Research on an Improved Ant Colony Algorithm Fusion with Genetic Algorithm for Route Planning

Harshith Jupuru, Indian Institute of Information Technology Sri City, Chittoor, AP, India
harshith.j19@iiits.in

Prem Kumar Reddy Nesanuru, Indian Institute of Information Technology Sri City, Chittoor, AP, India
premkumarreddy.n19@iiits.in

***Abstract*—This document is about the ant colony optimization algorithm which is combined with a genetic algorithm for improving performance and convergence rate. The genetic algorithm was used for initializing the pheromone levels of route nodes which will be used further by the ant colony optimization. The output taken by the genetic algorithm after doing selection, cross over, mutation, is considered as the prior information for the ant colony optimization algorithm. Compared to genetic algorithm or ant colony optimization alone, ant colony algorithm fusion with genetic algorithm gave better results when we checked for the traveling salesman problem with 10, 20, 40 nodes. Also proposed approach requires less number of iterations for convergence when compared with genetic algorithm or ant colony algorithm alone. The proposed improved algorithm shows much better performance than others when the interior nodes of the interesting region increase. As the improved algorithm needs fewer iterations so the calculation time also decreases.**

Keywords—route planning; genetic algorithm; ant colony algorithm; fusion algorithm

I. Introduction

For many e-commerce companies it requires an optimal way to find the route for dispatching and delivering goods/products. ALso for communication networks, transportation, robotics, artificial intelligence have dependencies with the route planning to solve other problems of above mentioned domains. There are many algorithms for solving route planning problems such as genetic algorithm, simulated annealing algorithm, Tabu search algorithm, immune algorithm, particle swarm algorithm etc. Also many researchers have proposed different combinations of above mentioned algorithms for solving the same. Now we focused on combining the ant colony algorithm with genetic algorithm. This improved algorithm only considers the distance between the locations/points. It won't consider the speed of objects moving from one point to another, weight of the load/payload if any and waiting or queue time. The next section i.e, section II explains about the mathematical model for route planning problem. Section III shows the simulation and results and finally section 4 describes the conclusion.

II. Methods

A. Traveling salesman problem

Let's consider V nodes which can be locations or any points and E is a set of edges between V vertices. C is the matrix with $C_{ij}>0$ as distance between nodes i,j. The aim is to reach all the places only once with the least distance.

B. Ant colony algorithm

This algorithm is inspired from the behavior of ants for their food search and other activities. Ants use the pheromone levels released by other ants for selecting their path to reach their destination. They use pheromones as their medium of communication. More number of ants move in a path, the more the pheromone levels in that path. Based on the positive feedback the shortest path is found. Also at the same time a table called taboo table is used for tracking the states which are already visited in order to prevent them from being visited again. So initially a set of m ants is taken as a population and each ant's path can be a solution. Those m ants can be started at any one of the n nodes. Two factors are involved in choosing the next node by the ants. One is pheromone level concentration in the path and other is a heuristic information which is the distance between the two nodes, the current node and the next possible node. We can initialize the pheromone levels with a constant value $\tau_{ij}(0) = c$ where c is constant and τ_{ij} is the pheromone level on the edge i,j. So the transition probability of moving from node i to node j is expressed as

$$P_{ij}^{k}(t) = \{(^{[\tau ij(t)]^{\alpha}.[\eta ij(t)]^{\beta}})/(\Sigma[\tau ij(t)]^{\alpha}.[\eta ij(t)]^{\beta}\ for\ s\in Jk(t))$$

when $j \in J_k(i)$ or else 0 otherwise.

Here we are giving α, β values which represents effect of pheromone levels on the edge and heuristic value respectively. Here we are calculating the probability of moving from i to j if j is not yet reached by ant k and for calculating so we are taking pheromone levels on edge i to j with power of α and heuristic value on edge i to j with the power of β and multiplying them. Then the result is divided by summation of such terms for all other nodes which are yet to reach by ant k. If ant k already reached node j then the probability of moving from node i to j is 0. When all nodes are reached by ant k then the taboo table for ant k contains all available nodes and it returns to the initial point of the traversal route. After all ants traverse all nodes, the pheromone levels on all routes gets updated using the formula:

$$\boldsymbol{\tau}_{ij}(t+n) = (1-\rho).\boldsymbol{\tau}_{ij} + \Delta\boldsymbol{\tau}_{ij}$$

where ρ lies between 0 and 1 ($0<\rho<1$) which indicates the degree of pheromone evaporation on route i,j and (1-ρ) represents the pheromone persistence coefficient, $\Delta\tau_{ij}$ represents the increment of pheromone on edge i,j in this iteration which can defined as

$\Delta\tau_{ij} = (\Sigma\Delta\tau_{ij}^{k}$ for k=1 to m) (i.e, for all ants) which means all the ants moving through i to j are taken and calculated the $\Delta\tau_{ij}^{k}$ value which is equal to the reciprocal to the length of path moved by ant k.

