IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 4.819
Volume 5, Issue 1, May 2021

# Comparative Analysis of Genetic Algorithm with Swarm Intelligence algorithms

**Ganga Devi D**
Assistant Professor, Department of Computer Applications
The Standard Fireworks Rajaratnam College for Women, Sivakasi, India

**Abstract:** *Traveling Salesman problem is the most recognized NP-hard problem for the researches in the field of computer science which focused on optimization. TSP finds the minimum travelling distance between given set of cities by traversing each of these cities only once except the starting city. This study discusses the salesman problem by presenting Particle Swarm Optimization Algorithm (PSO), Ant Colony Optimization algorithm (ACO) and Genetic Algorithm (GA) which intends to solve the problem. In this paper, these algorithms are applied on the benchmark TSP dataset Oliver 30. Also, the paper provides comparison between these algorithms and this comparison helps to choose the better algorithm.*

**Keywords:** Travelling Salesman Problem; NP-Hard; Particle Swarm Optimization Algorithm; Ant Colony Optimization algorithm; Genetic Algorithm; Oliver 30.

## I. Introduction

### 1.1 Meta-Heuristic Algorithms

Meta-heuristic optimization techniques have become very popular over the last two decades. Within the field of computer, Particle Swarm Optimization (PSO) [1], Ant Colony Optimization (ACO) [2], and Genetic Algorithm (GA) [3] are fairly well-known algorithms. Such optimization techniques have been applied in various inter-disciplinary fields of study. The meta-heuristics algorithms are common due to its main features as follows:

First, meta-heuristics are fairly simple because they have been inspired by the physical phenomena, animals' behaviours, or evolutionary concepts. This simplicity helps the computer scientists to simulate different natural concepts. This simplicity assists in learning meta-heuristics quickly and apply them in solving problems.

Second, without any special changes in the structure of the algorithm, meta-heuristics algorithms can be applied to different problems. Third, due to the derivation-free mechanisms of meta-heuristic algorithms, there is no need to calculate the derivative of search spaces to find the optimum. Thus, the meta-heuristics are highly suitable for real problems with unknown derivative information.

Finally, it has the superior abilities to avoid local optima because it allows them to avoid stagnation in local solutions. It may search the entire search space extensively. Thus the meta-heuristic algorithms are good choice for the real optimization problems.

### 1.2 Travelling Salesman Problem

Travelling salesman problem is a kind of an optimization problem which is used to find the shortest path of a given set of cities with the total distance travelled should be minimized. It should visit each city at least once and should return back to the starting city. Even though TSP is a simple problem, it should be a challenging problem too. Many AI algorithms such as Particle Swarm Optimization [4], adaptive bee colony [5], genetic algorithm (GA) [6], ant colony optimization (ACO) [7], simulated annealing [8] and Grey Wolf Optimizer Algorithm [9] have been explored to address the requirements of the TSP. These work gives the interest to researchers in exploring new algorithms or hybrid algorithms to attain best performance in terms of time and cost.

In this paper the Oliver 30 benchmark data set is used to solve the problem. Oliver30 is a commonly used benchmark for the Travelling Salesman Problem (TSP) [10]. The shortest cycle length is 423.741 when using the exact distance between each city. If each distance is rounded to the nearest integer, the shortest cycle length is 420.

## II. Methodology

### 2.1 Particle Swarm Optimization Algorithm

Particle swarm optimization [1] is a heuristic global optimization method put forward originally by Doctor Kennedy and Eberhart in 1995. In PSO, each single solution is like a 'bird' in the search space, which is called 'particle'. These particles are randomly initialized. All particles have its own fitness values to be optimized. These fitness values are calculated by the fitness functions. All particles also have velocities which direct the flying direction of the particles. During flight, each particle updates its own velocity and position based on the best experience of its own and the entire population. These updating values control the particle swarm to move toward the region with the higher objective function value, and finally all particles will gather around the point with the highest objective value.

