Ant Colony Optimisation for Travelling Salesman Problem for N cities

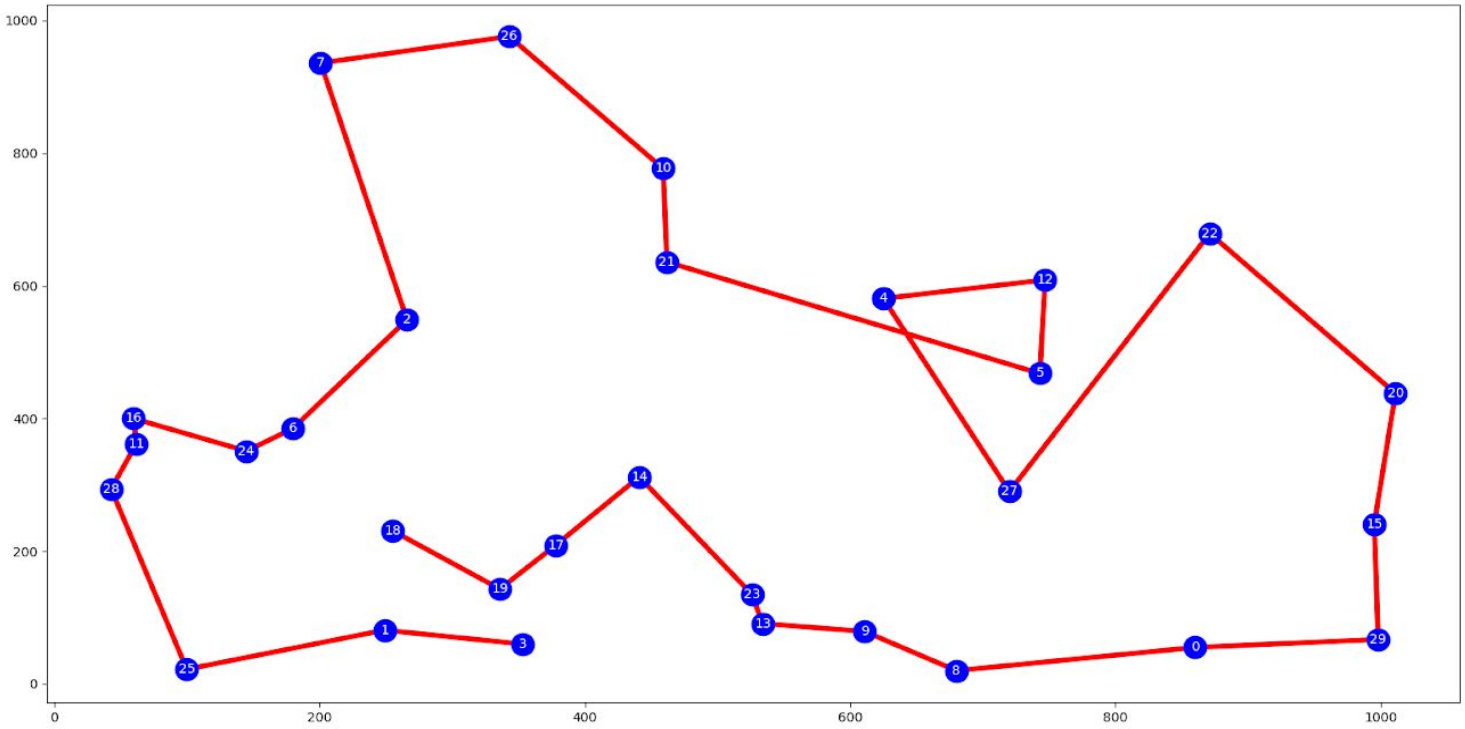
- **Assumptions:**
    a. There's a path between every city to every other city.
    b. Convergence is achieved when the cost of the path is same 3 times in a row.
- **Methodology:**
    a. Randomly initialize (x,y) coordinates for N cities.
    b. Calculate distance matrix
    c. Place an ant at some random city
    d. Move ant to all cities
    e. Update pheromone
    f. Repeat steps (c - e) for all ants
    g. Return optimal path
- **Algorithm:**

```
For i-generations
        For all ants
                For all cities
                        Let ant move based on Pij
                End loop
        Calculate path_cost, path[ ]
        End loop
        Update Pheromone
End loop
```

