An Overview of Minimum Shortest Path Finding System Using Ant Colony Algorithm

Nikita G. Mugal
Final year
Department of Information,
Technology,Yeshwantrao Chavan College Of Engineering ,
Nagpur ,India.

Priyanka A. Solanke
Final year
Department of Information Technology,Yeshwantrao
Chavan College Of Engineering ,
Nagpur,India.

Rinku Dhadse
Final year
Department of Information Technology,Yeshwantrao
Chavan College Of Engineering ,
Nagpur ,India.

Roshani Chandekar
Final year
Department of Information Technology,Yeshwantrao
Chavan College Of Engineering ,
Nagpur ,India.

Priyanka Jaiswal
Department of Information Technology,
Yeshwantrao Chavan College Of Engineering ,
Nagpur ,India.

***Abstract*— This paper proposes a stochastic optimization method, based on Ant Colony Optimization (ACO), for the optimal choice of transformer sizes to be installed in a distribution network. In nature, ants use pheromones to communicate with each other, allowing relatively simple behaviors performed by single ants to combine and result in complicated actions exhibited by the colony as a whole. The emergence of these complicated behaviors has been applied to the computer science world, and this project aims to make these principles available to both novice and experienced users through the creation of a usable interface for interacting with an ant colony simulation environment. Path finding involves solving a planning problem with agents seeking optimal paths from a start state to a goal state. The path finding process involves utilizing the full state space information available to agents to find the least expensive route to the goal. To solve optimization problems in path finding System, many methods have been proposed. One of these methods is to mimic ant's behavior in path finding problem, formally named as Ant Colony Optimization (ACO) algorithm. This paper provide us shortest path along with other paths, from that user may select shortest path or any optimal path which is beneficial to reach particular destination and also provide the shortest path along with the transportation system. Path Finding System provides us the map which shows the shortest path form source to destination.**

Keywords— ***Multi-path, optimality, path planning***

I. INTRODUCTION

Optimization is one of the most important tasks of engineers, which the engineer asked to design new, better, more efficient and less expensive systems as well as to devise plans and procedures for the improved operation of existing systems in both industrial and the scientific world. Many methods and algorithm have been proposed to solve the problem of path finding, In this paper, system proposes a refined approach of ant colony algorithm and applies it to the optimal path finding problem. Path finding involves solving a planning problem with agents seeking optimal paths from a start state to a goal state. Path finding is a system which is used to find the shortest path between routes. These Systems help us to find the shortest path and to reduce our time for reaching to the destination. Ant Colony Optimization (ACO) is a population-based approach for solving combinatorial optimization problems that is inspired by the Foraging behavior of ants and their inherent ability to the shortest path from a food source to their nest ACO is the result of research on computational intelligence approaches to combinatorial optimization originally conducted by Dr. Marco Dorigo,in collaboration with Alberto Colony and Vittorio Manie. In this paper, we build up a new multi-colonies ant algorithm, called General Multiple Ant Colonies Algorithm (GMAC). In this new approach, we apply principles of ACO algorithm such as Rank based Ant Colony,Best- worst ants and Elitist ants system to both sub-colonies and great colony. Purposes of the combination are to take advantages of each algorithm's properties to reduce processing time as well as increase performance of optimal path finding. Some comparisons among GMAC and an improved ACO are implemented to show the dominance of the new algorithm over the single-ACO algorithm.

II. LITERATURE SURVEY

1.Relationship between Genetic Algorithms and Ant Colony Optimization Algorithms.

Authors- Osvaldo G´omez Universidad Nacional de Asunci :- Genetic Algorithms (GA) have been used to evolve computer programs for specific tasks, and to design other computational structures. The recent resurgence of interest in AP with GA has been spurred by the work on Genetic Programming (GP). GP paradigm provides a way to do program induction by searching the space of possible computer programs for an individual computer program that is highly fit in solving or approximately solving the problem at hand. The genetic programming paradigm permits the evolution of computer programs which can perform alternative computations conditioned on the outcome of intermediate calculations, which can perform computations on variables of many different types, which can perform iterations and recursions to achieve the desired result, which can define and subsequently use computed values and subprograms, and whose size, shape, and complexity is not specified in advance.