In the PSO, the velocity and position of each particle change according the following equality. The detailed operation of particle swarm optimisation [1] is given below:

***Step 1****: Initialize Pi and Vi, and set pbest = Pi for i = 1, 2… s.*

***Step 2****: Evaluate each particle pbest for i = 1, 2… s.*

***Step 3****: Let gbest to be the best particle in {pbest1, pbest2,….. pbests}*

***Step 4:*** *For i = 1, 2… s. do:*

*Update Vi according to:*

*V =wV +c r (pbest -Y ) + c r (gbest-Y ) (4)*

*Update Pi according to:*

*Pi = Pi + Vi (5)*

***Step 5****: Go to Step 3, and repeat until convergence. Where w inertia weight factor; c1, c2 self-confidence factor and swarm-confidence factor, respectively; r1, r2 two random numbers uniformly distributed between 0 and 1. If Pi is better than pbest , then pbest =Pi*

***Step 6****: Go to Step 4, and repeat until convergence.*

### 2.2 Ant Colony Optimization Algorithm

Ant Colony Optimization (ACO) algorithms were introduced around 1990 [2]. These algorithms were inspired by the behaviour of ant colonies. Ants are a kind of social insects, being interested in the colony survival rather than individual survival. The ACO algorithm is inspired from ants' behaviour of finding the shortest path from their nest to food. When searching for food, ants initially explore the area surrounding their nest in a random manner.

While moving from one place to another, ants leave behind a chemical pheromone trail on the ground. Other ants are guided by these pheromone smell. Ants tend to choose the paths marked by the strongest pheromone concentration. Whenever an ant finds food, it evaluates the quantity and the quality of the food and carries some of it back to the nest. During the return trip, the quantity of pheromone that an ant leaves on the ground may depend on the quantity and quality of the food. The pheromone trails will guide other ants to the food source. The indirect communication between the ants via pheromone trails enables them to find shortest paths between their nest and food sources. Psuedocode for ACO model is given below:

```
Initialize
While stopping criterion not satisfied do
    Position each ant in a starting node
    Repeat
      For each ant do
          Choose next node by applying the state transition rule
          Apply step by step pheromone update
      End for
    Until every ant has built a solution
    Update best solution
    Apply offline pheromone update
```

*End while*

### 2.3 Genetic Algorithm

Genetic Algorithm was introduced by John Holland along with his colleagues and students in the mid - 1970s, at the University of Michigan[3]. The genetic are appropriately known as optimization technique which is totally based on natural evolution. It is based on the idea of survival of the fittest. GA randomly selects the solution and choose the fittest solution and create a new generation from the fittest solution which is better than the previous generations. GA method consists of the following sequences:

- **Initialization**: Random solutions are selected from given population to generate an initial population.
- **Fitness Function**: The solutions are evaluated using some fitness function. The fitness function generates fitness value for each of the individual solution. The function provides largest and smallest values for each of the individuals. If the individual has a larger fitness value then result will be a better solution but if a smaller value is obtained then solution obtained will not be better.
- **Selection**: Selection process is done based on the fitness value. Higher the fitness value,
- is higher the chance of selection. The individuals which have the highest fitness value are chosen to produce the offspring. Selection can be done using different selection techniques such as Steady state, Roulette Wheel, Rank selection and Tournament selection.
- **Crossover**: In crossover new generations are created by recombining the solutions chosen in the selection process.
- **Mutation**: Mutation is a simple search operator that is applied to genetic algorithm after crossover. Mutation is applied after crossover process is completed.
- **Termination**: This whole process continues until a stopping criterion is met or it executes certain number of iterations.

## III. EXPERIMENTAL ANALYSIS

The above algorithms are implemented to apply them to Oliver 30 TSP instances. The experimental results are presented in Table 1 and Table 2.

**Table 1:** Comparison between PSO, ACO & GA based on Execution Time

| Population Size \ Algorithm | PSO | ACO | GA |
|---|---|---|---|
| 100 | 5.1567 | 60.6411 | 0.5136 |
| 60 | 2.8526 | 40.5874 | 0.3937 |
| 50 | 2.3704 | 35.7102 | 0.3116 |
| 40 | 1.8392 | 31.7690 | 0.2801 |
| 30 | 1.8369 | 27.0660 | 0.2303 |
| 20 | 0.9097 | 23.0520 | 0.1967 |
| 10 | 0.5068 | 17.8616 | 0.1746 |

**Table 2:** Comparison between PSO, ACO & GA based on Total Distance Travelled

| Population Size \ Algorithm | PSO | ACO | GA |
|---|---|---|---|
| 100 | 625.1414 | 430.37 | 423.74 |
| 60 | 636.8850 | 435.60 | 423.74 |
| 50 | 649.3776 | 435.66 | 423.74 |
| 40 | 725.1110 | 430.94 | 423.91 |

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 4.819
Volume 5, Issue 1, May 2021

| | | | |
|---|---|---|---|
| 30 | 720.3168 | 434.47 | 423.9117 |
| 20 | 751.5029 | 431.58 | 424.46 |
| 10 | 915.4857 | 454.45 | 424.46 |

The Table1 concludes the best results obtained by solving TSP using Particle Swarm Optimization, Ant Colony Optimization and Genetic algorithm based on the Execution Time based on different population sizes. Whereas the Table2 concludes the best results based on the total distance travelled based on different population sizes.