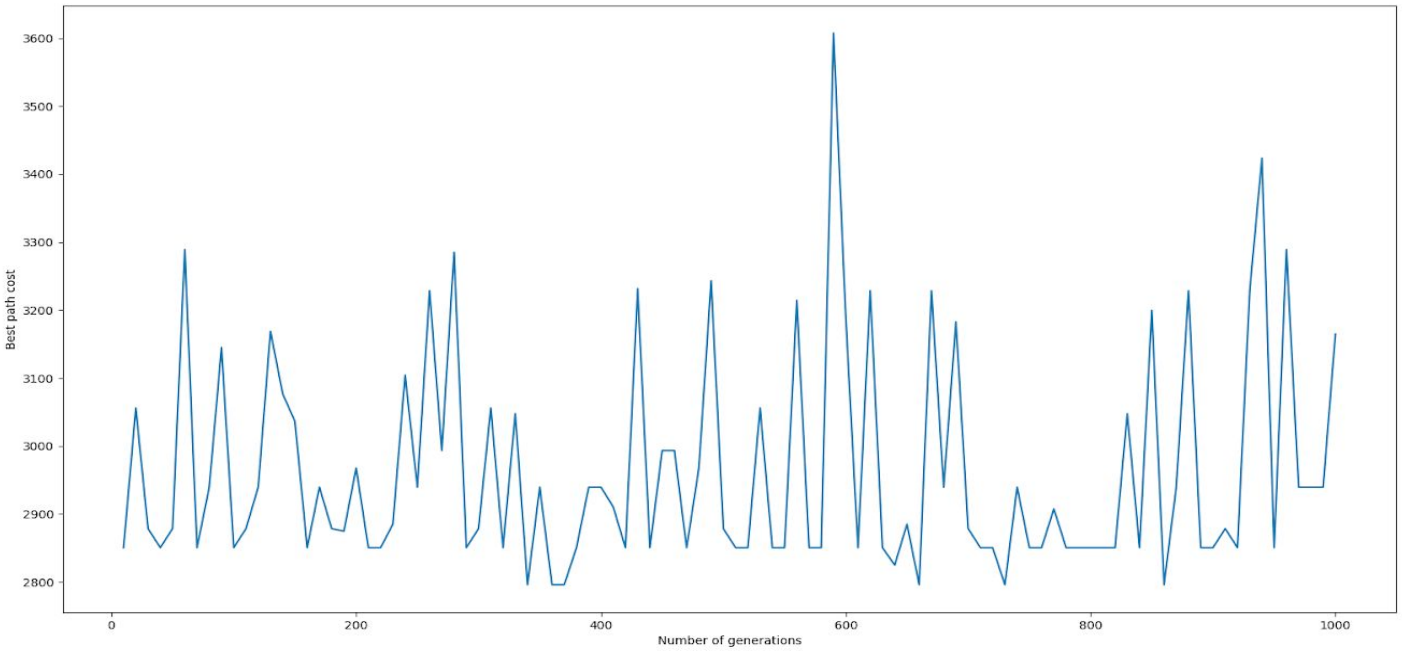
- **Observations:**
    - Increasing decay rate slows the rate of convergence.
    - Increasing alpha value increases the rate of convergence.
    - Increasing beta value increases the rate of convergence.
    - For the same number of cities, increasing the number of ants results in faster convergence.
    - Best path cost does not improve significantly on increasing number of generations.

- **Results:**

Solution for TSP (N=30):

