# Robot Path Planning using An Ant Colony Optimization Approach: A Survey

Alpa Reshamwala
Assistant Professor, Computer Engineering Department
MPSTME, SVKM's NMIMS University
Mumbai, India

Deepika P Vinchurkar
M. Tech Student, Computer Engineering Department
MPSTME, SVKM's NMIMS University
Mumbai, India

***Abstract*— Path planning problem, is a challenging topic in robotics. Indeed, a significant amount of research has been devoted to this problem in recent years. The ant colony optimization algorithm is another approach to solve this problem. Each ant drops a quantity of artificial pheromone on every point that the ant passes through. This pheromone simply changes the probability that the next ant becomes attracted to a particular grid point. The techniques described in the paper adapt a global attraction term which guides ants to head toward the destination point. The paper describes the various techniques for the robot path planning using the Ant colony Algorithm. The paper also provides the brief comparison of the three techniques described in the paper.**

***Keywords*— Path planning; Ant colony algorithm; collision avoidance.***

## I. INTRODUCTION

Path-planning can be described as the task of navigating a mobile robot around a space in which a number of obstacles that have to be avoided. Optimal paths could be paths that minimize the amount of turning, the amount of braking or whatever a specific application requires. Path-planning requires a *map* of the environment and the robot to be aware of its *location* with respect to the map. A reliable navigation algorithm must be able to

- Identify the current location of the robot,
- Avoid any collisions,
- Determine a path to the object.

Mobile robot navigation problem is a challenging problem, and a number of studies have been attempted, resulting in a significant number of solutions. Three major concerns regarding robot navigation problems are efficiency, safety and accuracy. The main scope of the path finding problem involves the efficiency and safety issues. The path finding problem can be overcome by combining global path planning and local path planning.[4]. The robot path planning methods could be classified into different kinds based on different situations. Depending on the environment where the robot is located, the path planning methods can be classified into the following two types as shown in Figure 1.

- Robot path planning in a static environment which contain only the static obstacles in the map; and
- Robot path planning in a dynamic environment which has static and dynamic obstacles in the map.

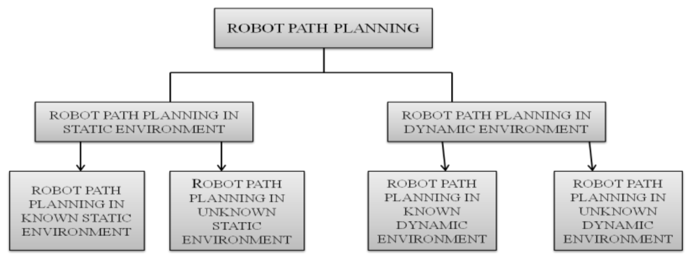


Fig. 1. Classification of the robot path planning methods.

Each of these two types could be further divided into two sub-groups depending on how much the robot knows about the entire information of the surrounding environment:

- Robot path planning in a clearly known environment in which the robot already knows the location of the obstacles before it starts to move.
- Robot path planning in a partly known or uncertain environment in which the robot probes the environment using sensors to acquire the local information of the location, shape and size of obstacles and then uses the information to proceed local path planning.

## II. REVIEW OF LITERATURE

Yogita Gigras, Kusum Gupta [1] proposed algorithm for collision avoidance using backtracking and used the ant colony algorithm for finding the optimum shortest path to reach to the destination. Buniyamin N., Sariff N., Wan Ngah W.A.J., Mohamad Z.[2] worked together to solve the Robot Path Planning(RPP) problem. They proposed the accurate representation of heuristic and visibility equations of state transition rules. The proposed algorithm was applied within a global static map having feasible free space nodes. Michael Brand, Michael Masuda, Nicole Wehner, Xiao-Hua Yu [3] investigated the application of ACO to robot path planning in a dynamic environment. They compared two different pheromone re-initialization schemes and describe the best of them based on the simulation result. O. Hachour[4] proposed algorithm for path planning of autonomous mobile robot in an unknown environment. The robot travels within the environment sensing and avoiding obstacles that come across its way to the target station. Daniel Angus [5] modified the existing Ant System meta heuristic by including three parameters: cost, visibility and pheromone. Based on this a new algorithm for the Shortest Path Ant Colony Optimization

