the studies on parameters control, some other researches are conducted to improve the algorithm performance by hybriding PSO with other techniques like the simplex method [9] and tabu search [10], or by using a comprehensive learning strategy [12], an orthogonal learning strategy [13], or a replacement strategy [14]. Moreover, due to the simple concept and low computational cost, PSO has been proposed into various kinds of real-world applications, like aircraft flight planning [15], wireless sensor networks design [16], power electronic circuit optimization [17], job shop scheduling [18], traffic flow forecasting [19], router CPU time management [20], and PID control design [21], etc.

Among the amount of work on PSO as mentioned above, researchers have found that the PSO algorithm performance is significantly depended on the algorithm parameters, especially the acceleration coefficients [6]-[8]. In traditional PSO, the acceleration coefficients $c_1$ and $c_2$ are both set to be 2.0 [2]. Even though some studies have been conducted and argued that they are unnecessary to be 2.0 for good performance [6]-[8], they were designed by the researches’ experiences. Few literatures based on experimental study of the two parameters can be found.

In order to make clear how the acceleration coefficients $c_1$ and $c_2$ affect the algorithm performance on different problems, and make clear how to set these two parameters to obtain better algorithm performance, this paper the first time to make a comprehensive investigation on...
the acceleration coefficients $c_1$ and $c_2$ through a set of 13 unimodal and multimodal benchmark functions. Based on the comprehensive experimental results, this paper will summarize the values selection rules of these two parameters and the promising combination of $c_1$ and $c_2$ for better performance.

The rest of the paper is organized as follows. Section 2 introduces the PSO algorithm framework and makes a brief survey on the work on the acceleration coefficients. Section 3 makes the experimental study on the acceleration coefficients based on a set of 13 benchmark functions. Experimental results are discussed and the selection rules for setting the values of $c_1$ and $c_2$ are summarized. At last, Section 4 concludes this paper and highlights the future research work.

2. Particle Swarm Optimization

2.1. Algorithm Framework

In PSO, a swarm of particles are used to represent the potential solutions, and each particle $i$ has two vectors, the velocity $V_i = [v_{i1}, v_{i2}, \ldots, v_{iD}]$ vector and the position $X_i = [x_{i1}, x_{i2}, \ldots, x_{iD}]$ vector. Here $D$ means that the solution is in $D$-dimension space. In the initialization, the velocity and position of each particle are set randomly within the search space. During the evolutionary process, the particle $i$ is evaluated according to its present position. If the present fitness is better than the fitness of $pBest_i$, which stores the best solution that the $i$th particle has been explored so far, then the $pBest_i$ will be replaced by the current solution (include the position and fitness). At the same time of determining $pBest_i$, the algorithm selects the best $pBest_i$ of the swarm to be the globally best, which is regarded as $gBest$. Then, the velocity and position of each particle will be updated using the following two equations (1) and (2).

$$v_{id} = \omega v_{id} + c_1 r_{1d}(pBest_id - x_{id}) + c_2 r_{2d}(gBest_d - x_{id})$$ (1)

$$x_{id} = x_{id} + v_{id}$$ (2)

where $\omega$ is the inertia weight, and $c_1$, $c_2$ are the acceleration coefficients; $r_{1d}$ and $r_{2d}$ are two separately generated uniformly distributed random numbers in the ranges $[0, 1]$ for the $d$th dimension. After updating the velocity and position, iteration goes on until the stop condition is met.

2.2. Acceleration Coefficients $c_1$ and $c_2$

In order to gain a clear view that how the PSO works, the equation (1) for velocity update should be considered carefully. The first part of Eq. (1) represents the influence of the previous velocity, and the inertia weight $\omega$ is demonstrated to be important for the performance. The second part is considered as the “self-cognitive” component, which represents the memory and the personal thinking. This component encourages the particles to move towards their own best positions found so far. The third part of (1), however, is known as the “social-influence” component, which represents the cooperation of the swarm in finding the global optimal solution. This component always pulls the particles towards the globally best particle found so far.

The Eq. (1) shows that both the values of $c_1$ and $c_2$ have significant influences on the flying behavior. If $c_1$ and $c_2$ are too small, the particles have to use many generations to reach the good region of the search space. If $c_1$ and $c_2$ are too large, the particles will change their fly velocity dramatically and cannot converge to a certain region.

Kennedy and Eberhart [3] suggested the fixed values 2.0 for both $c_1$ and $c_2$. As the acceleration coefficients are multiplied by a uniform random value in range $[0, 1]$, they give a mean value of 1.0 to make the particle “over-fly” the target about half the time. However, the authors of [3] claimed that further researches should be carried out to show whether there were some other optimal values other than 2.0 for better performance. From a psychological standpoint, the cognitive component coefficient ($c_1$) and the social component coefficient ($c_2$) should be time-varying values in order to get various exploration and convergence abilities in...
different evolutionary states. Therefore, it is significant and promising to research how to set these two parameters for better performance.

