Report for Assignment I

1. Description about Two Algorithms

1.1 PSO Algorithm

The first one is PSO algorithm (CEC2017-BoundContrained). In this algorithm, the maximum generation, the number of particles(ps), and the number of dimensions are obtained from the main function, while the parameter runs、acceleration factor(cc)、and the inertia factor(iwt) can be selected by people to learn the relationship between different parameters and results of functions used.

The first parameter to be introduced is runs, it controls the run number of each function and it can influence the result. When function number is 3, the value remains at around 300 in figure 1, but for function 4 in figure2, it is seen that when the runs equals to 30, the value is not steady enough, it means that different numbers of runs could be effective.

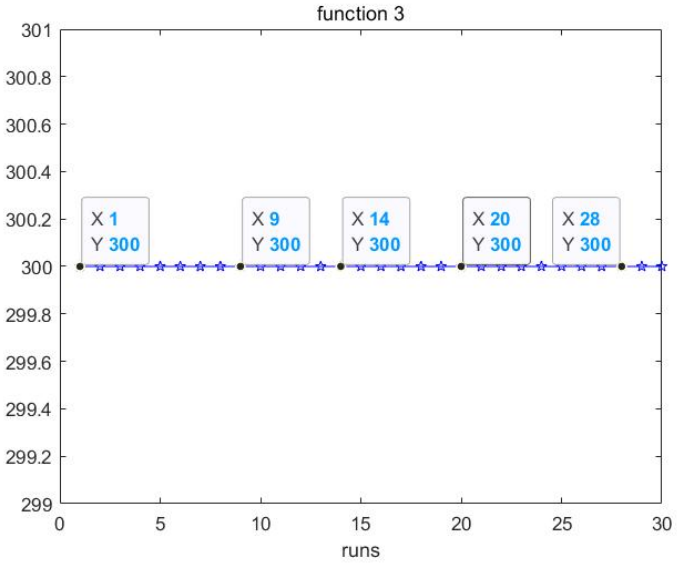

Figure 1

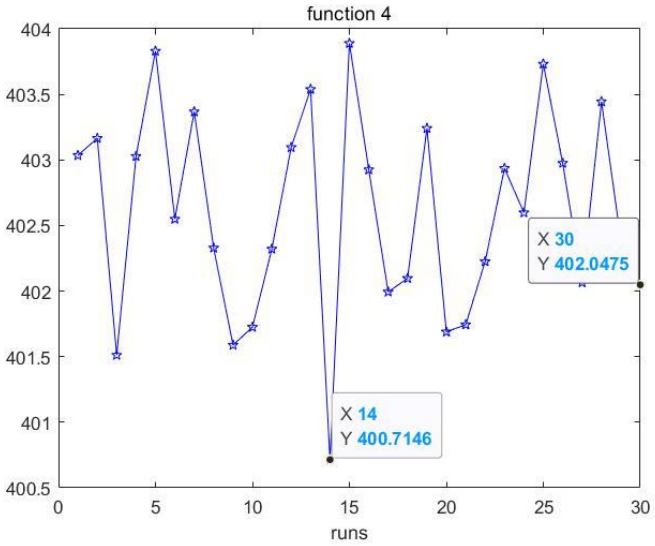

Figure 2

The second parameter to be introduced is learning factor cc, which parameter influences the updating of velocity vectors, so it will influence the direction for particle to travel. And the original codes about it are shown:

```
aa=cc (1) .* rand(ps,D) .* (pbest-pos) + cc(2) .* rand(ps,D) .* (gbestrep-pos);
vel=iwt(i). * vel + aa.
pos=pos + vel.
```

During this period, it probably changes the local and global search ability.

The third parameter iwt(inertia factor) can also influence the local and global search ability, the original code is:

iwt=0.9-(1:me). *(0.5./me).

vel=iwt(i). *vel+aa.

1.2 CLPSO algorithm

The second one is CLPSO algorithm(2006-IEEE-TEC-CLPSO). Important parameters are acceleration constants, Pc, and iwt. The influence of acceleration constants and iwt is similar as that of PSO algorithms, while the effect of Pc is a little different. For crossover probability Pc, it is an improvement of PSO, it helps to generate the pbest, which can decrease premature convergence to some degree.