(SPACO) was developed. The most important parameter included in this algorithm to solve shortest path problem is visibility. M.Dorigo, C. Blum [6] in Ant Colony Optimization theory: A Survey discussed the theoretical results of Ant Colony Optimization algorithms. They analyzed convergence results, connection between ACO algorithm and random gradient ascent within the model based search. Shahram Saeedi and Iraj Mahdavi[7] formulated a mathematical model to obtain the shortest path using Ant Colony Optimization. The model required calculation of shortest path between sources to target minimizing cost in the absence of any obstacle. Vinay Rishiwal et al. [8] proposed application of Ant Colony Optimization algorithm to find optimal paths in terrain maps. The algorithm uses penalty maps of the terrain maps as an input. The Terrain features such as land, forest etc are identified with different colors. Transition probability maintains a balance between pheromone intensity and heuristic information. Yee Zi Cong et al. [9] solved the mobile robot path planning problem using ACO algorithm. Each map consisted of static obstacles in different orientations. Each map was represented in a grid form with equal number of rows and columns. Song-Hiang Chia et al. [10] used Any Colony Optimization algorithm to solve the mobile robot path planning problem in such a way that the artificial ant reaches the target point from source point avoiding obstacles. The problem was modeled in a grid platform.

## III. PATH PLANNING TECHNIQUES

Path planning can be achieved through various different methods. In this section we describe the various techniques for path planning.

### A. *Particle swarm optimization (PSO):-*

Particle swarm optimization (PSO) method is relatively a new population-based intelligence algorithm and exhibits good performance in optimization problems. In the optimization process, the particles become more and more similar, and gather into the neighborhood of the best particle in the swarm, which makes the swarm prematurely converged possibly around the local solution. PSO do not guarantee an optimal solution is ever found. More specifically, PSO does not use the gradient of the problem being optimized.

### B. *Genetic algorithm (GA) :-*

Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

In a genetic algorithm, a population of strings called chromosomes or the genotype of the genome, which encodes candidate solutions called individuals, creatures, or phenotypes to an optimization problem, evolves toward better solutions. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

### C. *Tabu Search:-*

Fred Glover proposed in 1986 a new approach, which he called tabu search, to allow hill climbing to overcome local optima. The basic principle of tabu search is to pursue the search whenever a local optimum is encountered by allowing non-improving moves; cycling back to previously visited solutions is prevented by the use of memories, called tabu lists, which record the recent history of the search. Tabu search (TS) is based on the premise that problem solving, in order to qualify as intelligent, must incorporate adaptive memory and responsive exploration.

### D. *Simulated Annealing (SA)*

Simulated annealing (SA) is a random-search technique which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure and the search for a minimum in a more general system. Simulated annealing was developed in 1983 to deal with highly nonlinear problems. SA approaches the global maximization problem similarly to using a bouncing ball that can bounce over mountains from valley to valley.

### E. *Reactive Search Optimization (RSO)*

Reactive Search Optimization (RSO) advocates the integration of machine learning techniques into search heuristics for solving complex optimization problems.. Reactive Search Optimization also addresses a scientific issue related to the reproducibility of results and to the objective evaluation of methods. *Reactive Search* is a methodology for solving hard optimization problems, both in the discrete and continuous domain, based on the integration of machine learning and optimization in an online manner.

### F. *Ant Colony Algorithms*

The Ant Colony Optimization Algorithm is a relatively recent approach to solving optimization problems by simulating the behavior of real ant colonies. The Ant Colony System (ACS) models the behavior of ants, which are known to be able to find the shortest path from their nest to a food source.. Ants accomplish this by depositing a substance called a pheromone as they move. This chemical trail can be detected by other ants, which are probabilistically more likely to follow a path rich in pheromone. This trail information can be utilized to adapt to sudden unexpected changes to the terrain, such as when an obstruction blocks a previously used part of the path (Figure 2).

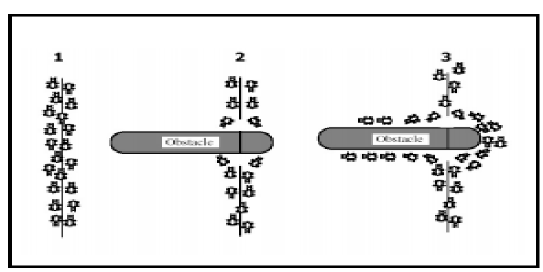


Fig. 2. Obstacles between Ants nest and Food

The shortest path around such an obstacle will be probabilistically chosen just as frequently as a longer path - however the pheromone trail will be more quickly reconstituted along the shorter path, as there are more ants moving this way per time unit (Figure 3).

