Michihgan Technological University

Computational Intelligence
CS 5821

Fall 2019

CI FINAL EXAM

Particle Swarm Optimization

Shashank Pathrudkar

December 19, 2019

Contents

1 Abstract

This work involves use of Particle Swarm Optimization(PSO) to find minima of 5 functions: Dejong, Rastrigin, Ackley, Rosenbrock, Branins. Best parameters for PSO are determined by grid search method. Study of time required for PSO to converge is also presented. Extension of PSO (modified PSO) is used to improve the results. Further, a comparison of PSO and Genetic Algorithm(GA) is presented.

2 Introduction

Particle Swarm Optimization(PSO) is an optimization method which makes use of numerous candidates in search space to find optima. PSO iteratively updates position of candidates by evaluating candidate fitness and velocity. PSO involves tuning of parameters which decide weightage of social and cognitive velocity in finding final velocity.

3 Problem 1

Assume that you want to optimize some fitness function $f(\bar{x}), \bar{x} \in R^d$, where the values of $\bar{x}$ are the in range $[-xrange/2 : xrange/2]$. Develop a Matlab or Python function for particle swarm optimization with the following syntax:

$$function[fxbest, xbest] = myPSO(funchandle, xrange, dim) \tag{1}$$

where $fxbest$ is the best fitness function value, $xbest$ is a column vector that is the associated best value of the input vector $\bar{x}$, $funchandle$ is the handle of the function being optimized (eg. $funchandle = @dejong$), $xrange$ is a scalar, which is the range of the input values, and dim is the dimensionality of the fitness function. Note that your internal code should automatically (or deterministically) define the termination criteria or other necessary aspects. No other inputs should be passed to your code.
You will use the three benchmark functions (Dejong, Rastrigin, Ackley) that you used for the GA algorithm as well as two others (Rosenbrock, Branins).

Note: In this report, MSE is mean squared error; $xrange$ is 30; and number of dimension is 5 if not mentioned specifically.

3.1 Pseudo-code

(5 points) Give a psuedo-code for your algorithm. Describe the major parameters of your PSO. Someone should be able to reproduce your algorithm perfectly from your description. Note that code should not be used to describe your algorithm. Equations, text, and psuedo-code are acceptable.

Algorithm 1: Particle Swarm Optimization to Find Minima of a function

---

function myPSO
Input : $funchandle, xrange, dimensions$
Output: $fxbest, xbest$
Initializations: Generate n random samples (x_{id}) between [-xrange/2 : xrange/2]
while *iter<2000 and change in* f_{gbest} *over 500 iterations*$>10^{-6}$ **do**
    Calculate fitness for all samples $\mathbf{f(x_{id})}$
    Find best fitness (minimum) (f_{gbest})
    Calculate change in f_{gbest} over 500 iterations
    Store location of sample with best fitness (p_{gd})
    if *sample fitness* $\mathbf{f(x_{id})}$ < *previous best sample fitness* $\mathbf{f(p_{id})}$ **then**
        update sample best location (p_{id});
    end
    $V_{id} = \omega * V_{id} + c_1 * rand() * (p_{id} - x_{id}) + c_2 * rand() * (p_{gd} - x_{id})$
    $x_{id} = x_{id} + V_{id}$
end

3.2 Finding optimum value of social and cognitive parameters

(10 points) Run your PSO for each of the five benchmark functions (use 3 or more dimensions for at least three of the benchmarks) for a variety of parameter values. For the cognitive and social parameters, do a grid search over parameter values [0; 0.2; 0.4; . . . . 2] and fill in the following table for each benchmark, where each cell includes the mean fitness squared error $(f(\bar{x}) - f_{best})^2$ over 10 runs, where $f(\bar{x})$ is the fitness value your PSO found and f_{best} is the known best fitness value for that benchmark. If you have additional parameters in your PSO, make sure you describe how you chose that parameter value, e.g., the momentum constant or the constriction factor.

In this section we find out optimal value of social and cognitive paramater using grid search method. Apart from social(c2) and cognitive paramter(c1), PSO contains momentum constant and constriction factor. Constriction factor is set to 1 and momentum constant is set to 0.1 before proceeding to tuning of c1 and c2.

Following tables show value of Mean squared error(MSE) over 10 runs for 5 benchmark functions. MSE is calculated as follows

$$MSE = \frac{1}{n}(f(\bar{x}) - f_{best})^2 \tag{2}$$

where, n is number of runs.

