Particle Swarm Optimization

Introduction

The goal of this project was to investigate how different parameters affect particle swarm optimization algorithms trying to solve two different problems. The first problem was:

$$Q(p_x, p_y) = 100 \cdot \left(1 - \frac{pdist}{mdist}\right)$$

This includes only one local/global maximum at (20, 7). The second problem had two maxima, one local at (-20, -7), and the global maximum at (20, 7). This problem was as follows:

$$Q(p_x, p_y) = 9 \cdot \max(0, 10 - pdist^2) + 10 \cdot \left(1 - \frac{pdist}{mdist}\right) + 70 \cdot \left(1 - \frac{ndist}{mdist}\right)$$

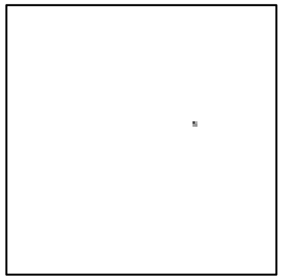
A total of ten different tests (five runs each) were conducted for each of these problems, utilizing different values for the parameters: number of epochs, number of particles, inertia, the cognition parameter, and the social parameter. The number of epochs determines how many times each particle updates. The particle number dictates how many particles are randomly placed in the space at the beginning of the run. Inertia controls how much weight a particle's previous velocity has on its current velocity being calculated. The cognition and social parameters determine how a particle converges on the "best" point in the space. The cognition parameter affects how quickly a particle will converge to the best cell *it* has found in the space, whereas a higher social parameter affects how quickly it converges to the best cell *any* particle has found in the space.

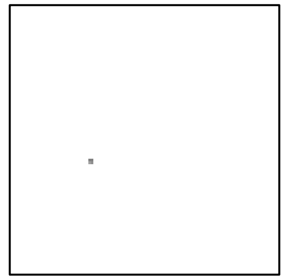
To perform these calculations, a Python script was written that created a .pgm image file after each update. These .pgm files were then combined into a .gif displaying the progress of the space over each epoch. These .gif files are included in the submission, but obviously can not be displayed in the report.

Analysis

For each test, the parameters as well as the (most common of 5 runs) final epoch's space for problem 1, then problem 2, will be shown.

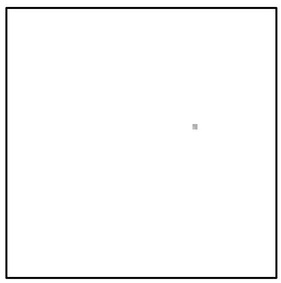
Test 1	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.75
Cognition Parameter	2.00
Social Parameter	2.00

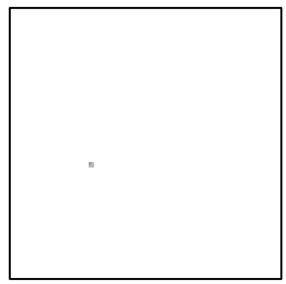

Problem 1


Problem 2

Test 1 was used as a control. The parameters were all set to fairly default values so the behavior of this test can be used as a baseline to compare other tests to. The global maximum was found for problem 1, with an X-error of 0.00437 and Y-error of 0.0436. However, only 1 of 5 runs found the global maximum for problem 2. This could be due to a lack of particles.

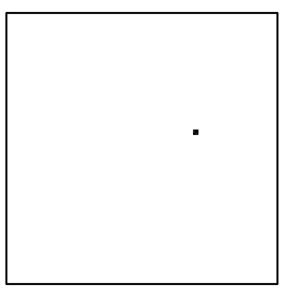
Test 2	**Value**
Number of Epochs	200
Number of Particles	10
Inertia	0.75
Cognition Parameter	2.00
Social Parameter	2.00

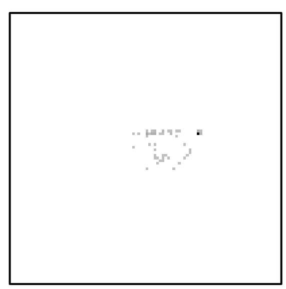

Problem 1


Problem 2

In Test 2, the number of particles was decreased to 10. This had very little effect on the output. It is hard to discern in the images, but a higher concentration of particles will result in darker spot in a plot, so the final images' spots are slightly lighter than in Test 1, since there are fewer total particles occupying the space. Error for Test 2 was lower in most cases than Test 1, and the same situation happened as in Test 1, where Problem 1 found the global maximum but Problem 2 only had one run to find the global max.