2. Description about How Important Parameters is Tuned

2.1 Parameters Tuned for PSO Algorithm

For PSO algorithms, Firstly, selecting 5 functions to use during parameters adjusting period. Running this algorithm (runs=1000, maximum generation=5000), the performance of function 3 to function 12 is in figure 3.

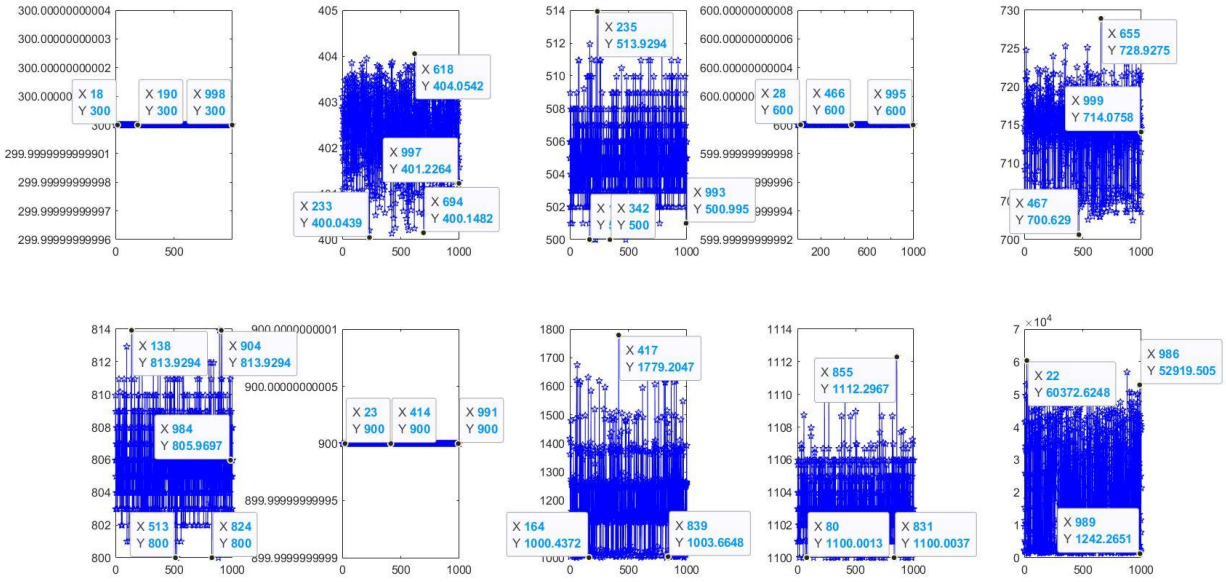

Figure 3

Selecting function 4 to function 8 to test the performance before and after the change of parameters.

Firstly, I change cc to test function 4 to function 8. The performance of it is in figure 4 and table 1.