Some attempts have been conducted in the literatures on the studies of $c_1$ and $c_2$. Suganthan [6] found that for different problems, there existed different combinations of values for $c_1$ and $c_2$, which were not necessary to be 2.0, that yielded good performance. Ratnaweera et al. [7] proposed a linearly decreasing strategy for the acceleration coefficient $c_1$ and a linearly increasing strategy for the acceleration coefficient $c_2$ during the evolutionary process. Zhan et al. [8] further adaptively controlled the values of $c_1$ and $c_2$ according to different evolutionary states.

However, a systemic study on the parameters $c_1$ and $c_2$ to investigate their influences on PSO performance and to provide some selection rules of their values for good performance is still in great need. This paper makes the first attempt to make such an experimental study based on 13 unimodal and multimodal benchmark functions.

3. Experimental Studies

3.1. Experimental Settings

The experiments are based on thirteen benchmark functions as listed in Table 1 [22]. The functions are divided into two groups, with 6 unimodal functions ($f_1$-$f_6$) and 7 multimodal functions ($f_7$-$f_{13}$). All functions are with $D=30$ dimensions and their search ranges are given in Table 1.

### Table 1. Thirteen benchmark functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Search Range</th>
<th>Name</th>
<th>Comments*</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1(x) = \sum_{i=1}^{D} x_i^2$</td>
<td>[-100, 100]$^D$</td>
<td>Sphere</td>
<td>U &amp; S</td>
</tr>
<tr>
<td>$f_2(x) = \sum_{i=1}^{D}</td>
<td>x_i</td>
<td>+ \prod_{i=1}^{D}</td>
<td>x_i</td>
</tr>
<tr>
<td>$f_3(x) = \sum_{i=1}^{D} (x_i - \frac{1}{x_i})^2$</td>
<td>[-100, 100]$^D$</td>
<td>Quadric</td>
<td>U &amp; L</td>
</tr>
<tr>
<td>$f_4(x) = \max_i (</td>
<td>x_i</td>
<td>, 1 \leq i \leq D)$</td>
<td>[-100, 100]$^D$</td>
</tr>
<tr>
<td>$f_5(x) = \sum_{i=1}^{D} (x_i + 0.5)^2$</td>
<td>[-100, 100]$^D$</td>
<td>Step</td>
<td>U &amp; S</td>
</tr>
<tr>
<td>$f_6(x) = \sum_{i=1}^{D} \text{random}(0,1)$</td>
<td>[-1.28, 1.28]$^D$</td>
<td>Noise</td>
<td>U &amp; S</td>
</tr>
<tr>
<td>$f_7(x) = \sum_{i=1}^{D} [100(x_i - x_i^2)]^2 + (x_i - 1)^2$</td>
<td>[-10, 10]$^D$</td>
<td>Rosenbrock</td>
<td>M &amp; L</td>
</tr>
<tr>
<td>$f_8(x) = \sum_{i=1}^{D} -x_i \sin(\sqrt{</td>
<td>x_i</td>
<td>})$</td>
<td>[-500, 500]$^D$</td>
</tr>
<tr>
<td>$f_9(x) = \sum_{i=1}^{D} [x_i^2 - 10 \cos(2\pi x_i) + 10]$</td>
<td>[-5.12, 5.12]$^D$</td>
<td>Rastrigin</td>
<td>M &amp; S</td>
</tr>
<tr>
<td>$f_{10}(x) = -20 \exp(-0.5 \sqrt{\frac{1}{D} \sum_{i=1}^{D} x_i^2} - \exp(\frac{1}{D} \sum_{i=1}^{D} \cos 2\pi x_i)) + 20 + e$</td>
<td>[-32, 32]$^D$</td>
<td>Ackley</td>
<td>M &amp; S</td>
</tr>
<tr>
<td>$f_{11}(x) = \frac{1}{4000} \sum_{i=1}^{D} x_i^2 - \sum_{i=1}^{D} \cos(\sqrt{</td>
<td>x_i</td>
<td>}) + 1$</td>
<td>[-600, 600]$^D$</td>
</tr>
<tr>
<td>$f_{12}(x) = \frac{\pi}{D} [100\sin^2(\pi x_i) + \sum_{i=1}^{D} (x_i - 1)^2 [1 + 10\sin^2(\pi y_i)]] + (y_i - 1)^2 + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$</td>
<td>[-50, 50]$^D$</td>
<td>Generalized Penalized</td>
<td>M &amp; S</td>
</tr>
<tr>
<td>$f_{13}(x) = 0.1 [\sin^2(3\pi x_i) + \sum_{i=1}^{D} (x_i - 1)^2 [1 + \sin^2(3\pi y_i)]] + (y_i - 1)[1 + \sin^2(2\pi y_i)] + \sum_{i=1}^{D} u(x_i, 5, 100, 4)$</td>
<td>[-50, 50]$^D$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
* U, M, S, and L stand for Unimodal, Multimodal, Variables Separable, and Variables Linkage, respectively

Experiments on each benchmark function can be briefly described as follows. When each function is optimized by PSO, we make parameters $c_1$ and $c_2$ span from 0.5 to 3.5 separately by 0.1 increased each step. Therefore, many couples of $c_1$ and $c_2$ form (exactly 31 by 31 couples). PSO variants with a specific couple of $c_1$ and $c_2$ values optimize the function for 100 independent times. The mean final fitness value of these 100 runs is calculated. A matrix is used to store this mean value and then a matrix with 31 rows by 31 columns forms. We plot this matrix to a contour figure with the $x$-coordinate standing for $c_1$ from 0.5 to 3.5 and $y$-coordinate...
standing for $c_2$ from 0.5 to 3.5. This contour figure can compare the performance PSO with different couples of $c_1$ and $c_2$.

