Computational Intelligence Solution for Market Pricing

Edward Evans

November 2018

1 Introduction

This paper will try to disprove the hypothesis:

> "The Particle Swarm Optimization (PSO) algorithm, on average, does not perform better than the Artificial Immune System (AIS) algorithm when given the market pricing problem in terms of revenue."

As a result of disproving that statement, this paper will try to prove:

> "The Particle Swarm Optimization (PSO) algorithm, on average, does perform better than the Artificial Immune System (AIS) algorithm when given the market pricing problem in terms of revenue."

To do so each algorithm will be implemented and run multiple times on a selection of different seeds. The revenues collected from each execution will then be compared against the opposing algorithm's results. These results will determine if the above hypotheses are true.

2 Artificial Immune System

2.1 Method

The Artificial Immune System (AIS) algorithm can solve hard problems by mimicking the processes that an immune system found within the body uses. AIS starts by creating an initial random population. Each member of the population, when applied to the market pricing problem, is a random price for each product, in a list.

The initial population is then cloned. The members of the cloned population are then mutated. For each member, the mutation varies based on how close to the optimal revenue that member is. If a member is far away from the best revenue, more of the prices are then changed.

After the mutations, the best members are selected to continue to the next iteration. The worst members are replaced with new random solutions. The steps, after and including the cloning step, are then repeated until the time allowed for execution is completed.

2.2 Improvements

To improve AIS some parameters can be tuned, to maximise the revenue for selected random seeds. For AIS there are four values that can be altered, the clone factor size, population size, replacement size and best fitness. To find the best values for each of these, AIS is run with the same selection of seeds each time only changing one of the parameters. The value for each of these parameters that returns the highest average revenue, then becomes the optimal parameter value. For example, in table 1, the highest average revenue seen is 5087.223 and is given by a clone factor size of 4.

Clone Factor Size	Avg. Revenue
1	5068.4065
2	5088.559
3	5064.6615
4	5087.223
5	5054.092
6	5052.047
7	5059.9535
8	5068.1345
20	5020.7795
50	4978.6845
100	4930.1405

Table 1: A table showing how the clone factor size affects the average revenue of the Artificial Immune System algorithm

Population Size	Avg. Revenue
3	5085.687
4	5094.63649999999
5	5082.489
6	5078.117
10	5015.176
15	5024.93949999999
20	5046.69399999999
25	4986.696
30	5045.11799999999

Table 2: A table showing how the population size affects the average revenue of the Artificial Immune System algorithm

After completing the optimisations for each parameter in AIS, the optimal values for this set of seeds are:

Clone Size Factor: 2

Population Size: 4

Replacement Size: 2

Best Fitness: 5250

Replacement Size	Avg. Revenue
1	5092.4515
2	5124.686
3	5114.244
4	3506.6665

Table 3: A table showing how the replacement size affects the average revenue of the Artificial Immune System algorithm

Best Fitness	Avg. Revenue
5000	5082.1465
5250	5128.714
5500	5103.8535
5750	5117.8855
6000	5091.116
7000	5072.9555
8000	5094.6365

Table 4: A table showing how the best fitness given affects the average revenue of the Artificial Immune System algorithm

3 Particle Swarm Optimization

3.1 Method

The Particle Swarm Optimisation algorithm (PSO) can be used on continuous problems. PSO starts by creating a random population of particles. A particle contains a solution to the problem and other details, such as a velocity and its previous best solution.

With each iteration, the best solution within the population is found. Each particle then has its velocity calculated based on inertia, its previous best solution and the best solution found by the entire population. Then each particle's solution is changed by applying the velocity calculated. This results in each particle swarming around the best solution, in the hope a particle finds a better solution to the problem.

3.2 Improvements

PSO can be optimised by changing the population size, inertia, the cognitive coefficient and social coefficient. To find the best value for each of theses parameters, only the parameter wanting to be improved is changed whilst the rest of the parameters remain the same. After trialing a set of values for the parameter, the one which returns the highest average revenue is used.

After optimising the four parameters for PSO, the optimal values for the selection of seeds the algorithm are run on are as follows:

Population Size: 2000

Inertia: 0.9

Cognitive Coefficient: 0.8

Social Coefficient: 0.8

Population Size	Avg. Revenue
100	5645.2315
200	5707.546
300	5777.8645
400	5788.269
500	5763.8675
600	5824.505
700	5825.435
800	5841.489
900	5826.2565
1000	5835.1
2000	5853.6085

Table 5: A table showing how the population size affects the average revenue of the Particle Swarm Optimization algorithm

Inertia	Avg. Revenue
0.1	55388.274
0.2	5393.3815
0.3	5402.6655
0.4	5523.3715
0.5	5599.2075
0.6	5685.5295
0.7	5838.55
0.721	5866.4915
0.8	5901.375
0.9	5945.044
1	4880.5175

Table 6: A table showing how the inertia affects the average revenue of the Particle Swarm Optimization algorithm

4 Results

After optimising each algorithm, they were both run with the same selection of seeds. The resulting revenues for each algorithm can be seen in table 9. These results are then inputted into Welch's t-test. The result gives a p-value of 0.000465. This p-value can then be compared against the significant level, which is 0.05. As the p-value is less than the significant level, the first hypothesis that PSO is no better, on average than AIS has been successfully disproved. This means that PSO does do better than AIS, on average, for the given problem set.

