ECM3412 Nature Inspired Computation: Ant Colony Optimisation Report

690037391, University of Exeter, UK

Bin Packing Problem 1

In Bin Packing Problem 1 (BPP1), there are 10 bins and 500 items to be placed within them. The weights of each item follows a linear relationship the weight is equal to the item number (i.e. the first item, item 1, has a weight of 1). All experiments for BPP1 where run sequentially (i.e. each of the 5 trials were run one after another) rather than in parallel (i.e. all 5 trials are run at the same time).

Time (seconds)	Fitness
55801	938
55819	933
54204	950
56780	885
56835	1012

Average Time	Average Fitness
55887.8	943.6

Table 1. p = 100 and e = 0.9, Start Time: Fri Dec 24 2021, 15:06:33 - End Time: Mon Dec 27 2021, 05:00:14

Time (seconds)	Fitness
55851	1055
55780	1102
54173	682
56866	1074
56809	961

Average Time	Average Fitness
55895.8	974.8

Table 2. p = 100 and e = 0.5, Start Time: Fri Dec 24 2021, 15:06:26 - End Time: Mon Dec 27 2021, 04:59:16

Time (seconds)	Fitness
5710	1144
5637	1171
5822	1150
6209	1260
6010	1102

Average Time	Average Fitness
5877.6	1165.4

Table 3. p = 10 and e = 0.9, Start Time: Fri Dec 24 2021, 15:06:35 - End Time: Fri Dec 24 2021, 23:16:24

Time (seconds)	Fitness
5703	1005
5627	987
5812	1294
6189	1335
6026	1073

Average Time	Average Fitness
5871.4	1138.8

Table 4. p = 10 and e = 0.5, Start Time: Fri Dec 24 2021, 15:06:31 - End Time: Fri Dec 24 2021, 23:15:46

Bin Packing Problem 2

In Bin Packing Problem 1, there are 50 bins and 500 items to be placed within them. The weights of each item follows an exponential relationship the weight is equal to the item number squared (i.e. the final item, item 500, has a weight of $500^2 = 25000$). For Experiments 1 and 2, instead of running a single instance of my executable to completion, I ran 5 copies of my executable stopping each when a single trial had finished; hence, the total time elapsed for all 5 trials far exceeds the difference between the start time and end time of computation.

Time (seconds)	Fitness
459999	765653
458267	787265
446532	775642
456231	765453
432665	778643

Average Time	Average Fitness
450738.8	774531.2

Table 5. p = 100 and e = 0.9, Start Time: Mon Dec 27 2021, 11:55:53 - End Time: Sat Jan 01 2022, 19:42:33

Time (seconds)	Fitness
456994	792839
457260	765279
449562	772685
455963	752643
453782	757562

Average Time	Average Fitness
454712.2	768201.6

Table 6. p = 100 and e = 0.5, Start Time: Mon Dec 27 2021, 11:55:21 - End Time: Sat Jan 01 2022, 18:56:36

Time (seconds)	Fitness
28298	789882
32922	809275
35642	855987
30195	838947
32626	858936

Average Time	Average Fitness
31936.6	830605.4

Table 7. p = 10 and e = 0.9, Start Time: Sat Dec 25 2021, 00:34:38 - End Time: Sun Dec 26 2021, 20:55:59

Time (seconds)	Fitness
27084	838740
30836	808479
35169	794181
28999	843098
31296	865500

Average Time	Average Fitness
30676.8	829999.6

Table 8. p = 10 and e = 0.5, Start Time: Sat Dec 25 2021, 00:31:00 - End Time: Sun Dec 26 2021, 20:52:23

WHICH COMBINATION OF PARAMETERS PRODUCES THE BEST RESULTS?

For Bin Packing Problem 1, based on the average time to complete a trial, I believe that 10 ant paths with an evaporation rate of 0.5 produces the best results. Although it has the second worst average fitness of 1138.8, each trial takes on average 5871.4 seconds, which is approximately 1 hour 37 minutes and 52 seconds. The average fitness of Experiment 4 is only 195.2 higher than the best average fitness, 943.6 from Experiment 1. Furthermore, Experiment 1 has an average trial time almost ten times slower than experiment 4, at 55895.8 seconds (15 hours 31 minutes and 36 seconds), meaning it improves on the fitness by 195.2 at an extra cost of almost 14 hours. Where Experiment 4 took in total 8 hours 9 minutes and 17 seconds, Experiment 1 took 3 days 5 hours 37 minute and 19 seconds. Experiment 3 provided an average trial fitness of 1165.4 and average trial time of 5877.6 seconds (1 hour 37 minutes and 58 seconds), demonstrating the worst fitness for BPP1. Furthermore, Experiment 2 provided an average trial fitness of 974.8 and average trial time of 55895.8 seconds (15 hours 31 minutes and 36 seconds), demonstrating that 0.5 evaporation rate is not as effective for 100 ant paths as it was for 10 ant paths.