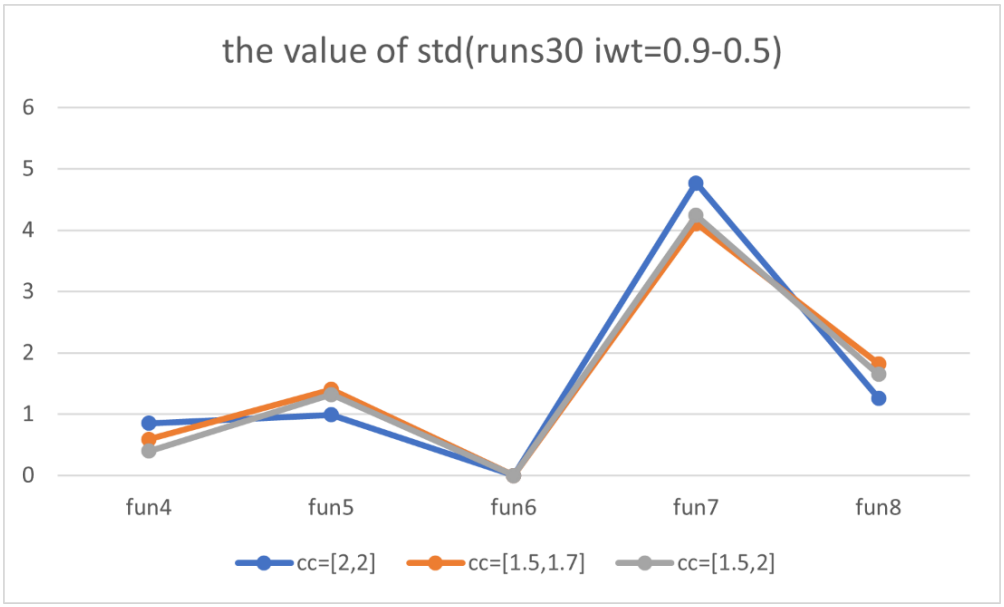

Figure 4

Table 1

performance of different set pf cc on fun4 to func8					
	function number	max	medium	min	std
cc= [2,2]	fun4	403	401.4	400	0.8492
	fun5	504	502.3	501	0.9892
	fun6	600	600	600	0
	fun7	715.5	710	702	4.769
	fun8	806	803.3	800	1.257
cc= [1.5,1.7]	fun4	402.4	400.8	400	0.5874
	fun5	506	503.1	501	1.401
	fun6	600	600	600	0
	fun7	717.4	712	703.6	4.108
	fun8	808	803.9	801	1.82
cc= [1.5,2]	fun4	401.7	400.6	400.2	0.3974
	fun5	506	502.8	501	1.317
	fun6	600	600	600	0
	fun7	717	712.5	702.6	4.241
	fun8	808	803.5	801	1.647

Secondly, I change iwt to see the convergence speed for different iwt range. It can be seen from the figure 5 to figure 8 that, with the decrease of iwt range, the convergence speed gets quicker. Figure 5 and 6 are for function 4, Figure 7 and 8 are for function 5.