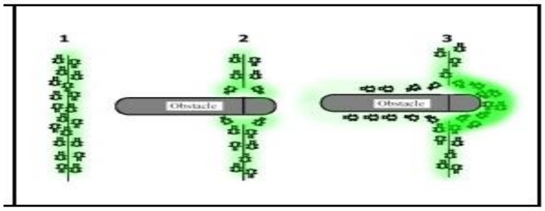


Fig. 3. Pheromone build-up allows ants to reestablish the shortest path.

Since the ants are more inclined to choose a path with higher pheromone levels, the ants rapidly converge on the stronger pheromone trail, and thus divert more and more ants along the shorter path. This particular behavior of ant colonies has inspired the Ant Colony Optimization algorithm, in which a set of artificial ants co-operate to find solutions to a given optimization problem by depositing pheromone trails throughout the search space. Existing implementations of the algorithm deal exclusively with discrete search spaces, and have been demonstrated to reliably and efficiently solve a variety of combinatorial optimization problems. Table 1 gives brief overview, of the three most successful algoritms: ant system (Dorigo 1992, Dorigo et al. 1991, 1996), ant colony system (ACS) (Dorigo & Gambardella 1997), and MAX-MIN ant system (MMAS) (Stützle & Hoos 2000). The historical order in which they were introduced

TABLE I. OVERVIEW OF THE THREE SUCCESSFUL ANT COLONY ALGORITHM

| Algorithm | Tour Construct | Evaporation | Pheromone | Updation |
|---|---|---|---|---|
| AS (Ant System) Dorigo et al. 1991, | random proportional rule | all arcs lowered with constant factor | $\tau_{ij} \leftarrow (1-\rho)\cdot\tau_{ij} + \sum_{k=1}^{m}\Delta\tau_{ij}$ | deposit on all arcs visited by all ants |
| ACS (Ant Colony System) Dorigo and Gambardella (1997). | Pseudo random proportional rule | only arcs of the best-so-far tour are lowered | $\tau_{ij} = (1-\varphi).\tau_{ij} + \varphi.\tau_0$ where $\varphi \in (0,1)$ is the pheromone decay coefficient | deposit only on arcs of the best so-far tour |
| MMAS (MAX-MIN Ant System) Stützle and Hoos (2000) | random proportional rule | all arcs lowered with constant factor | $\tau_{ij} \leftarrow (1-\rho).\tau_{ij} +.\Delta\tau_{bestij}$ where $\Delta\tau_{bestij} = 1/L_{best}$ | deposit only either by the iteration best-ant, or the best-sofar ant; interval [_min; _max] |

## IV. ALGORITHMS FOR ROBOT PATH PLANNING

### A. Path planning Algorithm[1]

The algorithm described below tries to avoid the collision and also suggest the steps to be followed during the occurrence of the obstacles.[1].

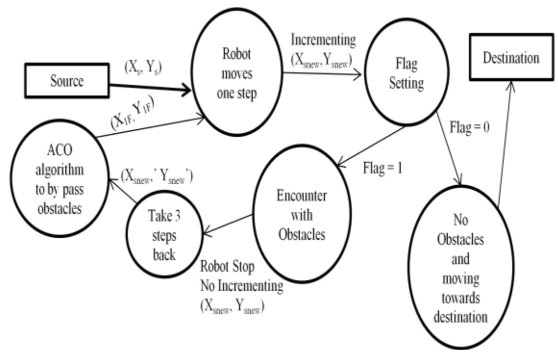


Fig. 4. Layout of robot path planning [1]

#### 1) Source

Robot start walking from a fixed source point ($X_s$,$Y_s$) .

#### 2) Robot Moves one step

The value of ($X_s$,$Y_s$) is changed to ($X_{snew}$,$Y_{snew}$) when the robot moves one step ahead by using the below equation:-

$$X_{snew} = X_{prev} + step * \cos(\theta) \quad (1)$$
$$Y_{snew} = Y_{prev} + step * \sin(\theta) \quad (2)$$

Where $X_{prev}$,$Y_{prev}$ denotes where the robot is currently situated. Robot's next position is determined by adding the product of step size and the cos(ɵ) and sin(ɵ).Where ɵ is dynamic angle and it can be calculated by:-

$$\theta = \tan^{-1} X_{prev} / Y_{prev} \quad (3)$$

#### 3) Flag Setting

Robot see the value of the flag, if its value is zero it means there is no obstacle and robot can take a one step ahead to the destination point.

