© 2015 IJEDR | Volume 3, Issue 2 | ISSN: 2321-9939

Hybrids of Ant Colony Optimization Algorithm- A Versatile Tool

[1] Preeti Tiwari,[2]Anubha Jain
[1]Senior Assistant Professor, [2] Head of Department
[1]Computer Science, [2] CS & IT
[1]International School of Informatics and Management, Jaipur, India
[2]The IIS University, Jaipur, India

***Abstract* - Ant Colony Optimization Algorithm is a meta-heuristic, multi-agent technique that can be applied for solving difficult NP-Hard Combinatorial Optimization Problems like Traveling Salesman Problem (TSP), Job Shop Scheduling Problem (JSP), Vehicle Routing Problem (VRP) and many more. The Positive Feedback Mechanism and Distributed Computing ability makes it very robust in nature. The artificial ants implement a randomized construction heuristic which makes probabilistic decisions as a function of artificial pheromone trails to solve the problems that are dependent on the input data. In spite of ACO having global searching ability and high convergence speed towards optimal solutions, it has some limitations like low population scattering ability and no systematic way of startup. To overcome these problems, various hybrids of ACO with other algorithms like Dynamic Programming, Genetic Algorithm and Particle Swarm Optimization have been proposed to provide better results than using ACO in isolation. This paper studies various approaches for the development of Hybrids of ACO Algorithm for different types of applications and effects thereof.**

***IndexTerms* - Ant Colony Optimization Algorithm, Pheromones, Genetic Algorithm, Particle Swarm Optimization**

I. INTRODUCTION

Ant colonies commonly known as social insect societies are distributed systems that, in spite of the simplicity of their individuals, present a highly structured social organization. The behavior of these ants is directed towards the survival of colony as a whole than that of a single individual of the colony [1]. Three Italian Scholars- Colorni A, Dorigo M and Maniezzo V developed Ant Colony Optimization Algorithm in 1991. Ant Colony Optimization Algorithm is a meta-heuristic, multi-agent technique that can be applied for solving difficult NP-Hard Combinatorial Optimization Problems like Traveling Salesman Problem (TSP), Job Shop Scheduling Problem (JSP), Vehicle Routing Problem (VRP), Signal Processing Problems, Artificial Life Problems and many more[2][3].

An interesting behavior of the ant colony is its indirect co-operative foraging activities that tend to choose their shortest path between food and nest on the basis of strong pheromone concentration. While searching for food, real ants use a material called pheromone to communicate with other ants. When an ant chooses to randomly travel from nest to food, they use this pheromone information as a guide to explore the search space. Fig. 1 presents the process by which ants choose their paths [5]. Some ants choose to travel on one side of the path while others randomly travel on the other side of the path.

Assuming that these ants crawl at the same speed, the trails with lack of pheromone concentration are less performed random walk. Ants traversing on shorter path increase the concentration of the pheromone much quickly than those ants traversing on longer paths. However, as soon as ants sense a trail with higher pheromone concentration in their vicinity, they tend to follow that path, thus reinforcing this trail and as a result the entire colony converges to the shortest route.

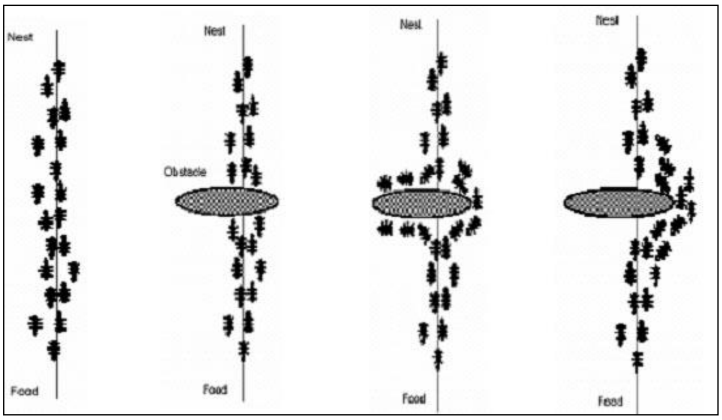

Fig. 1: Ants Traversal Path [5]