For Bin Packing Problem 2, again based on the average time to complete a trial, I believe that Experiment 4 provides the best results. Experiment 4 provides an average trial fitness of 829999.6 at an average trial time of 30676.8 seconds (8 hours 31 minutes and 17 seconds), which is just over 5 times longer than the same experiment for BPP1. The best average trial fitness found is from Experiment 2 at 768201.6, with an average time of 454712.2 seconds (5 days 6 hours 18 minutes and 33 seconds). This fitness is only 61798 lower than Experiment 4, but takes 424035.4 seconds (4 days 21 hours 47 minutes and 16 seconds) longer to reach, thus making Experiment 4 a better choice. Experiment 1 has an average fitness of 774531.2 and average time of 450738.8 seconds (5 days 5 hours 12 minutes and 19 seconds) and Experiment 3 has an average fitness of 830605.4 and average time of 31936.6 seconds (8 hours 52 minutes and 17 seconds).

WHAT DO YOU THINK IS THE REASON FOR YOUR FINDINGS IN SECTION ?

Firstly, I believe that my own algorithm is very inefficient and it could be greatly improved, making BPP2 much quicker to execute such that it does not take over 5 days for one trial to complete (thus making it so that I could execute it sequentially in a suitable amount of time and not 25 days). I believe that Experiment 4 for both problems is the most efficient as they both have a lower evaporation rate, meaning the best paths retain more of their pheromone levels than with an evaporation rate of 0.9. Furthermore, The experiments using only 10 paths per fitness evaluation are faster to execute as they do not explore as many paths, but suffer from lower explorability hence the higher fitness values.

HOW DO EACH OF THE PARAMETER SETTINGS INFLUENCE THE PERFORMANCE OF THE ALGORITHM?

The number of ant paths per fitness evaluation seems to be the most deciding factor between experiments, with the larger value of 100 causing a great increase in computation time (almost 10 times as long for BPP1 and almost 15 times longer for BPP2), creating almost unsustainable computation times in BPP2 experiments 1 and 2; if I had not run these experiments in parallel, they would have taken almost an entire month to fully compute, which is entirely unacceptable and indicative of the inefficiency of my own code. Furthermore, the evaporation rate seems to affect performance minimally, with larger evaporation rates increasing computation time; for example in BPP1, Experiment 3 takes 6.2 seconds longer than Experiment 4.

DO YOU THINK OTHER ALGORITHMS MIGHT HAVE PROVIDED BETTER RESULTS?

I believe that evolutionary algorithms could provide better results as they can more efficiently encode the data and manipulate it easier. In my own solution I make use of multiple C++ Vectors, as well as Vectors of Vectors which are very inefficient to iterate over. Furthermore, an Evolutionary Algorithm (EA) would not need to create a representation for a construction graph like for the Ant Colony Optimisation. In an EA, one could use permutation with separators representation [1] which represents a solution to the problem as a list of n items and l separators of bins, with $\{1, \cdots, n\}$ representing the items and $\{n+1, \cdots, n+l\}$ representing the separators; separators can appear in the first or last position of a bin and can appear together. For example, a solution for problem with 7 items and 3 bins might be $S_1 = (1, 3, 9, 8, 5, 2, 7, 10, 6, 4)$, where the items are partioned as such: $(1, 3)$, $(5, 2, 7)$, $(6, 4)$ [2]. Stawowy [2] proposes an evolutionary heuristic for the bin packing problem, developing a unique concept of separator movement during mutation, as well as using separator removal to reduce problem size. The experiments done in Stawowy's [2] paper show that the proposed heuristic is simpler than but comparable in performance to more complex heuristics such as Alvim et al.'s hybrid improvement heuristic [3].

REFERENCES

[1] D. R. Jorues and M. A. BeItrarno, "Solving partitioning problems with genetic algorithms," in *Proceedings of the 4th ICGA*, 1991, pp. 442–449.

[2] A. Stawowy, "Evolutionary based heuristic for bin packing problem," *Computers & Industrial Engineering*, vol. 55, no. 2, pp. 465–474, 2008.

[3] A. C. Alvim, C. C. Ribeiro, F. Glover, and D. J. Aloise, "A hybrid improvement heuristic for the one-dimensional bin packing problem," *Journal of Heuristics*, vol. 10, no. 2, pp. 205–229, 2004.