International Journal of Science and Research (IJSR)
ISSN: 2319-7064
ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426

Ant Colony Optimization for City Public Transport Route Design in Tanzania

Isaack Adidas Kamanga

Dar es Salaam Institute of Technology (DIT), Dar es Salaam, Tanzania
[1,2]Electronic and Telecommunications Engineering Department

Abstract: ***In this paper, I study and analyze a possibility of solving the problem of designing the most appropriate routes for public city buses optimizing both transport resources and QoS by covering as many areas as possible using as little number of buses as possible to overall reduce the unnecessary traffic jams while serving many users. I look to achieve this by employing Ant Colony Optimization (ACO), which belongs to the group of evolutionary techniques and I treat the problem as the well-known Travelling Salesman Problem (TSP), in this work, the nodes are assumed to be the bus stops and the connection of these nodes are the roads used by public buses. I also study the impact of some control parameters on the problem at hand by implementing ACO algorithm, To achieve optimal routes, the amount of virtual pheromones that shall attract the "Ants" which are the buses in this study, is the number of passengers at a particular location, a node with many pheromone attracts the buses, simulation is carried using 11, 101, and 201 and the results are compared. The quality of the solution is compared with the optimal solution. In this work I used Eil51 and Eil76 datasets to test the algorithm and compare with the known best results, the observations are recorded in the table under experiments and results discussion, the obtained results appeared to have deviations less than 6% to the best known solution. The simulation tool used in this work is MATLAB. I target to waken the debate on the implementation of city bus route in Tanzania because transportation policies and huge financial investments for road infrastructures and public bus system seem insufficient to solve the issue of urban mobility and its sustainability.***

Keywords: Bus stop, Public Transport, travelling salesman problem, SUMATRA, ant colony optimization (ACO), pheromone

1. Introduction

In Tanzania the city bus routes are designed by **SUMATRA** which stands for Surface and Marine Transport Regulatory Authority, by determining the number of residence at the particular area and the number of buses assigned to a nearby bus stops. This manual approach could result into many buses assigned to an area unnecessary while there could be a possibility to redesign the route by just adding the new areas to a system with the same number of buses. Travelling salesman problem (TSP) consists of finding the shortest route in complete weighted graph G with n nodes and n (n-1) edges; so that the start node and the end node are identical and all other nodes in this tour are visited exactly once as the "Ant" travels from the starting bus stop to an end bus stop. The most popular practical application of TSP are: regular distribution of goods or resources, finding of the shortest of costumer servicing route, planning bus lines etc., but also in the areas that have nothing to do with travel routes. I will also mention some of them as an illustration: The natural Ants follow the **pheromone** which is a chemical substance produced and released into the environment by them, affecting the behavior or physiology of others ants, and normally the routes with many pheromones is given a priority indicating that many ants have passed that route, for the case of bus route. In this work a bus stop is defined as a node where the bus has to stop an pick up the passengers.

2. Problem Statement

Let C be the matrix of shortest distances (dimension NxM), where n is the number of nodes of graph G. The elements of matrix C represents the shortest distances between all pairs of nodes (bus stops for this case) (i, j), i, j=1, 2, …,n. The travelling salesman problem can be formulated in the category programming binary, where variables are equal to 0 or 1, depending on the fact whether the route from bus stop i to bus stop j is realized (x_{ij}= 1) or not (xit= 0). Then, the mathematical formulation of TSP [2] is as follows (the idea of this formulation is to assign the numbers 1 through n to the nodes with the extra variables τij, so that this numbering corresponds to the order of the nodes in the route.

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \quad (1)$$

Whereby for true values of i and j

$$\sum_{i=1}^{n} x_{ij} = 1, j = 1, 2, \ldots n (i \neq j) \quad (2)$$

$$\sum_{j=1}^{n} x_{ij} = 1, i = 1, 2, \ldots n (i \neq j) \quad (3)$$

$$u_i - u_j + n x_{ij} \leq n - 1 (i, j = 2, 3 \ldots n, i \neq j) \quad (4)$$

$$x_{ij} \in \{0, 1\} i, j = 1, 2, \ldots n (i \neq j) \quad (5)$$

3. Employing the ACO in solving Travelling Salesman related Problems (TSP)

The Travelling Salesman Problem is one of the best known NP-hard problems, which means that there is no exact algorithm to solve it in polynomial time. The minimal expected time to obtain optimal solution is exponential and is never fixed. So, for that reason, we usually use heuristics to help us to obtain a so called best known solution which is used as a reference in this study. Many algorithms were applied to solve TSP with more or less success. There are various ways to classify algorithms, each with its own merits.

