Abstract
Ant Colony Optimization, a Swarm intelligence method which solves NP hard problems inspired from the behaviour of ant foraging (Searching for food) the heuristics they use and partial guidance by other ants in indirect format (Stigmergy). This paper developed to list out the variations of the ACO application, its variants with clear diagrammatic and graphical representation. Clear denotation of the techniques they have used to solve such problems, representation of problems, transformations, parameters used and advantage of the techniques and used variants. Through this article we identify some open suggestions with a certain interest of being solved in near future.

Keywords: Heuristics, Pheromone, Stigmergy

1. Introduction

1.1 Biological Background of ACO
Ant Colony Algorithm was been first proposed by Dorigo who is the person first invented Ant Colony Algorithm in Travelling Salesman Problem which is a NP Hard Problem. This Ant Colony Optimization technique comes under Swarm Intelligence which comes under Bio Inspired Algorithms.

1.2 Bio Inspired Algorithms
Biologically Inspired Algorithms (in short as Bio Inspired Algorithms) where the algorithms have been derived from the nature and this bio inspired algorithms have been applied to many fields. It had successful scale results. This bio inspired algorithms often touch artificial intelligence but it does not come under Artificial Intelligence. Bio Inspired Algorithm differs in the way that it uses an evolutionary kind of approach for learning which was been opposed by the Artificial Intelligence and the same will be happening in Artificial Intelligence with the name of creation. And there the programmer would be the creator who will imbues the method with its intelligence. Here the optimality will be obtained by evolution. And our Bio Inspired Algorithms uses more simple rules and uses those rules at each and every iteration and attains the optimality. Bio Inspired Algorithms are highly decentralized as well as it is a bottom up approach.

1.3 Swarm Intelligence
Swarm intelligence was been proposed and induced in the Artificial Intelligence by Beni and Wang in 1989. It was been applied in the context of cellular robotic systems. Swarm Intelligence are basically small agents or living organisms which will collectively work together and make optimality or the group work will end with some beneficiary report. Inspiration from those group works and turning it into the artificial things and making those ideas possible in the computers to attain some sort of goal is called swarm intelligence. The inspiration basically
from the nature and to be even crisper the inspiration from the biological systems is called swarm intelligence. A collective work of small agents (agents in the sense of living organisms) which will be beneficiary to the group and also gain some benefits from the group. The agents will be interacting with each other either direct or indirect. Here there were no centralized behaviours from any agents. Optimal solution will be attained only at the evolution of same set of behaviour so many iterations. Some of the examples include ant colonies, birds flocking, fish schooling, etc.

For example in Ant Colony Algorithm, a colony of artificial ants works or cooperates together in order to find an optimal solution for discrete optimization problems or for the problems which are uncertain in nature. In this Ant Colony Algorithm the cooperation with the other ants is much essential for finding the optimal solutions for any set of problems.

2. Overview of the Article

Ant Colony Optimization has been applied in many form likes optimizing a problem which is of fuzzy space, uncertainty problems and the NP hard Problems. This article presents various kind of situations and problems that the ACO can be applied to find out optimal solutions where the problem consists of many constraints. Fuzzy problems and uncertainty problems can also be optimized by ACO. For example Traffic Signalling Problem: in this the traffic signal will be safe until or unless the capacity of the signal increase and once it occurs there may be a chance of colliding somewhere either in the front or in the tail of the signal. It may result in a loss of huge timing may be also in days and days of time. ACO can be able to avoid these kinds of problems and provide a clear picture on how to handle those situations with some kind of simulation with the help of ants.

An introduction about ACO and its working is essential to know for travelling this article beyond this point. In this the representation of problem plays an important role to make it to be solved using ACO. Problems need to be represented in form of graph which is the basic and essential need to make it to get solved by ACO technique.

2.1 Earlier Surveys

Earlier Surveys on ACO gives a clear picture on the level of ACO at that point of time. First survey was been published in the year 2005, Ant colony optimization theory: A survey, by Marco Dorigo, et al. which gives the theoretical results on ACO. Second, a survey in the same year 2005, A Critical Survey of Ant Colony Optimization for Hard Discrete Optimization Problems, Buck Moritz this will be explaining about the behaviour of ants and the introduction of ACO by Marco Dorigo and finally the analysis of bibliography liked to the ACO metaheuristic. Third, Ant colony optimization: Introduction and recent trends, Christian Blum, clarifies the foraging behaviour of real ant colonies, the recent activities in the field of ACO at that time. There are present some of the ACO variants which was been introduced till that period of time. Figure 1 illustrates the graphical representation of previous year surveys on ACO.

In the year 2011, A survey on Parallel ACO, Martin Pedemonte, et al. allowing ACO algorithms to achieve high quality results in reasonable execution times in parallel computing techniques. Then in the year of 2012, Ant Colony Optimization-based bio inspired hardware survey and prospect Haibin Duan, et al. which will be discussing on the recent progress of ACO-based hardware for Travelling Salesman Problem, digital circuits, digital infinite impulse-response filters, and hardware oriented ACO with look up table and hardware/software partition. Next in the same year 2012, A survey: Ant Colony Optimization based recent research and implementation on several engineering domain B. Chandra Mohan, et al. have proposed a modified ACO model which can be applied in the Network routing problem and it has been then compared with the existing techniques which solves that problem previously.

2.2 Components of ACO

**Pheromone:** Pheromone is a chemical factor which is excreted by ants for benefit of following ants. In Artificial
ants these pheromones are used to find the optimal solution (path) in several kinds of problems which are uncertain and combinatorial in nature. These pheromone trails are volatile in nature. It will get evaporate after a particular amount of time. Likewise the artificial ants also have evaporation factor which helps to give less priority for non-optimal paths.