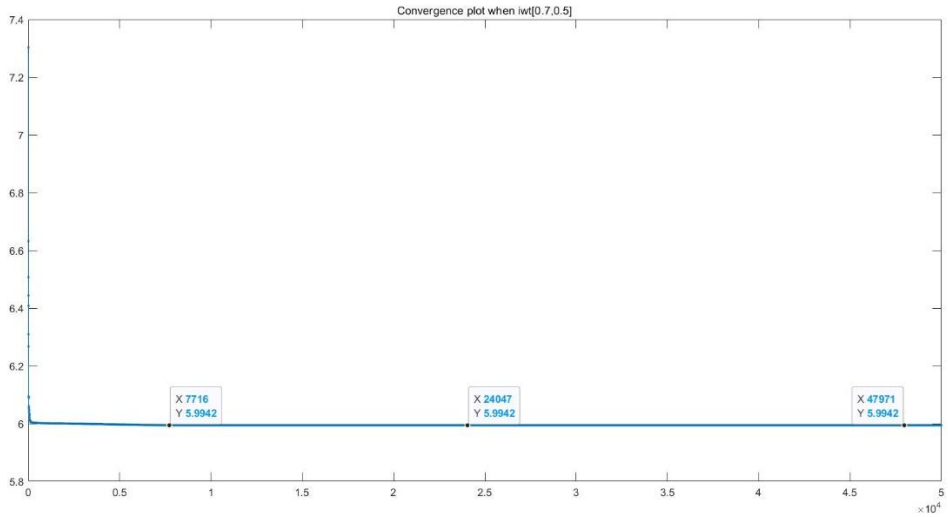

Figure 5

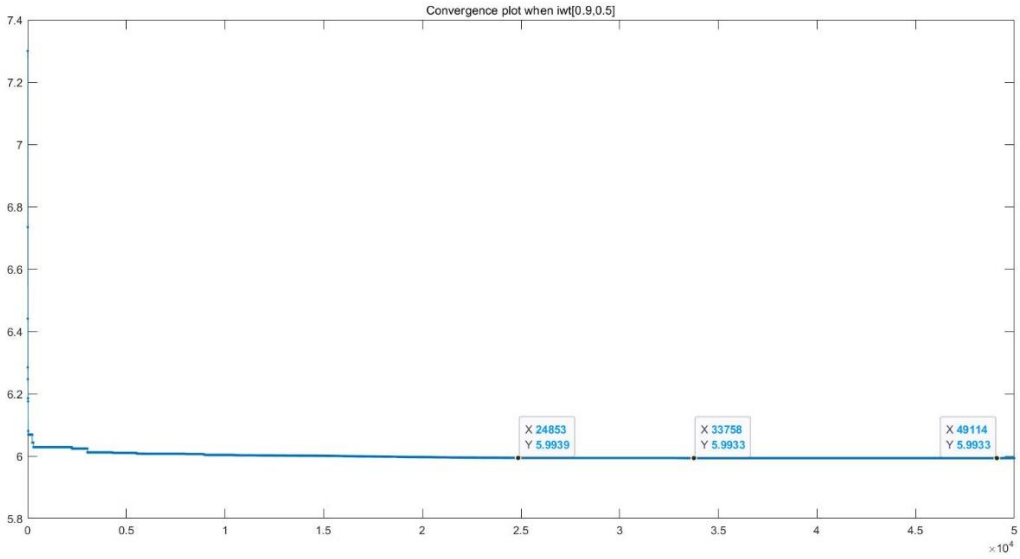

Figure 6

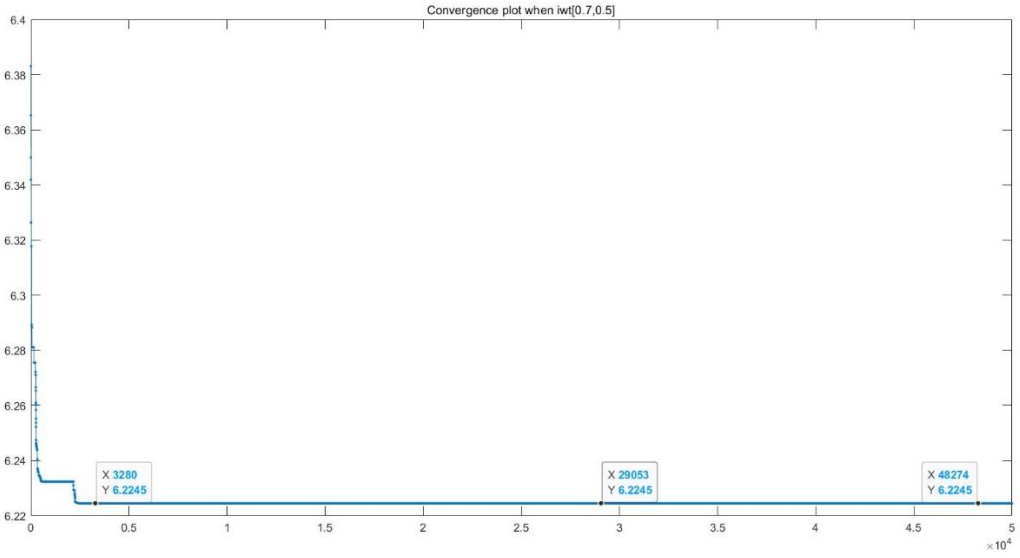

Figure 7

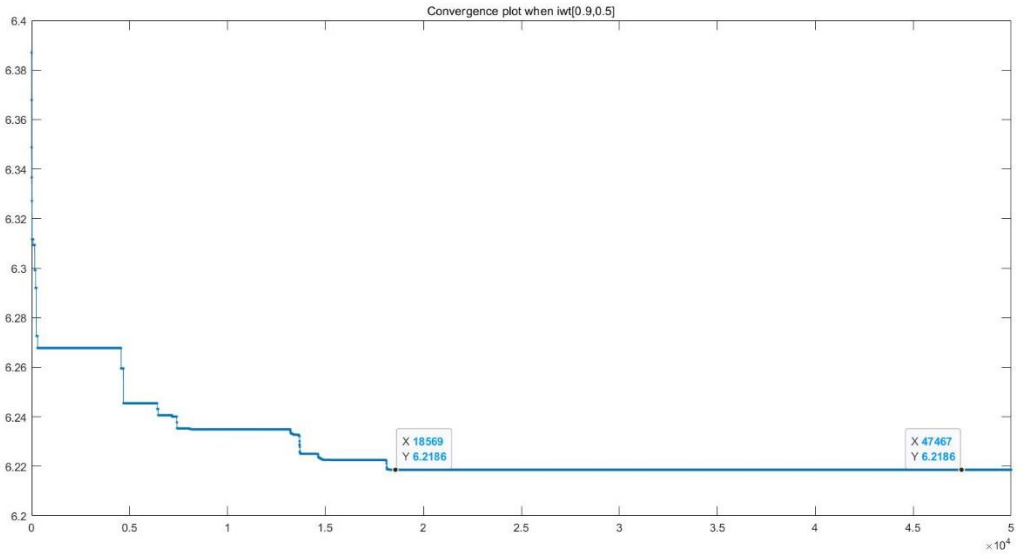

Figure 8

2.2 Parameters Tuned for CLPSO Algorithm

For second algorithm, to compare the performance of PSO and CLPSO algorithms, selecting the same batch of functions to use, which are function 4 to function 8.