The basic characteristic is the ability to reach optimal solution: exact algorithms or heuristics. The best known exact methods for solving TSP are: explicit enumeration, implicit enumeration, branch and bound method, cutting plane method and dynamic programming. These methods work well only for solving the problems with no more than 40-80 nodes (we suppose the use of one computer). For the practical relevance, it is necessary to solve the larger-scale problems with the help of heuristics. Heuristic methods vary from exacts methods in that they give no guarantee to find the optimal solution to the given problem (so that solution is called suboptimal), but in many cases this is the solution of good quality and we can obtain it in acceptable time. Heuristic methods are usually focused on solving the special type of problems. The significant part of heuristics comprises met heuristic methods, which differs from the classical methods in that they combined the stochastic and deterministic composition. It means that they are focused on global optimization, not only for local extremes. The big advantage of met heuristics is that they are built not only for solving a concrete type of problem, but they describe general algorithm in that, they show only the way, how to apply some procedures to become solution of the problem. This procedure is defined only descriptively, by black-box, and the implementation depends from the specific type of problem. The group of the most known met heuristics includes evolutionary algorithms, which are inspired by process in nature (for example genetic algorithms, particle swarm optimization, differential evolution, ant colony optimization, etc.).

4. Ant Colony Optimization for Solving the Travelling Salesman Problem

Ant colony optimization (ACO) belongs to the group of met heuristic methods. The idea was published in the early 90s for the first time. The base of ACO is to simulate the real behavior of ants in nature. The functioning of an ant colony provides indirect communication with the help pheromones, which ants excrete. Pheromones are chemical substances which attract other ants searching for food. The attractiveness of a given path depends on the quantity of pheromones that the ant feels. Pheromones excretion is governed by some rules and has not always the same intensity. The quantity of pheromones depends on the attractiveness of the route. The use of more attractive route ensures that the ant exudes more pheromones on its way back and so that path is more also attractive for other ants. The important characteristic of pheromones is evaporation. This process depends on the time. When the way is no longer used, pheromones are more evaporated and the ants begin to use other paths. What is important for ACO algorithm the moving of ants. This motion is not deterministic, but it has stochastic character, so the ants can find the path, which is firstly unfavorable, but which is ultimately preferable for food search. The important characteristic is that a few individuals continuously use non-preferred path and look for another best way. ACO was formulated based on experiments with double path model, where the quantification was made similar to Monte Carlo method. The base of this simulation was two artificial connections between the anthill and a food source. The simulation demonstrated that ants are able to find the shorter these two paths. A significant impact of this simulation was to quantify the behavior of ants. For practical use of ACO, it was necessary to project virtual ants. It was important to set their properties. These properties help virtual ants to scan the graph and find the shortest tour. Virtual ants do not move continuously; they move in jumps, which mean that, after a time unit, they will always be in another graph node. The absolved path is saved in ant memory. The created cycles are detected in ant memory. In the next tour, the ant decides on the base of pheromones power. Just because the property of pheromone evaporation, pheromones on shortest edges are stronger, because of the fact that the ant goes across these edges faster. Based on these facts we can mathematically describe the behavior of the virtual ants (Onwubolu & Babu, 2004, p. 712):

4.1. Establishing the public bus routes in the cities in Tanzania

In the Ant Colony Optimization algorithm the agents (ants) are placed on different nodes (usually it is used a number of ants equal to the number of nodes). And, the probability of choosing the next node (bus stop) is based on agents chose the next node using the equation known as the transition rule that represents the probability for ant k to go from stop i to stop j on the tth tour (equation no 6).

$$P_{ij}^{k}(t) = \frac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum_{l \in J_i^k}[\tau_{il}(t)]^{\alpha}.[\eta_{il}]^{\beta}} \tag{6}$$

In the equation, τij represents the pheromone trail and ηij the visibility between the two bus stops, while α and β are adjustable parameters that control the relative weight of trail intensity and visibility.