2. Heuristic shortest path algorithms for transportation applications.

Authors- L. Fua,∗ , D. Sunb, L.R. Rilettc :- In recent years there has been a resurgence of interest in the shortest path problem for use in various transportation engineering applications. This is directly attributed to the recent developments in Intelligent Transportation Systems (ITS), particularly in the field of in-vehicle Route Guidance System (RGS) and real time Automated Vehicle Dispatching System (AVDS) where there is a definite need to find the shortest paths from an origin to a destination in a quick and accurate manner. In a distributed RGS, an in-vehicle computer is commonly used to calculate the optimal route in a large traffic network. This paper first provides a brief overview of the optimal shortest path algorithms, which are often considered as the starting point for the development of many heuristic shortest path algorithms. The paper then examines four heuristic search strategies: (i) limit the area searched, (ii) decompose the search problem, (iii) limit the links searched, and (iv) some combination of above. The review summarizes the distinguishing idea of each search strategy and its algorithmic variations.

III. PROBLEM DEFINITION

Now a day all the users use the internet for searching the path between particular source and destination. This sites shows only one path between selected source and destination, But how user can believe those path show by the site is shortest? In many cases information provided by the site which related to particular path is not efficient and it takes so much time to understand by the user. So in that case user can fail to perform their operation. To overcome this problem we develop a site which helps to such users. In our system we provide a shortest path between the routes as well as the other path to reach out that particular destination. The basic idea to develop system is come from the ant colony algorithm. This concept helps us to make this site more user friendly and less time consuming. This system provides the transportation facility to users, according to requirement of user shortest path will display on map with their required transportation cost. The concept of finding path is also used in GPS system, but GPS system only shows Route between sources to destination, GPS system not provide various routes related Source and Destination. Some time it very difficult to understand navigation map which provide by the GPS system. There are many websites available which help to find out the shortest path but, that websites only shows one path for particular source to destination. The present system fail to give the shortest path it also take too much time for performing search operations and give the path.

Flaws in Present Working system**:-**

1. The navigation instruction provided by this system is very difficult to understand for the user.
2. This system only shows single path of a particular source and destination. Sometimes that path takes too much time to reach out the destination.
3. It is very time consuming system.
4. This site is not user friendly.

IV. OPTIMAL PATH PLANNING PROBLEM

A. Edge selection

An ant is a simple computational agent in the ant colony optimization algorithm. It iteratively constructs a solution for the problem at hand. The intermediate solutions are referred to as solution states. At each iteration of the algorithm, each ant moves from a state $x$to state y, corresponding to a more complete intermediate solution. Thus, each ant $k$computes a set $A_k(x)$of feasible expansions to its current state in each iteration, and moves to one of these in probability. For ant k, the probability $p_{xy}^k$of moving from state $x$to state $y$depends on the combination of two values, viz., the *attractiveness* $\eta_{xy}$of the move, as computed by some heuristic indicating the *a priori* desirability of that move and the *trail level* $\tau_{xy}$of the move, indicating how proficient it has been in the past to make that particular move.

The *trail level* represents a posteriori indication of the desirability of that move. Trails are updated usually when all ants have completed their solution, increasing or decreasing the level of trails corresponding to moves that were part of "good" or "bad" solutions, respectively.

In general, the kth ant moves from state $x$to state $y$with probability.

$$p_{xy}^k = \frac{(\tau_{xy}^{\alpha})(\eta_{xy}^{\beta})}{\sum_{y\in \mathrm{allowed}_y}(\tau_{xy}^{\alpha})(\eta_{xy}^{\beta})}$$

where

T_{xy} is the amount of pheromone deposited for transition from state $x$to y, 0 ≤ $\alpha$is a parameter to control the influence of τ_{xy}, $\eta_{xy}$is the desirability of state transition xy(*a priori* knowledge, typically $1/d_{xy}$, where $d$is the distance) and β≥ 1 is a parameter to control the influence of η_{xy}. $\tau_{xy}$and η_{xy}represent the attractiveness and trail level for the other possible state transitions.