#### 4) Encounter with obstacle

Whenever the robot encounter with obstacle, it has to stop moving. In our proposed work, twenty obstacle are generated randomly which is of rectangular shape. Number of obstacles is fixed which a constraint in our work is.

#### 5) Take three step back

Whenever the robot encounter with obstacle, robot stop moving and take three step back by using the following equation:-

$$X_{snew} = X_{prev} - 3 * step * \cos(\theta) \quad (4)$$
$$Y_{snew} = Y_{prev} - 3 * step * \sin(\theta) \quad (5)$$
$$\theta = \tan^{-1} X_{prev} / Y_{prev} \quad (6)$$

*6) Destination*

Finally robot has to reach at the point $(X_T,Y_T)$, which is fixed. Robot has to bypass the obstacle and by following optimal path has to reach to target point.

*7) Apply the ACO algorithm to bypass the obstacle*

ACO is used to find out the optimal one i.e. locally or globally optimal. This algorithm is implemented in two steps.

*a) In first step, the edge is selected on the basis of probability formula. Assume that ant k is located at node i, uses the pheromone deposited on the edge (i,j) to compute the probability of choosing next node$_{ij}$*

$$P_{ij} = \begin{cases} \dfrac{\tau_{ij}\alpha}{\sum_{j \in N_{i^{(k)}}} \tau_{ij}\alpha} & if\ j\ \varepsilon\ N_{i^{(k)}} \\ 0 & \text{otherwise} \end{cases} \qquad (7)$$

Where α denotes the degree of importance of pheromone trail and $Ni^{(k)}$ indicates the set of neighbor of ant k when located at node i except the last node visited by ant k, which helps to prevent the ant k for returning to the same node.

*b) In second step, once all the ants complete their tour, then global optimization of the pheromone trail takes place.*

$$\tau_{ij} = (1-\rho).\rho + \sum_{k=1}^{N} \Delta\tau_{ij}^{(k)} \qquad (8)$$

Where ρ Є (0,1) is the evaporation rate and $\Delta\tau_{ij}^{(k)}$ and Δ is the amount of pheromone deposited on the edge (i,j) selected by the best ant k. The aim of pheromone updating is to increase the pheromone value associated with optimal path. The pheromone deposited on arc (i, j) by the best ant k is $\Delta\tau_{ij}^{(k)}$. Where,

$$\Delta\tau_{ij^{(k)}} = \frac{Q}{L_k} \qquad (9)$$

Here Q is a constant and $L_k$ is the length of the path traversed by the best ant k. This equation is also implemented as:-

$$\Delta\tau_{ij}^{(k)} = \begin{cases} \dfrac{T_{best}}{f_{worst}} & if\ (i,j)\varepsilon\ global\ best\ tour \\ 0 & otherwise \end{cases} \qquad (10)$$

*B. Robot Path Planning Algorithm By Buniyamin N Et,Al[2]*

For the Robot Path Planning (RPP) purpose, the proposed path planning algorithm is a modification of the original ACO concept (also known as Ant Colony System) proposed by Marco Dorigo [2]. Figure 5 outlines the implementation of ACO for RPP of a mobile robot. The model and concept of the proposed algorithm is as follows:

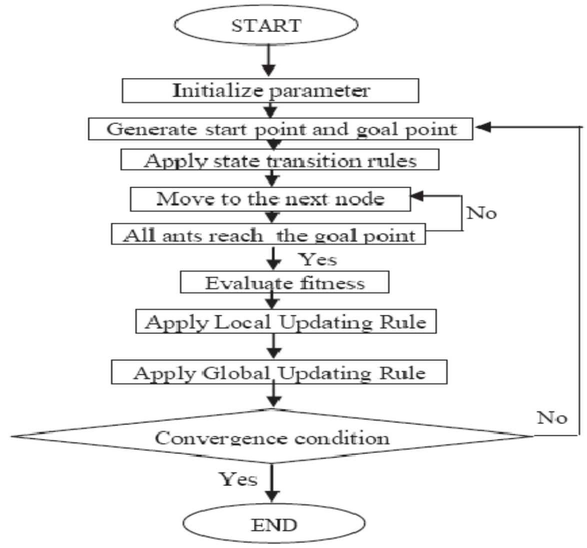


Fig. 5. Outline for the implementation of ACO for RPP of a mobile robot [2]

Starting from the start node located at x-y coordinate of (1,1), the Robot will start to move from one node to other feasible adjacent nodes.