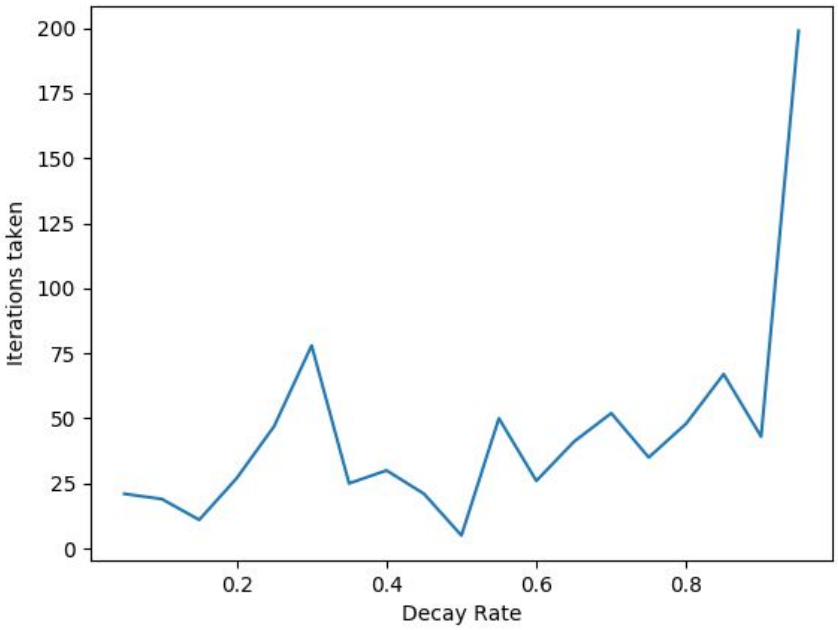


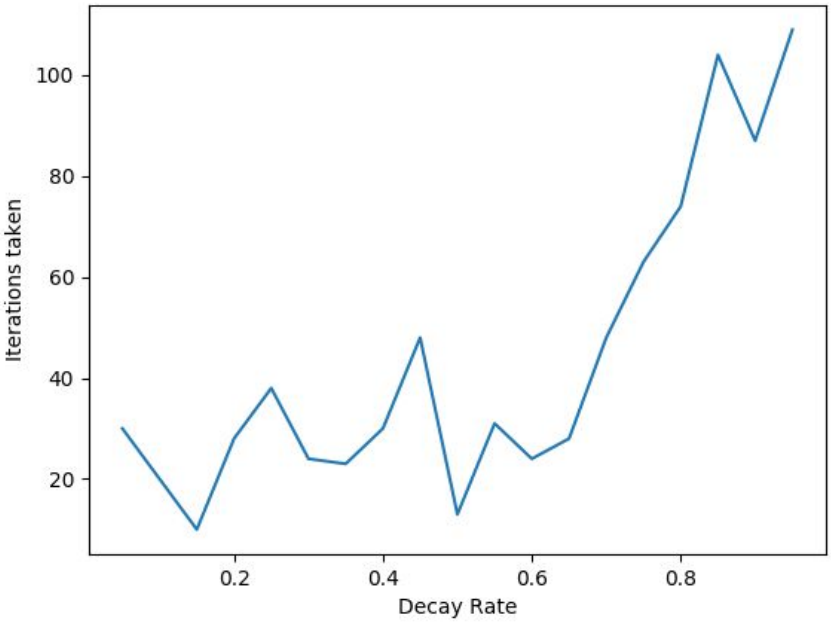
- Number of generations versus Best path cost



- **Inference:**
  - No significant relation found

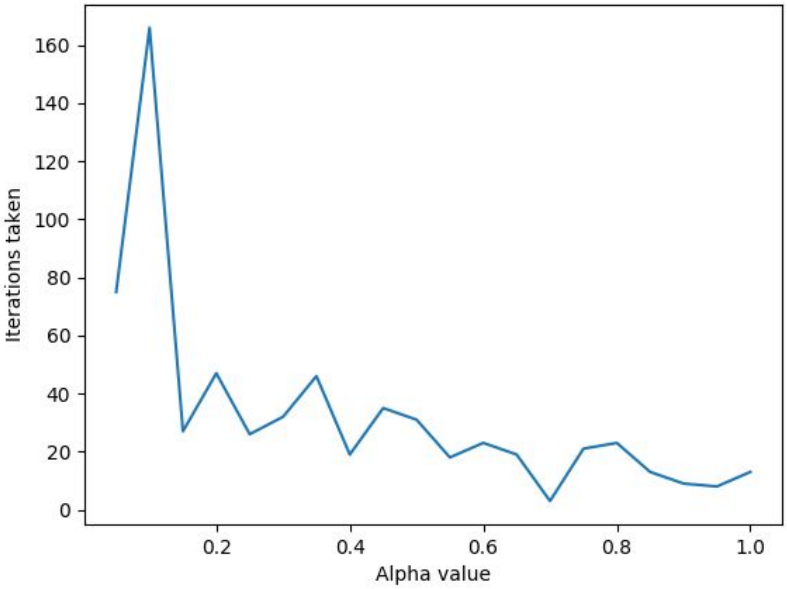
- **Decay Rate versus Number of iterations for convergence:**