**Initial value of Pheromone:** Quantity of pheromone that lay on all the paths that the ant travels. Initial pheromone values can be set to either 1 or 0. If 1 is the value of initial pheromone then the evaporation rate will be assigned is such a way that first it gets evaporated and then the pheromone will be laid over the path. If it set to 0 then 1st laying property will be made and then the pheromone evaporation factor will reduce the pheromone quantity deposited on the path.

**Stigmergy:** Stigmergy refers the indirect communication between ants via pheromone deposit on the path. It used by natural ants, when it secrets pheromone on the path, successor ant will note the places this ACO can be explored for solving.

**Pheromone Evaporation Rate:** Pheromone Evaporation Rate refers the pheromone decay in unit time. This evaporation rate will be implemented in the problem in terms of formula which will be represented usually in $\Delta$.

**Pheromone Decay coefficient:** it is a constant value which refers the decay of the pheromone in a constant manner.

### 2.3 Goals of ACO

Goal of ACO is to optimize the problems which are hard in nature. Many kinds of problems can be handled using ACO and few are NP hard problems which can be solved only in polynomial time. That kind of problems needs only the optimal solution and not exact best solution. In such kind of places this ACO can be explored for solving.

Figure 3 illustrates the flow of ACO algorithm on how it identifies optimal solution for NP hard problems.

**Finding shortest path:** Artificial ants are used to find optimal path from source node to destination node which depicting from real ants. Real ants will be searching for food from nest to food source. The way it builds the path would not be worse at first but it will be optimal in nature using the pheromone trail. This pheromone trail lay on the path will result in the optimal path and the same will be done using the computational source to find the optimality for any kind of problems. Real ants have the nature that it will move to food source in an optimal way and used to lay the pheromone which is chemical substance that will help the following ants to find the better path to travel based on the quantity laid by the previous ants. Pheromone will also have the evaporation capability so that the path which are travelled first and then omitted.

---

**Figure 2.** Components and Results of ACO.

**Figure 3.** Flow chart of ACO algorithm.
due to non-optimality will get evaporated. Real ants used
to move to food source by crossing in-between adjacent
nodes that it can choose to reach the final food source.

The adjacent node chosen ways: Real ant chooses its path
in two ways. One, it chooses the next adjacent path to be
travelled using the laid pheromone trails by previous ants
which travelled in that path. The next way is using some
of the heuristics or in some random manner it chooses the
next path to be travelled from one node to another node.
In the same way the artificial ants chooses its arc or path
to travel from node i to node j using the pheromone quantity
that are laid on the path. It was done by previous iteration
ants and it also can choose a random path if the program
allows those artificial ants to do so. Artificial ants have
some other qualifications that the real ants don’t have in
it. Real ants don’t have the lookahead capability to predict
its future path. It just either chooses the path based on the
pheromone quantity laid on the path or in some random
manner. But the artificial ants have the capability of looka-
head and it can be able to go in a path which gives always a
complete solution as well as the optimality attained by the
artificial ants will be better. Because of no lookahead the
real ants would be in trouble of complete local optimality.

Apart from lookahead, artificial ants19 have some
other qualifications to attain optimality. Artificial ants
have some set of internal memory which will be hav-
ing the past actions that were been made in the previous
iterations or in the current iteration the path travelled
from source node to destination node. This memory will
be very useful in online pheromone trail update. In the
online pheromone trail update the ants will construct the
complete tour and then only the pheromone update will
be made. And in that place while updating the path from
the destination to source the internal state will be helpful
in order to update the path.

3. Traditional Approaches of ACO

ACO has been introduced in the year 1991 by Dorigo and
from that date to till date there were many algorithms
came to existence. Initial was Ant Systems and after that
Max-Min and a lot more have been derived according to the
problem where it has been introduced.

3.1 Ant System

Ant System (AS) was proposed in 19918-10 and first
deployed to the Travelling Salesman Problem. The
problem is to make artificial ants to move from one node
of the graph to the other (i.e. to make the artificial ants to
travel from one city to another so that all cities needs to
be visited with minimal cost) and the algorithm executed
for t times where t has been considered as the iteration
number. For every iteration m ants build a tour executing
n steps in which state transition rule is applied. When AS
was been proposed it was been made in 3 versions namely
ant-density, ant-quantity and ant-cycle. The differences
between first 2 versions i.e. ant-density and ant-quantity
are nothing of big. But when comparing first 2 versions
with 3rd version then there comes notable differences. In
first 2 versions ant deposit the pheromone in all the paths
they travel from node i to j where i and j considered as
adjacent nodes to each other which will help the follow-
ing ants for guidance. While in 3rd version update using
pheromone trail will be made over the edges only when
complete tour has been constructed. Also the edge will
be laid with pheromone only if the quality is better than
previous tour. When comparing these 3 versions, version
3 provides best results on comparing 1st 2 versions and
the 3rd version gives better result in AS versions. So it has
been declared as usable one and the other two versions
were abandoned.

There were 2 main steps that are there in Ant System
and they are Tour Construction and Pheromone Update
on the nodes that the ant travels.

Tour Construction: At initial state each ant will be put
on to some random chosen city. For each step of con-
structing the tour, ant k will be given a state transition
rule (probabilistic action). The probability of ant k moves
from current city i to adjacent city j will be of the form,

\[ p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha[n_{ij}]^\beta}{\sum_{k \in K_i} [\tau_{ij}(t)]^\alpha[n_{ij}]^\beta} \]

where \( \eta_{ij} = 1/d_{ij} \) is the heuristic value that is available, \( K_i \)
is the kneeborder of ant k, and when \( \alpha = 0 \) mostly the
neighbor cities will be selected. If \( \beta = 0 \) only pheromone
amplification will be working.

