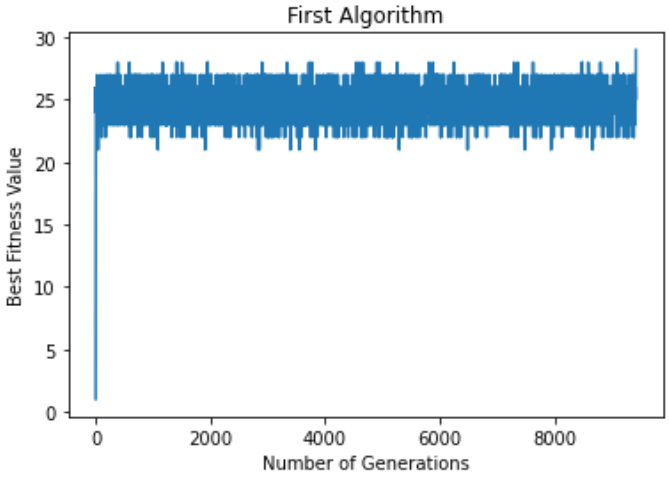
Artificial Intelligence Assignment 1

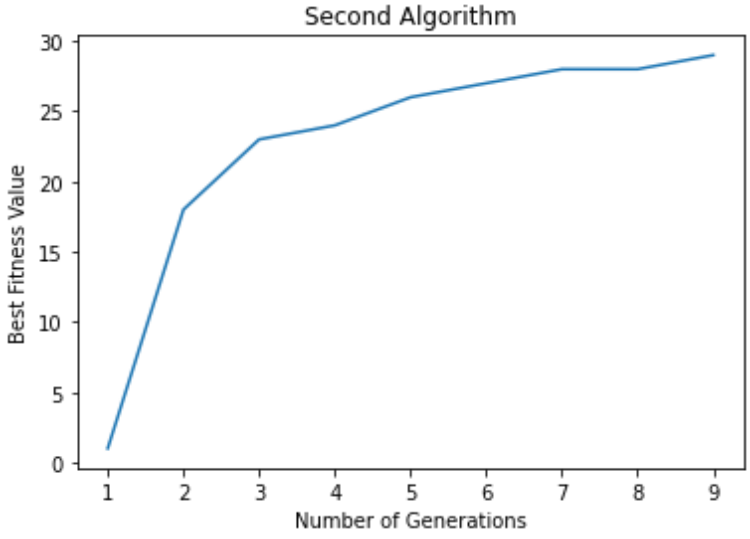
Saurabh Wandhekar
2018A7PS0157G

1) 8-Queens Problem

Output of First Algorithm: This algorithm took around 10000 generations to reach the optimal solution.



Output of Second Algorithm: This algorithm could reach the optimum solution in less than 10-20 generations.



Modifications that worked:

1) Size of population was increased from 20 to 200: This ensured more diversity in the population and hence the algorithm could converge faster to the optimal solution.

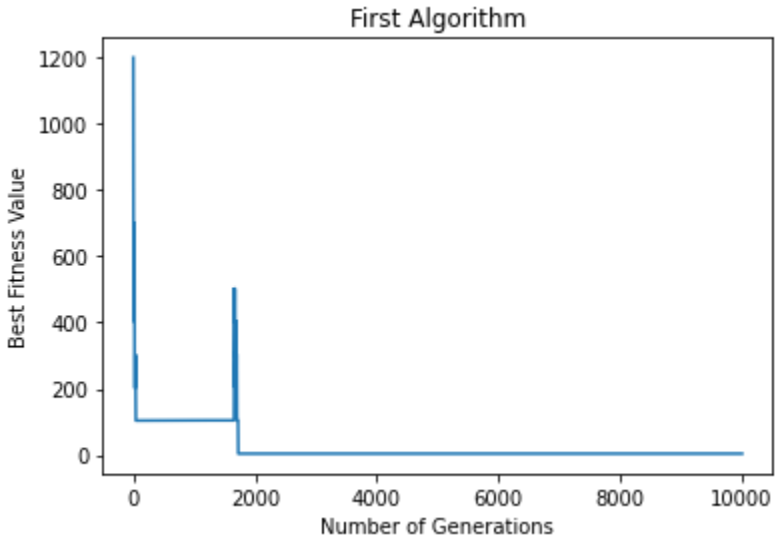
2) Reproduction/crossover function was changed:
   A random subset of the first parent is chosen and that subset occurs in the child in the exact same position as in the parent. Now, the rest of the elements of the child comes from the second parent, but it is ensured that if the elements in the subset of parent 1 are present in parent 2 then their first occurrence in parent 2 is not included in the child. Ex: parent1: [1,2,3,***4,5,6***,7,8] parent2: [4,5,3,2,4,8,1,6] child: [3,2,4,**4,5,6**,8,1]
   This ensures that most of the queens are in different rows in the child.

3) 2 children were created from 2 parents and the child with the best fitness value was placed in the new population. I.e. once subset is taken from parent 1 and in the next child, subset is taken from parent 2. This ensured the best fit child was added to the new population.

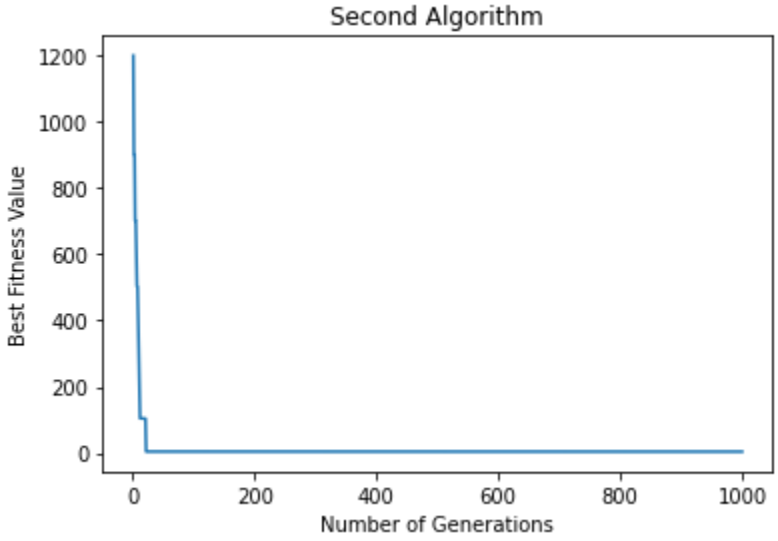
4) Mutation rate was lowered from 0.3 to 0.05: Due to the updated crossover function it wasn't necessary to perform mutation that often.

2) Travelling Salesman Problem

Output of First Algorithm: This algorithm took around 2000 generations to reach the optimal solution.



Output of Second Algorithm: This algorithm took around 25-30 generations to reach the optimal solution.



Modifications that worked:

1) Size of population was increased from 20 to 200: This ensured more diversity in the population and hence the algorithm could converge faster to the optimal solution.

2) 2 children were created from 2 parents and the child with the best fitness value was placed in the new population. I.e. once subset is taken from parent 1 and in the next child, subset is taken from parent 2. This ensured the best fit child was added to the new population.

Modifications that did not necessarily work:

1) Choosing one parent with highest probability and the other parent with random probability. Gave a solution with more or less the same number of generations.

2) Double mutation with lesser mutation rate than the first. The solution was not converging in this case.