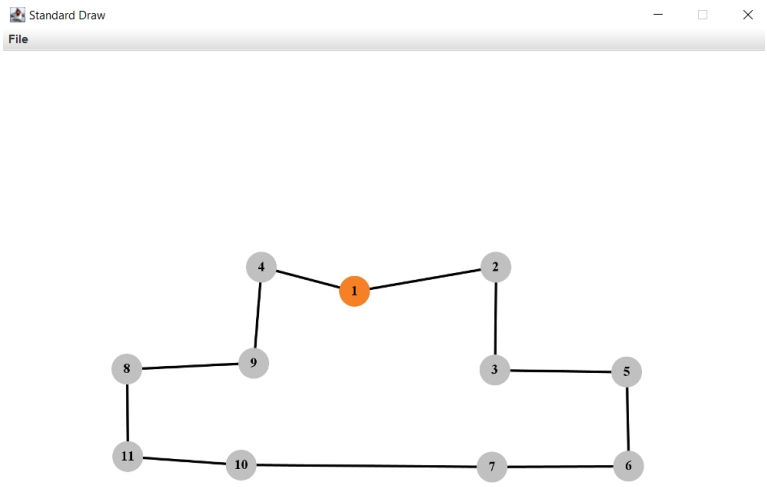
Migros Delivery using Ant Colony Optimization

Outputs of Brute force Method:

Input file 1:



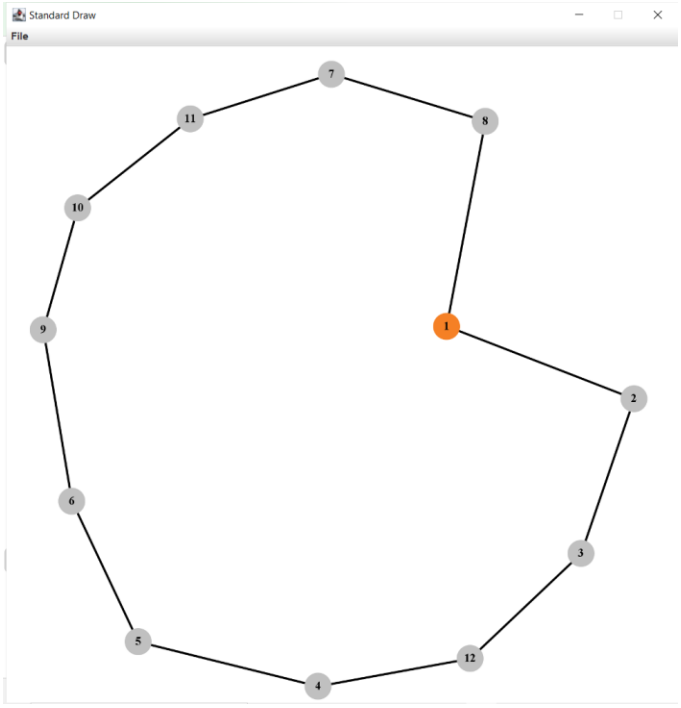
Method: Brute-force Method

Shortest Distance: 1,79529

Shortest Path: [1, 4, 9, 8, 11, 10, 7, 6, 5, 3, 2, 1]

Time it takes to find shortest path: 0.584 seconds

Input File 2:



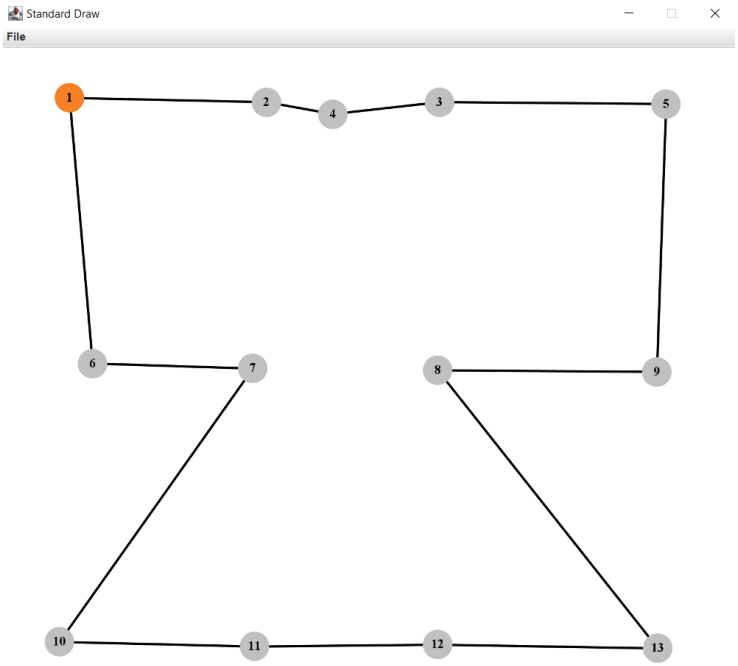
Method: Brute-force Method

Shortest Distance: 2,93588

Shortest Path: [1, 8, 7, 11, 10, 9, 6, 5, 4, 12, 3, 2, 1]

Time it takes to find shortest path: 2.151 seconds

Input file 3:



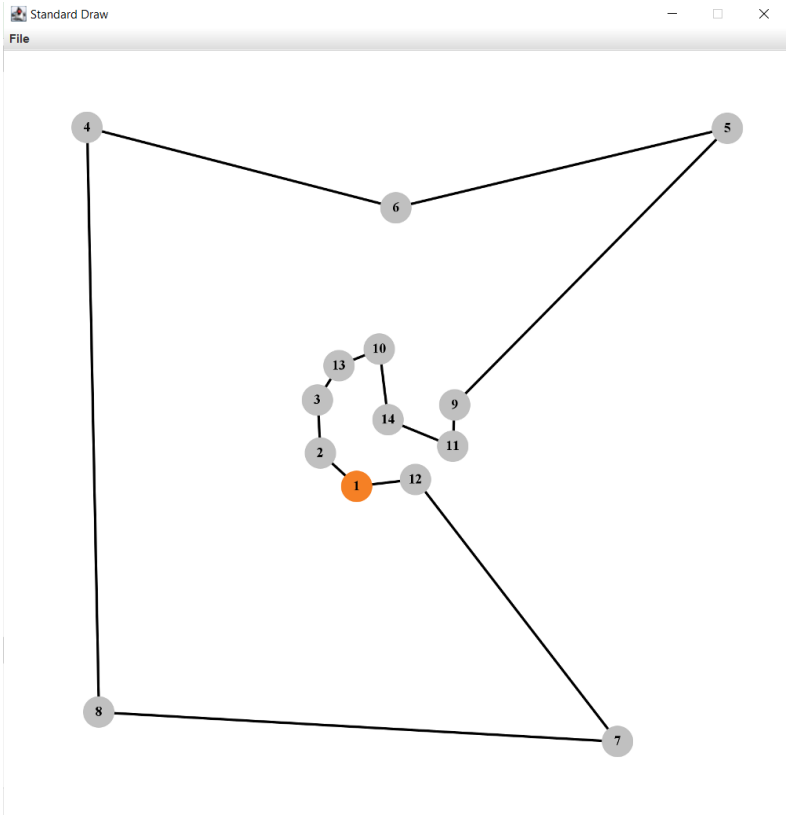
Method: Brute-force Method

Shortest Distance: 3,80292

Shortest Path: [1, 2, 4, 3, 5, 9, 8, 13, 12, 11, 10, 7, 6, 1]

Time it takes to find shortest path: 18.374 seconds

Input file 4:



Method: Brute-force Method

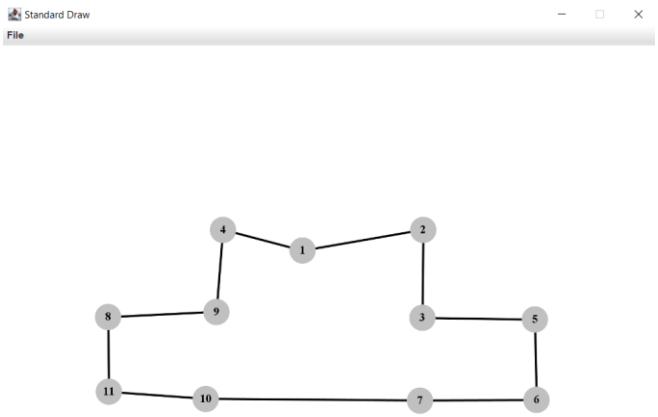
Shortest Distance: 3,71091

Shortest Path: [1, 2, 3, 13, 10, 14, 11, 9, 5, 6, 4, 8, 7, 12, 1]