Pheromone Update: After all ants construct the path
pheromone update will be made. This will be done after
decreasing the pheromone strength on entire arc by a
constant factor and then deposit of pheromone will be
made as follows

\[ \tau_{ij}(t+1) = (1-\rho) \tau_{ij}(t) + \sum_{k=1}^{m} \Delta \tau_{ij}^k(t) \]
where $\rho$ is the pheromone trail evaporation and it will lie in between 0 and 1, $\Delta \tau^k_{ij}(t)$ is the amount of pheromone deposited on the arc it travelled and it is defined as

$$\Delta \tau^k_{ij}(t) = \begin{cases} 
\frac{1}{L^k(t)} & \text{if arc} \ (i, j) \ \text{is used by ant} \ k \\
0 & \text{otherwise}
\end{cases}$$

$L^k(t)$ is the length of the tour of ant $k$.

### 3.2 Ant Colony System

Ant colony system was developed by Dorigo in 1997. ACS differs in 3 forms from AS. First, ACS uses more aggressive action choice rule than AS. Second, the pheromone is updated only to arcs which belong to global-best solution. Third, in all iterations whenever the ant moves from a city $i$ to $j$ some pheromone will be deduced in constant from the arc. In the following we present these alterations in more detail.

**Tour Construction:** Unlike the transition rule (i.e. probabilistic action) for choosing next city in AS, ACS chooses the next city to be travelled using pseudo random proportional action choice rule. From city $i$ to $j$ an ant $k$ move with a probability of $q_k$, for which $\tau_{ij}(t)[\eta_{ij}]^\rho$ is maximal. It means the probability of $q_k$ i.e. the best possible move from node $i$ to $j$ will be guided by pheromone trail or by heuristics (exploitation of learned knowledge). With the probability of $(1-q_k)$ ant performs exploration of arcs as like AS.

**Global Pheromone Update:** In ACS only the global best solution will get updated by pheromone after each iteration and global best ant will be doing it. The global best ant will be identified based on the solution that it returns after each iteration. Here the update formula would be

$$\tau_{ij}(t + 1) = (1 - \rho) \cdot \tau_{ij}(t) + \rho \cdot \Delta \tau^{gb}_{ij}(t)$$

where $\Delta \tau^{gb}_{ij}(t) = 1/L^{gb}$.  

**Local Pheromone Trail Update:** Unlike AS ACS performs local pheromone trail update after each ant travels an arc. This update will change the quantity of pheromone on a particular arc which will make the following ant to choose some other arc. It results in exploration of more arcs. In this manner the unvisited arcs exploration will be increased and the local pheromone update will be done by

$$\tau_{ij} = (1 - \zeta) \cdot \tau_{ij} + \zeta \cdot \tau_0$$

where $\zeta$ and $\tau_0$ are the 2 parameters and $\zeta$ will be in the interval between 0 and 1.

### 3.3 Max-Min Ant System

In Max Min Ant System (MMAS), the solution is constructed in the same way just like AS. Qualifying’s made in MMAS over AS are totally 3 in number. First is to exploit the best solution found during the iteration, after each iteration only one ant will be allowed to update the pheromone on the path that it travelled. This idea is just like in ACS. There are two possibilities of ants that update the pheromone trail and they are

i. Iteration best ant: ant which finds best solution in current iteration will be considered as iteration best ant.

ii. Global Best ant: ant which found the best solution from the beginning of the trial.

Second, to avoid stagnation of search space the pheromone trail is limited between the intervals $[\tau_{min}, \tau_{max}]$ on each solution. Third, the pheromone trails will be initially set to $\tau_{max}$ for exploration of more search space.

After all ants construct the solution the pheromone trails will be updated in the form

$$\tau_{ij}(t + 1) = (1 - \rho) \cdot \tau_{ij}(t) + \rho \cdot \Delta \tau^{gb}_{ij}$$

where $\Delta \tau^{gb}_{ij}(t) = 1/L^{gb}$.

Ant that is allowed to update the pheromone may be either Iteration best or Global best ant. If an arc often used by the ants to travel then that arc will take part in the global best solution since it receives most pheromone obviously. The use of any one solution i.e. either the Iteration best solution or Global best solution has a reason behind it. When using only the Global best ant to update the path then the search may concentrate too fast around this solution and the exploration of better paths may get limited which will lead to get trapped in poor quality of solutions. This can be nullified when choosing the Iteration best ants to update the pheromone since its solution may differ from iteration to iteration and many arcs can be explored.

**Trail Limits:** Lower and upper trial limits were imposed in MMAS in order to avoid the search stagnation. It will limit the ant when it is about to choose an arc to travel by eliminating that arc if it does not fall between the interval. The elimination is because that arc may not end in good quality of solution which has been learnt using the previous history of pheromone trails.

### 3.4 Model Induced Max-Min Ant System

This algorithm is a kind of hybrid algorithm which was been developed for Asymmetric Travelling Salesman
Problem (ATSP). It combines the complete analytical results of ATSP problem and Max-Min Ant Colony Optimization algorithm (MM-ACO)\(^{15}\). The contribution of MIMM-ACO is of on two aspects

i. Adjusted transition probabilities are developed by replacing static biased weighting factors with dynamic ones. The dynamic weighting factor is closely dependent on partial solution that ant has constructed. Idea behind it is that it favors the choice of edges with small residual cost instead of small actual cost. As a byproduct non optimal arcs will be identified at each step of tour construction using dual information derived from solving associated Assignment Problem (AP) and these arcs will be discarded from future consideration.

ii. A terminal condition is determined analytically based on the state of pheromone matrix structure. The result comes with a necessary condition for obtaining one optimal solution.

Algorithm of MIMM-ACO uses AP to calculate residual cost and PATCH algorithm is used to repair the solution given by AP to get lower bound. Then with the help of lower bound residual cost matrix \((Z_{AP})\) is found. This PATCH algorithm will return \(1^{st}\) candidate solution for MIMM-ACO (s1). Then the termination condition \(t_0\) and minimum pheromone value will be set in the following form

\[
\tau_{\text{min}} = \frac{1}{(\frac{1}{2})f(s_1) + (\frac{1}{2})Z_{AP}}
\]

and the gap will be set. After all the things initial minimum pheromone will be calculated dynamically and set. Then the tour will be constructed and then the local search will get used using 2-OPT heuristics. Best solution will be updated and also the gap.