*8) The robot will then take the next step to move randomly based on the probability given by equation (11) :*

Probabilityij (t) = Heuristic$_{ij}$ (t)*Pheromone$_{ij}$(t)
=[(1/distance between vector start point to next point and start point to reference line to goal)β * (trail/ Σtrail)σ] (11)

Where Heuristic (t) indicates every possible adjacent nodes to be traversed by the robot in its grid position at every t time. The quantity of Pheromone$_{ij}$ (t) is an accumulated pheromone between the nodes when the robot traverses at every t time. Therefore, the probability equation depends on both values where it will guide the robot to choose every possible node in every t time. Each time robot construct a path from one node to another, the pheromone amount will be reduced locally by the given evaporation rate using the formula of update local rules as shown below:

$$T_{ij^{(new\ trail)}} \leftarrow (1-\rho) * T_{ij^{(old\ trail)}} \qquad (12)$$

where ρ=evaporation rate

This equation shows that each time the robot move from one node to another node, the amount of local pheromone will be updated in parallel.

This process is important to prevent the map from getting unlimited accumulation of pheromone and enables the algorithm to forget a bad decision that has been previously made.

Once the robot found its path to goal, the fitness of robot will be calculated. This covers the calculation of distance or path cost each robot takes to traverse from start point to goal point by using derivation of objective function for RPP below:

$$Distance = \sqrt[2]{(y_2 - y_1)^2 + (x_2 - x_1)^2} \quad (13)$$

The fitness value will then be used for the process of global update. When all robots reach the destination, the robots will update the value of pheromone globally based on the fitness found by each robot by using Equation (14) below.

This process will be repeated as the path being traverse by robots in each generation is determined using this global value. During the process, the path with the shorter distance will be chosen as the probability to be chosen is higher compared to the path with the longer distance. The equation of global updating is derived in (14) and (15) below:

$$t_{ij} \leftarrow t_{ij} + \sum \Delta t_{ij}^{k} \quad (14)$$

Where $\Delta\ t_{ij}^{k}$ is the amount of pheromone of robot m deposits on the path it has visited. It's defined as below:

$$\Delta\tau_{ij(k)} = \frac{Q}{L_k} \quad (15)$$

Where Q is number of nodes and $L_k$ is the length of the path $P_k$ built by the robots.

The process will be repeated from Step 1 to Step 5 until the process converges. The process will stop when all robots traverse the same path that shows the shortest path to goal has been found.

### C. *Robot Dynamic Path planning by Michael Brand, et.al[3]*

In this section, the proposed ant colony optimization algorithm is applied for robot path planning in a grid network. Since our goal is to find the shortest path between the starting and ending positions, the total path length is chosen to be the cost or reward associated with each possible solution. The simulation starts with a "clean" environment; i.e., there is no obstacle in the original network. The upper-left comer is chosen to be the starting point and the lower-right comer is chosen to be the destination. All the pheromones are initialized as 0. The ant colony algorithm is then applied to find the shortest path and pheromones are deposited. A computational flow chart is shown in Figure6.

Consider a network where ants can travel between different nodes. Using pheromone deposits, the probability that an ant k located in node i will choose to go to another node in the network is given by the equation (16)

$$p_{ij}^{k} = \left|\begin{cases} \dfrac{(\tau_{ij}^{k})^{\alpha} . (\eta_{ij}^{k})^{\beta}}{\sum_{l \epsilon N_i^k} (\tau_{il}^{k})^{\alpha} . (\eta_{il}^{k})^{\beta}} & if\ j \epsilon N_i^k \\ 0 & if\ j \notin N_i^k \end{cases}\right| \quad (16)$$

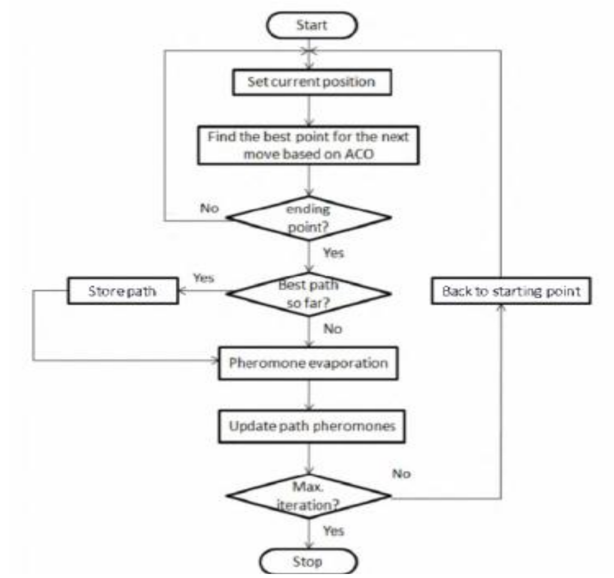


Fig. 6. Computational flow chart of ACO[3].

