Implementation of Query Optimization using Heuristic and Ant Colony Algorithms

Priya Patidar[1] Dr. Pankaj Sharma[2]
[1]Research Scholar [2]Professor
[1,2]Department of Computer Science & Engineering
[1,2]SIMS, M.P, India

Abstract— to implement query optimization methods such as Heuristic Greedy based optimization, Iterative Improvement based cost optimization and Ant Colony optimization algorithms. Show Comparison of cost, execution time and response time between Heuristic based optimization, Ant Colony Optimization and Modified Ant Colony optimization algorithms. After implementation of Heuristic based optimization, Ant Colony Optimization and Modified Ant Colony optimization algorithms, found that Modified Ant colony taken less computation time compare to others. All three methods Implemented using Java language.

Keywords: Query Optimization, Heuristic-Based Optimizers, Ant-Colony, Modified Ant Colony

I. INTRODUCTION

Query process is the method by which the query results are retrieved from a high-level query like SQL or OQL.

Generally, the query optimizer can't be accessed directly by any users, once queries are submitted to database server, and parsed by the parser, then passed to the query optimizer where optimization happens.

Query processing denotes to the range of activities elaborate in retrieve data from databases. The activities involve translation of queries in high-level database languages into looks that can be used at the objective level of the file system, a range of query-optimizing changes, and actual evaluation of queries [4].

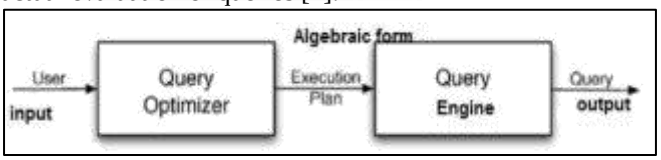

Fig. 1: Query Optimization

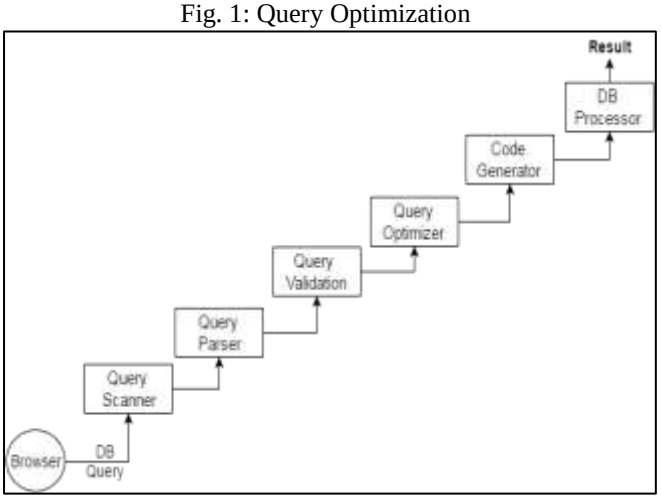

Fig. 2: Query Optimization Process

Query process is the method by which the query results are retrieved from a high-level query like SQL or OQL. Generally, the query optimizer can't be accessed directly by any users, after queries are accepted to database server, and construed by the parser, then passed to the query optimizer where optimization happens.

II. PROBLEM DOMAIN

Query optimization is the efficient way to executing a SQL query. By studying the complete candidate ideas the lowest cost plan is considered as the best execution plan that an optimizer efforts to generate for a SQL statement. As SQL is a non-procedural language, so the optimizer is free to rebuild, reorganize, and process in any order [3]. Problems in Query optimization are as

- Query rewriting:
  - Transformations from one SQL query to another one using semantic properties.
- Selecting query execution plan:
  - Single query blocks (I.e., S-P-J blocks)
  - Join enumeration
- Cost estimation:
  - To compare between plans we need to estimate their cost using statistics on the database.

III. QUERY OPTIMIZATION METHODS

A. *Heuristic Greedy based optimization:*

Heuristic optimization transforms the query-tree b using a set of rules that typically (but not in all cases) improves execution performance.

1) Perform selection early (reduces the number of tuples)
2) Perform projection early (reduces the number of attributes)
3) Perform most restrictive selection and join operations (i.e. with smallest result size) before other similar operations.

Heuristic Greedy based optimization Algorithms steps,

1) First the parsed query must pass the query transformer inside the optimizer the query transformer rewrite the query using heuristic [2]
2) Achieve selection and projection as early as feasible
3) Predicate pushdown
4) Sub query nesting

B. *Ant Colony based optimization:*

Ant Colony Optimization is based on artificial System. Ant colony Optimization system based of real Ant Colonies. Using Ant Colony Optimizer to solve discrete optimization problem.