### 3.5 Quantum Ant Colony Optimization

Quantum Ant Colony Optimization\(^\text{16}\) main motivation is to introduce the Q-bit representation and quantum rotation gate into ACO. It is to develop a discrete binary ACO algorithm and implement the hyper-cube framework.

The working of QACO is as follows:

After setting all the initialization of parameters the initial pheromone with \(\alpha = \beta = 1/2^{1/2}\) is

\[
\tau = \begin{bmatrix}
\tau_{1a} & \tau_{2a} & \cdots & \tau_{ma} \\
\tau_{1\beta} & \tau_{2\beta} & \cdots & \tau_{m\beta}
\end{bmatrix}
\]

A random number \(p\) is generated and it is compared with \(p_0\) which is the probability parameter.

If \(p<p_0\), the solution of \(i^{th}\) ant of \(j^{th}\) bit is

\[
\text{solution}_{i,j} = \begin{cases}
0 & \text{if } \tau_{j,a} \leq \tau_{j,a} \\
1 & \text{if } \tau_{j,a} > \tau_{j,a}
\end{cases}
\]

If \(p>p_0\), the solution of \(j^{th}\) ant of \(j^{th}\) bit is determined using the threshold function

\[
\eta_i(x) = \begin{cases}
0 & c < c_0 \\
1 & c \geq c_0
\end{cases}
\]

Then calculate the best function and then the termination condition satisfied and if the termination is not satisfied then update the pheromone density with the form

\[
\begin{bmatrix}
\tau_{ia}' \\
\tau_{i\beta}'
\end{bmatrix} = R(\theta) \begin{bmatrix}
\tau_{ia} \\
\tau_{i\beta}
\end{bmatrix} = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix} \begin{bmatrix}
\tau_{ia} \\
\tau_{i\beta}
\end{bmatrix}
\]

\(\theta\)-represents the rotation angle.

### 3.6 Cunning Ant System

Cunning Ant System (CAS) is different from traditional ACO’s. In traditional ACO algorithms ants will generate the solution based on the present pheromone trail \(\tau_i(t)\). But Cunning Ant System\(^\text{17}\) differs from those ACO algorithms in construction the solutions. It constructs the solution by borrowing a part of the solution from existing solutions. The remaining solution will be built based on traditional ACO algorithms as usual \(\tau_i(t)\).

Ant that gives donation of solution to a borrowing ant is called as donor ant (d-ant) and the ant that borrows the solution is called cunning ant (c-ant).

At a particular iteration \(t\) c-ant \(k,t\) borrows a part of existing solution from d-ant \(k,t\). Then c-ant \(k,t\) is compared with d-ant \(k,t\). The best among the two will be continued in the next iteration and then the pheromone density will be updated in the following form

\[
\tau_{ij}(t+1) = \rho \tau_{ij}(t) + \sum_{k=1}^{m} \Delta^*_{ij} \tau_{ij}(t)
\]

\[
\Delta^*_{ij}(t) = \frac{1}{C^*_{k,t}} \cdot \text{if } (i,j) \in \text{ant}^*_{k,t}; \ 0 : \text{otherwise}
\]

where \(C^*_{k,t}\) is the fitness value of \(\text{ant}^*_{k,t}\).

In cAS the maximum value of pheromone trail will be calculated of the form
\[ \tau_{\text{max}}(t) = \frac{1}{1 - \rho} \sum_{k=1}^{m} \left| C_{k,t} \right| \]

where the minimum is of the form

\[ \tau_{\text{max}}(t) = \frac{1}{1 - \rho} \frac{1}{C_{t}^{\text{best-so-far}}} \]

where \( C_{t}^{\text{best-so-far}} \) is the fitness of best-so-far solution at the iteration \( t \).

### 3.7 Cooperative Genetic Ant System

This algorithm is just like a hybrid algorithm which combines both the Genetic Algorithm and Ant System and presents here as Cooperative Genetic Ant System\(^1\) (CGAS). This algorithm executes both GA and AS concurrently and cooperatively. This gives a better chance in achieving global solution of TSP. This algorithm was been deployed \(^1\) in TSP.

Initial construction of tour is by using AS method. Using those new constructed tour the GA has been initialized. Then the construction of new solutions by AS and construction of new generations using GA has been made simultaneously. Among those two set of best solutions the global best will be chosen and then updating the pheromone will be done. After reaching the termination condition or global best the algorithm will get end\(^1\).

In this CGAS the next city will be selected using the formula

\[ j = \left\{ \begin{array}{ll} \min c(i) & \text{if } j \in s_k, \text{ar } f \max \tau_{ij}, \\rho \theta_i \text{ otherwise} \end{array} \right. \]

\( i \) – the current city, \( j \) – next city to be visited, \( c(i) \) the sorted list of cities with respect to adjacency.

### 4. Literature Survey

A close review of latest ACO and its variants implemented papers were been discussed here and the results of all those papers were been represented in fishbone and round diagrams to show the working methodology and consideration factors (parameters used) were been listed and the upcoming section shows the way that ACO applied in various NP hard problems to find the optimal results in polynomial time\(^5\).