Dejong	**Cognitive Parameter**										
Social Parameter	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
0	512.88	419.78	454.38	471.00	370.18	329.25	280.86	408.54	336.83	196.02	445.01
0.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 1: MSE with varying social and cognitive parameters for Dejong function with $xrange = 30$ and $n_{dimensions} = 5$

Rastrigin	**Cognitive Parameter**										
Social Parameter	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
0	2748.73	3681.46	2477.15	2590.88	3130.76	2470.69	3088.03	3536.07	2500.55	2937.20	2037.95
0.2	4.59	3.57	3.86	2.58	3.49	2.57	3.17	1.29	1.53	1.08	1.26
0.4	4.91	4.57	2.77	2.88	3.27	4.16	2.48	1.39	0.90	0.20	0.20
0.6	5.69	4.87	3.09	3.67	4.06	1.29	1.09	0.72	1.20	0.40	0.40
0.8	3.57	6.64	2.38	6.24	3.66	2.67	1.09	2.87	0.59	0.79	0.10
1	4.55	5.35	2.57	4.65	3.17	4.45	1.29	2.87	1.39	0.59	0.30
1.2	9.31	5.84	6.34	13.56	7.03	5.44	2.38	2.38	1.39	1.58	0.20
1.4	5.74	5.05	6.63	6.53	6.34	3.27	3.17	0.30	0.69	0.99	0.40
1.6	4.26	7.82	7.23	4.26	2.77	3.96	4.26	0.69	1.29	0.99	1.09
1.8	5.94	5.54	4.16	2.47	1.09	2.77	1.48	0.69	0.49	0.30	0.30
2	2.87	5.74	2.97	4.55	2.67	4.16	0.30	0.40	0.10	0.20	0.30

Table 2: MSE with varying social and cognitive parameters for Rastrigin function with $xrange = 30$ and $n_{dimensions} = 5$

Ackley	**Cognitive Parameter**										
Social Parameter	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
0	73.92	68.46	66.30	72.40	50.48	58.95	59.52	55.53	66.27	51.98	59.92
0.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 3: MSE with varying social and cognitive parameters for Ackley function with $xrange = 30$ and $n_{dimensions} = 5$

Rosenbrock	**Cognitive Parameter**										
Social Parameter	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
0	3.33E+07	6.72E+07	4.77E+07	5.46E+07	4.01E+07	5.02E+07	3.39E+07	1.43E+07	2.91E+07	4.19E+07	4.45E+07
0.2	1.22	8.88	6.06	6.22	4.02	3.63	5.81	2.64	3.62	2.60	0.87
0.4	2.38	7.95	6.31	4.18	4.42	7.74	2.04	10.24	2.56	1.79	3.17
0.6	5.90	9.26	4.22	5.49	4.38	5.46	8.51	4.77	2.55	3.05	4.34
0.8	9.07	7.92	6.90	4.84	5.21	16.44	6.60	4.42	1.63	2.38	4.70
1	5.91	8.00	10.10	5.66	5.16	6.18	7.93	3.97	2.66	5.96	15.21
1.2	7.56	19.51	16.58	17.28	12.57	15.06	8.08	12.53	19.55	12.25	13.63
1.4	13.15	31.56	7.57	14.40	3.70	6.26	11.18	2.51	4.03	23.06	2.06
1.6	39.72	14.16	1.64	8.37	6.88	24.33	6.14	2.65	3.26	4.88	1.72
1.8	6.06	14.81	4.51	7.58	0.89	7.72	5.92	0.00	4.64	0.00	3.09
2	3.72	7.10	0.09	0.36	0.06	5.21	0.00	0.00	1.55	0.00	0.00

Table 4: MSE with varying social and cognitive parameters for Rosenbrock function with $xrange = 30$ and $n_{dimensions} = 5$

Branins	**Cognitive Parameter**										
Social Parameter	**0**	**0.2**	**0.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**1.6**	**1.8**	**2**
0	0.03	0.02	0.02	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 5: MSE with varying social and cognitive parameters for Branins function with $xrange = 30$ and $n_{dimensions} = 2$

3.3 Finding optimum value of momentum constant and number of samples

(10 points) Using the best overall parameter values for the cognitive and social parameters from part (b), explore the performance of your PSO with the five benchmarks by varying the other parameters in your code (choose two parameters). Do a grid search over possible values of these two parameters—e.g., momentum constant (starting at 0) and constriction factor—and make similar tables as you did for part (b).