Here the ant cycle model uses global information to update pheromone on the traversal route.

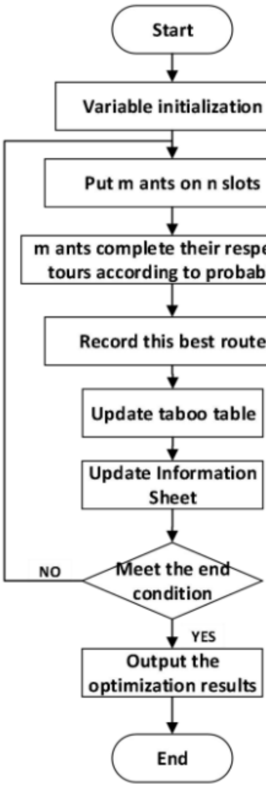

Fig.1. Flow chart represents the Ant Colony Optimization

C. Genetic algorithm

Genetic algorithm mimics the evolutionary mechanism of species in nature and uses corresponding simulation operators to implement genetic processes. It follows the rules of *natural selection, survival of the fittest* which are discussed in the book "The Origin of Species" by Charles Darwin. Genetic operations include selecting superior individuals, exchanging genes between individuals to generate new individuals, and mutations in individual genetic information to produce mutations in new individuals, accordingly improving the adaptability of offspring individuals. Initially based on fitness of population, select good individuals from one generation which is called as a mating pool to undergo the process of crossover in which the selected individuals are assigned randomly or using any heuristic to exchange the chromosome pairs between them with a set crossover probability value. Then all the individuals mutate with a set mutation probability. Mutation here means swapping of two nodes in the traversal path of the individual. For crossover there are many variants for the order representation method. The individuals which traversed all nodes and retained and those which fail to meet the requirements are eliminated. By this gradually optimizing the adaptability of new individuals in next generations can better meet the requirements of route planning problems.

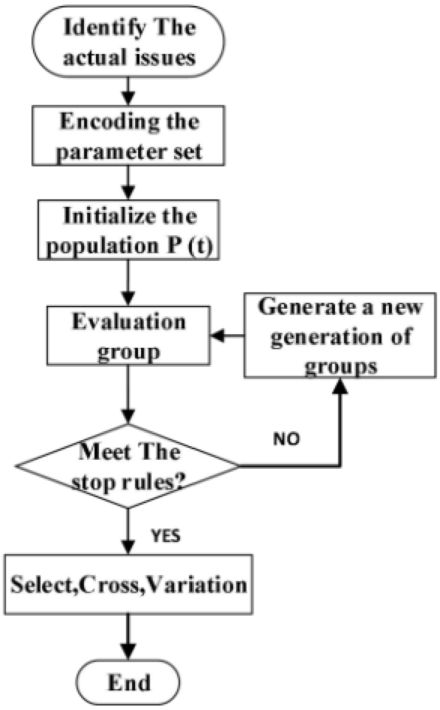

Fig.2. Flow chart represents the Genetic Algorithm

D. Elite Ant System

This elitism is an improvement on the basic ant colony system algorithm. In this algorithm the best solution in this case the best ant path is represented by T^{bs} and when this route modifies the pheromone trajectory it releases more pheromone to increase the effect of positive feedback. So the pheromone updation formula can be modified as follows:

$\Delta\tau_{ij} = (\Sigma(\Delta\tau_{ij}^{k} + e\Delta\tau_{ij}^{bs})$ for k=1 to m) (i.e, for all ants)

where e is a parameter which influences the effect of the best solution positive feedback. Here also we take

$\Delta\tau_{ij}^{bs} = 1/L_{bs}$ for $(i,j) \in T^{bs}$ or else 0

where L_{bs} is the traversal length of optimal path T_{bs}.