Ant colony optimization for RDF chain queries for decision support, Alexander Hogenboom\(^2\) have suggested a new way to improve the data accessing speed faster than before. The previous algorithms did not give reasonable results and also the process is static in a dynamic environment where the data changes dynamically. The proposed work resolves scheduling problem and sequential ordering. And the way he achieved the aim is by applying the ACO foraging tech of ant the path has been developed in a shorter format and whenever the data changes its source address the ACO adapts accordingly. To achieve this result he have taken the parameters as the cost of each joins and the number of joins that were taken into account for resulting a particular query. The way he have transformed the ACO into the suggested system is he has taken the joins as the vertices of a graph where the ant travels along the path from vertex to another vertex and the path the ant chosen will be the flow of join and the cost will be calculated in such a way. The pros and cons of this paper: Pros: It handles the change over the data source and latency differences. Cons: For more join operations a lot of combinations of possible solutions are computed. (Just for 20 join operations 720,218 possible candidate solutions are formed)\(^3\).

Jing Xiao\(^3\), have suggested a way for solving the Software Project Scheduling Problems using the ant colony optimization (SPSP). The need of this technique is to make the project scheduling crisper by adding more parameters. Using the suggested idea of this paper we can reduce the project scheduling. This idea has been suggested since the previous work does not give accurate timing because, the previous works on this SPSP does not include the cost associated with the employees. The employee's several skills also taken into account in this SPSP solving using ACO\(^4\). It has been done by considering the employee dedication, Number of skills present in each employee, utilization of employee skills completely during the regular working hours. The parameters taken into account for solving the SPSP using ACO are the number of tasks, skills required number of employees, Vertex set, arc set, workload per task, etc. The transformation of the SPSP problem according to the suggested idea of solving SPSP using ACO is the employee's dedication is taken in format of vertices with edges. The dedication of the employees are scattered in the form of vertices of a graph. The employees are placed as columns. The ant travel in such a way it chooses one set of dedication from each employee and finally finishes the task. The pros and cons of this paper, Pros: The Results are accurate since we didn't leave any of the parameter and because of giving weight-age for the parameters which are in need. Cons: Only the employee dedication has been calculated on the
whole Project Scheduling. For this alone they are taking a lot of effort. When comes on calculating the whole SPSP the computational time would be more.

Jiajia He\textsuperscript{22}, suggested an idea for optimizing the traffic signal timing using Ant Colony Optimization. Need of this technique is to handle the traffic flow in rush hours and free hours. This idea has been suggested because, the time delay is more when there is less flow of traffic and the number of stops also more and the traffic capacity is less or constant even in rush hours. It has been done by optimizing the signal timing, by reducing the time delay and number of stops in free hours, and by improving the traffic capacity in Rush hours. The parameters taken in this paper are saturation flow, time delay, number of stops, traffic capacity and the weighting coefficients of all these parameters. The transformation of the Traffic signal problem which needs to be solved using ACO to the suggested idea is the ants are taken as vehicle that crosses the signal. The pheromone that it secrets along the path will be considered as the path or optimal solution to handle the traffic signal at various situations like What to do when there is low traffic? And how to optimize the time at Large flow of traffic? A new formula has been derived along with the coefficients of those parameters and also the parameters are taken into account. Implement the formula and the simulation with some numerical values has been done. The pros and cons of this paper are Pros: The evaporation of pheromone has been considered as one of the idea and the iteration has been done in some other paths which may lead to optimal solution. Cons: If the pheromone updating has not been done in optimal solution which convergence to optimality that drops down may lead to non optimal solution\textsuperscript{5}.

Julei Ding\textsuperscript{23} have suggested a new idea for solving the Vehicular Routing Problem with Time Windows using an improved ant colony optimization (VRPTW)\textsuperscript{6}. The need of this technique is to improve the vehicular routing problem which needs an appropriate way to travel via the vehicles. This idea has been suggested because the previous algorithms often get trapped in local optima and premature convergence and low search speed. It has been done by adjusting the pheromone to avoid local optima, the disaster operator to search the solution space completely, to minimize the candidate list in order to reduce the computational time and finally the use of Solving Algorithm and Interchange Algorithm in order to improve the convergence speed\textsuperscript{7}. The parameters that were taken into account to solve the VRPTW are the number of customers, density of pheromones, density of visibility, and by taking the candidate list as $1/4^\text{th}$ of the total customers. The VRPTW problem which needs to be solved using improved ACO has been transformed to the suggested idea, each constraint in VRPTW is encoded with a formula, each parameter is initialized with a numerical value and the evaluation is made using a format, ants have been taken as vehicles, Customers have been taken as Nodes, and the distance between 2 customers will be considered as the travelling time from one city to other. The pros and cons of this paper, Pros: The Pheromone density value set dynamically avoids the local optima. Cons: Using these many algorithms increase the computational time heavily which gives very minimal reduction of errors compared with previous algorithm\textsuperscript{8}.

Figure 4. Fish Bone diagram illustrates the consideration factors that all the NP hard problems have been taken to find out optimal solution and the technologies used to solve the problem and also the comparison of the implemented algorithm with existing ones. The partial rectangle indicates the author of that particular paper who solved that NP hard problem using the ACO variant and compared with other variants\textsuperscript{9}.

JieBai\textsuperscript{24}, have suggested a model induced max-min ant colony optimization for solving asymmetric traveling salesman problem. The need for this technique is to utilize the partial solution, to deduce the word probabilistic and to increase the computational speed\textsuperscript{10}. It has been done using analytical proof, by utilizing the best among the lower bound partial solutions, and the bound of the max min ACO reduces the search space. The parameters that have been taken into account are the vertex of a graph, edge between two vertices, the cost to travel from one node to the next node, and the partial solution that produced which have some cities visited and some cities unvisited. Ants are put into each city (i.e. Vertex), the travelling between nodes i to j has been calculated and kept and it has been considered as actual cost, The Residual cost has been calculated using Assignment Problem and used for further consideration of the Partial Solution. Pros and cons of this paper, Pros: By choosing the adjacent nodes instead of taking all nodes for computation, it reduces nearly half of the computational time. Cons: The computational time is more since we use Assignment Problem (AP) for calculating residual cost of order $O n^2$ and also the PATCH algorithm to resolve the Assignment Problems values\textsuperscript{11}.