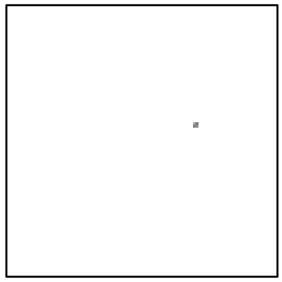
Test 3	**Value**
Number of Epochs	200
Number of Particles	100
Inertia	0.75
Cognition Parameter	2.00
Social Parameter	2.00

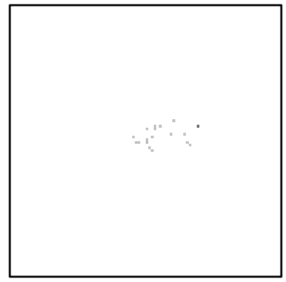

Problem 1


Problem 2

Test 3 shows that increasing the number of epochs does make it more likely for Problem 2 to find the global maximum. It is fairly easy to see though that Problem 2's algorithm for finding the global maximum has flaws. The particle count should not have to be increased very high for the algorithm to work properly. Error on average was greater than 8 and 3 for X and Y axes, respectively. As an aside, it is easier to tell in Problem 1's space that a larger concentration of particles in one cell causes said cell to get darker.

Test 4	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.20
Cognition Parameter	2.00
Social Parameter	2.00

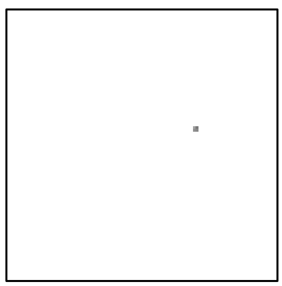

Problem 1


Problem 2

In Tests 4 and 5, inertia is being modified. This was the first test in which a problem converged well before epoch 200. Problem 1 on average had converged within a threshold of 0.01 error by epoch 75. This makes sense, with a lower inertia weight, particles are capable of making sharper “turns” and can be more precise in their movement. 80% of the runs for Problem 2 found the global maximum, but all reached epoch 200 with large amounts of error. This can probably still be attributed to the issues with the algorithm, rather than lowering the inertia.

Even though lowering the inertia has a direct positive impact on convergence, inertia is present in nature and as such cannot be completely ignored in particle swarm optimization algorithms.

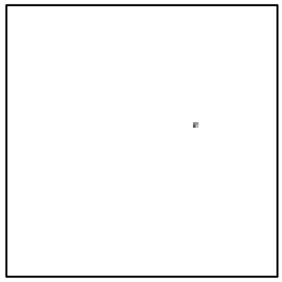
Test 5	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	1.00
Cognition Parameter	2.00
Social Parameter	2.00


Problem 1

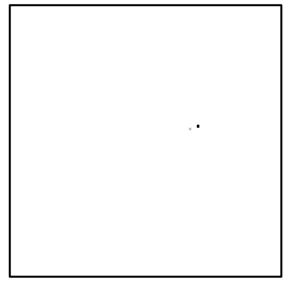

Problem 2

Although the images look fairly similar to previous tests, both problems resulted in a larger error than normal. When viewing the .gif files, you can see that even after a particle has reached the area around a maximum, it constantly overshoots the actual maximum value when trying to converge. This is because with such a high inertia, particles have a hard time stopping in place by “turning” around.