The fundamental approach underlying ACO is an iterative process in which a population of simple agents repeatedly constructs candidate solutions. These solutions construction process is probabilistically guided by heuristic information on the given problem instance as well as by a shared memory containing experience gathered by the ants in previous iteration. This mechanism is known as Positive Feedback Mechanism that enables the use of artificial intelligence techniques on exponentially grown solution spaces to generate Global Optimization Solutions [4].

The artificial ants build a solution by traversing the fully connected construction graph G = (V, E) where V is the set of vertices and E is the set of edges. The ants move from vertex to vertex along the edges of the graph to build partial solutions. The artificial ants in ACO implement a randomized construction heuristic which makes probabilistic decisions as a function of artificial pheromone trails to solve the problems that are dependent on the input data.

II. ALGORITHM

The algorithm given below (Fig. 2) explains the working of Ant Colony Optimization Problem with Metaheuristics that iterates over three phases [6]:

A. ConstructAntSolution: *m* artificial ants construct multiple solution elements from a finite set of available solution components.

B. ApplyLocalSearch: Once solutions have been constructed, and before updating the pheromone, this function improves the solutions obtained by the ants through a local search.

C. UpdatePheromones: It increases the pheromone concentration values associated with promising solutions, and also decreases the pheromone concentration values associated with bad solution through pheromone evaporation.

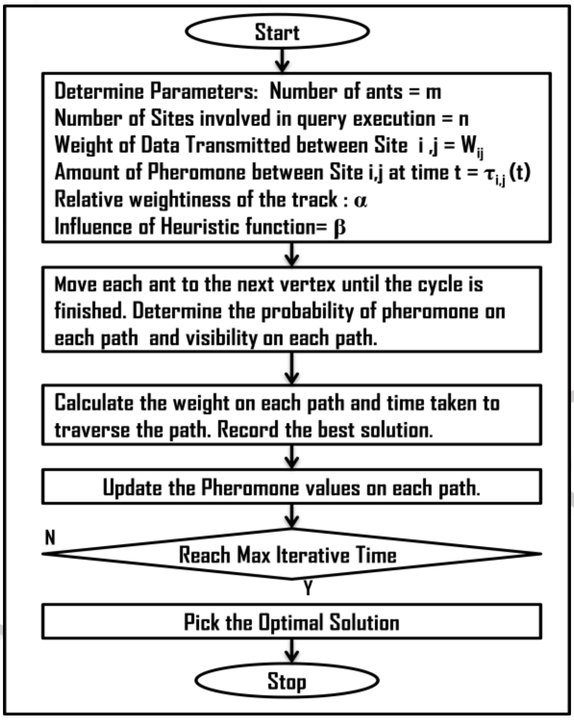

Fig. 2: Meta-heuristic Algorithm for ACO [6]

III. LIMITATIONS OF ACO

In spite of ACO having special properties like Parallel Processing, Distributed Computing, Global Search ability, high convergence speed, to name a few, ACO has some limitations associated with it [7][8][10]:

A. The initial formation needed by ACO does not have a systematic way of startup. The ants randomly run in all directions in search of food hence leading to low population scattering ability. This leads to an increase in the time taken to generate optimal solutions.

B. The convergence speed of ACO is lower at the beginning because of little pheromone concentration on the traversal paths of the ants. This also leads to an increase in time taken to create a set of promising solutions. However the convergence speed increases towards optimum answer because of positive feedback mechanism.

To overcome these shortcomings, various hybrids of ACO with other algorithms like Dynamic Programming, Genetic Algorithm and Particle Swarm Optimization Algorithm have been proposed to provide better results for Combinatorial Optimization Problems. This paper studies the hybrids of ACO, its analyses and effectiveness for various complex problems.

