Ant-Based Clustering In Literature Review On To Optimize Join Queries in Distributed Database Using Ant Colony Algorithm with Evolutionary Approach

Manish Singhal & Dr. Pushpneel Verma
Computer Science & Engineering Department, Bhagwant University
Ajmer (Rajasthan)

Abstract: Query optimization in scattered databases plainly required in many aspects of the optimization process, this is not only increases the cost of optimization, but also changes the trade-offs involved in the optimization process radically. It describes the unnaturally growth of query optimization methods from uniprocessor relational database systems to analogous database systems. With the progression of Computer Networks and increase in size of databases, the conveyance of databases has led to the maturity of Distributed Database over multiple machines where distribution of the database is Transparent to the users. However, when ACO is implement in Distributed Database queries, the Initial Information required by ACO to produce an optimal result set is not systematic and organized which leads to slower convergence speed in the beginning of the processing to generate an optimal solution.

Keywords- Scattered Database, Query Optimization ,Ant Colony Optimization AlgorithmOptimization Strategies.

Introduction: The role of distributed query processor is to map these Distributed Transactions (set of queries) onto a relational algebra format, expressed on relational calculus, into a sequence of relational algebra operators expressed on data fragments (Bernstein, 1981). The implementation of the doubt in DDBMS is much more complex than accomplish it in a centralized database as a huge number of processing phases and internal procedures are involved in processing and carrying out of the query. When a query enters a distributed environment, it is first parsed, checked on integrity constraints and validated semantically and syntactically by Semantic Data Controller. The Query Decomposer then decomposes the validated query into a sequence of relational operations expressed in relational algebra form on the Global Conceptual Schema.

The Data Localization Phase uses the Materialization and Access Planning Program to map the Global intangible Schema to the Fragmentation Schema and Allocation Schema of the distributed data. Query Processing in DDBMS implements join query operation on multiple

relations. A join query is defined on different Query Execution Plans that results in transfer of data between sites via different paths or in different join orders. In the Global Query Optimizer phase, various equivalent plans of the query are constructed in the search spaces on the basis of permutations on the order of relational operators. These plans have different transmission speed, and the execution order can affect the quantity of data transferred .The distributed database query optimizer evaluates these plans according to the predefined cost model that can be either based on the data statics or cost of transmission of intermediate results or speed of communication network or time taken to accomplish the query or CPU processing speed or a combination of any of these. The fundamental mission of the Global Query Optimizer is to generate the Join Order that represents the "Best" Query Execution Plan (QEP). The Global Execution Monitor ensures that the execution of the doubt at the Local Query Optimization Phase takes place as per the best plan generated by the Global Query Optimizer.

The Query Optimization at the Local Query Optimization Phase is supported by the Local Conceptual Schema or Local Internal Schema. It works on the Access Path Selector and selects the best path locally to access any data item. This is supported by Local Query Run-Time Support Processor that acts as an edge between the Database Buffer and Operating System so as to enhance the processing speed and helps the Local Query Executor to the query from the underlying database. The performance of the query principally depends on the competency of the Search Strategy implemented by the Distributed Database Management Software and its ability to handle complex queries.

Figure 1.1: Ants walking around the graph

Query Optimization In Distributed Database :

High presentation low-cost PC hardware and high speed LAN/WAN technologies make distributed database systems an attractive research area where query optimization is an important notion. Query Optimization in Distributed Database consists of four phases :