Test 6	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.75
Cognition Parameter	0.25
Social Parameter	2.00



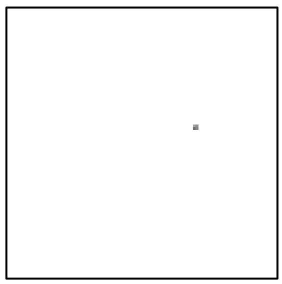
Problem 1



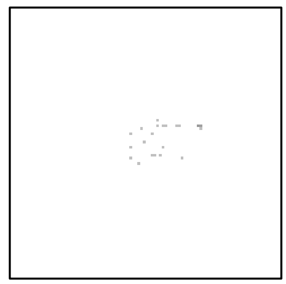
Problem 2

Tests 6 and 7 analyze the cognition parameter. Problem 1 converges much faster with a lower cognition, with the last epoch averaging at 120. Problem 2 converges with a lower error than normal to the global maximum, but on 2/5ths of the runs, it converges to the local maximum. This local convergence happens much quicker, within 120 epochs, whereas the convergence to the global maximum iterates through all 200 epochs without finishing.

Test 7	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.75
Cognition Parameter	3.75
Social Parameter	2.00



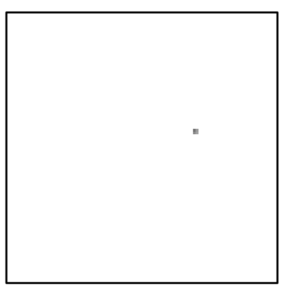
Problem 1



Problem 2

Increasing cognition causes much slower convergence with larger error. This could be due to a few reasons. One of the main ones is that with a high cognition and semi-high social parameter, particles may become “confused” as to which parameter to follow. Some particles may be broadcasting that the local maximum is the point to converge to, therefore it requires a larger number of particles at the global maximum to ensure full convergence.

Test 8	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.75
Cognition Parameter	2.00
Social Parameter	0.25

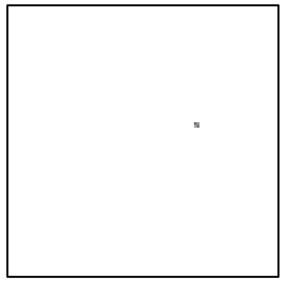


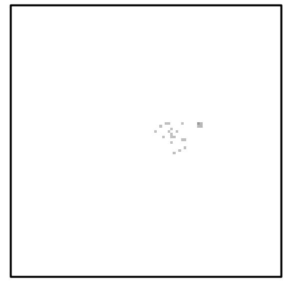
Problem 1

Problem 2

Tests 8 and 9 modify the social parameter. Even though both tests 7 and 8 have a cognition parameter that is 1.75 higher than their social parameter, their results differ. Test 8’s Problem 1 converges within the error threshold on average after 120 epochs. Problem 2 in Test 8 performs worse, however, with only 2/5 runs finding the global maximum in comparison to Test 7’s 3/5. From this we can assume that a higher sociality is required for a better chance at converging to the global maximum.

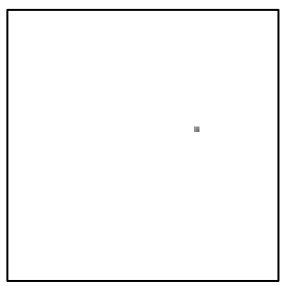
Test 9	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.75
Cognition Parameter	2.00
Social Parameter	3.75

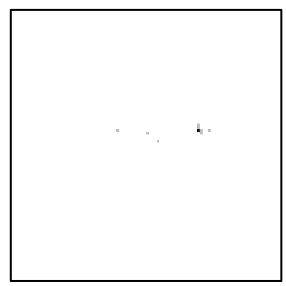

Problem 1


Problem 2

In Test 9, the particles in Problem 1 took longer to converge than in Test 8, however the error spread was much less in Problem 2 than in Test 8. A higher social parameter may cause confusion in a particle. Since particles are relying so heavily on finding the current global best position, they are less likely to be confident in their own personal best they've found. This decreases the likelihood of particles exploring other areas, and therefore decreases the chance that the global maximum is found.

Test 10	**Value**
Number of Epochs	200
Number of Particles	30
Inertia	0.75
Cognition Parameter	2.00
Social Parameter	2.00


Problem 1


Problem 2

This final test maintained the default parameters, but increased the maximum velocity from 1 to 3. This changed very little as far as rate of convergence goes, but the particles in Problem 2 did have a higher probability of finding the global maximum with smaller error. Since the particles can move faster, they can explore more ground quicker and therefore have a higher chance of multiple stumbling across the global maximum, which will attract other particles.