At the beginning there is the same value of pheromone in all the edges. Based on the transition rule, which, in turn, is based on the pheromone and the visibility (distance between nodes), some paths will be more likely to be chosen than others. When the algorithm starts to run each agent (ant) performs a tour (visits each node), the best tour found until the moment will be updated with a new quantity of pheromone, which will make that tour more probably to be chosen next time by the ants. Number of public buses moving in the registered route by SUMATRA will thus depends on the amount of pheromone on the graph edges of the nodes which are the bus stop, a research on the amount of passengers at each bus stop is a significant for this case. Now let us assume the ant is the bus, the probability pi k of transition of a virtual ant from the node i to the node k is given by formula (7). We assume the existence of internal ant's memory.

$$P_i^k = \frac{\tau_i^{\alpha} + \eta_i^{\beta}}{\sum(\tau_{N^i}^{\alpha} + \eta_{N^i}^{\beta})} \tag{7}$$

Where by

$\tau_{N^i}^{\alpha}$ Indicates the attractiveness of transition in the past

$\eta_{N^i}^{\beta}$ Adds to transition attractiveness for ants,

N^i Set of nodes connected to point i i without the last visited point before point i

α, β Adjustable parameters that control the relative weight of trail intensity and visibility

The real ants display the behavior enlightened in figure 1 below, A and B shows the movements of the ants from the nest to food source and back to the nest when there is no obstacle for our case it is how the buses flow through the freshly registered route and back to starting bus stand, C and D displays how the ants behaves when the obstacle introduced to the system, analog to the new registered bus stop which is not along the registered route.

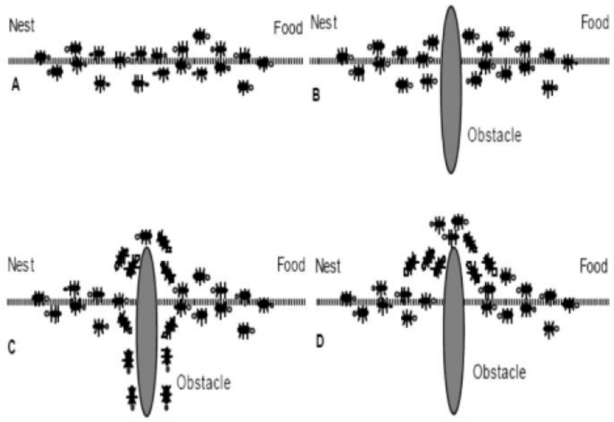

Figure 1: How real ants behaves in their movements

a) Real ants follow a path between nest and food Source, An obstacle appears on the path: Ants choose
b) Whether to turn left or right with equal probability, Pheromone is deposited more quickly on the
c) Shorter path, All ants have chosen the shorter path.

4.2. Reverse

Virtual ant is using the same reverse path as the path to the food resource based on his internal memory, but in opposite order and without cycles, which are eliminated. After elimination of the cycles, the ant puts the pheromone on the edges of reverse path according to formula (7).

$$\tau_{ij}^{t+1} = \tau_{ij}^{t} + \Delta\tau^{t} \qquad (7)$$

Where by

τ_{ij}^{t} Value of pheromone in step t,

$\Delta\tau$ Value of ants saved pheromones in step t,

Value $\Delta\tau$ can be constant or they can be changed depending on solution quality.

4.3. Evaporation of Pheromones

At last, the pheromones on the edges are evaporated. The evaporation helps to find the shortest path and provide that no other path will be assessed as the shortest. This evaporation of pheromones has an intensity ρ (8).

$$_{ij}^{t+1} = (1-\rho)\tau_{ij}^{t} \qquad (8)$$

This formula is applied on all graph edges with intensity ρ (interval (0, 1)). On this knowledge we can compose an algorithm of ACO, which can be used for solving the travelling salesman problem. It is necessary to keep information about quantity of pheromones τij in memory, which has stochastic character and actually represents state of graph scan. Further on, there is a need to memorize the edge costs, or information derived from this information (ηij). Information of pheromones value τij is changing during the simulation, but values of ηij stay the same during the calculation. Virtual ants use this information during their moving across the graph. On the basis of these considerations, we can describe the ACO algorithm as system of steps (Chu, 2009, p. 107):

1) Ants scan graph G. The aim of this scan is to find an optimal solution
2) Every ant has its own memory, which is used for saving information about travelled path (for example about travelled nodes). This memory can also serve to ensure constraints or to evaluate of the solution.
3) The process begins in state xs k and has one or more ending constraints ek. Let the actual state of an ant be the state xr = (xr–1, i) and no ending constraint is complied, so the ant moves to node j in neighborhood of the state Nk(xr) and the ant moves to the new state (xr, j) X. In case that some ending constraint is complied with, the ant ends with process of scan. The transition to a state that represents unacceptable solution is usually banned by appropriate implementation of internal ant memory.
4) The next ant motion depends on the probability, which is calculated on the base of pheromone quantity on edges of graph, and it also takes into consideration its local memory and the acceptance of this step.
5) If the ant can to add new component of graph GC, it can update the value of corresponding pheromone information (information is bound with corresponding edge, or aim node).
6) The ant can update pheromone values after reverse path construction by editing associate pheromone values.