IV. HYBRIDS OF ACO

This paper integrates various algorithms with ant colony algorithm which provides a systematic way of startup in the beginning and to initialize pheromone distribution so that the speed, time and precision can be increased of the solutions. Section IIIA analyzes hybrids of ACO with Dynamic Programming, Section IIIB explores the hybrids with Genetic Algorithm and Section IIIC discusses the hybrids with Particle Swarm Optimization Algorithm.

A. Hybrid of ACO and Dynamic Programming

Query Optimization Problems in databases were being solved by implementing Dynamic Programming (DP) which simplifies a complicated problem by breaking it down into simpler sub-problems and decision making steps in a recursive manner. It follows "Divide and Conquer" approach for programming. In Distributed Database Management System with the increase in the number of relations, the number of joins also increases in the query. This leads to the creation of a large number of Query Execution Plans which can be considered as viable solutions for query processing. Dynamic Programming leads to extended execution time with large memory requirements. This leads to the problem of traceability as the size of the problem increases exponentially. Tansel et.al. [9] proposed a hybrid Algorithm of DP-ACO (Dynamic Programming-Ant Colony Optimization) as a Search Algorithm in Distributed Database for query optimization problem of multi way chain equijoin queries. Dynamic Programming alone can perform optimization on seven relations but DP-ACO have proved to be viable solution by producing good execution plans with 15 way join queries within limited time and very limited memory space.

B. Hybrid of ACO and Genetic Algorithm

Based on the principles of Charles Darwin-"Survival of Fittest" [10], Genetic Algorithms (GA) are powerful random search techniques based algorithms that implement the process of natural selection and genetic selection to solve problems of multiple disciplines. Proposed by John Holland in 1992, Genetic Algorithm belongs to a large class of Evolutionary Algorithm that generate solutions to optimization problems using techniques like representation, selection, evaluation, inheritance, mutation, and crossover [11]. GA has strong adaptability, robustness, quick global searching ability with higher population scattering ability for extensive amplitude of answers [7].

A Genetic Algorithm functions by generating a large set of possible solutions called chromosomes to a given problem. It then evaluates each of those solutions, and decides on a "fitness level" for each solution set [12]. Each chromosome consists of genes. GA adopts a method for moving from one population of chromosomes (strings of bits) to a new population by using operators of recombination, mutation and inversion to breed new solutions. The parent solutions that are more "fit" are more likely to reproduce offspring, than those that were less "fit". The design of GA includes [13]:

1. Initialization: Many individual solutions are randomly generated to form an initial population which is depended on the nature of the problem. These populations are often in the form of bit strings called chromosomes or genotypes.

2. Selection: A portion of the existing population is selected to breed a new generation of parents and off- springs on the basis of "Fitness-Function" which is defined over the genetic representation to measure the quality of the represented solution.

3. Genetic Operators: The three Genetic Operators viz. crossover, mutation and inversion are used to produce the next generation population of solutions.
   a. Crossover: It is primary Genetic Operator that combines segments of more than one parent solution to generate new off-springs. This can be achieved by randomly selecting cut points in the parents and concatenating their segments.
   b. Mutation: It generates a new solution by randomly modifying one or more gene values of the existing solution. This is to prevent falling all solutions in population into local optimum of solved problems.
   c. Inversion: It generates a new solution of off-springs by reversing the gene order of the existing solution. This can be achieved by choosing two cut points at random and switching them.

4. Termination: Over successive iterations, the solutions in the pool are evaluated using the Fitness-Function and solutions with food schemata are retained to generate optimal solution off-springs. The Genetic Algorithm stops when either the fixed number of generations is reached or the Computational Time/Cost Budget is reached or the percentage difference between the performance of best and worst solution is negligible.

Fig.3 given below represents the general flowchart of GA.