B. Pheromone update

When all the ants have completed a solution, the trails are updated by

$$\tau_{xy} \leftarrow (1-\rho)\tau_{xy} + \sum_k \Delta\tau^k_{xy}$$

where τ_{xy} is the amount of pheromone deposited for a state transition xy, ρ is the *pheromone evaporation coefficient* and $\Delta\tau^k_{xy}$ is the amount of pheromone deposited by kth ant, typically given for a TSP problem (with moves corresponding to arcs of the graph) by

$$\Delta\tau^k_{xy} = \begin{cases} Q/L_k & \text{if ant } k \text{ uses curve } xy \text{ in its tour} \\ 0 & \text{otherwise} \end{cases}$$

where L_k is the cost of the kth ant's tour (typical length) and Q is a constant.

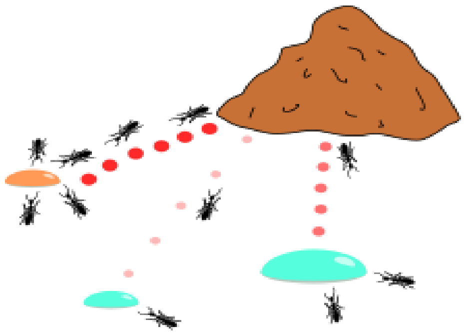

Figure 1:Ant travel for searching a food

The first ACO algorithm was called the Ant system and it was aimed to solve the travelling salesman problem, in which the goal is to find the shortest round-trip to link a series of cities. The general algorithm is relatively simple and based on a set of ants, each making one of the possible round-trips along the cities. Fig(2) shows rules to update pheromone or move to other point. At each stage, the ant chooses to move from one city to another according to some rules:

1. It must visit each city exactly once;

A distant city has less chance of being chosen (the visibility);

2. The more intense the pheromone trail laid out on an edge between two cities, the greater the probability that edge will be chosen;
3. Having completed its journey, the ant deposits more pheromones on all edges it traversed, if the journey is short;
4. After each iteration, trails of pheromones evaporate.
5.

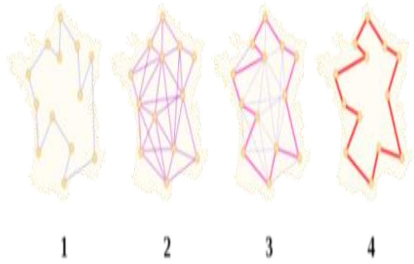

Figure2: Intensity of pheromone updating

V. GENERAL MULTIPLE ANT COLONIES ALGORITHM

In this section we would like to present about new algorithm in path planning, which is so-called General Multiple. Ant Colonies Optimization Algorithm. The word "general" here refers to integration of principles of several ACO Algorithms into one general algorithm to utilize advantages of these algorithms. In this algorithm, a number of colonies is utilized to find solutions. The process of searching solutions in each sub-colony is not much different from original ACO and ACO, except some formulas are changed. GMAC solution generating process is described in part A. Then, part B and part C talk about choice rule and pheromone updating rule for sub-colonies in the algorithm. Finally, part D and part E introduce methods applying to Great Colony to rank sub colonies and update general pheromone.