5. Experiments and Results Discussion

The moving of ants provides the parallel and independent search of the route with the help of dynamical change of pheromone trail. The ant represents an elementary unit with the ability to learn, and due to collective-cooperative work with other members of population, it is able to find acceptable solution to the given problem. For experiment, we used the problem of 32 cities in Slovakia. We were able to get an optimal solution to that problem with the help of GAMS (Solver Cplex, 17498 iterations, and optimal length of route 1453 km). Secondly, we try to solve that problem using ACO algorithm (6 functions in Matlab _information, Ants_primaryplacing, Ants_cycle, Ants_cost, Ants_traceupdatin and script Main (MATLAB, 2012). Chu (2009) recommends parameters α= 1, β=5. The number of iterations was set to 1000 (iter=1000) simultaneously with the changing number of ants (m = 100, 1000, 5000, 10000) scanning the graph the number of ants (m) were changed as well. With the number of ants equal to 100 (m=100), the algorithm finds the tour with length k=1713, that differs from optimal solution over than 17%. The searching process is shown in Figure 1.

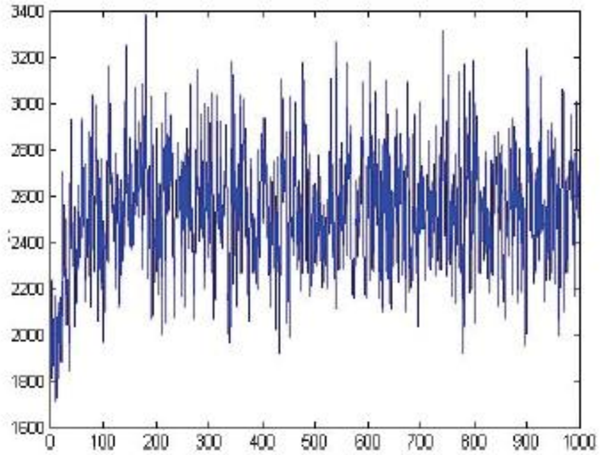

Figure 1: Simulation using 11 nodes

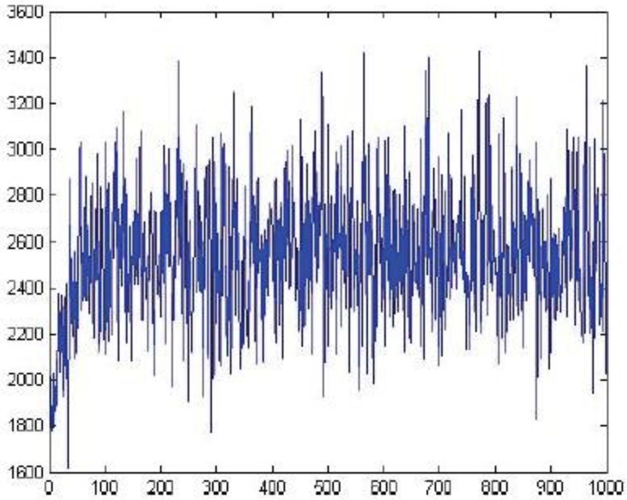

Figure 2: Simulation using 101 nodes

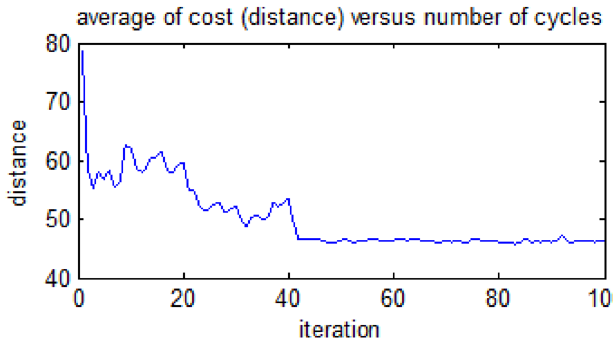

Figure 3: Average route cost when nodes was set to 101

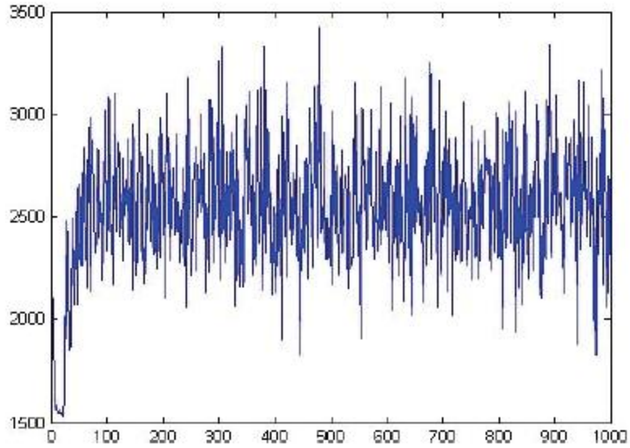

Figure 4: Simulation using 201 nodes

Subsequent simulations were realized with m=1000. The result was the tour with length k=1621 in 34th iteration (difference 11.56% from optimal route). The running of search is shown in Figure 2.

Further on, we set the number of ants to 5000 (m=5000). Algorithm ACO finds the tour with length k=1532 in 21st iteration (difference 5.44% from optimal route).

At last, the number of ants was set to 10000 (m=10000). Algorithm ACO find the tour with length k=1465 in 242nd iteration, (deviation from optimal solution was 0.83%).

Based on the experiments, it can be concluded that the quality of solutions depends on the number of ants. The lower number of ants allows the individual to change the path much faster. The higher number of ants in population causes the higher accumulation of pheromone on edges, and thus an individual keeps the path with higher concentration of pheromone with a high probability. The final result differs from optimal solution by 12 km (deviation is less than 0-9 %). The great advantage over the use of exact methods is that ACO algorithm provides relatively good results by a comparatively low number of iterations, and is therefore able to find an acceptable solution in a comparatively short time, so it is useable for solving city public bus route problems.