Social Coefficient	Avg. Revenue
0.7	5913.1325
0.8	5945.9055
0.9	5940.58
1.0	5923.2675
1.1	5893.0075
1.1193	5945.044
1.2	5850.541

Table 7: A table showing how the social coefficient affects the average revenue of the Particle Swarm Optimization algorithm

Cognitive Coefficient	Avg. Revenue
0.7	5894.6765
0.8	5937.2155
0.9	5921.913
1.0	5916.749
1.1	5910.546
1.1193	5913.1325
1.2	5928.1135

Table 8: A table showing how the cognitive coefficient affects the average revenue of the Particle Swarm Optimization algorithm

Seed	PSO	AIS	Seed	PSO	AIS	Seed	PSO	AIS
0	4850.3	4094.41	550	7004.77	5944.18	1100	8163.81	7002.96
50	5662.84	4736.69	600	5488.78	4317.06	1150	7257.54	6056.27
100	6623.14	5811.5	650	4531.03	3912.82	1200	4425.77	3783.32
150	6721.28	5766.99	700	6749.55	5747.99	1250	4718.91	3671.85
200	5299.78	4755.44	750	8205.17	7005.42	1300	5378.81	4437.59
250	6291.76	5491.72	800	5095.29	4467.36	1350	7106.14	5952.53
300	4404.94	3425.23	850	4333.09	3669.42	1400	6751.13	5478.17
350	6321.99	5508.57	900	6261.3	5588.94	1450	6670.19	5418.55
400	7054.8	6048.05	950	5297.22	4195.41	1500	8361.72	7382.26
450	5971.35	4960.98	1000	5051.75	4297.68	1550	4956.78	4087.16
500	6822.01	5887.2	1050	6211.78	5319.49	1600	5197.07	4238.38
Avg.	6037.63	5104.8967						

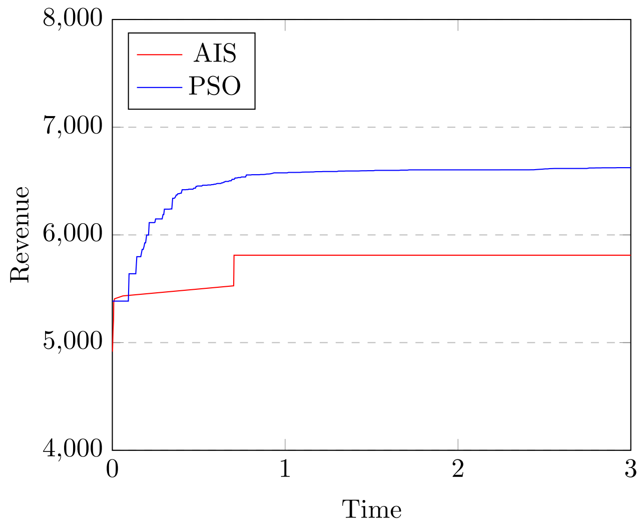
Table 9: A table showing the resulting revenues of each algorithm with different random seeds

Figure 1, shows how each algorithm improves during its execution, when given the a seed of 100. It can be seen that PSO returns the highest revenue.

When PSO starts its revenue is similar to the starting revenue of AIS. Within the first second of execution the revenue improves the most, then for the later two seconds it only makes minor improvements.

AIS follows a similar trend to PSO, improving at the beginning then improving less later in its execution. The sudden jumps in revenue may be a result of a good random solution replacing one of the worse solutions in the population.

Figure 1: The highest revenue found throughout each algorithm's execution. (Seed: 100)



5 Conclusion

This paper has stated the steps required to implement the Particle Swarm Optimisation algorithm and Artificial Immune System algorithm and the improvements that can be made to increase the resulting revenue for a given set of problems.

The results from each algorithm were then compared using Welch's t-test, this gave a result lower than the significance level. Due to this result the null hypothesis

> "The Particle Swarm Optimization (PSO) algorithm, on average, does not perform better than the Artificial Immune System (AIS) algorithm when given the market pricing problem in terms of revenue."

can be rejected. Therefore the alternative hypothesis

> "The Particle Swarm Optimization (PSO) algorithm, on average, does perform better than the Artificial Immune System (AIS) algorithm when given the market pricing problem in terms of revenue."

has been proved.