For the other experimental settings, the PSO algorithm has a population size of 20 particles, with the inertia weight $\omega$ being 0.5, and with the maximal generation number being 2000.

For the other experimental settings, the PSO algorithm has a population size of 20 particles, with the inertia weight $\omega$ being 0.5, and with the maximal generation number being 2000.

![](figure2.png)

**Figure 2.** The Contour Plot for Different Couples of $c_1$ and $c_2$ on Multimodal Functions, the Higher the Color in the Color Indicating Bar, the Better (Smaller) Value it Represents.

### 3.2. Experimental Results Discussions

The contour figures for the unimodal functions ($f_1$-$f_6$) are given in Fig. 1 and the contour figures for the multimodal functions ($f_7$-$f_{13}$) are given in Fig. 2.

The Fig. 1 and Fig. 2 show that for most of the functions, the sum of $c_1$ and $c_2$ lay in the interval $[3.5, 4.5]$ would result in good solutions. Therefore, the traditional values of 2.0 for...
both $c_1$ and $c_2$ which make the sum 4.0 are also a good choice for PSO. The figures show that no matter for unimodal functions or multimodal functions, the contours share a similar characteristic that the figures are divided by the contour lines that are nearly parallel to the diagonal crossing the left-up corner to the right-down corner, and the value becomes worse and worse as the contour lines being further away from this main diagonal. That is to say, the combination of $c_1$ and $c_2$ should be chosen under the rule that their sum is near 4.0, more specifically, being in the interval of $[3.5, 4.5]$, if we want to obtain good performance.

Moreover, the figures for the unimodal functions yield a characteristic that if the value of $c_1$ is relative smaller, the value of $c_2$ can span in a relative wider range while the opposite conclusion can be drawn if the value of $c_1$ is relative larger. For example, Fig. 1(b) shows that, when $c_1$ is set as 1.0, $c_2$ can be set approximately from 2.25 to 3.25 (the width is about 1.00) to get good solution, whilst if $c_1$ is set as 3.0, $c_2$ should be clamped in a relative narrower range (approximately from 0.75 to 1.5, the width is about 0.75) if we want to get the good solution.

However, such a characteristic is not very evident for some multimodal functions. For example, the Fig. 2(c) shows that the good values for $c_2$ are in the interval $[3.0, 3.5]$ when $c_1$ is 0.5, whilst the good values for $c_2$ are in the interval $[0.5, 1.5]$ when $c_1$ is 3.0. Therefore, the span range for $c_2$ is even larger when the value of $c_1$ is larger. This means that setting $c_1$ relative large may be good for the PSO performance on multimodal functions.

Observed from the Fig. 1 and Fig. 2 and by the discussions as above, we can conclude three rules for choosing good values for $c_1$ and $c_2$ as follows.

1) The sum of $c_1$ and $c_2$ should be clamped in the interval $[3.5, 4.5]$;  
2) The $c_1$ value can be relative small while the $c_2$ value can be relative large when optimizing unimodal problems;  
3) The $c_1$ value can be relative large while the $c_2$ value can be relative small when optimizing multimodal problems.

4. Conclusion and Future Work

A comprehensive experimental study on the acceleration coefficients $c_1$ and $c_2$ has been carried out in this paper, and the rules for choosing $c_1$ and $c_2$ are summarized from the experimental results. The experimental results show that the sum of the $c_1$ and $c_2$ values should be clamped in the interval of $[3.5, 4.5]$ for a PSO algorithm to obtain good performance.

It is expected that the parameter setting rules for $c_1$ and $c_2$ can be used by researchers who want to enhance the PSO algorithm via controlling the acceleration coefficients. For example, even though some studies have been conducted on controlling the values of $c_1$ and $c_2$ in the literatures [6][7][8], it seems that the study in [6] did not use constraints for $c_1$ and $c_2$, the study in [7] just simply decreased $c_1$ from 2.5 to 0.5 while increased $c_2$ from 0.5 to 2.5, and the study in [8] clamped the sum of $c_1$ and $c_2$ in $[3.0, 4.0]$. These studies may be further enhanced if the parameter setting rules obtained in this paper are used in these existing studies. In the future work, further experimental studies based on more complex benchmarks, such as those proposed in CEC2005, will be carried out for more comprehensive investigations. Moreover, researches on using the parameter setting rules for the acceleration coefficients $c_1$ and $c_2$ obtained in this paper will be carried out to enhance the traditional and existing improved PSO variants’ performance.

5. Acknowledgements

This work is partly supported by the Industry-Academia-Research Cooperation Project of Guangdong Province in China under Grant No. 2011B090400008.

6. References