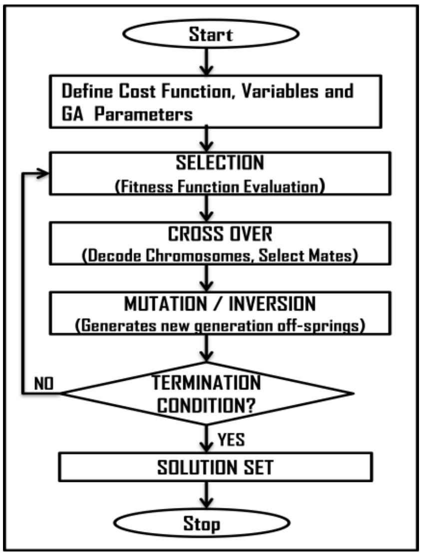

Fig 3: General Flowchart of Genetic Algorithm [11]

The Genetic Algorithm has some deficiencies [14]:

The convergence speed of GA is slower towards optimum solution as compared to ACO.

The "better" solution is only in comparison to other solutions. As a result, the stop criterion is not clear in every problem.

The above study concludes that though Genetic Algorithm has quick global searching ability but it suffers from premature convergence and low convergence speed towards optimum solution. Ant Colony Optimization Algorithm has high converging speed on the optimization path through pheromone accumulation and renewal [14] but has low convergence speed at the beginning for there is only little pheromone difference on the path at that time [15].

To overcome the deficiencies of Ant Colony Optimization and Genetic Algorithm, a combined solution of both is being proposed. The fusion algorithm adopts the strengths of GA and ACO to generate optimal solutions in minimum amount of time and cost [15][16]. It develops enough advantage of the two algorithms. The hybrids of ACO and GA proposed by scientists for different types of applications are given below as follows:

- A combination of GA and BWAA (Best Worst Ant Algorithm) was proposed that utilizes the global searching of genetic algorithm to initialize pheromone distribution in ACO and Ant Colony Algorithm ensures the precision of the solution by using the positive feedback mechanism to achieve high convergence speed and low execution time in Multi-Join Query Optimization in relational database to seek the best join order among the tables. The Hybrid Algorithm proved to be a viable solution to generate time efficient results for relational queries to gain optimal solution. This algorithm can also be applied to handle NP-Hard Problems of query optimization of Distributed Database [16].

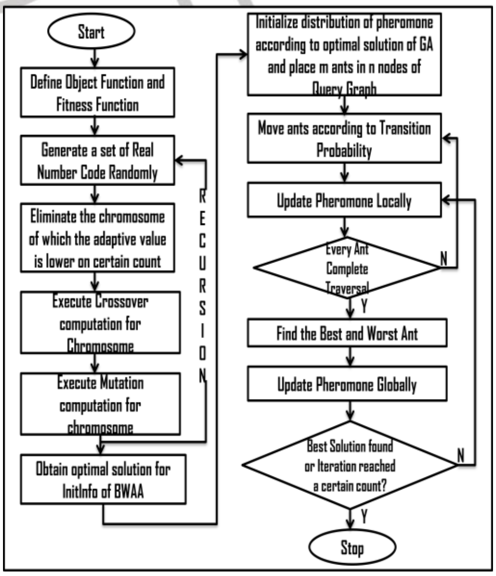

Fig. 4: Hybrid Algorithm of GA-ACO [16]

- Zohu et.al [17] proposed a Fusion Algorithm of Genetic Algorithm and Ant Colony Optimization Algorithm to resolve the optimal solution of a bi-level multi-objective optimization model in Traffic Signal Control Problem (Fig. 5). In the bi-level optimization model, the objective function of the upper level is to minimize the automotive exhaust emission at the intersection and objective of lower-level is to balance the traffic flow using user-equilibrium model. The problem is handled by breaking down of pollution parameter into running emission and idle emission at the entrance of each intersection.