Time it takes to find shortest path: 266.252 seconds

Outputs of Ant Colony Optimization Method:

Input file 1:

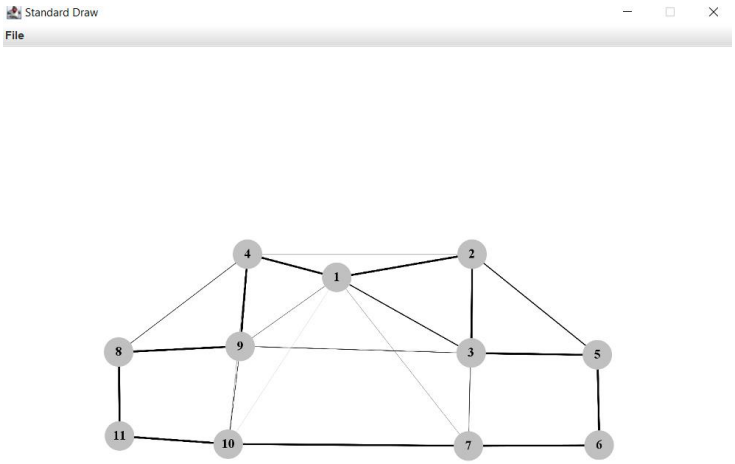


Method: Ant Colony Algorithm

Shortest Distance: 1,79529

Shortest Path: [1, 4, 9, 8, 11, 10, 7, 6, 5, 3, 2, 1]

Time it takes to find the shortest path: 0.362 seconds



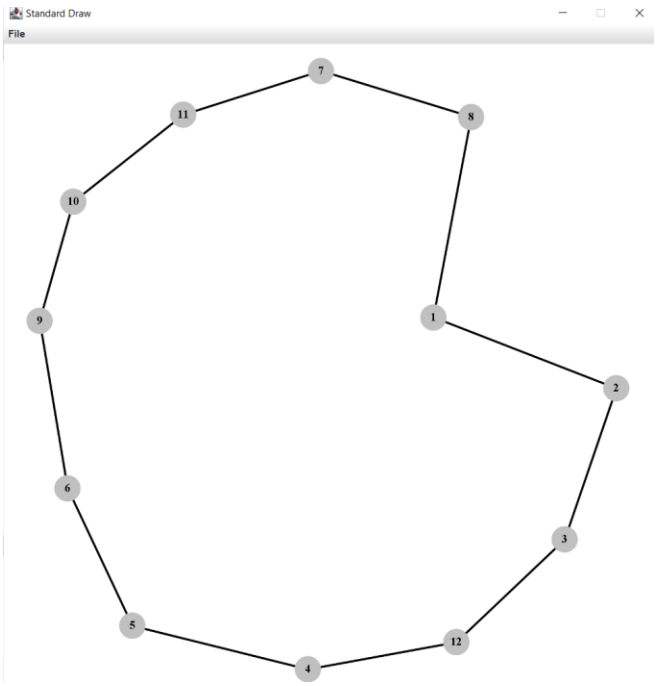
Method: Ant Colony Algorithm

Shortest Distance: 1,79529

Shortest Path: [1, 4, 9, 8, 11, 10, 7, 6, 5, 3, 2, 1]

Time it takes to find the shortest path: 2.195 seconds

Input file 2:

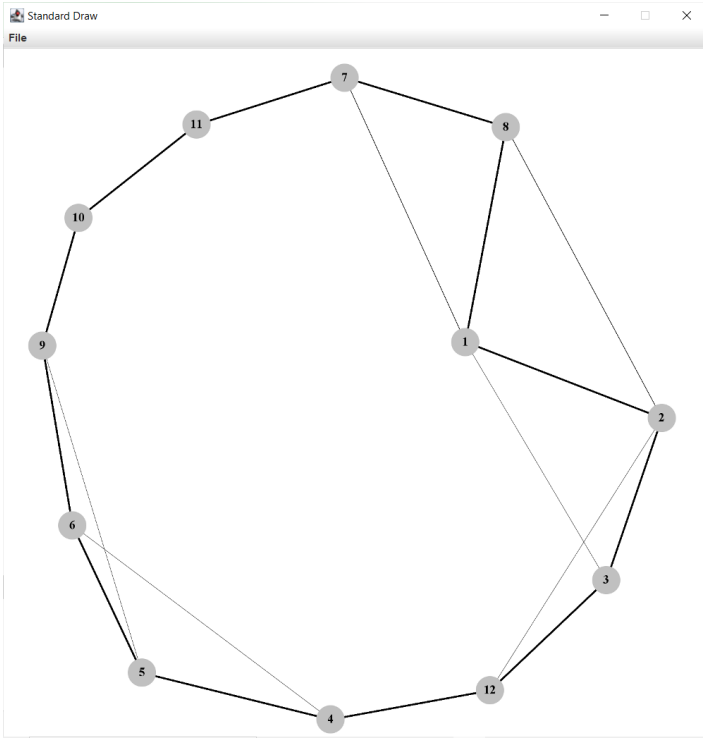


Method: Ant Colony Algorithm

Shortest Distance: 2,93588

Shortest Path: [1, 8, 7, 11, 10, 9, 6, 5, 4, 12, 3, 2, 1]

Time it takes to find the shortest path: 0.422 seconds



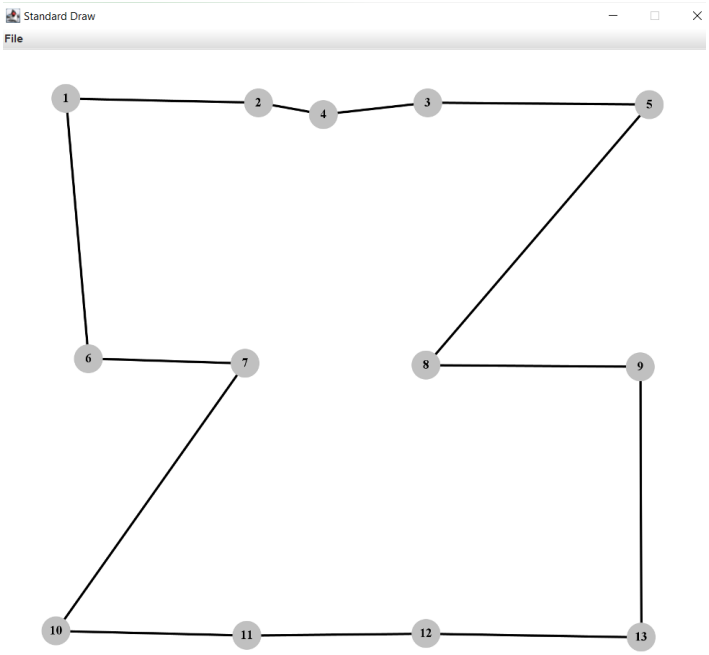
Method: Ant Colony Algorithm

Shortest Distance: 2,93588

Shortest Path: [1, 8, 7, 11, 10, 9, 6, 5, 4, 12, 3, 2, 1]

Time it takes to find the shortest path: 2.4 seconds

Input file 3:

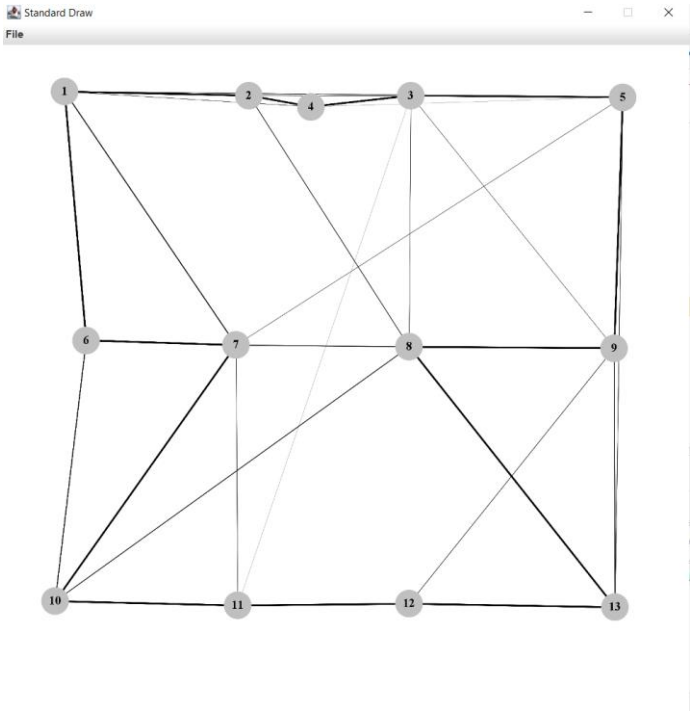


thod: Ant Colony Algorithm

Shortest Distance: 3,80853

Shortest Path: [1, 2, 4, 3, 5, 8, 9, 13, 12, 11, 10, 7, 6, 1]