Ant Colony Query optimization algorithms steps:

1) Step 1. Initialize parameters. The parameters of the proposed algorithm are initialized.

2) Step 2. Classification of nodes: These nodes are divided into the center node and boundary node according to the geographic coordinate information of nodes. The classified result is saved and sorted in order to prepare for the subsequent running of the algorithm.
3) Step 3. The m ants are randomly placed into n nodes, and this node is added into Tabu list of the ant.
4) Step 4. After the ants have completed a choice, the path length is calculated. Then the respective Tabu list is modified. Repeat step-3 until the completed touring of the ant. The current optimal path length is saved, and the global optimal path is updated in this iteration.
5) Step 5. Update the pheromone: The pheromone on the optimal path is globally updated according to the equation in the improved updating rules of pheromone.
6) Step 6. Iteration control: Set the iterative counter return to Step 4. Otherwise, the algorithm is terminated, and the optimal solution is output.

C. Modified Ant Colony based optimization:

Modified Ant Colony Query optimization algorithms steps:

1) Step 1. Initialize parameters: The parameters of the proposed algorithm are initialized.
2) Step 2. Classification of nodes: These nodes are divided into the center node and boundary node according to the geographic coordinate information of nodes. The classified result is saved and sorted in order to prepare for the subsequent running of the algorithm.
3) Step 3. The m ants are randomly placed into n nodes, and this node is added into Tabu list of the ant.
4) Step 4. For each ant, when Tabu table is not null, the selection probability to the next node is calculated according to the equation in the dynamic movement rules of ant. Then this node is added into the Tabu list, and the pheromone is locally updated.
5) Step 5. After the ants have completed a choice, the path length is calculated. Then the respective Tabu list is modified. Repeat Step 4 until the completed touring of the ant. The current optimal path length is saved, and the global optimal path is updated in this iteration.
6) Step 6. Update the pheromone: The pheromone on the optimal path is globally updated according to the equation in the improved updating rules of pheromone.
7) Step 7. Iteration control: Set the iterative return to Step 4. Otherwise, the proposed algorithm is terminated, and the optimal solution is output.

IV. RESULT ANALYSIS.

In the experiment, of Heuristic based optimization, Ant Colony Optimization and Modified Ant Colony optimization algorithms implements on java platform. Machine configured with core i5 processor, 4GB RAM and Window-7 OS. MYSQL 5.5 used as database with PHPMyadmin. Compare between Heuristic based optimization, Ant Colony Optimization and Modified Ant Colony optimization algorithms on the basis of computation time.

List of Complex Queries

- select name,instructor.deptname,building from instructor,department where instructor.deptname=department.deptname;
- select name, courseid from instructor, teaches where instructor.ID= teaches.ID;
- select name, courseid from instructor natural join teaches;
- SELECT name, title FROM (instructor NATURAL JOIN teaches)JOIN course USING (courseid)

Below Table show Computation time for different queries run on machine.

Query	Heuristic based optimization	Ant Colony Optimization	Modified Ant Colony optimization
Query-1	127	105	90
Query-2	128	120	94
Query-3	130	110	90
Query-4	128	108	91
Query-5	131	111	88

Table 1: Computation Time Comparison between Optimization Methods

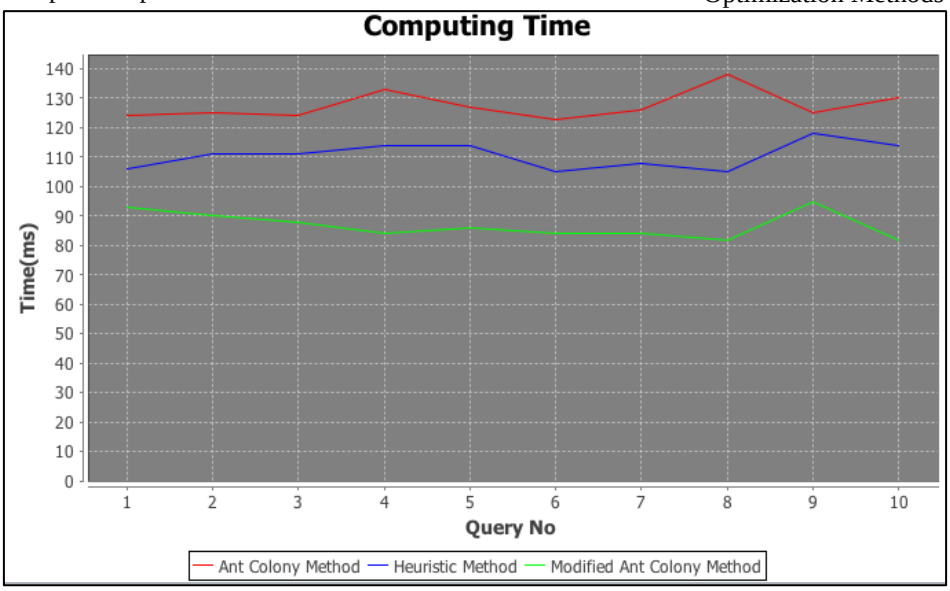

Fig. 3: Computation Time Comparison between Optimization Methods

Fig. 4: Query result using Heuristic Algorithm.