Firstly, I change cc to test function 4 to function 8. The performance of it is in figure 9 and table 10.

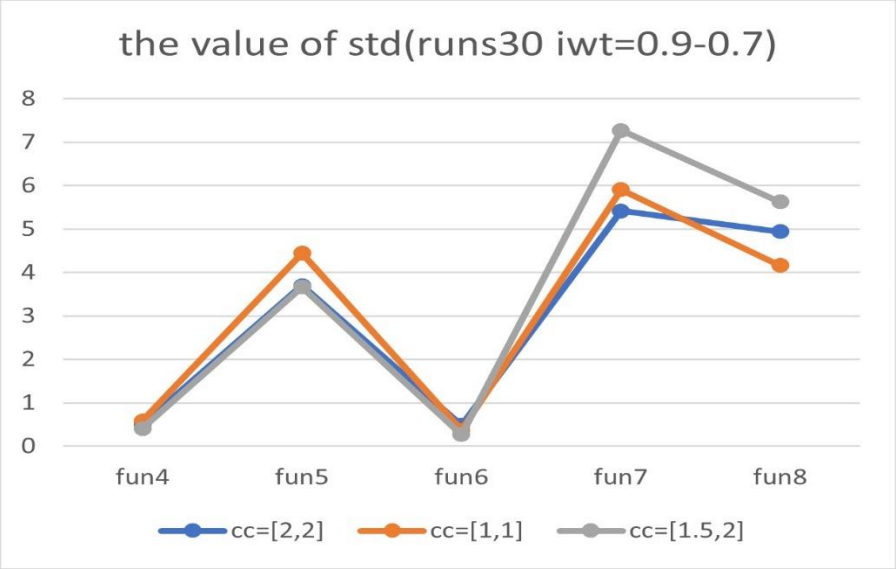

Figure 9

performance of different set pf cc on fun4 to func8					
	function number	min	max	medium	std
cc= [2,2]	fun4	406.8	409.2	407.4	0.5031
	fun5	519	532.8	526.7	3.699
	fun6	601	603	601.9	0.4856
	fun7	739.8	765.9	755.9	5.416
	fun8	812.6	840.1	828.2	4.94
cc= [1,1]	fun4	406.2	408.8	407.6	0.5814
	fun5	515.7	532.9	523.7	4.443
	fun6	601.1	602.7	602	0.3656
	fun7	742.5	768.8	756	5.909
	fun8	820.9	838.3	830.5	4.16
cc= [1.5,2]	fun4	406.2	408.1	407.4	0.4086
	fun5	513	531.1	525.5	3.667
	fun6	601.4	602.3	601.9	0.2684
	fun7	742.2	767	753.6	7.274
	fun8	8121	837.6	828.4	5.624

Table 2

Secondly, I change iwt to see the convergence speed for different iwt range. Figure 10 and 11 are for function 4, Figure 11 and 12 are for function 5.

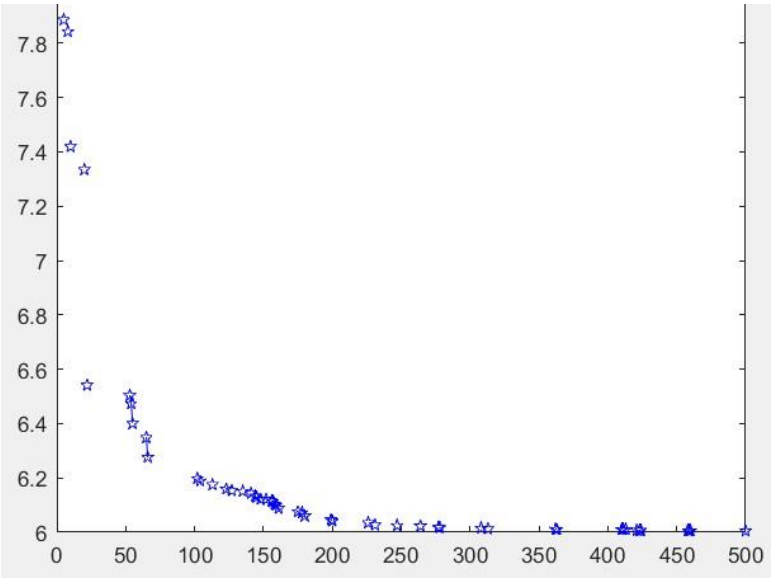

Figure 10(iwt=0.9)