E. Ant colony algorithm fusion with GA

Genetic algorithm main use is better search process with simple genetic operators like selection, crossover, mutation etc. and Ant colony algorithm is used for optimization by simulation of biological characteristics. But traditional genetic algorithm disadvantage is poor local search ability and premature convergence. In both of the above algorithms the process moves parallelly and in ACO each ant is independent of another ant's path and because of global cumulative positive feedback for updation of pheromone the evolution process proceeds quickly. ACo is robust and requires few parameters to set. Even ant colony systems have a major role played by information rich routes in updation of pheromone which may lead to inadequate search time and stagnation.

Proposed Approach

So based on the advantages and disadvantages of both genetic algorithm and ant colony algorithm, it is better to use fusion of the above mentioned algorithms for better results. The genetic algorithm runs with random population and after it converges its evolution information is used for the initialization of pheromone value required by the ant colony optimization algorithm. The improved algorithm has three parts. One is the Genetic algorithm then the ant colony algorithm then uses additional release of pheromone to increase the positive feedback when best path edges pheromone levels are updating. Steps followed are as follows:

First part: Genetic algorithm

1. Initialize required parameters for genetic algorithm
2. Create the initial population. In TSP the population is random permutations of order of cities viewed.
3. Fitness calculation for each individual.
4. Selection of mating pool based on fitness values
5. Do crossover and mutation with set probabilities
6. New population generated
7. Stopping criteria met move to next step else go to step 3
8. Record current population and evaluate the best solution.
9. Convert population evaluation info to initial pheromone value required by ant colony algorithm.

Second part: Ant colony algorithm

1. Initialize the pheromone values on all edges of the path and calculate the distances between every node with every other node.
2. Calculate the transition probabilities
3. Determine the next node to move for each ant based on the transition probabilities and taboo table containing visited nodes.
4. Repeat the above two steps until all ants reach all the nodes exactly for one time i.e, generate a route to traverse all the cities.
5. Sort all routes with respect to the route length to find the shortest path and determine the optimal ant for this iteration.
6. Global update of pheromone levels of all the edges and also consider the additional pheromone release based on the optimal ant's path for this iteration.
7. Repeat the steps from 2 to 7 till we reach stopping criteria which is either number of iterations or the threshold set for the objective function or best solution appears with no betterment in it for continuous iterations.
8. Finally the fusion algorithm ended and we found the optimal solution.

III. Simulation and Results

The parameters set for the improved ant colony algorithm fusion with genetic algorithm for route planning are as follows:

(1) For ant colony algorithm, the population size is 50, the pheromone influencer α is taken 1, the pheromone evaporation coefficient is set as 0.1, maximum iterations are 200.
(2) For genetic algorithm the population size is 200, the maximum iterations or generations are 2000.
(3) Ant colony algorithm fusion with genetic algorithm gave better results than that of both individual genetic algorithm and ant colony algorithm. The fusion algorithm obtains the optimal solution with less number of iterations.

To get the objective value to 536.5, it took 1680 iterations for the genetic algorithm, 60 iterations for ant colony algorithm and ACO-GA algorithm's starting distance itself is less than 536.5. Following are the plots between the distance taken for traversal through all 40 nodes vs number of iterations. For each algorithm the plots are given as follows:

Following image shows plotting of all 40 cities (left) and graph showing fitness evolution of genetic algorithm (right). For 40 nodes, the shortest distance given by genetic algorithm to cover all cities exactly once is 536.53 in 1680 iterations. Similarly for 40 nodes, the shortest distance given by ACO is 519.0369 in 97 iterations whereas ACO-GA algorithm gave the same shortest distance in 78 iterations itself. For 20 nodes also ACO-GA algorithm did better than the other 2 individual algorithms. And ACO-GA is very useful when the nodes are larger. So we can consider ACO-GA is better than the other 2 algorithms. We can compare the 3 algorithms for 20 and 40 nodes in the below given tables and the path given by each algorithm are plotted along with its fitness evolution in finding the best path.