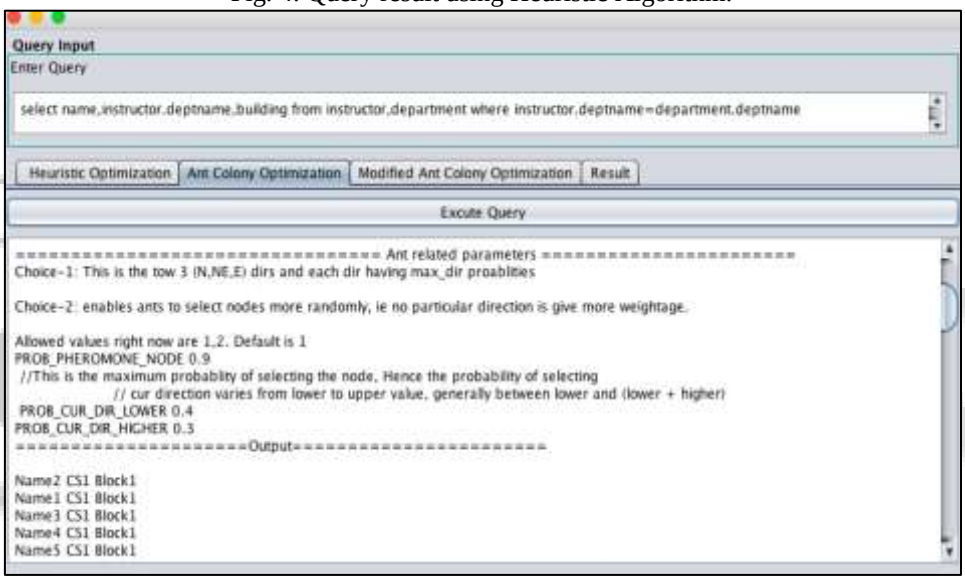

Fig. 5: Query result using Ant Colony Algorithm.

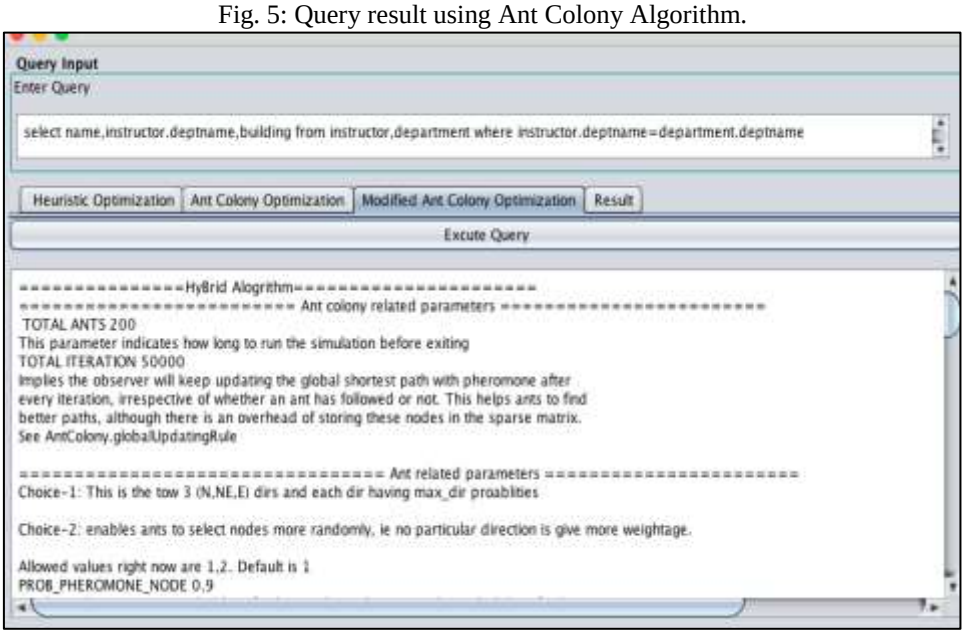

Fig. 6: Query result using Ant Colony Algorithm.