Where pheromone levels are denoted by $\tau^{k}_{ij}$ .The summation in the denominator considers possible choices (or neighboring nodes) in the set $N_i^k$ when the ant is at node i. α, β and $\eta_{ij}^{k}$ are usually application dependent; where $\eta_{ij}^{k}$ represents the heuristic information, and the values of α and β weigh the importance of the pheromone and heuristic values. When β = 0, $(\eta_{ij}^{k)\ \beta}$ then the probability only depends on the pheromone levels; on the other hand, when α = 0, the probability only depends on heuristic values that is, the node that is the closest one to the current node has the highest probability of being selected.

The pheromone levels of the path (from node i to i), can evaporate with a percentage ρ (also called the evaporation rate):

$$\tau_{ij} \leftarrow (1 - \rho) . \tau_{ij} \quad (17)$$

Where $0<\rho< 1$. After pheromone evaporation occurs, the new pheromone levels are updated with the additional pheromone laid by the ants that just crossed the path:

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \quad (18)$$

Where Ck is the associated cost or reward of ant k for choosing this path.

$$\Delta\tau_{ij}^{k} = \frac{1}{C^k} \quad (19)$$

## IV. Comparison Between Algorithms

The brief overview of the different proposed algorithm and the facts that allow them to differ from each other are described in this section. In this Table 2. section compares the proposed algorithm based on the variations done by each algorithm in the basic ant colony algorithm which helps to overcome the drawback of the basic ant colony algorithm and provide us with the better solution for robot path optimization.

TABLE II. COMPARISON BETWEEN ALGORITHMS

| Properties | Path planning Algorithm[1] | Robot Path Planning Algorithm by Buniyamin N et,al[2] | Robot Dynamic Path planning by Michael Brand, et.al[3] |
|---|---|---|---|
| Objective Of Algorithm | Collision Avoidence | To find an optiml path based on distance time and number of itertions | To find the Shortest Path |
| Use of Flag | Yes. use the flag value to indicate the presence of the obstacles. | No. | No. |
| Action on obstacle occurrence | Allow the robot to move three steps back if the obstacles are detected | The path containing an obstacle is considered as unfeasible in the initial stage only. | Reinitialization of the pheromone in the network is done. |
| Average Time Taken | 27.911018 sec. | 63 seconds | 100 seconds |

## V. APPLICATIONS OF ANT COLONY ALGORITHM

In recent years, the interest of the scientific community in ACO has risen sharply. The use of an algorithms providing exponential time worst complexity is often infeasible in practice, thus ACO algorithms can be useful for quickly finding high quality solutions. This section describes the applications of the Ant colony algorithm in various fields.

### A. Applications To NP-Hard Problems

ACO has been tested on probably more than one hundred different NP-hard problems. The problems include the sequential ordering problem, openshop scheduling problems, some variants of vehicle routing problems, classification problems, and protein–ligand docking. Many of the tackled problems can be considered as falling into one of the following categories:

- *routing problems:-* as they arise, for example, in the distribution of goods;
- *assignment problems*, where a set of items has to be assigned to a given number of resources subject to some constraints.
- *scheduling problems*, which–in the widest sense–are concerned with the allocation of scarce resources to tasks over time; and
- *subset problems*, where a solution to a problem is considered to be a selection of a subset of available items.

### B. Applications To Telecommunication Networks

ACO algorithms have shown to be a very effective approach for routing problems in telecommunication networks where the properties of the system, such as the cost of using links or the availability of nodes, varies over time. ACO algorithms were first applied to routing problems in circuit switched networks. A well-known example is AntNet[11].

### C. Applications To Industrial Problems

The first to exploit algorithms based on the ACO metaheuristic is EuroBios (www.eurobios.com). They have applied ACO to a number of different scheduling problems such as a continuous two-stage flow shop problem with finite reservoirs.

Another company that has played, and still plays, a very important role in promoting the real-world application of ACO is AntOptima. AntOptima's researchers have developed a set of tools for the solution of vehicle routing problems whose optimization algorithms are based on ACO.