Here we vary momentum constant and number of samples

Dejong	**Momentum Constant**										
Samples	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.88E+30	2.35E+63	1.30E+72
100	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.34E+22	1.97E+53	6.28E+88
250	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	5.37E+10	9.48E+46	1.11E+76
500	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.70E+05	2.07E+40	3.18E+71
1000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	5.31E+35	3.38E+67
2000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.34E+31	5.53E+63

Table 6: MSE with varying momentum constant and number of samples for Dejong function with $xrange = 30$ and $n_{dimensions} = 5$

Rastrigin	**Momentum Constant**										
Samples	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
50	31.08	12.47	24.65	10.49	5.15	4.26	1228.30	4.06E+04	1.24E+31	3.42E+65	8.01E+91
100	14.35	9.31	12.28	3.17	4.65	2.47	1621.61	3071.54	2.34E+24	1.82E+57	1.97E+80
250	2.87	6.53	2.67	3.37	2.18	1.39	553.53	1878.39	7.04E+11	2.23E+45	6.98E+75
500	1.09	3.66	1.48	2.08	0.89	0.69	431.10	953.62	2.87E+06	7.98E+40	2.59E+71
1000	0.20	0.30	0.30	0.20	0.20	0.30	51.33	614.04	8883.85	3.46E+35	7.64E+66
2000	0.00	0.00	0.20	0.20	0.10	0.10	0.00	501.99	2666.53	1.14E+31	1.92E+63

Table 7: MSE with varying momentum constant and number of samples for Rastrigin function with $xrange = 30$ and $n_{dimensions} = 5$

Ackley	**Momentum Constant**										
Samples	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
50	1.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	424.24	429.71	430.26
100	0.27	0.00	0.00	0.00	0.00	0.00	0.00	0.00	423.65	427.15	423.46
250	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	180.43	419.72	419.49
500	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	23.44	416.07	411.32
1000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.43	411.79	413.03
2000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	407.97	408.98

Table 8: MSE with varying momentum constant and number of samples for Rastrigin function with $xrange = 30$ and $n_{dimensions} = 5$

Rosenbrock	**Momentum Constant**										
Samples	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
50	8.20	49.39	1615.85	5.18	3.09	0.00	0.00	4.08E+45	2.41E+72	1.59E+129	4.45E+175
100	6.72	4.62	5.24	0.00	4.64	1.55	0.00	1.12E+30	1.37E+76	7.92E+114	1.76E+171
250	3.09	4.64	1.55	3.09	4.64	0.00	0.00	0.00	4.64E+64	4.45E+110	6.38E+154
500	0.00	3.09	0.00	0.00	0.00	0.00	0.00	0.00	2.33E+71	1.38E+97	5.86E+145
1000	1.55	1.55	1.55	0.00	0.00	0.00	0.00	0.00	5.33E+62	2.40E+95	3.81E+136
2000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	86.41	4.01E+94	1.03E+129

Table 9: MSE with varying momentum constant and number of samples for Rosenbrock function with $xrange = 30$ and $n_{dimensions} = 5$

Branins	**Momentum Constant**										
Samples	**0**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1**
50	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.75E+39	1.91E+63	4.97E+100	1.16E+141
100	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.06E+11	1.42E+67	5.51E+107	1.75E+133
250	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.17E+27	5.88E+74	1.34E+101	3.02E+125
500	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.24E+41	5.01E+71	1.41E+100	8.46E+129
1000	0.00	0.00	0.00	0.00	0.00	0.00	834.65	2.05E+36	8.17E+68	2.43E+85	1.30E+135
2000	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.19E+30	1.08E+69	1.04E+100	6.35E+123

Table 10: MSE with varying momentum constant and number of samples for Branins function with $xrange = 30$ and $n_{dimensions} = 2$

3.4 Reporting best parameter values

(10 points) Using the results from parts (b) and (c) and present the squared error for each of the five benchmarks for the best overall parameter choice. That is, if you had to choose one set of parameter values, what would you choose. Report the parameter values and the five squared errors. Note, you can just copy the squared error values from the experiments you did in (b) and (c). Comment on these parameter values. Are they the best for all the benchmarks? If they aren't the best for all the benchmarks, is there something different about the benchmarks that the parameters aren't optimal for?

Results presented above clearly show that the best parameter values are:
Social paramter = 2
Cognitive Paramter = 1.8 or 2
Momentum constant = 0 to 0.6
Number of samples = 2000

Function	**Squared Error**
Dejong	0.0
Rastrigin	0.0
Ackley	0.0
Rosenbrock	0.0
Branins	0.0

Table 11: Mean squared error for benchmark functions

Table 11 shows Squared error for given 5 benchmark functions. $xrange$ was set to 30 and number of dimensions were taken as 5. Best parameter obtained from previous analysis were used during optimization.