V. CONCLUSION

The implementation of selection and projection operations in the relational algebra expression, to avoid the direct Cartesian product operation, then combines a series of selection and projection before and after it together to reduce the size of intermediate relations, thus to achieve optimization. We compare query optimization methods such as Heuristic based optimization, Ant Colony optimization algorithms and Modified Ant Colony optimization algorithms. Show Comparison execution time and response time between Heuristic based optimization Ant Colony optimization algorithms and Modified Ant Colony optimization algorithms.

In future any new hybrid algorithm apply for query optimization, which give better result from existing algorithm.

REFERENCES:

[1] Duy-Hung Phan et al."A Novel, Low-latency Algorithm for Multiple Group-By Query Optimization" ICDE 2016 Conference, IEEE, 978-1-5090-2020-1/16 2016 IEEE.
[2] Vishal P. Patel, Hardik R. Kadiya "Optimization of Large Join Query using Heuristic Greedy Algorithm " IJCAT - International Journal of Computing and Technology Volume 1, Issue 1, February 2014 www.IJCAT.org
[3] Myungcheol Lee et al. "A JIT Compilation-based Unified SQL Query Optimization System" 978-1-5090-3765-0/16/ ©2016 IEEE
[4] Saurabh gupta ,Gopal Singh Tandel ,Umashankar Pandey , "A Survey on Query Processing and Optimization in Relational Database Management System ", International Journal of Latest Trends in Engineering and Technology (IJLTET) ,Vol. 5 Issue 1 January 2015 , ISSN: 2278-621X
[5] Dr. G. R. Bamnote Professor & Head Dept. of CSE, PRMITR, Badnera, India ,Prof. S. S. Agrawal ,Asst. Prof Dept. of CSE, COE & T, Akola, India ," Introduction to Query Processing and Optimization ", International Journal of Advanced Research in Computer Science and Software Engineering , Volume 3, Issue 7, July 2013 ISSN: 2277 128X
[6] A. K. Giri and R. Kumar, "Distributed query processing plan generation using iterative mprovement and simulated annealing," 2013 IEEE 3rd International Advance Computing Conference, pp. 757-762, Feb. 2013.
[7] T. Kumar, V. Singh and A. K. Verma, "Distributed query processing plans generation using genetic algorithm," International Journal of Computer Theory and Engineering, pp. 38-45, 2011.
[8] Melanie Mitchell, "An introduction to Genetic Algorithms", Prentice Hall of India, 2004
[9] Hsiung Sam, Matthews James, "An introduction to Genetic Algorithms", 2000 http://www.generation5.org/content/2000/ga.asp
[10] S. J. Xue, J. Zhang and X. D. Xu, "An improved algorithm based on ACO for cloud service PDTs scheduling", Advances in Information Sciences and Service Sciences, vol. 4, no. 8, (2012), pp. 340-348.
[11] Y. C. Li, L. P. Liu, S. J. Zhou and Z. C. Yang, "Dynamic resource scheduling in construction project group management based on improved ACO algorithm", Energy Education Science and Technology Part A: Energy Science and Research, vol. 31, no. 2, (2013), pp. 1111-1116.
[12] E. Walid, E. A. Haikal, A. Ajith and A. M. Alimi, "A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP.Applied Soft Computing Journal", vol. 25, no. 12, (2014), pp. 234-241.
[13] J. H. Shen and J. Chen, "An improved ACO based RWA algorithm and its application in wavelength converter allocation issue of the intelligent optical networks", Journal of Computational Information Systems, vol. 10, no. 8, (2014), pp. 3341-3349.
[14] J. B. Escario, J. F. Jimenez and J. M. Giron-Sierra, "Ant colony extended: Experiments on the travelling salesman problem", Expert Systems with Applications, vol. 42, no. 1, (2015), pp. 390-410.
[15] F. Li and M. L. Jin, "GACO: a GPU-based high performance parallel multi-ant Colony Optimization algorithm", Journal of Information and Computational Science, vol. 11, no. 6, (2014), pp. 1775-1784.
[16] Z. L. Zhang, C. Gao, Y. X. Liu and T. Qian, "A universal optimization strategy for ant colony optimization algorithms based on the Physarum-inspired mathematical model", Bioinspiration and Biomimetics, vol. 9, no. 3, (2014), pp. 1-7.
[17] M.DorigoandL.M.Gambardella,"Antcoloniesforthetravellingsalesmanproblem",Biosystems,vol. 43, no. 2, (1997), pp. 73-81