a) Query Decomposition

b) Data Localization

c) Global Optimization

d) Local Optimization

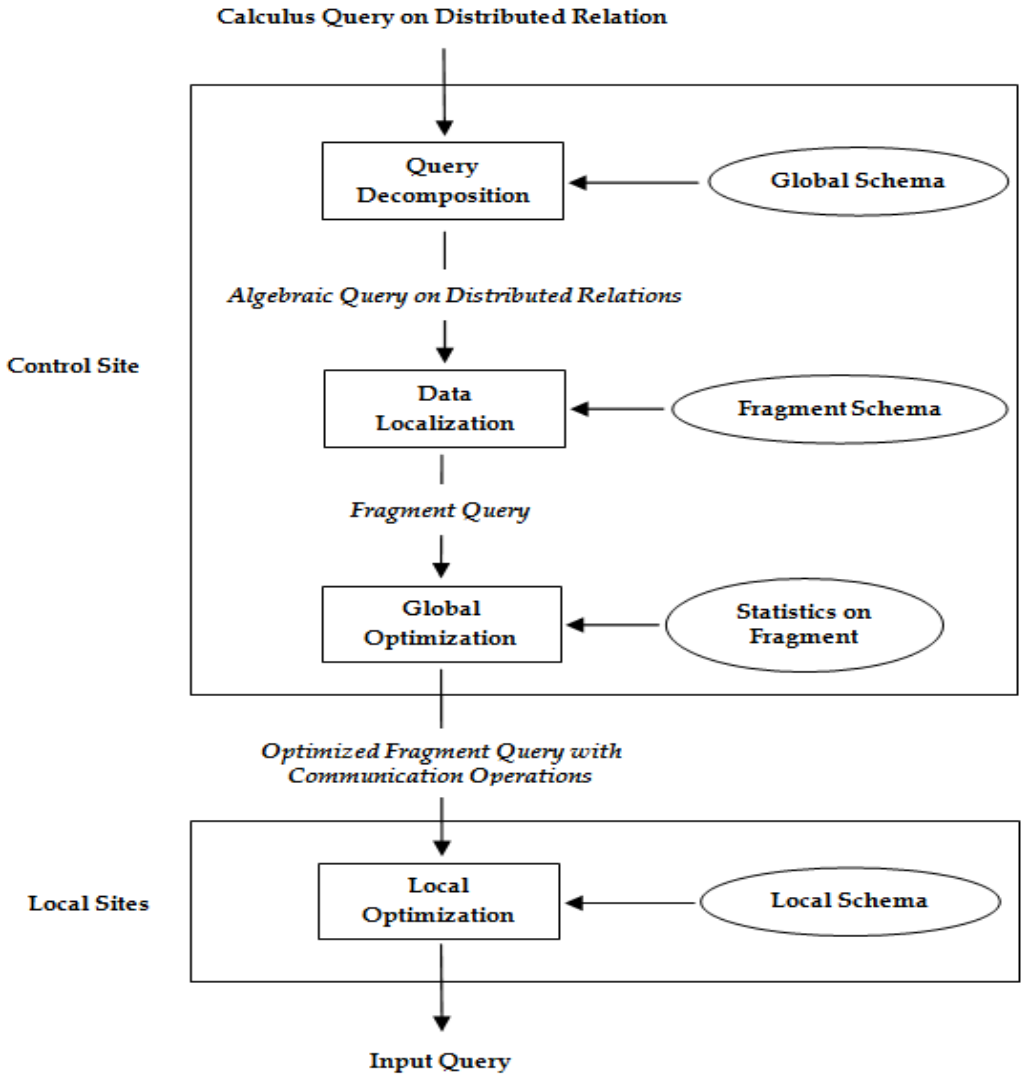

Fig No. 1.2 Scattered Query Optimization

Data Localization refers to the availability of query data at the local site for processing. The Global Optimization mainly consists of influential the best execution site for local sub query and finding the best inter-site operator scheduling. The confined Optimization is mainly alert on optimizing each local sub doubt on each site. A lot of research was also prepared on various factors like optimization algorithms, search space, execution strategies and cost model to optimize a query . Join procedure is the most important operation in Scattered Database Environment so as to salvage data from multiple sites . With the enhance in the size of the database and the number of joins, the complexity of the Query Execution Plan (QEP) and Join Strategy also increases. The complexity of query optimization is determined by a number of alternative QEPs which grows exponentially with the number of relations involved in the query because a single query can be joined in several ways. Since all execution plans are equivalent in terms of their final output with a difference in cost and amount of time that they need to run, it is necessary to optimize these query plans, join orders and join methods in modeling query processing. Enumerative optimization strategies are chiefly dealing with the join query to settle on the best plan to execute the query. The task of Query Optimizer is to:

a) Find out the Order of the Execution of Relational Operators.

b) Find out the access methods for pertinent relations.

Ant Colony Algorithm: Ant colony algorithms are becoming popular approaches for solving combinatorial optimization problems in the literature.As ants have weak global insight of its environment, an ant moves at random when no pheromone is available. The ant does not choose its direction based on the level of pheromone exclusively, but also considers the neighborhood of the nest and of the food place, respectively, into account. This allows the discovery of new and potentially shorter paths.