Gaifang Dong\textsuperscript{18}, suggested an idea for solving the Traveling Salesman Problem (TSP) using Cooperative Genetic Ant Systems (CGAS). The need of this technique is to make the diversity in all iteration in order to explore
more solution space in TSP. This idea has been suggested since in previous algorithm only the best known solution space has been taken into the next iteration which will leave the unsearched nodes as such which will lead to get into local optima12. It has been done by computing Genetic Algorithm (GA) new population in all iterations in parallel and comparing it with the Ant System (AS) best known solutions and taking the minimum value in these two to the next iteration. The existing problem has been converted to a form where it can be solved using the suggested idea in the following manner, The GA and AS computed in parallel and the comparison is made. The minimum value is replaced with the old value and taken to the next iteration. The pros and cons of this paper are,

Pros: More Solution space is explored using GA which yields global optimum. Cons: Iterations give good results only at a particular parameters value.

R.F. Tavares Neto25, suggested an idea to solve the Permutational Flowshop Scheduling Problem with outsourcing allowed using Ant Colony Optimization (FSACO). Need of this technique is to find the jobs to be outsourced with the constraint of budget cost for
outsourcing\textsuperscript{13}. This idea has been suggested because, without this algorithm the problem need human interaction and the budget constraint which needs more computational time to find a solution. The proposed method has been done using ACO algorithm for finding the job to be outsourced by giving weight age to all the constraints and using Max-Min Ant System for updating the pheromone which results in better convergence. The parameters used are the processing time for a particular job at a particular machine, number of operations which will be proportional to machines, outsourcing cost of particular job, outsourcing budget cost, outsourcing lead time for a particular job, pheromone density, and the visibility of ants\textsuperscript{15}. The existing problem has been transformed to the suggested idea in the following manner; a graph structure of the problem and represented the FSACO algorithm is applied in 2 stages. The first stage is to deduce whether the job to be outsourced or not. The second stage is to order the in-processing job with the machine. The pros and cons of this paper, Pros: Since Invariance of results over certain period of time is not there we can use it for other problems also. Cons: this idea cannot be used of Multiple Tardiness problem and all\textsuperscript{14}.

Zhengxing Huang et al.\textsuperscript{26}, have suggested an idea for solving Resource Allocation Optimization in Business Process Management using ACO\textsuperscript{16}. The need of this technique is to reduce the complexity in allocating the resources and Data Object to do a particular task. It has been done by using ACO the optimal path has been calculated along with considerations of constraints. The parameters used for solving this problem are the set of processes, the set of resources and its availability, the cost of each process and the date object that is available. The transformation of problem which needs to be solved using ACO is in the following manner; the operations, Resources and data objects are available in a tree like structure, the ants are allowed to travel in the path in that tree, the Maximum availability of pheromone of the iteration will be chosen as best path until any other best solution found\textsuperscript{17}. The Pros and cons of this paper; Pros: The global optimal solution with some constraints including Availability and cost of the operations has been derived. Cons: For Multi-tasks with minimum number of resources cannot be done efficiently\textsuperscript{18}.

Figure 5 shows the numerical values of the output that was been generated because of implementation of ACO variants on NP Hard problems\textsuperscript{19}. The innermost circle indicates that all the work were been a part of ACO. And the next circle from the innermost indicates the problems that were been solved using ACO. The next 2 circles indicate the output like duration time, hit ratio when implementing ACO variants onto the problem\textsuperscript{20}. Outermost circle indicates variants of ACO which were been implemented on the respective NP hard problems.

5. Success and Error Rate

Success and error rate of the papers that taken for calculating Hit rate, Duration, Efficiency are represented in the form of graph and compared with the other success and error rate of other NP hard problems in order to find out the performance of ACO algorithm\textsuperscript{21}. It has been compared in 2 sets as Permutational Flowshop Problem, Vehicular Routing Problem, Asymmetric Travelling Salesman Problem, Travelling Salesman problem s set1. This has been done so that the comparison and the representation of graph will be meaningful since the margin between set1 and the next set having different margins of success and error rates\textsuperscript{22}.

Figure 6 illustrates success rate of PFSP, VRP, ATSP, and TSP when applying ACO variants to find out optimal solution.

Figure 7 illustrates error rate of PFSP, VRP, ATSP, and TSP when applying ACO variants to find out optimal solution. Error rate of Set 1 is of the range from 0 to 2.34 % to a maximum but set 2 ranges up to 80% of error rate\textsuperscript{23}.

Set 2 consists of the values of RDF chain queries decision support problem, Software Project Scheduling Problem, Traffic Signal Timing Optimization Problem, Task Operation Management. These values of error and success rate were been compared in order to find the performance of ACO on NP hard problems\textsuperscript{24}.

Figure 8, and 9 illustrates success and error rate of RDFCQ, TSTO, TOM, and SPSP when applying ACO variants to find out optimal solution.

6. Literature Review Methodology

We began our search for article in the literature concerning ACO by exploring a lot of ways via Google search. ACM Digital Survey, Science direct, IEEE using the combination of keywords (Example: ACO implemented on various domains, ACO techniques, ACO solved papers, NP hard problems, etc.)\textsuperscript{25}. A lot more review has been made on internet for the addition of papers to the corpus. And for citation of reference papers we have undergone a set of search on all the surveys and new techniques devised on ACO.
Among those a lot of papers have been filtered since they may not support for the literature survey on ACO. And the initial analysis of papers, we have identified a lot of novel approaches on new techniques of ACO but they are not with any kind of experimental proof over a problem. Instead of a practical application they have proved it with theorem based and benchmark problems and hence they are not included in this literature survey.26

6.1 Time Line Diagram of Surveyed Papers
Figure 10 shows the time line diagram of surveyed papers which are discussed in this paper. Time line diagram
Figure 5. Whether job R.F. Tavares or Neto, 0.1. 2. 3. 4. 8. 9. 9 9 9 9 9 9 0 0 illustrates 10% 2 3 4 4 8 9 lrd problems represented in the fo

Figure 6. Success rate of Set 1.