  The algorithm utilizes the solutions of genetic algorithm as the initial value of ant colony algorithm. The global searching capability of genetic algorithm and the positive feedback of ant colony algorithm work together to converge faster and to gain optimal solution. The computed result showed that the model and the fusion algorithm are effective in optimizing signal timing and reducing automotive exhaust emission.

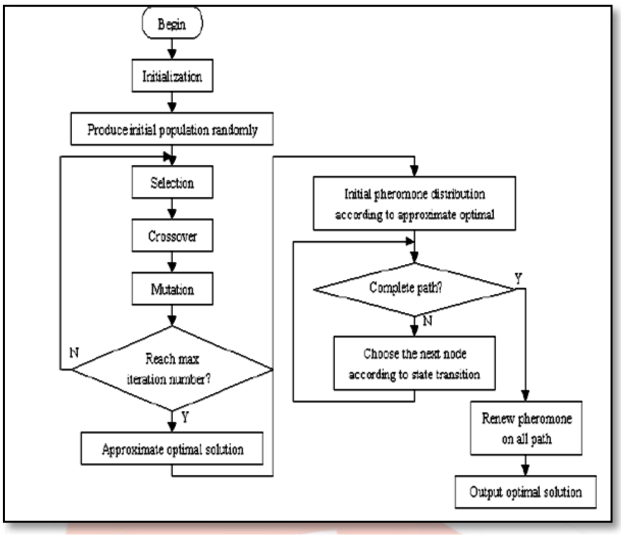

Fig 5: Fusion Algorithm of GA-ACO for Traffic Signal

- Aravind and Micheal [18] proposed a hybrid of ACO and GA to find the shortest path with help of distance-value in Wireless Mesh Network. WMN is a promising wireless technology for several emerging and commercially interesting applications like broadband home networking, community and neighborhood networks, coordinated network management, intelligent transportation systems which provide wireless broadband service access at a reasonable cost. The algorithm implemented ACO to maintain the routing table in a simple and understandable form and explored the solution space in multiple directions at once to find all possible paths. These set of paths obtained as output by ACO were given as input to the GA. GA works efficiently for solving problems where the solution space is huge and time taken to search exhaustively is very high. The genetic algorithm undergoes the selection, crossover and mutation process and generates the result. The result contains only one path which is optimal among the shortest path. The algorithm of the proposed hybrid of ACO-GA is given below.
  - START
  - If (current node = destination node) then
    - save the path & update the routing table
  - else if (nodes have same distance values)
    - Consider all the nodes of same value
    - Update the routing table & Send it to next node
  - else if (nodes available)
    - Update the routing table &Send it to next node
  - Repeat these steps until it reach the destination node.
  - Read number of paths from saved path (n = number of paths)
    - fitness = **Evaluate fitness** (cost)
    - Perform Cross Over
    - Perform Mutation & generate New Node
    - fitness solution = correct path
  - print correct path
  - END
- Na Li et.al [19] proposed a hybrid of GA and ACO for Vehicle Routing Problems (VRP) with the objective to minimize the servicing cost without exceeding the capacity of the vehicles. A typical VRP can be stated as designing of least cost routes from a central depot to a set of dispersed points (customers, stores, cities, warehouses, etc.) with various demands wherein each customer is serviced exactly once. VRP is similar to dynamic logistics resource allocation problem where the mechanism of allocating logistics resources (containers, vehicles etc.) are quickly responded to the variety of customer order demands which changes in short-term time intervals. In the given algorithm (Fig. 6) the GA performs a series of selection, crossover and mutation and then pheromone update of ACO is performed to move out of local optima. The experimentation

result showed the algorithm achieved significant results by reducing the number of iterations and increasing the convergence speed of optimization and hence taking less time to solve large-scale VRP.