## VI. CURRENT RESEARCH TOPICS IN ACO

A significant part of research on ACO is still concerned with applications. However, increasing attention is and will be given to even more challenging problems that, for example, involve *multiple objectives*, *dynamic modifications* of the data, and the *stochastic nature* of the objective function and of the constraints.

### A. Dynamic optimization problems

Dynamic problems are characterized by the fact that the search space changes during time. Hence, while searching, the conditions of the search, the definition of the problem instance and, thus, the quality of the solutions already found may change. For this problem, ACO algorithms belong to the state-of-the-art techniques [11][12].

An ACS algorithm has also been applied to dynamic vehicle routing problems, showing good behavior on randomly generated as well as real-world instances.

### B. Stochastic optimization problems

In stochastic optimization problems, some variables have a stochastic nature. The probabilistic traveling salesman problem (PTSP) was the first stochastic problem tackled by ACO algorithms. The first ACO algorithm for this problem was proposed by Bianchi et al.[13] Further ACO algorithms for the PTSP have been proposed by Branke and Guntsch[14] , Gutjahr[15][16], and Birattari et al[17].

### C. Multi-objective optimization

Multiple objectives can often be handled by ordering or weighting them according to their relative importance. In the two-colony ACS algorithm for the vehicle routing problem with time window constraints[18] and in the MMAS for the bi-objective two-machine permutation flow shop problem , the multi-objective optimization problem is handled by ordering the objectives; differently,

### D. Continuous optimization

ACO algorithms have been applied to continuous optimization. When an algorithm designed for combinatorial optimization is used to tackle a continuous problem, the simplest approach would be to divide the domain of each variable into a set of intervals [21][22]. Research in this direction is currently ongoing.

## VII. Conclusion

In this paper, ACO is used to find the shortest navigational path of mobile robot avoiding obstacles to reach the target station from the source station. In this paper, the results of detailed investigation of ACO algorithms being applied to a path optimization problem have been presented Overcoming the limitations of the algorithms represent a challenge for future research. No matter how many obstacles are present, this algorithm does not devote an excessive amount of time in iteration process. ACO approach takes some unnecessary steps, so that the algorithm does not return the best solution. Furthermore, a global attraction term had to be added to lead ant to reach the goal point. Eliminating this term may cause not only the ant wander around in the map, but also the ant may become stuck at a point which will prevent the ant .i.e. robot from reaching to the goal. This paper also gives the comparison about the different algorithm described in the paper. Based on the comparison we can state that the Path Planning Algorithm by Yogita Gigras et.al. is better than the other two based on the average time taken.

## References

[1] Yogita Gigras, Kusum Gupta , "*Artificial Intelligence in Robot Path Planning",* International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2, May 2012

[2] Buniyamin N., Sariff N., Wan Ngah W.A.J., Mohamad Z., "*Robot global path planning overview and a variation of ant colony system algorithm*", International Journal Of Mathematics And Computers In Simulation.

[3] Michael Brand, Michael Masuda, Nicole Wehner, Xiao-Hua Yu, "*Ant Colony Optimization Algorithm for Robot Path Planning*", 2010 International Conference On Computer Design And Appliations (ICCDA 2010).

[4] O.Hachour "*Path Planning of Autonomous Mobile Robot"*, International journal of systems applications, engineering & development, Issue 4, Volume 2, 2008

[5] Daniel Angus, *"Solving a unique Shortest Path problem using Ant Colony Optimization",* Communicated by T.Baeck

[6] Marco.Dorigo & Christian Blum , *"Ant Colony Optimization Theory : A Survey"*

[7] Shahram Saeedi and Iraj Mahdavi, *"Using Ant Colony Optimization for Shortest Path Problem"*

[8] Vinay Rishiwal , Mano Yadav, K.V. Arya, *"Finding Optimal Paths on Terrain Maps using Ant Colony Algorithm",* International journal of computer theory and engineering, Vol. 2, No. 3, June 2010

[9] Yee Zi Cong and S. G. Ponnambalam, *"Mobile Robot Path Planning using Ant Colony Optimization",* 2009 IEEE/ ASME International Conference on Advanced Intelligent Mechatronics, Suntec Convention and Exhibition Center, Singapore, July 14-17, 2009

[10] Song-Hiang Chia, Kuo-Lan Su, Jr-Hung Guo, Cheng-Yun Chung*, "Ant Colony System Based Mobile Robot Path Planning",* 2010 Fourth International Conference on Genetic and Evolutionary Computing.