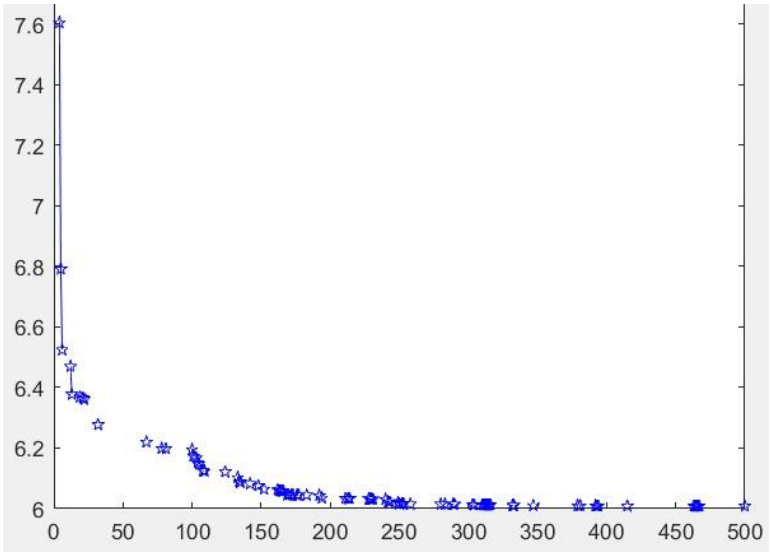

Figure 11 (iwt=0.7)

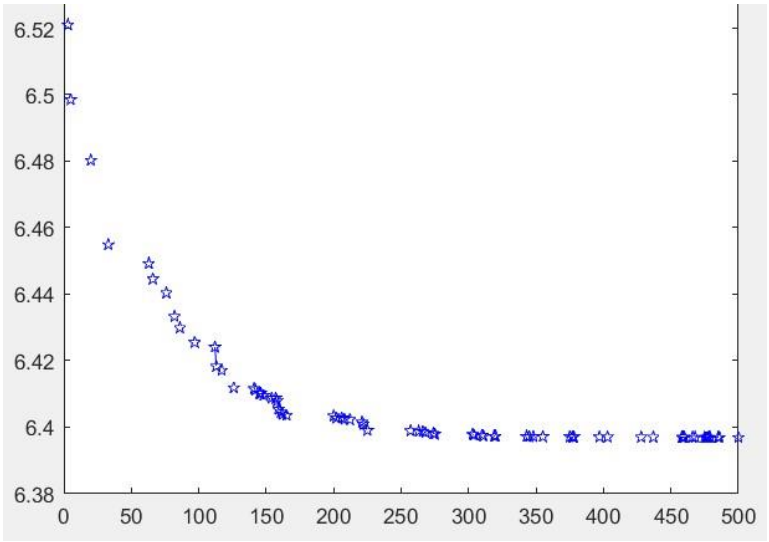

Figure 12 (iwt=0.9)

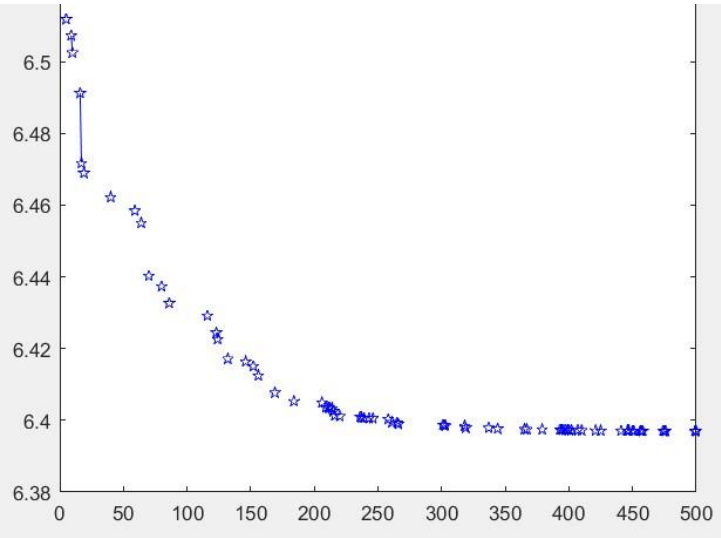

Figure 13 (iwt=0.7)

Thirdly, I select different pc to draw the standard deviation figure.