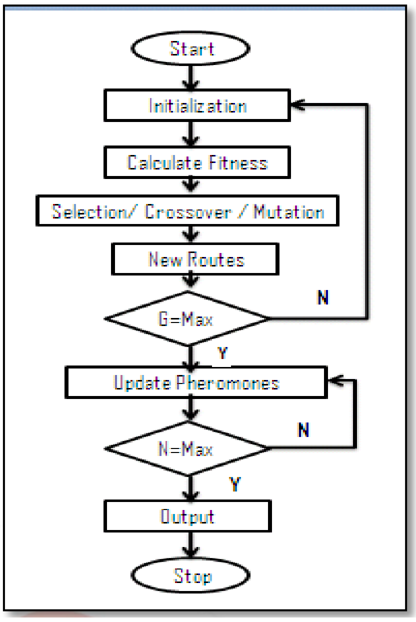

Fig 6: Hybrid Algorithm of GA-ACO for VRP [17]

- A hybrid algorithm of Genetic Algorithm and Ant Colony Optimization (GA-ACO) was introduced by [15] to solve the problems of optimization of join ordering (only nested loop joins considered) in relational database queries by overcoming the shortcomings of both the algorithms. The hybrid algorithm initially adopts Genetic Algorithm to give pheromone to distribute and then it makes use of ant colony algorithm to give the precision of the solution. This is achieved by first creating a set of execution plans by Artificial Ants. These plans are considered as initial populations of GA. Pheromones in routes are updated (Fig. 7). For every iteration, each ant firstly seeks an answer, then these answers are passed to GA for recombination and the new generation created is again passed to the ant colony for further improvisation. This improves the convergence rate of the produced offspring in each generation and also produces an answer closer to the optimum one with enhanced optimal speed.

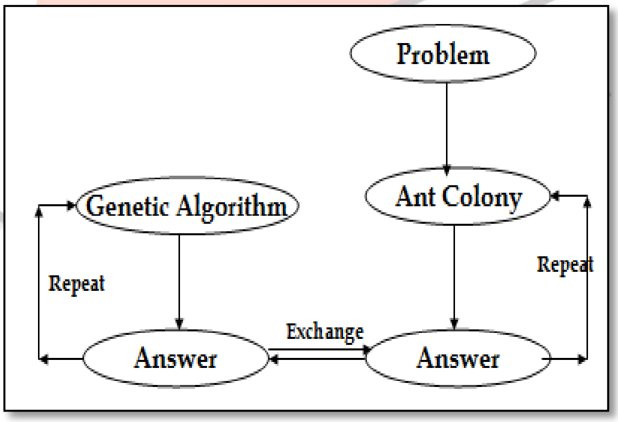

Fig. 7: Diagram for Hybrid GA-ACO Algorithm in DBMS [15]

- A Hybrid of ACO-GA, known as GACO was proposed by [20] for Travelling Salesman Problem by combining certain steps of GA and ACO. In this method, GA evaluates and preserves the fittest ant in each cycle in every generation and only unvisited cities will be assessed by ACO. The table given below shows the hybridization performed on parameters, variables and algorithmic steps of both the algorithms. It was proved experimentally that the performance of GACO was significantly improved and its time complexity was fairly equal to the GA and ACO.

ACO	GA	GACO
Hybridization on Parameters and variables of the algorithms		
Number of cycles	Number of generations	Number of cycles
Number of ants	Population size	Number of ants; it is equal to the population size
Tabu list	Chromosome	Tabu list and Chromosome. Tabu list is the representation of solution. Chromosome is indirect representation of solution. It represents the remaining city to be visited by the ant based on equation (2)
Tabu list and chromosome are the representations of solution		
Hybridization on steps of the algorithms		
Initialization: Pheromone	Initialization: first population	Initialization: Pheromone and first population
Initiate first city of each ant		Initiate first city of each ant
Finding a route of each ant according to equation (2)		Finding a route of each ant according to equation (2) & chromosome
Finding shortest route		Finding shortest route
	Evaluation	Evaluation process is hybridized on finding a route
Updating pheromone matrix and shortes route of each cycle	Updating population	Updating population, pheromone matrix and shortes route of each cycle