A. GMAC Algorithm Process [5]

The proposed algorithm's path planning process is described as follows:

1. Set parameters, initialize pheromone trails.
2. Create M colonies, each colony has N ants. The initial point of every ant is attributed to point S.
3. For every ant, apply rule in part B to choose next positions till they reach destination point D.
4. For every ant, compute the next possible positions set *ki N* .In the algorithm, the number of next possible positions around a position is 8.
5. Update pheromone trails for each sub-colony by using pheromone trails updating rule in part C.
6. After every sub-colony does its tours, calculate rankings of sub colonies by applying the method in part D.
7. Then do update general pheromone in sub-colonies with the formula in part E.
8. After a number of iterations for GMAC, we compare rankings of sub-ant colonies last time. Then choose the result of ACO that get highest ranking.
9. Every step stops as it meets terminal conditions such as out of max number of movement steps, etc...

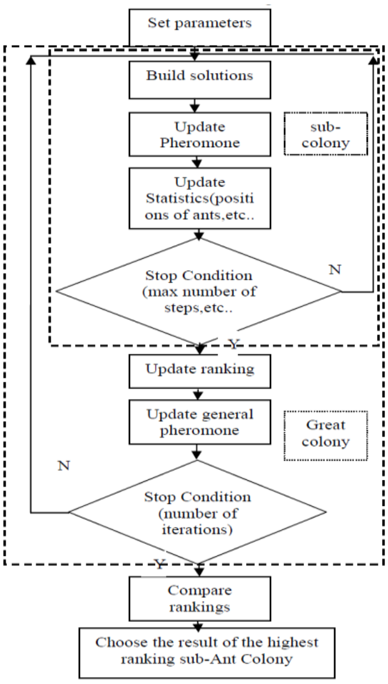

Figure 3:Structure of GMAC Algorithm

B. Next position choice rule

Suppose at time t, the *kth* ant is standing at node i (xki,yki).It has to choose a next point in the map to go. The formula for *kth* ant to choose node j (xki ,yki) in the algorithm is (2):

$$p_{ij}^{k} = \frac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum_{l \in N_i^k} [\tau_{il}(t)]^{\alpha}.[\eta_{il}]^{\beta}} + \alpha * \Omega_{ij} ;$$

As we introduced in part II, the function includes two terms: local term and global term. The purpose of the global term here (*) *ij* α Ω is to instruct agents to follow desired directions so that the agents are able to reach the destination point in a limited number of steps [14] .

C. Pheromone trails updating rule in sub-ACO

After moving to one node, agents (ants) continue to calculate the next node by using the choice rule as describe in part B. This process will be repeated till agents reach node D or terminal conditions are satisfied.

Pheromone value of edges will be updated after ants leave. To do this job, here we apply elitist strategy for edges caused by elitist ants and punishment strategy for edges caused by worse ants. It means that we will add extra pheromone to edges belong to best paths and subtract pheromone from edges belong to worse paths per iteration:

$$\tau_{ij} \leftarrow (1-\rho).\tau_{ij} + \sum_{k=1}^{m} \Delta_{\tau_{ij}}^{k} + \Delta\tau_{ij}^{*} \quad (5)$$

With

$$\Delta\tau_{ij}^{*} = \begin{cases} \sigma.\dfrac{Q}{L^{*}} & \text{,if edge(i,j) is part of the best solution found so-far;} \\ -\sigma.\dfrac{Q}{L^{*}} & \text{,otherwise;} \end{cases}$$

σ : Number of elitist ants. It is suggested that the performance of the algorithm is good if σ is set equal to number of ants in the colony.

D. Method of ranking sub-colonies: Elitist and Worst ACO update

After every colony has finished its own finding process, the best paths that they found are compared with each other. The *vth* sub-colony will be rewarded some extra points if the best path found by itself is shorter than others'. Otherwise, it will be penalized. The formula is as follows:

$$\gamma_v \leftarrow \gamma_v + \Delta\gamma \, (6)$$

Where

$$\Delta\gamma = \begin{cases} \Delta\gamma & \text{, if the best tour found by } v^{th} \text{ is better than others';} \\ -\Delta\gamma, & \text{otherwise;} \end{cases}$$

$\Delta\gamma$ is a parameter and is set by designers

γ_v is ranking of v^{th} sub-colony at the time of calculation.