The Figure 1, Figure 2 and Figure 3 depict the solving TSP using the heuristic algorithms PSO, ACO and GA respectively. The map represents the solution history of those algorithms.

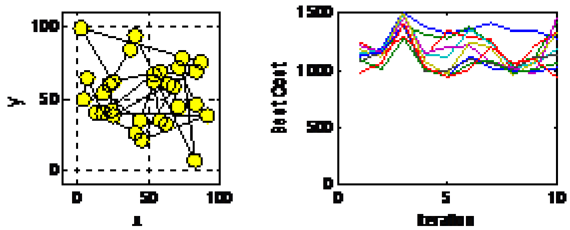


**Figure 1:** Solution using PSO

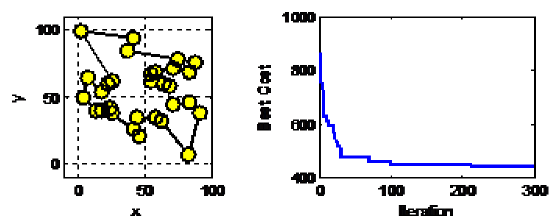


**Figure 2:** Solution using ACO

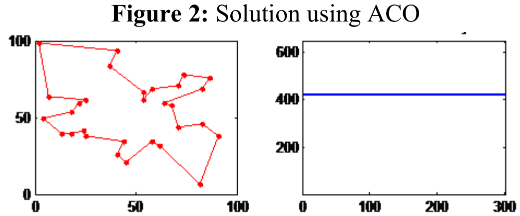

**Figure 3:** Solution using GA

## IV. Conclusion

The TSP can be solved using the heuristic algorithms PSO, ACO and GA. The result in the table depicts that the PSO algorithm doesn’t converge to the resultant value. The ACO algorithm obtains the convergence to some extent but it takes a lot of execution time on comparison. GA proves to give better result than other algorithms. It also seems that the convergence depends on the population size of the algorithm.

## References

**[1].** Kennedy, J. and R.C. Eberhart. 1995. “Particle Swarm Optimization”. Proceedings IEEE International Conference on Neural Networks, IV, p. 1942-1948.

**[2].** Dorigo M, Di Caro G, Gambardella LM, “Ant algorithms for discrete optimization”. Artificial Life, 5(2), 1999, 137-72.

**[3].** J.H.Holland, ”Genetic algorithms and the optimal allocation of trials”, SIAM Journal of Computing, Vol.2, no.2, 1973, pp. 88-105.

**[4].** Xuesong Yan, Can Zhang et al.”Solve travelling salesman problem using particle swarm optimization algorithm”, International Journal of computer science issues,Vol.9, issue 6,No.2, November 2012, ISSN:1694-0814.

**[5].** Ivan BrezinaJr.,ZuzanaCickova, “Solving the Travelling Salesman Problem using the Ant colony Optimization”, Management Information Systems, Volume 6, Issue 4,2011, pp. 10-14.

**[6].** Saloni Gupta, Poonam Panwar, “Solving Travelling Salesman Problem Using Genetic Algorithm”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 6, June 2013, pp. 376-380.

**[7].** Nishant Pathak, Sudhanshu Prakash Tiwari, “Travelling Salesman Problem Using Bee Colony With SPV”, International Journal of Soft Computing and Engineering, July 2012, ISSN: 2231-2307.

**[8].** Zicheng Wang, XiutangGeng, Zehui Shao,“An Effective Simulated Annealing Algorithm for Solving the Traveling Salesman Problem”, Journal of Computational and Theoretical Nanoscience, 2009.

**[9].** Ameen Shaheen, Azzam Sleit& Saleh Al – Sharaeh,” Travelling Salesman Problem Solution Based on Grey Wolf Algorithm over Hypercube Interconnection Network”, Modern Applied Science; Vol. 12, No. 8;2018, ISSN 1913-1844, E-ISSN 1913-1852, Published by Canadian Center of Science and Education.

**[10].** I. M. Oliver, D. J. Smith and J. R. C. Holland, “A study of permutation crossover operators on the travelling salesman problem,” in Proceedings of the Second International Conference on Genetic Algorithms on Genetic algorithms and their application, 1987, pp. 224-230.