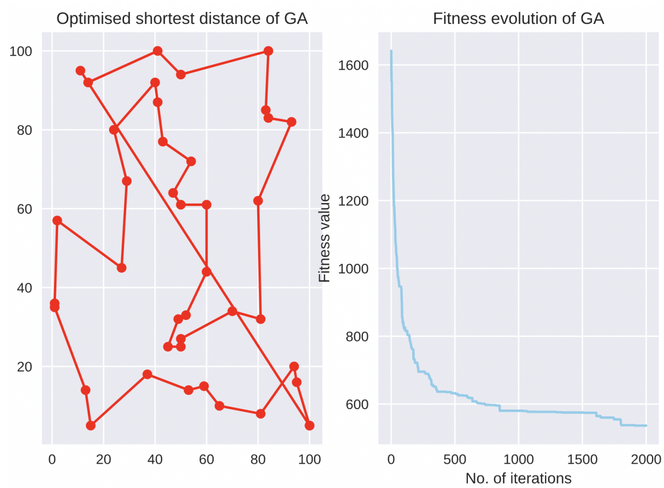

Fig.3. The distance and the fitness evolution of GA

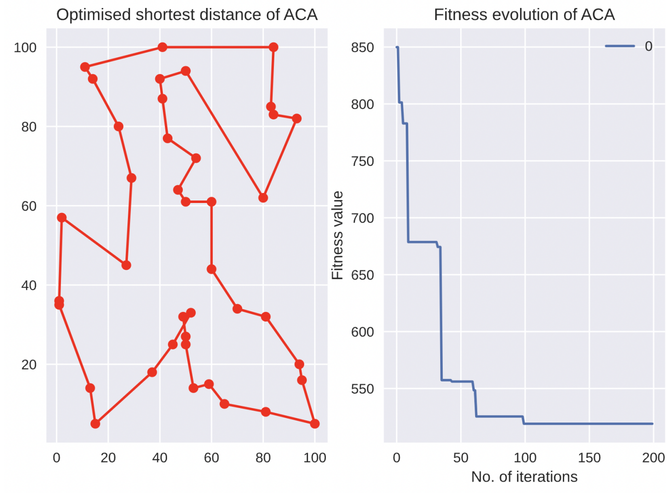

Fig.4. The distance and the fitness evolution of ACO

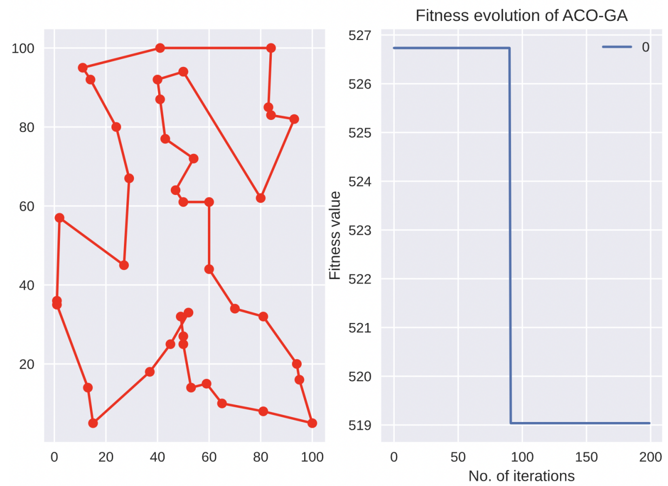

Fig.5. The distance and the fitness evolution of ACO-GA

We can compare different algorithms for constant distance length with different number of nodes/locations as follows:

TABLE I. THE ROUTE DISTANCE OF ALGORITHMS SIMULATES

Distances	20 nodes	40 nodes
GA	381.0763157215	536.5307
ACO	357.2168851867	519.0369
ACO-GA	357.2168851867	519.0369

Tab.II THE ITERATIONS OF ALGORITHMS

Iterations	20 nodes	40 nodes
GA	780	1680
ACO	50	97
ACO-GA	33	78

Based on the results we can say that ACO-GA (ant colony optimization fusion with genetic algorithm) converges very quickly when compared with only GA or only ACO as the number of nodes increases. Also ACO-GA algorithm found optimized route distances and shorter paths for randomly generated locations and also with less number of iterations.

Conclusion

The aim of the project is to find a better way for route optimization to cover all the nodes exactly once which we can refer to as traveling salesman problem. To escape from falling into local optimum, low efficiency, we improves the way of pheromone update in an ant colony algorithm by proposing a fusion of ant colony algorithm with genetic algorithm in which the pheromone initialization is done in a much better way using the population evolution information obtained from genetic algorithm for faster optimization. The experimental results indicated that the ACO-GA algorithm obtains the shorter route distance with the interior nodes of the interesting region increases when compared with GA and ACO algorithms both. Moreover, the fusion algorithm obtains an optimal solution with fewer iterations and hence the calculation time cost also decreases.

References

[1] Chen, X., & Dai, Y. (2020, June). Research on an improved ant colony algorithm fusion with genetic algorithm for route planning. In 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC) (Vol. 1, pp. 1273-1278). IEEE.

[2] Lecture slides by Dr. R. Shathanaa mam in SCEAI course.