Figure 7. Error rate of set 1.

Figure 8. Success rate of set 2.

Figure 9. Error rate of set 2.

Considering the number of employees and their skills. Allocating the tasks to the employees. The number of tasks that can be handled by an employee in a day. Considering only the working time for a day. The dedication factor of an employee.

Figure 10. Time Line Diagram.

R.F. Tavares Neto, et al. (2011)
6. Discussions

This literature survey reviewed the problems that have been optimized by ACO. Each problem varied in the way it represented and the way it has been solved or handled.

7. Observations

Figure 10 shows the number of papers surveyed and a total of 8 papers range from the year 2011 to 2013. The given survey reflects the knowledge on ACO, with deep finding of how ACO algorithm works, the heuristic handling, exploration of paths are all reviewed. And also the Assignment Problem, PATCH algorithm, Tabu Search, Saving Algorithm, Disaster Operator, Interchange Algorithm and some more new terms have been defined. The main theme focused on how ACO evaluated its working and its results of discussed problems. The consideration factors would be execution time, solution cost, solution quality, the initial population and many more things have been projected.

8. Research Directions

ACO has been considered as the stochastic probabilistic search which gives a lot of research directions on NP hard problems. Through this survey work the problem representation, the path exploration and exploitation, termination conditions, pheromone variations, usage of heuristics have been discussed. The future research directions can be in the way of paralleling the execution and exploration of more search space, efficient way of using the heuristics, the pheromone evaporation factor, initial candidate solution, initial number of ants chosen in the way the premature convergence should not occur.

9. Glossary

**Stigmergy:** Indirect communication which leads to some useful information regarding food or source of interest.

**Candidate Solution:** initial solution given by the user or from previous iteration for continuing the process.

**Swarm Intelligence:** An Artificial Intelligence (AI) technique based on the collective behavior in decentralized, self-organized systems.

**Stochastic Optimization:** New solutions will be attained at every iteration.

**Meta Heuristics:** Heuristics about a Heuristic.

**Polynomial Time:** Time taken to solve the problem would be based on the problem size.

---

**Table 1. Tabular column of correlation of ACO variants and NP hard Problems**

| Author Name              | Pheromone Updating Methods | AS | ACS | Max Min | SPU | RPU |
|--------------------------|----------------------------|----|-----|---------|-----|-----|
| Alexander Hogenboom et al. |                            |    |     |         |     |     |
| Jing Xiao, et al.        |                            | ✓  |     |         |     |     |
| Jiajia He, et al.        |                            |    |     | ✓       |     |     |
| iulei Ding, et al.       |                            |    |     | ✓       |     |     |
| JieBai, et al.           |                            |    |     |         | ✓  |     |
| Gaifang Dong, et al.     |                            | ✓  |     |         |     |     |
| R.F. Tavares Neto, et al.|                            | ✓  | ✓   | ✓       |     |     |
| Zhengxing Huang, et al.  |                            | ✓  | ✓   | ✓       | ✓  |     |

---

**Keywords**

| Paper Authors            | Keywords                                                                 |
|--------------------------|--------------------------------------------------------------------------|
| Alexander Hogenboom, et al. | RDF chain query optimization, Ant colony optimization, Genetic algorithm, Iterative improvement, Simulated annealing |
| Jing Xiao, et al. (2013)    | Scheduling, Automatic software management, Software project scheduling, Ant colony optimization |
| Jiajia He, et al. (2012)    | Signal timing optimization, Ant colony algorithm, Webster algorithm, Time delay, Number of stops, Traffic capacity |
| iulei Ding, et al. (2012)   | Ant colony optimization, Vehicle routing problem with time windows, Combinatorial optimization problem |
| JieBai, et al. (2012)       | Hybrid evolutionary algorithm, Ant colony optimization, ATSP, Model induce |
| Gaifang Dong, et al. (2012) | Ant colony optimization, Ant system, Genetic algorithm, Traveling salesman problem |
| R.F. Tavares Neto, et al. (2011) | Ant colony optimization, Flowshop scheduling problem, Outsourcing |
| Zhengxing Huang, et al. (2012) | Ant Colony Optimization, Business Process, Optimization, Resource Allocation, Task Operation Model |
Variants of Ant Colony Optimization-A State of an Art