E. General pheromone updating rule

This rule is carried out after each sub-colony did its tour and their ranking comparison finished. In the proposed algorithm, the sub-colony that gets highest ranking is allowed to keep its pheromone trails value, and sub-colonies that get lower ranking have to change their pheromone indexes according to this formula:

$$\tau_{ijlo} \leftarrow \tau_{ijhi} + \tau_{ijlo}(1-a) \; (7)$$

Here *ijhi* τ is pheromone value of the sub-colony that gets highest ranking. *ijlo* τ is pheromone value of sub-colonies that get lower ranking. A is a parameter value depending on the ratio of lowest ranking to highest ranking among sub-colonies at the time of calculation.

VI. CONCLUSION

In the Path Finding system, System shows the multiple path to reach the destination. The benefit is that user knows there are many ways to reach out that destination so he can believe on this system and also we will provide complaint facility so user can chat with admin. This system is easy and fast and also provides an easy way to communicate with the administrator. Provides quick way to know details about the paths. This system will also provide map with the multiple paths and shortest path. Allows only authenticated users to create an account. Ant Colony Algorithm is a heuristic method. Therefore, for a long time, it means that to get most optimal path, we need a very large number of ants to explore the map and find optimal solutions. Some researchers use a number of ants equal to 100,200 ants or even larger to solve path finding problem. It is a big problem if we do experiments for this algorithm. Reducing number of ants has same meaning with save money and time. Ant Colony Algorithm offers this chance. We only need 10 or 20 ants to get near-optimal results, while for previous ACO approaches, we need hundreds of ants to obtain them.

REFERENCES

1) Zlochin, M., Dorigo, M: Model-based search for combinatorial optimization: comparitive study.
2) Blum, C., Sampels, M.: When model bias is stronger than selection pressure. InGuerv_os, J.M.
3) Blum, C., Dorigo, M.: Deception in ant colony optimization. In Dorigo, M., et al.eds.: 4th International Workshop on Ant Colony Optimization and Swarm Intelli-gence, ANTS 2004.
4) Pathfinding in Open Terrain S. D. Goodwin, S. Menon†, R. G. PriceCentre for Interactive Digital Entertainment Research School of Computer Science, University of Windsor Windsor, Ontario
5) A Refined Ant Colony Algorithm for Optimal Path Planning Duc-Hien Nguyen Department of Electrical Engineering National Cheng Kung University Tainan City, Taiwan Jyh-Ching Juang Department of Electrical Engineering National Cheng Kung University Tainan City,Taiwan
6) Heuristic shortest path algorithms for transportation applications: State of the art L. Fua,∗, D. Sunb, L.R. Rilettc aDepartment of Civil Engineering, University of Waterloo, Waterloo, ON, Canada
7) Parallelization Strategies for Ant Colony Optimisation on GPUs Jos´e M. Cecilia, Jos´e M. Garcıa Computer Architecture Department University of Murcia 30100 Murcia, Spain
8) Ant Colony versus Genetic Algorithm based on Travelling Salesman Problem Mohammed Alhanjouri Belal Alfarra
9) Ant Colony Optimization Algorithm Nada M. A. Al Salami
10) Ant Colony Optimization applied to Route Planning using Link Travel Time predictions Rutger Claes DistriNet Labs Katholieke University Leuven Leuven
11) Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem TR/IRIDIA/1996-*5 Université Libre de Bruxelles* Belgium Marco Dorig*o IRIDIA*
12) Performance Comparison between Ant Algorithm and Modified Ant Algorithm Shaveta Malik# #CSE International University Faridabad, Haryana-121001 INDIA
13) Improved Ant Colony Algorithm for Global Optimal Trajectory Planning of UAV under Complex Environment Guanjun Ma Haibin Duan Senqi Liu ,Beihang University, Beijing 100083, P. R. China
14) Relationship between Genetic Algorithms and Ant Colony Optimization Algorithms Osvaldo G´omez Universidad Nacional de Asunci.