Fig. 8: Diagram for Hybrid GA-ACO Algorithm in DBMS [20]

C. Hybrid of ACO and Particle Swarm Optimization Algorithm

The birds in a flock are symbolically represented as particles in the PSO Algorithm. These particles are considered as simple agents “flying” through a problem space. A location of the particle in the multi-dimensional problem space represents one solution for the problem and when the particle moves to a new location, a different problem solution is generated. This solution is evaluated by a fitness function that provides a quantitative value of the solution’s utility. The velocity and direction of each particle moving along each dimension of the problem space is altered with each generation of movement. In combination, the particle’s personal experience and its neighbors’ experience influence the movement of each particle through a problem space. Shang Gao et al [21] proposed a hybrid of ACO and PSO to solve Traveling Salesman Problem. PSO adopts statistics method to generate several initial better solutions and passes this information to ACO for the distribution of pheromone. ACO makes use of this pheromone accumulation and renewal to generate multiple solutions. These solutions are passed on to PSO which performs across and mutation operation to generate effective solutions. The advantage of PSO algorithm method is the use of self information, individual best information and global best information to generate effective and optimal results.

V. IMPROVEMENT ANALYSIS OF HYBRID OF ACO

The above study observes the following improvements in the given applications by implementing the hybrids of Ant Colony Optimization Problems.

Table 1 Different Application of Hybrids of ACO

Nature of Application	**Hybrids of ACO Implemented with**	**Improvements Observed**
Query Optimization in Databases / Distributed Databases Management Systems	Dynamic Programming [9] Genetic Algorithm [15][16]	• Removes problem of Traceability • Improved Convergence Rate. • Large number of joins can be handled • Reduced cost of query execution
Traffic Signal Controlled Problem	Genetic Algorithm [17]	❖ Optimized Signal Timing ❖ Reduced Automative Exhaustive Emission
Wireless Mesh Network Problem	Genetic Algorithm [18]	❖ Easily managed routing table ❖ Shortest path retrieved
Vehicle Routing Problem	Genetic Algorithm [19]	❖ Reduced number of iterations ❖ Increased Convergence speed ❖ Reduced Time
Travelling Salesman Problem	Genetic Algorithm [20] Particle Swarm Optimization [21]	❖ Improved Time Complexity ❖ Shortest path in minimum time

VI. CONCLUSIONS

The results proved that hybrids of ACO are effective and viable in various applications The Hybrids of ACO proved to be beneficial in the following way:

a) Improved the Initial Convergence Speed of the ACO execution.
b) Increased the Population Scattering Ability for obtaining optimal solutions.
c) Helped in developing new methods Using Artificial Intelligence to solve "ill-structured" problems like optimiztion etc.
d) Reduced Handle Time And Space Complexity when the size of the query and the number of Joins increases.
e) Solved the problems of Tractability using Positive Feed Back mechanism.

VII. REFERENCES

[1] Nada M. A. Al Salami , "Ant Colony Optimization Algorithm", UbiCC Journal, Volume 4, Number 3, August 2009

[2] Enxiu Chen1 and Xiyu Liu, "Ant Colony Optimization - Methods and Applications-Multi-Colony Ant Algorithm", Google Scholar, April 2011, http://cdn.intechweb.org/pdfs/13584.pdf

[3] Book: Marco Dorigo, Thomas St¨utzl, "The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances", 1991.

[4] Marco Dorigo, Mauro Birattari,Thomas St¨utzle, "Ant Colony OptimizationArtificial Ants as a Computational Intelligence Technique", IEEE Computational Intelligence Magazine, November 2006.

[5] Enxiu Chen1 and Xiyu Liu, "Ant Colony Optimization - Methods and Applications-Multi-Colony Ant Algorithm", Google Scholar, April 2011, http://cdn.intechweb.org/pdfs/13584.pdf.