10. References

1. Beni G, Wang J. Swarm Intelligence in Cellular Robotic Systems. Proceed. NATO Advanced Workshop on Robots and Biological Systems; Tuscany: Italy. 1989 Jun. p. 26–30.
2. Dorigo M, Blum C. Ant colony optimization theory: A survey. Theoretical Computer Science 344. 2005. p. 243–78.
3. Buck M. Critical Survey of Ant Colony Optimization for Hard Discrete Optimization Problems. 2005.
4. Vijayaragavan SP, Karthik B, Kiran Kumar TVU, Sundar Raj M. Robotic surveillance for patient care in hospitals. Middle - East Journal of Scientific Research. 2013; 16(12):1820–4. ISSN: 1990-9233
5. Blum C. Ant colony optimization: Introduction and recent trends. Physics of Life Reviews. 2005. p. 353–73.
6. Pedemonte M, Nesmachnow S, Cancela H. A survey on parallel ant colony optimization. Applied Soft Computing 11. 2011. p. 5181–97.
7. Duan H, Yu Y, Zou J, Feng X. Ant Colony Optimization-based bio inspired hardware: survey and prospect. Transactions of the Institute and Control. 2012. p. 318–33.
8. Chandra Mohan B, Baskaran R. A survey: Ant Colony Optimization based recent research and implementation on several engineering domain. Expert Systems with Applications 39. 2012. p. 4618–27.
9. Colorni A, Dorigo M, Maniezzo V. Distributed Optimization by Ant Colonies. Proceedings of the First European Conference on Artificial Life (ECAL 91). Elsevier; 1991. p. 134–42.
10. Vijayaragavan SP, Karthik B, Kiran Kumar TVU, Sundar Raj M. Analysis of chaotic DC-DC converter using wavelet transform. Middle - East Journal of Scientific Research. 2013; 16(12):1813–9 ISSN: B27.
11. Dorigo M. Optimization, Learning, and Natural Algorithms. [PhD Thesis], Dip. Elettronica, Politecnico di Milano, 1992.
12. Dorigo M, Maniezzo V, Colorni A. Positive Feedback as a Search Strategy. Technical Report 91-016, Dip. Elettronica, Politecnico di Mi-lano, 1991.
13. Dorigo M, Gambardella LM. Ant Colony System: A cooperative learning approach to the travelling salesman problem. IEEE Transactions on Evolutionary Computation. 1997; 1(1):53–66.
14. Gambardella LM, Dorigo M. Solving Symmetric and Asymmetric TSPs by Ant Colonies. Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC’96), IEEE Press; 1996. p. 622–7.
15. Stautzle T, Hoos HH. The MAX-MIN ant system and local search for the travelling salesman problem. In: Baeck T, Michalewicz Z, Yao X. Editors. Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC’97). 1997. p. 309–14.
16. Vijayaraghavan K, Nalini SPK, Prakash NU, Madhankumar D. Biomimetic synthesis of silver nanoparticles by aqueous extract of Syzygium aromaticum. Materials Letters. 2012; 75:33–5. ISSN; 0167-577X.
17. Stautzle T, Hoos HH. Improvements on the Ant System: Introducing the MAX-MIN Ant System. In: Albrecht RF, Smith GD, editors. Artificial Neural Networks and Genetic Algorithms. Springer Verlag; Wien New York. 1998. p. 245–9.
18. Bai J, Yang G-K, Chen Y-W, Hua L-S, Pan C-C. A model induced max-min ant colony optimization for asymmetric travelling salesman problem. Applied Soft Computing. 2012; 13(3):1365–75.
19. Wang L, Niu Q, Fei M. A novel quantum ant colony optimization algorithm. Springer-Verlag Berlin Heidelberg. 2007. p.277-86.
20. Vijayaraghavan K, Nalini SPK, Prakash NU, Madhankumar D. One step green synthesis of silver nano/microparticles using extracts of Trachyspermum ammi and Papaver som-niferum. Colloids and Surfaces B: Biointerfaces. 2012; 94:114–7. ISSN: 0927-7765.
21. Tsutsui S. cAS: Ant Colony Optimization with Cunning Ants. Springer-Verlag Berlin Heidelberg. 2006. p. 162–77.
22. Dong G, Guo WW, Tickel K. Solving the travelling salesman problem using cooperative genetic ant systems. Expert Systems with Applications 39. 2012. p. 5006–11.
23. Michel R, Middendorf M. An island model based Ant System with look ahead for the shortest super sequence problem. Proceedings of PPSN-V, Fifth International Conference on Parallel Problem Solving from Nature, Berlin: Springer-Verlag. 1998. p. 692–701.
24. Hogenboom A, Frasinar F, Kaymak U. Ant colony optimization for RDF chain queries for decision support. Proceedings of Expert Systems with Applications, Science Direct; 2012.
25. Xiao J, Ao X-T, Tang Y. Solving software project scheduling problems with ant colony optimization. Proceedings of Computers and Operations Research 40, Science Direct; 2013. p. 33–46.
26. Kulanthaivel L, Srinivasan P, Shanmugam V, Periyasamy BM. Therapeutic efficacy of kaempferol against AFB1 induced experimental hepatocarcinogenesis with reference to lipid peroxidation, antioxidants and biotransformation enzymes. Biomedicine and Preventive Nutrition. 2012; 2(4):252–9. ISSN: 2210-5239.
27. He J, Hou Z. Ant colony algorithm for traffic signal timing optimization. Proceedings of Advances in Engineering Software 43, Science Direct; 2012. p. 14–8.
28. Ding Q, Hu X, Sun L, Wang Y. An improved ant colony optimization and its application to vehicle routing problem with time windows. Neurocomputing, Science Direct; 2012.
29. Bai J, Yang G-K, Chen Y-W, Hua L-S, Pan C-C. A model induced max-min ant colony optimization for asymmetric...
travelling salesman problem. Applied Soft Computing, Science Direct; 2012.

30. Tavares Neto RF, Godinho Filho M. An ant colony optimization approach to a permutational flowshop scheduling problem with outsourcing allowed. Computers and Operations Research 38, Elsevier Jounal. 2011. p. 1286–93.

31. Huang Z, Lu X, Duan H. A Task Operation Model for Resource Allocation Optimization in Business Process Management. IEEE Transactions on Systems, Man, and Cybernetics- Part A: Systems and Humans. 2012 Sep; 42(5).

32. Kimio T, Natarajan G, Hideki A, Taichi K, Nanao K. Higher involvement of subtelomere regions for chromosome rearrangements in leukemia and lymphoma and in irradiated leukemic cell line. Indian Journal of Science and Technology. 2012 April; 5(1):1801–11.

33. Cunningham CH. A Laboratory Guide in Virology. 6th ed. Minnesota: Burgess Publication Company; 1973.

34. Sathish Kumar E, Varatharajan M. Microbiology of Indian Desert. In: Sen DN, editor. Ecology and Vegetation of Indian Desert. India: Agro Botanical Publishers; 1990. p. 83–105.

35. Varatharajan M, Rao BS, Anjaria KB, Unny VKP, Thyagarajan S. Radiotoxicity of Sulfur-35. Proceedings of 10th NSRP; India. 1993. p. 257–8.

36. 01 Jan 2015. Available from: http://www.indjst.org/index.php/vision