Time it takes to find the shortest path: 0.402 seconds



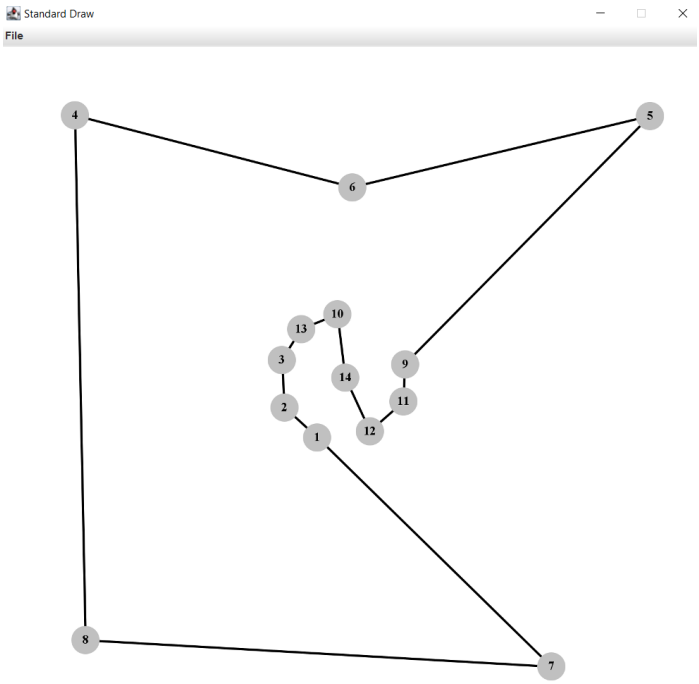
Method: Ant Colony Algorithm

Shortest Distance: 3,80292

Shortest Path: [1, 2, 4, 3, 5, 9, 8, 13, 12, 11, 10, 7, 6, 1]

Time it takes to find the shortest path: 2.79 seconds

Input file 4:

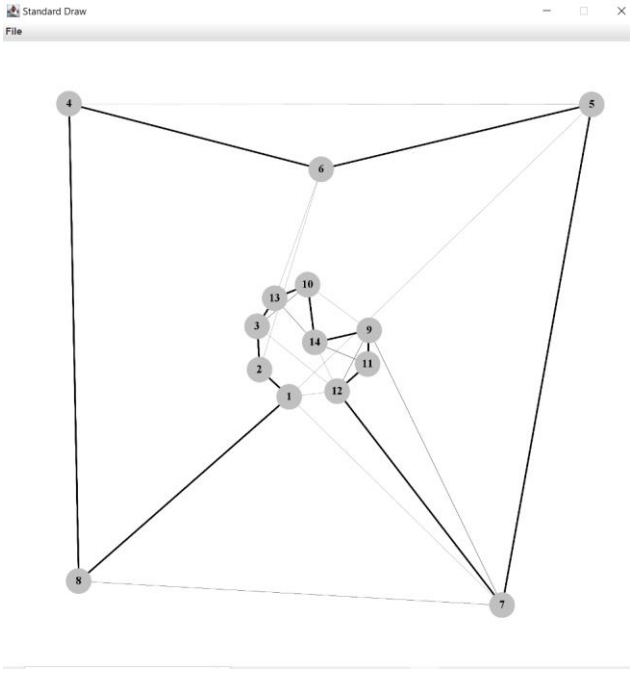


Method: Ant Colony Algorithm

Shortest Distance: 3,73714

Shortest Path: [1, 2, 3, 13, 10, 14, 12, 11, 9, 5, 6, 4, 8, 7, 1]