[6] Marco Dorigo, Mauro Birattari,Thomas St¨utzle, "Ant Colony Optimization-Artificial Ants as a Computational Intelligence Technique", IEEE Computational Intelligence Magazine, November 2006.

[7] Zainudin Zukhri and Irving Vitra Paputungan, "A Hybrid Optimization Algorithm Based On Genetic Algorithm And Ant Colony Optimization, International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 4, No. 5, September 2013.

[8] M. Naohoghdem, " Query Optimization in Distributed Database using Hybrid Evolutionary Algorithm", Intl. Conference on Information Retrieval and Knowledge Management, pp 125-130, 2010.

[9] Tansel Dokeroglu, Ahmet Cosar, "Dynamic Programming with Ant Colony Optimization Metaheuristic for optimization of Distributed Database Queries", ISCIS:26th International Symposium on Computer and Information Sciences, IEEE, Vol. 2 , pp.107-113, 2011.

[10] M. Gregory, "Genetic Algorithm Optimization of Distributed Database Queries", IEEE World Congress on Computational Intelligence, pp 271-276, 1998.

[11] Book: Gen, M. & Cheng, R.: Genetic Algorithms and Engineering optimization, New York: Wiley (2000).

[12] Prof. Swati V. Chande1, Dr. Madhavi Sinha2, Genetic Algorithm: A Versatile Optimization Tool, BVICAM'S International Journal of Information Technology, © BIJIT – 2008.

[13] M. Naohoghdem, " Query Optimization in Distributed Database using Hybrid Evolutionary Algorithm", Intl. Conference on Information Retrieval and Knowledge Management, pp 125-130, 2010.

[14] Kangshun Li, Lanlan Kang, Wensheng Zhang, Bing Li, "Comparative Analysis of Genetic Algorithm and Ant Colony Algorithm on Solving Traveling Salesman Problem", IEEE International Workshop on Semantic Computing and Systems, 2008.

[15] H. Kadhkhodaei, F. Mahmoudi, "A Combination Method for Joining Ordering Problem in Relational Database using Genetic Algorithm and Ant Colony", IEEE Transactions on Granular Computing, pp 312-317, 2011.

[16] Y. Zhou, W.Wan, J. Liu, "Multi-Joint Query Optimization of Database Based on the Integration of Best-Worst Ant Algorithm and genetic Algorithm", Wireless Mobile Computing (CCWMC 2009), IET Intl. Communication Conference, pp 543, 2009.

[17] Zhou Shen-Pei1, Yan Xin-Ping, "The Fusion Algorithm of Genetic and Ant Colony and Its Application", Fifth International Conference on Natural Computation, IEEE Computer Society, 2009.

[18] S. Aravind, Mr. G. Michael, "Hybrid of Ant Colony Optimization and Genetic Algorithm for Shortest Path in Wireless Mesh Network**s",** JGRCS, Vol3, number 1, 2012.

[19] Na Li, Shoubin Wang, Yulan Li, "A Hybrid Approach of GA and ACO for VRP", Journal of Computational Information Systems 7: 13 (2011) 4939-4946 1553-9105, Binary Information Press, December, 2011.

[20] Zainudin Zukhri and Irving Vitra Paputungan, "A Hybrid Optimization Algorithm Based On Genetic Algorithm And Ant Colony Optimization, International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 4, No. 5, September 2013.

[21] Shang Gao1, Ling-fang Sun, Xin-zi JIANG, e-zong TANG, " Solving Traveling Salesman Problem by Ant Colony optimization -Particle Swarm Optimization Algorithm", DCABES PROCEEDINGS, pp 426-429, 2006

[22] Zar Chi Su Su Hlaing, May Aye Khine , "An Ant Colony Optimization Algorithm for Solving Traveling Salesman Problem", International Conference on Information Communication and Management, Singapore, IPCSIT vol.16 (2011), IACSIT Press.