3.5 Evaluating functions for 10 different sets of parameter values

(10 points) Choose 10 different sets of parameters that work well for the five benchmarks. Make sure the parameter choices are somewhat diverse. Run the PSO for these 10 different sets of parameters and time your code. Report the results. Comment on the results. Is there something you can conclude from this experiment?

Nomenclature used in following table is:

- C1: Cognitive paramter
- C2: Social parameter
- w: Momentum Constant
- n: Number of samples
- a: Function Dejong
- b: Function Rastrigin
- c: Function Ackley
- d: Function Rosenbrock
- e: Function Branins

C1	**C2**	**w**	**n**	**Mean Squared Error**					**Time (seconds)**					
				a	**b**	**c**	**d**	**e**	**a**	**b**	**c**	**d**	**e**	Σ
1.8	1.6	0	1500	0.00	0.99	0.00	0.21	0.00	0.17	0.22	0.21	0.29	0.10	1.00
2	2	0.2	2000	0.00	0.99	0.00	0.00	0.00	0.13	0.21	0.20	0.27	0.07	0.88
1.8	1.8	0.3	1000	0.00	0.00	0.00	0.00	0.00	0.09	0.11	0.12	0.11	0.05	0.48
2	1.6	0.4	1500	0.00	0.00	0.00	0.00	0.00	0.09	0.13	0.15	0.13	**0.07**	0.57
1.6	2	0.5	2000	0.00	0.00	0.00	0.00	0.00	**0.13**	**0.18**	0.20	0.16	0.06	0.73
2	2	0	2000	0.00	0.00	0.00	0.00	0.00	0.11	0.15	0.16	0.25	0.05	0.73
2	1.8	0.1	1500	0.00	0.00	0.00	0.00	0.00	0.08	0.16	**0.12**	**0.31**	0.04	0.72
1.6	2	0.2	2000	0.00	0.99	0.00	0.00	0.00	0.13	0.30	0.17	0.21	0.05	0.86
1.8	2	0.3	1000	0.00	0.00	0.00	0.00	0.00	**0.06**	**0.09**	**0.21**	**0.10**	**0.03**	0.49
2	1.8	0.4	1500	0.00	0.00	0.00	0.00	0.00	0.08	0.14	0.18	0.12	0.05	0.58

Table 12: MSE and time required for different set of parameters for 5 different functions

In table 12 we can see that for certain set of parameters we get 0 MSE. By considering time required to converge we have a slowest set of parameters and fastest set of parameters. It is obvious that more time would be required for higher number of samples which can be seen in table 12. Fastest values are highlighted in yellow and slowest values are highlighted in blue. Parameter values for which we get zero MSE for all functions are only considered.

3.6 Plots of Squared Error vs Iteration

(5 points) Choose the slowest and fastest parameter choices from part (e) and plot the squared error of the PSO versus iteration. Note you can do this by plotting the

Parameter	Fastest Set	Slowest Set
c1	1.8	1.6
c2	2	2
omega	0.3	0.5
n	1000	2000

Table 13: Table showing selected fastest and slowest paramter set

squared error of the global best vs. iteration. Comment on the results.

Figures presented in this section show plot of Squared errors vs Iteration for all 5 benchmark functions. Fastest and slowest performing parameter set which can be obtained from table 12 are used in the optimization before plotting squared error vs iteration.

One of the stopping criteria used is based on change is f_{gbest} over 500 iterations, therefore everytime we require more than 500 iterations to converge. But from plots we can see that for all benchmark functions convergence is obtained within 100 iterations. That's why the plots are shown till 100 iterations.

From the plots we can see that slowest parameter set take few more iterations to make Squared error zero. Both the parameter sets make squared error 0 as they contain well suited individual parameter values.

