Abstract

The Bin Packing Problem (BPP) is solved using an evolutionary algorithm. There are six experiments with different parameters to observe the behaviour of the algorithm and impact of each parameter in solving the problem.

Introduction

The Bin Packing Problem (BPP) is an optimization problem. It has a goal to assign a number of items n that have different weights to be packed into a bin in which the total number of bins is minimised. BPP is implemented along with an evolutionary algorithm (EA). The idea of EA is to develop methods to transform individuals and use selection mechanisms inspired by nature. The pseudocode of the algorithm can be seen in figure 1:

Experiment

There are two bin packing problems that have been implemented
BPP1: In the problem, there are 500 items with the weight of each item is **2i** and number of bins b=20
BPP2: In the problem, there are 500 items with the weight of each item is **i^2** and number of bins b=100

Components of Evolutionary algorithm

Fitness Function: fitness values are derived from the difference between the heaviest bin and lightest bin. It is to evaluate the chromosome from being the optimal solution.

Parent selection method
Binary Tournament Selection is used where the chromosomes that have the best fitness value is chosen to become the selected parents.

Single-Point Crossover: It is used to choose a random crossover point and the genes from two parents are interchanges to create new children.

Mutation: The type of mutation that is used is Multi-Gene Mutation. Parameter Mk is used where a gene is chosen randomly and changed to random value.

Replacement: Replace the children within the population when the solutions have better fitness values.

Termination Condition: Run the algorithm once and stop when 10,000 fitness evaluations have been reached

There are six experiments with each of them having five trials. Every experiments contains differents parameters:

M: The number of genes that can changed from mutation

N: The size of the population. The initial population that is set for the experiments are 10 and 100

Results

Table1: Results crossover & operator M1 and population size 10.

	BPP1	BPP2
Trial 1	44	332056
Trial 2	22	297648
Trial 3	36	310593
Trial 4	44	368447
Trial 5	36	326641
Average	36.4	327077.0

Table 2: Results of experiment 2 crossover & operator M1 and population size 100.

	BPP1	BPP2
Trial 1	216	532783
Trial 2	194	545682
Trial 3	222	611220
Trial 4	162	517504
Trial 5	182	523314
Average	195.2	546100.6

Table 3:Results of experiment 3 crossover & operator M5 and population size 10.

	BPP1	BPP2
Trial 1	456	332056
Trial 2	476	297648
Trial 3	400	310593
Trial 4	490	368447
Trial 5	418	326641
Average	448.0	318768.6

Table 4:Results of experiment 4 crossover & operator M5 and population size 100.

	BPP1	BPP2
Trial 1	744	468774
Trial 2	702	480423
Trial 3	552	512291
Trial 4	756	474760
Trial 5	684	521118
Average	687.6	491473.2

Table 5: Results of experiment 5 operator M5 and population size 10.

	BPP1	BPP2
Trial 1	3056	795322
Trial 2	4040	801467
Trial 3	3126	842960
Trial 4	4222	819534
Trial 5	3632	795807
Average	3615.2	811018.0

Table 6:Results of experiment 6 crossover and population size 10.

	BPP1	BPP2
Trial 1	1434	795850
Trial 2	1920	813486
Trial 3	5302	807075
Trial 4	3884	786248
Trial 5	2996	813248
Average	3107.2	803181.4

Plots

figure1:

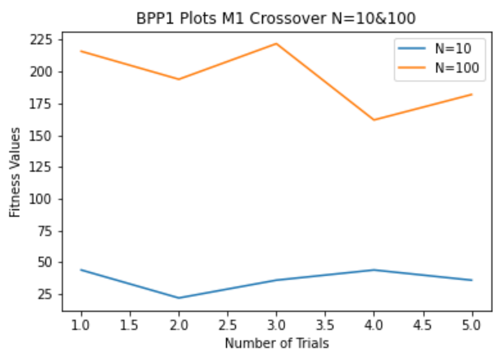


Figure 2:

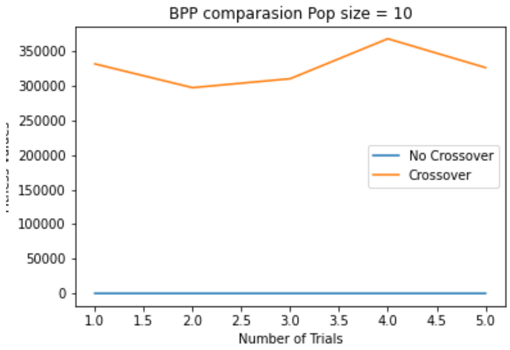
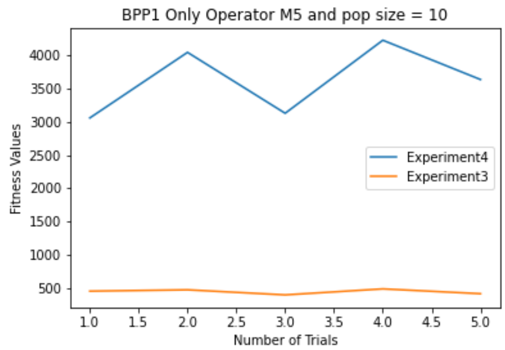


Figure 3:



Question

Question 1: Which combination of parameters produces the best results for BPP1 and BPP2?

BPP1, the parameters of EA with crossover & operator M1 and population size 10 produces the best results
BPP1, the parameters of EA with crossover & operator M5 and population size 10 produces the best results

Question 2: Why do you think this is the case?

The important phases of evolutionary algorithms consist of having population initialization, fitness calculation, selection, crossover and mutation. The combination of parameters in BPP1 and BPP2 have similarity which are the number of population and the inclusivity of all phases of the evolutionary algorithm.

It can be said that all the phases and the population sizes have impacts on the performance of the algorithm.
Parent selection, mutation, and crossover influences the diversity of the population. It increases the ability to explore different candidates of solutions and avoid getting stuck in local solutions.

The population size affects the performance of the algorithm based on the speed of each iteration. It impacts the ability to search for the best solution in the search space. A large population size can help the algorithm to explore more candidates more thoroughly and lead to better solutions.

However, as we see from the results, the best parameters use smaller population size. This is because the number of time iterations needed is less in a small population. Global optimum solution is reached faster by having quicker convergence.

Question 3: What was the effect when you removed mutation? What about crossover?

Both crossover and mutation are essential mechanisms for evolutionary algorithms. Therefore, removing both would result in an ineffective and a worse performing algorithm.

Crossover is crucial in generating genetic diversity. Diversity of the population affects the performance of the algorithm. By crossover, the algorithm is able to evaluate population convergence, combine solutions from different candidates and search for optimal solutions more efficiently. Removing crossover, the algorithm would do local mutation. The optimal solution is reached from randomised local search which is harder to escape from local optimum. As you can see in figure 2 that algorithm without crossover results in higher fitness values.

Without mutation, the algorithm creates worse solutions. This can be seen through the results from all tables where fitness is changing. Mutation is used to maintain diversity in the population and helpful to avoid the local optimum solution. It is because low diversity could slow the convergence to the global optimum solution.

Question 4: Which other nature-inspired algorithms might be effective in optimising the BPP problems? Explain your choice(s).

I think ant colony optimization can be used instead of an evolutionary algorithm. Ant colony optimization(ACO) is a meta-heuristic technique to solve optimization problems to find optimum solutions through graphs. Several steps are needed to implement ACO: defining pheromone trails, defining heuristic function, identify effective mechanisms to update pheromone trails, and compute fitness function.

ACO might perform better for BPP because the pheromone trails along with other steps allow the search spaces to decrease sizes and concentrate on exploring fewer solutions quicker where solutions are located.

Further Experiment

There are several types of crossover and parent selection for evolutionary algorithms.

I would do uniform crossover instead of single point crossover. It is because the uniform requires randomly chromosomes within the population. Therefore, candidate solutions would be more diverse, could expand the search space and find a better global optimum.

Roulette wheel selection could be used instead of tournament selection. This would show better results because it uses probability to choose parents. Fitter individuals would have a higher probability of being chosen since the beginning. Therefore, it would increase the performance of the algorithm. Moreover, it allows more individuals to be evaluated as parents.

Conclusion

Evolution algorithm is implemented to solve bin packing problems with various parameters and methods. It has been found that there are many factors that influence the performance of the algorithm such as crossover, mutation, and population sizes. This problem is not limited to evolution algorithms, Ant Colony Optimization can also be implemented as the more efficient algorithm.