[11] G. Di Caro and M. Dorigo, "AntNet*: Distributed stigmergetic control for communications networks*," Journal of Artificial Intelligence Research, vol. 9, pp. 317–365, 1998.

[12] G. Di Caro, F. Ducatelle, and L. M. Gambardella, "*AntHocNet:An adaptive nature-inspired algorithm for routing in mobile ad hoc networks,*" European Transactions on Telecommunications, vol. 16, no. 5, pp. 443–455, 2005.

[13] L. Bianchi, L. M. Gambardella, and M. Dorigo, "An ant colony optimization approach to the probabilistic traveling salesman problem," in *Proceedings of PPSN-VII*, ser. LNCS, J. J. Merelo *et al.*, Eds., vol.2439. Springer Verlag, 2002, pp. 883–892.

[14] J. Branke and M. Guntsch, "New ideas for applying ant colony optimization to the probabilistic TSP," in *Applications of Evolutionary Computing, Proceedings of EvoWorkshops 2003*, ser. LNCS, G. R. Raidl *et al.*, Eds., vol. 2611. Springer Verlag, 2003, pp. 165–175.

[15] W. J. Gutjahr, "A converging ACO algorithm for stochastic combinatorial optimization," in *Proc. SAGA 2003*, ser. LNCS, A. Albrecht and T. Steinh¨ofl, Eds., vol. 2827. Berlin, Germany: Springer Verlag, 2003, pp. 10–25.

[16] ——, "S-ACO: An ant based approach to combinatorial optimization under uncertainty," in *Ant Colony Optimization and Swarm Intelligence,4th International Workshop, ANTS 2004*, ser. LNCS, Dorigo *et al.*, Eds., vol. 3172. Berlin, Germany: Springer Verlag, 2004, pp.1–12.

[17] M. Birattari, P. Balaprakash, and M. Dorigo, "ACO/F-Race: Ant colony optimization and racing techniques for combinatorial optimization under uncertainty," in *MIC 2005: The 6th Metaheuristics International Conference*, K. F. Doerner *et al.*, Eds. Vienna, Austria: University of Vienna, Department of Business Administration, 2005, pp. 107–112.

[18] L. M. Gambardella, E. D. Taillard, and G. Agazzi, "MACS-VRPTW: A multiple ant colony system for vehicle routing problems with time windows," in *New Ideas in Optimization*, D. Corne *et al.*, Eds. McGraw Hill, London, UK, 1999, pp. 63–76.

[19] K. Doerner, R. F. Hartl, and M. Reimann, "Competants for problem solving – the case of full truckload transportation," *Central European Journal for Operations Research and Economics*, vol. 11, no. 2, pp. 115–141, 2003.

[20] Iredi, D. Merkle, and M. Middendorf, "Bi-criterion optimization with multi colony ant algorithms," in *First International Conference on Evolutionary Multi-Criterion Optimization, (EMO'01)*, ser. LNCS, E. Zitzler *et al.*, Eds., vol. 1993. Springer Verlag, 2001, pp. 359–372.

[21] K. Socha, "ACO for continuous and mixed-variable optimization," in *Ant Colony Optimization and Swarm Intelligence, 4th International Workshop, ANTS 2004*, ser. LNCS, M. Dorigo *et al.*, Eds., vol. 3172.Springer Verlag, 2004, pp. 25–36.

[22] K. Socha and M. Dorigo, "Ant colony optimization for continuous domains," *European Journal of Operational Research*, 2006, in press.

## Authors Profile

Ms. Alpa Reshamwala is currently an Asistant Professor in the Department of Computers at MPSTME, NMIMS University. She received her B.E degree in Computer Engineering from Fr. CRCE, Bandra, Mumbai University in 2000 and M.E degree in Computer Engineering from TSEC, Mumbai University in 2008. Her area of Interest includes Artificial Intelligence, Data Mining, Soft Computing – Fuzzy Logic, Neural Network and Genetic Algorithm. She has 20 papers in National/International Conferences/ Journal to her credit. She is also associated as an International Expert of International Journal of Electronics Engineering and Mobile Computing. She has a membership of International Association of Computer Science and Information Technology (IACSIT) and is also a student member of UACEE (Universal Association of Computer and Electronics Engineers)

Deepika Vinchurkar, pursuing M.Tech CS at Mukesh Patel School of Technology Management and Engineering from NMIMS Mumbai. Her areas of interest are Artificial Intelligence, Network Security Neural Netwoks, Robotics.