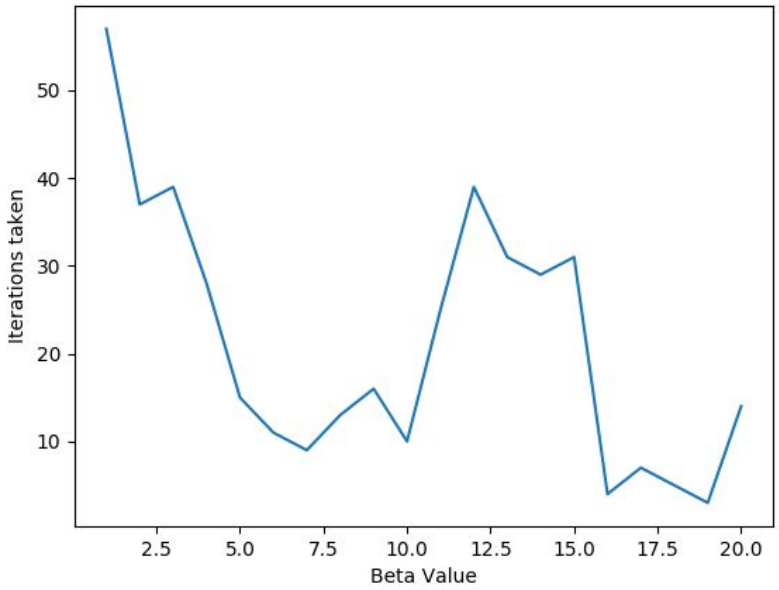




- **Inference:**
    - Convergence slows down with increasing decay rate.
    - This may be because the frequently travelled path information gest eliminated.

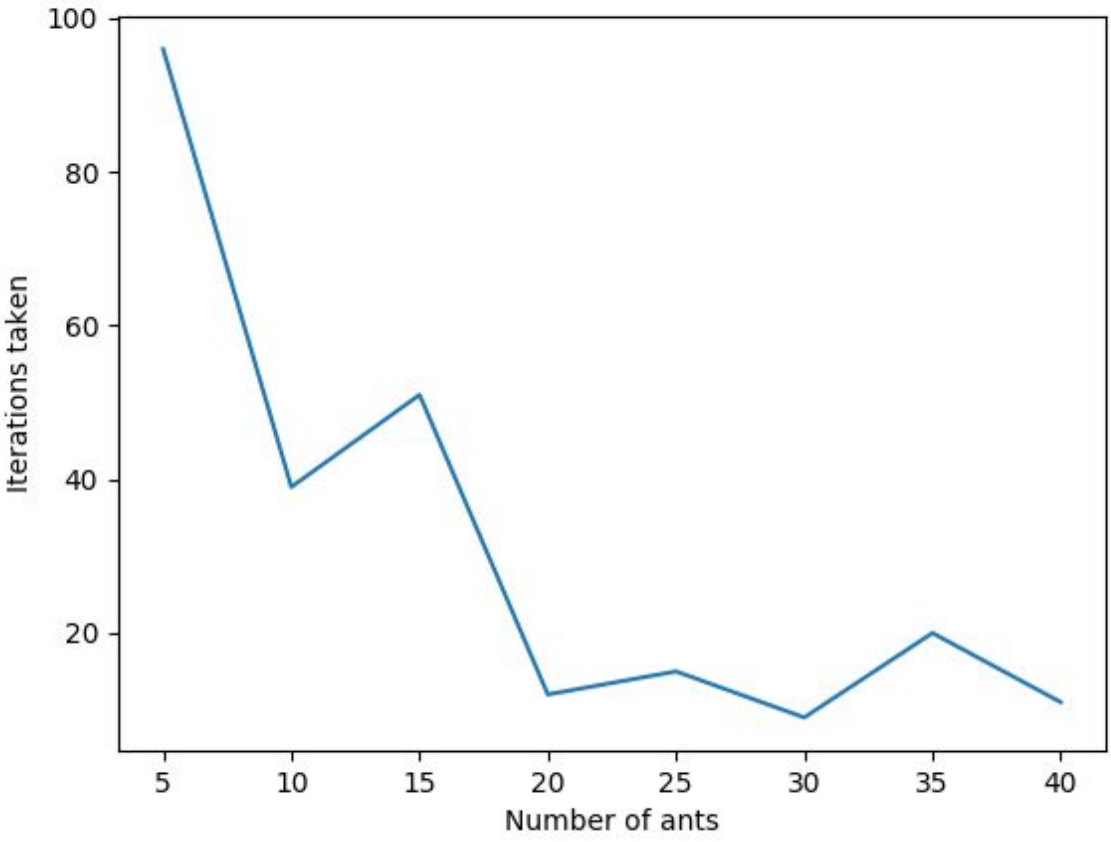
- **Effect of alpha and beta values on convergence :**





- **Inference:**
  - Increasing alpha value escalates convergence.
  - Increasing beta value escalates convergence.

- **Number of ants versus convergence:**



- **Inference:**
    - For the same number of cities, increasing the number of ants escalates convergence.
    - This is because more number of ants are able to find the best path quicker.