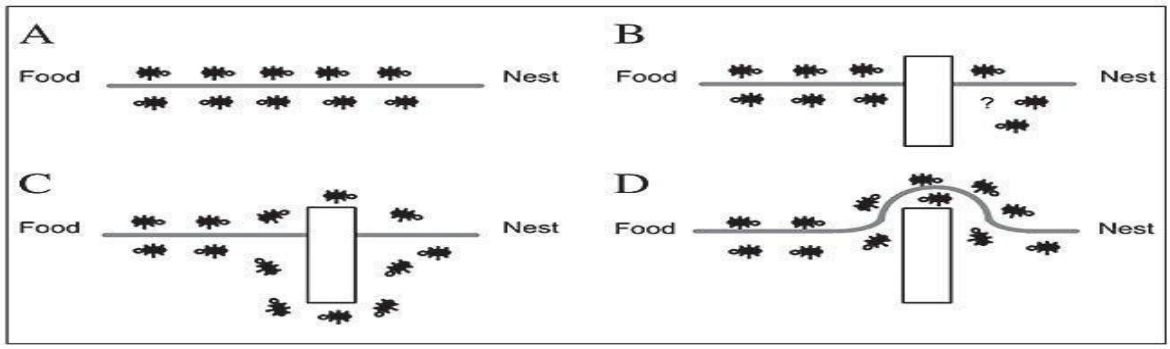

Fig 1.3: Ants Searching their paths

Pseudo code for Ant Colony Algorithm:

```
Generate a set of solutions over the search space
select the best k elements among the set of ants
repeat
        build pheromones from ants in s
        create new solutions according to pheromones information
        take the best k elements among s and the new solutions as new s
until termination criterion is met
```

Ant-Based Clustering In The Literature: Ant colonies provide a means to formulate some powerful nature-inspired heuristics for solving the clustering problems. Several clustering methods based on ant performance have been proposed in the literature. This section provides a brief portrayal of these methods.

Ant-based clustering algorithms are based upon the offspring sorting behavior of ants. Larval sorting and cadaver cleaning by ant was first modeled by Deneubourg et al. for accomplishing certain tasks in robotics. Their work was actually focused on clustering objects by using group of real world robots. Their model is known as basic model (BM). This model can be described as follows: The data items are indiscriminately scattered into a two-dimensional grid. The assessment to pick up or drop an item is random but is predisposed by the data items in the ant's immediate neighborhood. The possibility of dropping an item is increased if ants are delimited with similar data in the neighborhood. In this way, clustering of the elements on the 2D grid is obtained.

Literature Review: The Search Strategies implemented by the distributed database optimizer to extract appropriate query data in minimum computational time from the distributed database are of great concern. Also, the performance of the distributed database optimizer gets largely affected by the factors defined for it. Some of these factors are dependent on the architectural structure of DDBMS and some of them are dependent on the procedural behavior of DDBMS. These factors include.

(i) Allocation of data at multiple sites affecting the Data Transmission Cost.

(ii) Shape of Join Query Graph.

(iii) Order of Join Operation.

(iv) Order of Join Sites.

(v) Arrangement of Query Operators.

Among these factors, the most significant factor researched upon by implementing appropriate search strategies is the Order of Join Operation of the relations involved in the query. Large numbers of search strategies have been researched upon and the search is still in progress for finding a suitable Search Strategy for effectively optimizing large join queries. The review of literature covered in this chapter emphasizes on the Search Strategies implemented till now as query optimizer in Distributed Database Management System. This is done in an effort not only to show what has been done till present, but also to explain how the work carried out in this thesis correlates with other research.

According to (Matthias, Joachim W. Schmidt, 1984) query processing is defined as "range of activities involved in translating a high level language query into low-level language by a series of operations like query parser and validation, adaptation into relational algebra & relational calculus, creation of multiple query execution plans and processing the query by an efficient query optimizer". The distributed query optimizer first maps the distributed database with the SQL query defined on the global relations and then converts them into a string of relational operations that are defined on fragmented relations. Because the performance of the query is an important criterion of measurement of data retrieved from the underlying database, it has always received considerable attention in research. Query processing in scattered database consists of four generic phases where the input is relational calculus of the distributed query expressed on global conceptual schema, The query optimizer diverges in various traits like type of enumeration algorithm, optimization granularity, time taken to optimize queries, effect of representation of replicated fragments and implementation of semi-joins (Ozsu, 2001; Aljanaby et al., 2005).