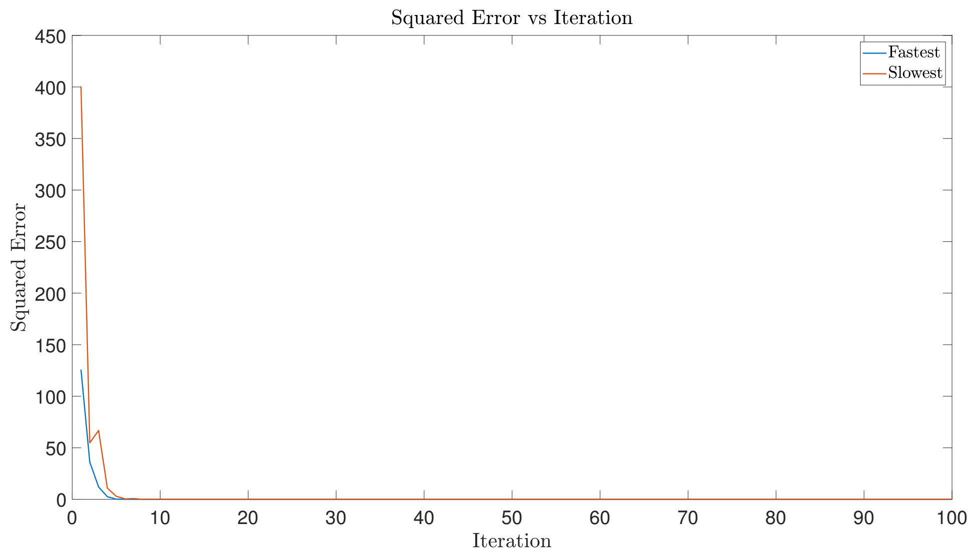

Figure 1: Plot of Sqaured Error vs Number of iteration for Dejong Function using slowest and fastest performing parameters

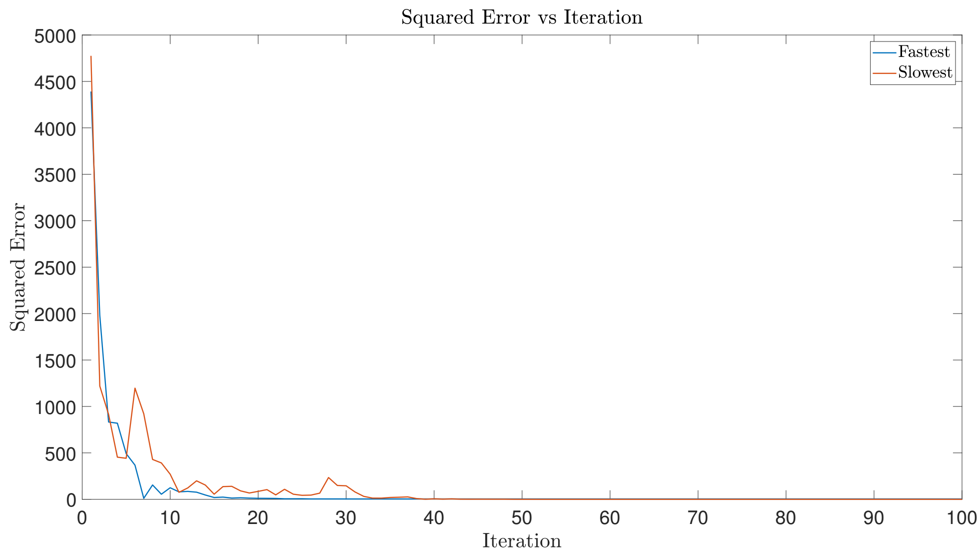

Figure 2: Plot of Sqaured Error vs Number of iteration for Rastrigin Function using slowest and fastest performing parameters