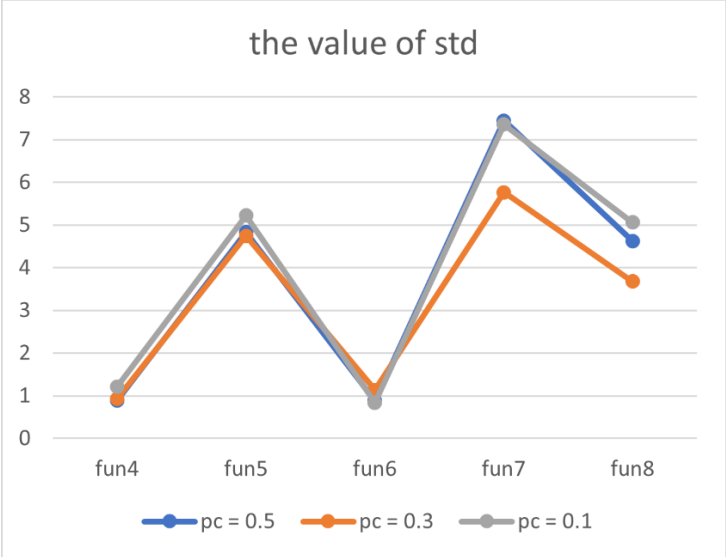

Figure 14

Table 3

performance of different set pf pc on fun4 to func8(cc=[2,2])					
	function number	min	max	medium	std
pc = 0.5	fun4	408.2	412	409.4	0.8882
	fun5	518.3	537.6	529.2	4.834
	fun6	602.6	606.2	604.8	0.886
	fun7	747.9	776.2	762.1	7.443
	fun8	825.6	843.9	834.3	4.628
pc = 0.3	fun4	408.2	412.3	409.6	0.9251
	fun5	521.9	539.9	530	4.75
	fun6	601.7	607.5	604.6	1.132
	fun7	749.9	774.1	762.6	5.767
	fun8	822.2	840	834.2	3.68
pc = 0.1	fun4	407.8	412.4	410.1	1.21
	fun5	516.2	537.6	530.1	5.224
	fun6	602.8	606	604.4	0.841
	fun7	743.6	772.7	758.9	7.36
	fun8	822.8	845.3	834.1	5.065

3. Conclusion

3.1 Conclusion about PSO Algorithm

Firstly, from figure 4 and table 1, we can speculate that when the difference gets larger between c1 and c2, the value of standard deviation becomes smaller for functions like function 4. Besides, the value of standard deviation when cc=[1.5,2] is among the other two configuration of cc, and for function 4 to function 8, it is possibly that cc=[1.5,2] can make the standard deviation more stable. However, this hypothesis is not suitable for each function. Then, It can be seen from the figure 5 to figure 8 that, with the decrease of iwt range, the convergence speed gets quicker. Figure 5 and 6 are for function 4, Figure 7 and 8 are for function 5. Besides, in PSO algorithm, convergence comes earlier more likely.

3.2 Conclusion about CLPSO Algorithm

Firstly, from figure 9 and table 2, we can speculate that when cc=[2,2], the performance of standard deviation is best. Secondly, It can be seen from the figure 10 to figure 13 that the CLPSO algorithm solve the premature convergence problem in PSO algorithm. Figure 10 and 1 are for function 4, Figure 12 and 13 are for function 5. Thirdly, from figure 14 and table 3, for function 5, 7 and 8, it is more difficult to make values like others and to reach relatively small values.

3.3 Comparison of PSO and CLPSO

Firstly, from figures and tables, PSO is better at finding the minimum for these specific functions, and the standard deviation. However, the CLPSO has some advantages. It deletes the global optimal item to avoid Oscillation. Besides, pbest is got by particle's own optimality combining with random parameters, which makes the whole direction for better values. Thirdly, rate of convergence of two algorithms are both influenced by iwt parameter. Then, the convergence time of them is different as well, and the PSO is more likely to get convergence.