Search Strategy: It is the enumeration algorithm that explores large search space to determine the optimal Query Execution Plan (QEP) for complex query based on information available on Data Statistics, Order of Join Operator and Data Sites involved in query. The QO problem has been addressed by various strategies like Deterministic Strategies, Randomized Strategies and Evolutionary Strategies. Building of QEP begins from the base relations and

proceeding by joining one or more relation at each step until a complete plan is obtained in the technique adopted by Deterministic Strategies. Once all plans are created, the "best" solution is searched. Randomized Strategies build QEP around a particular point at a time by searching for optimal solutions around it. However, there is no guarantee of optimal solution generated by it but it is capable of avoiding high cost of query optimization. Evolutionary algorithms have global search capabilities and are best suitable to combinatorial optimization problems.

In this era of digitization, where anytime and all time availability of information plays a vital role, it is important to focus on the mechanisms that can retrieve information in a very short span of time from large databases. The information provided should be accurate and reliable. Powerful algorithms are being designed that can accomplish queries involving large quantity of relatives and generate end result with minimum response time.

Refrences:

1. M.S.Chen and P.S. Yu, "Using Join Operations as Reducers in Distributed Query Processing", Proceedings of the 2nd International Symp. on Databases in Parallel and Distributed System, July 1990
2. S. Pramanik and D. Vineyard, "Optimizing Join Queries in Distributed Database", IEEE Trans., Software Eng., vol.14, pp 1391-1326, Sept. 1988
3. Marco Dorigo, Mauro Birattari,Thomas St¨utzle, "Ant Colony Optimization Artificial Ants as a Computational Intelligence Technique", IEEE Computational Intelligence Magazine, November 2006
4. 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
5. Marco Dorigo, Thomas St¨utzl, "The Ant Colony Optimization Metaheuristic:Algorithms, Applications, and Advances"
6. 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

7. 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
8. Gao Shang, Jiang Xin-zil, Tang Kezong, Yang Jingyu, “Hybrid Algorithm Combining Ant Colony Optimization Algorithm with Particle Swarm Optimization”, Proceedings of the 25th Chinese Control Conference, Harbin, 7-11 August, 2010
9. 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
10. Zehai Zhou, “Using Heuristics and Genetic Algorithms for Large Scale Database Query Optimization”, Journal of Information and Computing Science, Vol 2, No 4, pp- 261-280, 2007
11. Doshi P. and Raisinghani V., “Review of Dynamic Optimization Strategies in Distributed Database”, Electronics Computer Technology (ICECT), 3rd International Conference, April 2011 .
12. Favaretto D, E. Moretti, P. Pellegrini (2009) On the explorative behavior of MAX–MIN Ant System. In: Stutzle T, Birattari M, Hoos HH (eds) Engineering Stochastic Local Search Algorithms. Designing, Implementing and Analyzing Effective Heuristics. SLS 2009, LNCS, Springer, Heidelberg, German.
13. Fidanova, S., P. Marinov and M. Paprzycki (2013) Influence of the Number Of Ants On Multi-Objective Ant Colony Optimization Algorithm for Wireless Sensor Network Layout, In International Conference on Large-Scale Scientific Computing, Springer, Berlin, Heidelberg.
14. Gajjam, N. S. and S. S. Apte (2014) Reducing Execution Time of Distributed SELECT Query in Heterogeneous Distributed Database using Genetic Algorithm, International Journal of Computer Applications
15. Hei Y. and P. Du (2011) Optimal choice of the parameters of ant colony algorithm, Journal of Convergence Information Technology,
16. Abdelkader Hameurlain and Frank Morvan, “Evolution of Query Optimization Methods”, Trans on Large Scale Data and Knowl. Cent. Sys, I LNCS 5740, pp 211-242, 2009
17. Ozsu M.T. and Valdureiz P: “Principles of Distributed Database System”, 2nd Edition, Prentice Hall, Woodcliff’s, 1999

18. P.M.G. Apers, A.R. Hevner and S.B Yao, “Optimization Algorithm for Distributed Queries”, IEEE Trans. On Software Eng., Vol.SE-9, no.1, pp 56-68, Jan. 1983
19. A.R. Hevner and S.B.Yao, “Query Processing in Distributed System”, IEEE Trans., Software Eng., vol.SE-5, pp 177-187, May 1979