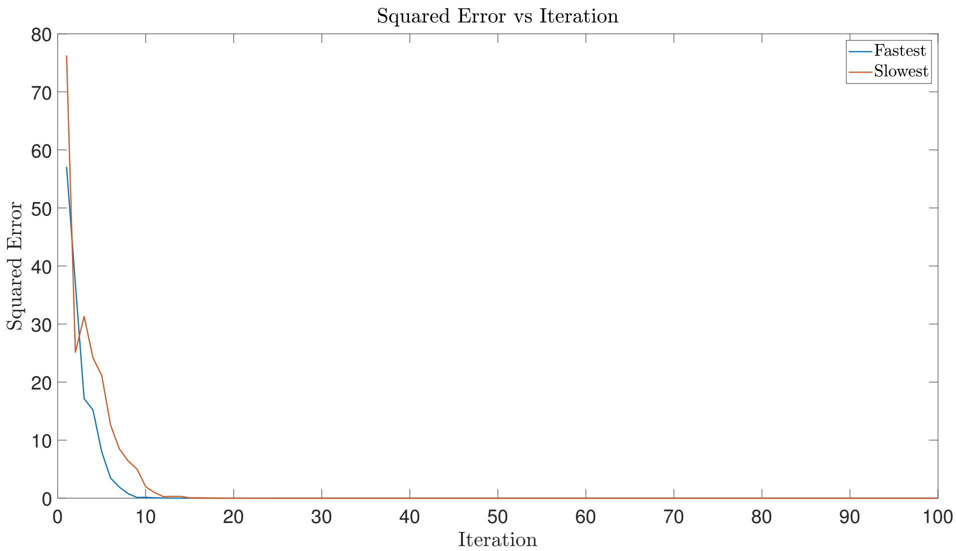

Figure 3: Plot of Sqaured Error vs Number of iteration for Ackley Function using slowest and fastest performing parameters

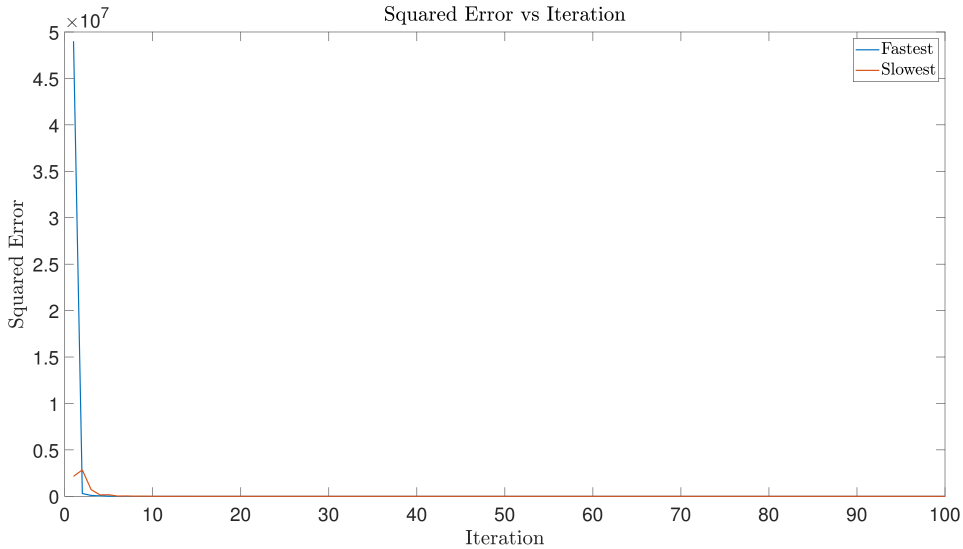

Figure 4: Plot of Sqaured Error vs Number of iteration for Rosenbrock Function using slowest and fastest performing parameters

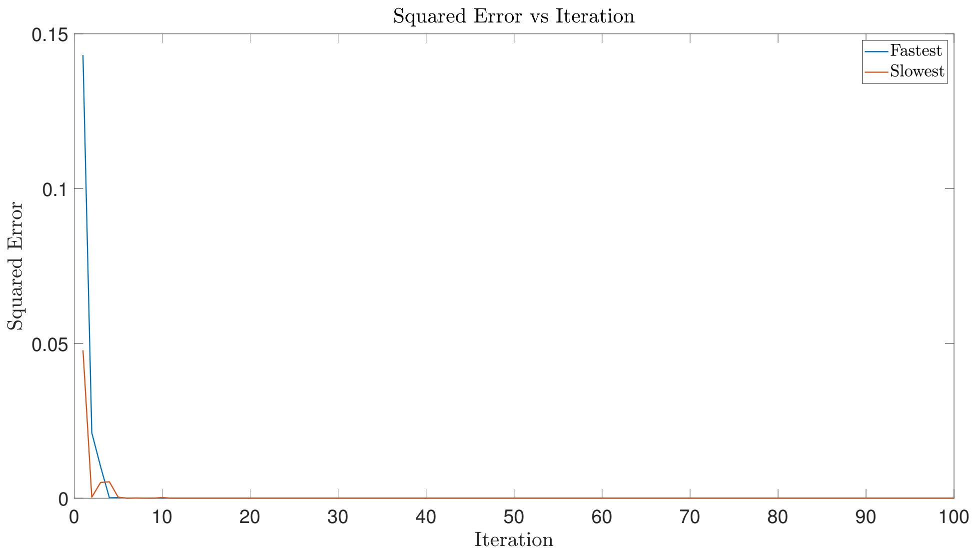

Figure 5: Plot of Sqaured Error vs Number of iteration for Branins Function using slowest and fastest performing parameters

3.7 Modified Particle Swarm Optimization (MPSO)

(20 points) Find a research paper on a modification of the PSO and implement this modification. A good place to look is the IEEE Trans. Evolutionary Computation. Describe this modification and run a set of experiments that shows how this compares to your previous results, addressing both squared error performance and algorithm speed. Make sure you run enough experiments to support your conclusion. You don't need to completely reproduce all of the results from parts (b)-(f), but if you write your code in a smart way, it certainly wouldn't be hard to plug in your modification to a script that just runs everything for you.