Time it takes to find the shortest path: 0.457 seconds



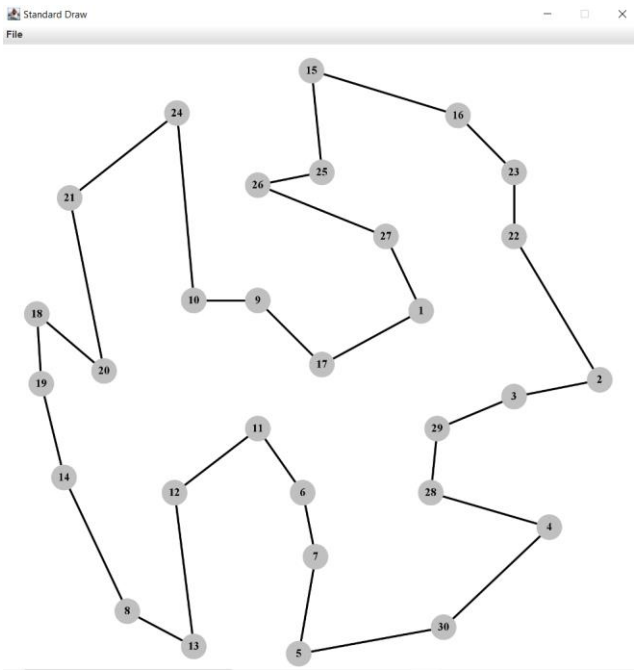
Method: Ant Colony Algorithm

Shortest Distance: 3,77015

Shortest Path: [1, 2, 3, 13, 10, 14, 9, 11, 12, 7, 5, 6, 4, 8, 1]

Time it takes to find the shortest path: 2.983 seconds

Input file 5:

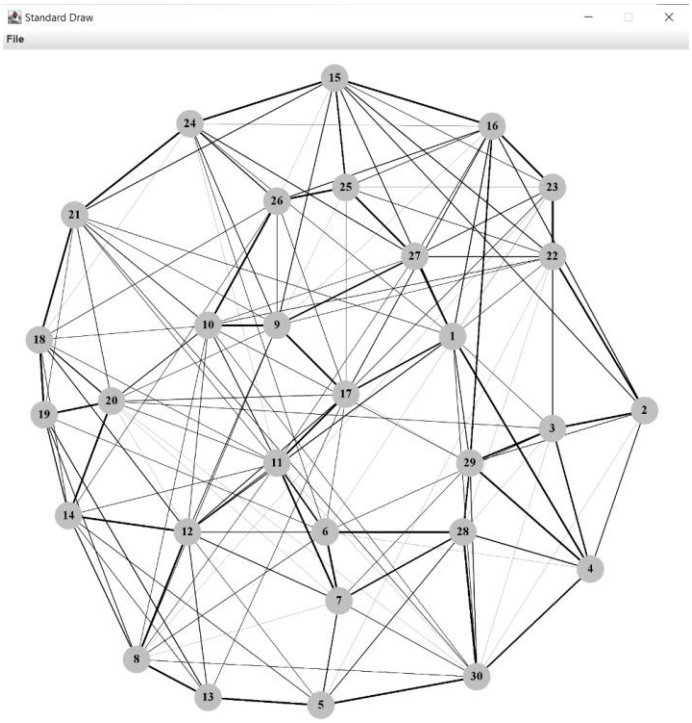


Method: Ant Colony Algorithm

Shortest Distance: 5,08847

Shortest Path: [1, 27, 25, 26, 15, 24, 21, 18, 20, 19, 14, 12, 8, 13, 5, 7, 6, 11, 10, 9, 17, 29, 3, 28, 30, 4, 2, 22, 23, 16, 1]

Time it takes to find the shortest path: 1.019 seconds



Method: Ant Colony Algorithm

Shortest Distance: 4,96418

Shortest Path: [1, 27, 9, 10, 20, 19, 18, 21, 24, 26, 25, 15, 16, 23, 22, 2, 3, 29, 28, 4, 30, 5, 13, 8, 14, 12, 11, 6, 7, 17, 1]

Time it takes to find the shortest path: 14.419 seconds

Best Ant Colony Hyperparameters:

Maximum iteration count = 100

Ant count per iteration = 50

Degradation factor = 0.5

Alpha = 0.8

Betha = 1.5

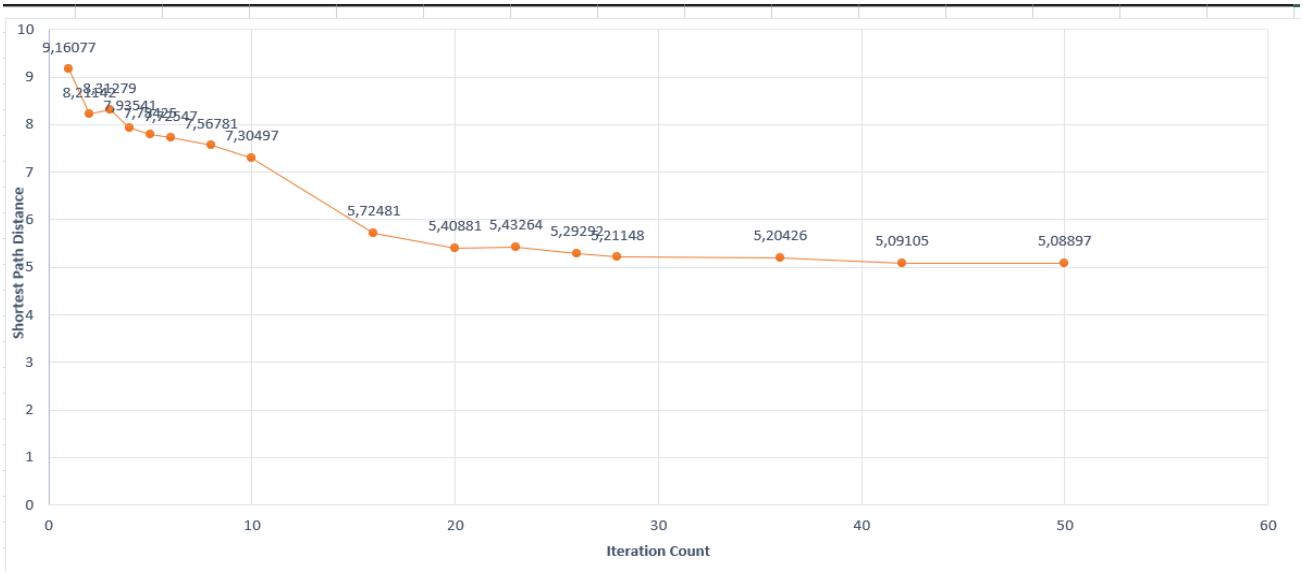
Initial Pheromone Intensity = 0.01

Q-Value = 0.0001

Comparison Of Methods:

Input File	Number of Houses and Migros	Brute-Force Time (seconds)	Ant Colony Time (seconds)	Speed Up Factor
Input1	11	0.598 (Distance :1,7952)	0.403 (Distance: 1,7952	1.48 times faster
Input2	12	1.967 (Distance:2,9358)	0.46 (Distance: 2,9358)	4.27 times faster
Input3	13	19.572 (Distance: 3,8029)	0.504 (Distance: 3,8029)	38,8 times faster
Input4	14	247.78 (Distance: 3,7109)	0.544 (Distance: 3,7371)	455,47 times faster
Input5	30	too long to compute	0.844 (Distance: 4,9815)	too many times faster

Shortest Distance vs Iteration Graph:



References of external resources:

[ant-colony-optimization · GitHub Topics · GitHub]

[Introduction to Ant Colony Optimization - GeeksforGeeks]

[Ant Colony Optimization | Baeldung]

Discussion About Advantages and Disadvantages of the Ant Colony Optimization Method:

Advantages of Ant Colony Method:

Ant colony method is much more efficient than brute force in larger problems. It uses a heuristic approach to find good solutions quickly through positive feedback and parallel computation.

Ant colony method is highly scalable in comparison to brute force. Suitable for distributed systems where each ant's search can be processed in parallel, reducing overall computation time as the problem scale increases.

Ant colony method is applicable to various problem types, including routing, scheduling, and resource allocation problems, where near-optimal solutions are acceptable.

Polynomial time complexity relative to the number of iterations and ants used, not directly tied to the factorial growth of the solution space.

Disadvantages of Ant Colony Method:

Ant colony method does not guarantee finding the optimal solution; it aims to find a good enough solution within a reasonable time frame.

While it scales better than brute force, it still requires significant computational resources for very large or complex networks, especially to maintain and update pheromone levels.

Ant colony method may require custom tuning of parameters for different types of problems, which can sometimes be complex to optimize effectively.

Although it is faster than brute force, still requires significant computation, especially as problem size and complexity grow.