Table 1: Comparison of ACO on route design to other methods on two datasets using 101 nodes

DataSets	Method 1 TS	Method 2 TSA	Method 3 GA	Best known	Results obtained
Eil51	206	207	207	212	207
Eil76	210	208	209	213	209

Table Keys: TS -Tabu search
ATS -Adaptive Tabu Search
GA -Generic Algorithm

6. Conclusion

My results are found to be competitive to other methods although they do not produce any best known solution. I observed for Eil51 and Eil76 dataset, the result is better than simulated annealing (SA) and annealing-genetic (AG) algorithm. Therefore, I conclude that ACO hyper-heuristics are comparable to other problem specific methods.

7. Acknowledgement

I would like to express my gratitude to my master's degree supervisor professor Dening Jiang who even though is far from dare s Salaam, provided assistant during data analysis and simulation of the results.

References

[1] Bell, J.E., McMullen, P.R., 2004. Ant colony optimization techniques for the vehicle routing problem. Advanced Engineering Informatics 1 (8), 41–48
[2] Brezina, I. (2003). Kvantitatívne metódy v logistike. Bratislava: Ekonóm.
[3] Chu, A. (2009). Metaheuristická metóda mravčej kolónie pri riešení kombinatorických optimalizačných úloh. Praha: Vysoká škola ekonomická v Praze.
[4] MATLAB. (2007, May 21). Solving tsp with ant colony system. Retrieved December 8, 2012, from The Math Works
[5] Onwubolu, G. C., & Babu, B. V. (2004). New Optimization Techniques in Engineering. Berlin-Heidelberg: Springer-Verlag
[6] Gambardella, L., Taillard, E., Dorigo, M., 1997. Ant Colonies for the QAP, Technical Report 97-4, IDSIA, Lugano, Switzerland

Author Profile

Isaack adidas Kamanga received the B.S.in Telecommunications Engineering from University of Dar Es Salaam in 2012, Masters of signal and information processing engineering from Tianjin University of Technology and Education, P.R China in 2018, with research base in computer vision, worked as Engineer in fields of RF, VSAT, Optic fiber network for, currently working as Assistant Lecturer and Consultant at Dar Es Salaam Institute of Technology (DIT), Tanzania.