Dongping Tian and Zhongzhi Shi have listed modifications of particle swarm optimization in their paper [1]. The paper suggest modification of momentum constant linearly with iterations. The formula for momentum constant as given in the paper is

$$\omega(t) = \frac{t_{max} - t}{t_{max}}(\omega_{max} - \omega_{min}) + \omega_{min} \tag{3}$$

where t_{max} = maximum number of iterations, t = iteration number

As mentioned in section 3.4 we have a range of momentum constant values where we obtained good results. From that range we set ω_{max} as 0.6 and ω_{min} as 0. Though maximum number of iterations is set as 2000 here I have used t_{max} as 500.

This modification of PSO ensures that the particles don't rapidly change position at every iteration. We want the particles to slow down as the the optimization progresses and particles come closer to the global minima. Therefore at 1-st iteration we have ω as ω_{max} and ω linearly goes on decreasing. This ensures that the velocity of particles reduces as optimization progresses and particles don't fly past the global minima.

Results obtained in section 3.3 show that simple particle swarm optimization works best for Dejong, Ackley, Branins but it performs not so good for Rosenbrock and Rastrigin. Therefore, Rosenbrock and Rastrigin were selected as comparing ground for Modified PSO and Simple PSO. Optimization is done over a $xrange = 30$ and 5 dimensions. Mean squared error over 10 runs for function Rastrigin and Rosenbrock are presented in tables 14 and 15 respectively.

Rastrigin	Cognitive Parameter									
Social Parameter	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
0	28477.35	12157.22	4998.78	3477.67	2309.56	1720.39	2193.58	1575.46	1996.59	1686.22	2444.37
0.2	2.57	2.97	2.57	2.38	3.66	0.69	0.20	0.49	0.00	0.10	0.00
0.4	12.57	3.66	3.66	6.73	1.88	1.68	1.88	0.99	0.49	0.00	0.10
0.6	13.07	5.64	4.26	2.57	3.96	1.88	1.29	0.99	0.30	0.20	0.10
0.8	7.42	6.63	5.25	2.28	2.08	1.19	2.28	1.88	0.30	0.10	0.10
1	7.52	2.38	4.55	5.64	1.88	1.68	0.69	0.49	0.10	0.20	0.00
1.2	2.87	2.77	1.88	1.98	1.29	0.20	0.49	0.40	0.10	0.00	0.00
1.4	1.88	5.54	1.29	0.79	0.00	0.40	0.20	0.10	0.00	0.00	0.00
1.6	1.19	2.08	0.79	0.49	0.59	0.30	0.00	0.10	0.00	0.00	0.00
1.8	0.40	0.49	0.40	0.69	0.59	0.10	0.10	0.00	0.00	0.00	0.00
2	1.19	1.09	0.79	0.59	0.40	0.10	0.10	0.10	0.40	0.10	0.00

Table 14: MSE with varying social and cognitive parameters for Rastrigin function with $xrange = 30$ and $n_{dimensions} = 5$ (Using MPSO)

Rosenbrock	Cognitive Parameter									
Social Parameter	0	0.2	0.4	0.6	0.8	1	1.2	1.4	1.6	1.8	2
0	1.46E+10	9.76E+08	6.83E+07	2.99E+07	5.16E+06	1.96E+07	1.10E+07	2.26E+06	1.08E+07	2.57E+06	2.04E+06
0.2	15.78	8.58	2.33	2.43	3.89	4.04	6.24	7.98	0.73	0.78	0.61
0.4	0.93	12.72	31.59	3.62	3.25	0.07	0.07	1.58	0.01	0.00	0.00
0.6	55.28	13.44	21.00	0.01	3.11	1.55	0.00	0.00	0.00	0.00	0.00
0.8	220.29	0.01	23.09	2.13	2.40	4.73	0.00	8.38	0.00	0.00	0.00
1	25.92	7.78	13.90	0.00	1.57	0.00	0.00	0.00	0.00	0.00	0.00
1.2	0.00	43.96	1.41	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.4	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.6	7.68	1.59	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
1.8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 15: MSE with varying social and cognitive parameters for Rosenbrock function with $xrange = 30$ and $n_{dimensions} = 5$ (Using MPSO)

When we compare table 2 and table 14 we can clearly see that the values of Means Squared Error have reduced. Still many values are non zero but we can clearly see that MSE values obtained using MPSO are closer to zero than MSE values obtained using PSO.

Similar comparison can be done between table 4 and table 15. We can clearly see that the MSE values have decreased significantly when we used MPSO.

Now we compare time required for PSO and MPSO

C1	C2	w	n	Mean Squared Error				Time (seconds)			
				Rastrigin		Rosenbrock		Rastrigin		Rosenbrock	
				PSO	MPSO	PSO	MPSO	PSO	MPSO	PSO	MPSO
1.8	1.6	0	1500	0.99	0.00	0.21	0.00	0.22	0.40	0.29	0.66
2	2	0.2	2000	0.99	0.00	0.00	0.00	0.21	0.33	0.27	0.43
1.8	1.8	0.3	1000	0.00	0.00	0.00	0.00	0.11	0.19	0.11	0.26
2	1.6	0.4	1500	0.00	0.00	0.00	0.00	0.13	0.21	0.13	0.26
1.6	2	0.5	2000	0.00	0.00	0.00	0.00	0.18	0.28	0.16	0.37
2	2	0	2000	0.00	0.00	0.00	0.00	0.15	0.26	0.25	0.40
2	1.8	0.1	1500	0.00	0.00	0.00	0.00	0.16	0.21	0.31	0.25
1.6	2	0.2	2000	0.99	0.00	0.00	0.00	0.30	0.29	0.21	0.37
1.8	2	0.3	1000	0.00	0.00	0.00	0.00	0.09	0.17	0.10	0.18
2	1.8	0.4	1500	0.00	0.00	0.00	0.00	0.14	0.21	0.12	0.28

Table 16: Comparison of time required for PSO and MPSO for Rastrigin and Rosenbrock

As we can see in table 16, time required for PSO is around 50% less than MPSO, but MPSO performs better than PSO when it comes to squared error.

3.8 PSO vs GA

(10 points) Compare the results of your PSO with your GA results. Which is better and in what way? Comment on the overall quality of the solutions, the speed of the algorithm, and also the complexity of the code.

In this section we compare performance of PSO and Genetic Algorithm. Again Rastrigin and Rosenbrock are selected as comparing grounds. Comparison is done on the basis of Squared error and time required to reach optima.

From table 17 we can see that for Rastrigin function GA gives better results than PSO when we compare squared error. But, GA requires a lot more time than PSO. GA algorithm has a very complex code that is why it takes a lot of time.

Table 18 shows squared error and time required for Rosenbrock function usinf PSO and GA. In case of Rosenbrock also, time required for GA is more than that of PSO. But, squared error obtained for PSO is less than that of GA.

From these results we can say that use of PSO and GA depends on the application we are using it for. Both the algorithms offer their unique advantages and disadvantages. When considered time, PSO is a clear winner.

Number of dimensions	Squared Error		Time(Seconds)	
	PSO	GA	PSO	GA
2	0.00	0.00	0.13	12.39
3	0.00	0.00	0.16	18.56
4	0.00	0.00	0.23	24.18
5	0.00	0.00	0.26	30.29
6	0.00	0.00	0.29	40.03
7	0.99	0.00	0.38	54.89
8	3.96	0.00	0.43	78.87
9	3.96	0.00	0.52	82.11
10	8.91	0.00	0.65	108.61

Table 17: Squared error and time required for optimization of Rastrigin function using PSO and GA

Number of dimensions	Squared Error		Time(Seconds)	
	GA	PSO	GA	GA
2	0.00	0.00	0.12	13.31
3	0.00	0.00	0.14	21.79
4	0.00	0.00	0.20	36.84
5	0.00	0.44	0.19	48.85
6	0.00	2.61	0.25	90.28
7	0.00	10.57	0.31	155.72
8	0.00	24.14	0.34	239.43
9	0.00	33.01	0.41	665.36
10	0.00	-	0.54	-

Table 18: Squared error and time required for optimization of Rosenbrock function using PSO and GA

References

[1] Tian, Dongping and Shi, Zhongzhi. MPSO: Modified particle swarm optimization and its applications. *Swarm and Evolutionary Computation*, 41(August 2017):49–68, 2018.