AIRO JOURNALS

Free / Unpaid
Peer Reviewed
Multidisciplinary
International

ISSN: 2320-3714
Volume:1 Issue:1
January 2022
Impact Factor 5.7
Subject Computer&Science

DISTRIBUTED QUERY OPTIMIZATION USING HYBRID ANT COLONY ALGORITHM

Manish Singhal[1,*], Prof. Pushpneel Verma[2,*]

[1]Research Scholar, Computer Science & Engineering Department,
Bhagwant University Ajmer (Rajasthan)
[2]Professor, Computer Science & Engineering Department, Bhagwant
University Ajmer (Rajasthan)

DECLARATION:: I AS AN AUTHOR OF THIS PAPER / ARTICLE, HEREBY DECLARE THAT THE PAPER SUBMITTED BY ME FOR PUBLICATION IN THIS JOURNAL IS COMPLETELY MY OWN PREPARED PAPER.. I HAVE CHECKED MY PAPER THROUGH MY GUIDE/SUPERVISOR/EXPERT AND IF ANY ISSUE REGARDING COPYRIGHT/PATENT/ PLAGIARISM/ OTHER REAL AUTHOR ARISE, THE PUBLISHER WILL NOT BE LEGALLY RESPONSIBLE. . IF ANY OF SUCH MATTERS OCCUR PUBLISHER MAY REMOVE MY CONTENT FROM THE JOURNAL..

Abstract: Appropriated information base is arising as an aid for huge associations as it gives better adaptability and straightforwardness contrasted with unified data set. As the information is becoming over the conveyed climate step by step, a superior circulated administration framework is needed to deal with this huge information. Question improvement is a course of discovering better inquiry execution plan from various accessible choices. As there a various destinations in disseminated data set having portions of the information, and the size of information isn't static, a powerful arrangement is expected to advance questions in appropriated data set. The mix of Ant Colony Algorithm and Genetic Calculation can be utilized to give a powerful methodology.

Keywords-Distributed database, query optimization, query execution engine, semijoin, ant colony algorithm, genetic algorithm etc.

Introduction:

Database is a collection of files or relations. The management of these data is done by a system called Database Management System (DBMS). There are two ways two manage these data. Centralized database holds all data on a central computer, all the data physically presents at one location.In centralized database approach the data is placed on central repository hence it is easy to access or extract data from multiple tables. The database query can be easily altered into set of relational algebra’s operation.

A distributed database is a database in which portions of the database are stored on multiple computers within a network. Though the data is distributed, database is still centrally administered as a corporate resource while providing local flexibility and customization. The network should allow the users to share the data; thus a user (or program) at location A must be able to access (and perhaps update) data at location B. The sites of a distributed system may be spread over a large area (e.g., a city or a country) or over a small area (e.g., a building or campus).

A major objective of distributed databases is to provide ease of access to data for users at many

different locations. To achieve this, the distributed database system must provide location transparency, which means that a user (or user program) using data for querying or updating need not know the location of the data. As data is distributed over several sites, it requires more efforts to transform a query and in distributed database.

Query Optimization

With distributed databases, data is distributed over different sites thus response to a query may require the DBMS to assemble data from several different sites (due to location transparency, the user is unaware of this need). An important task for the distributed database is how to process a query, which is affected by both the way a user, provides a query and the intelligence of the distributed

Challenges in Distributed Database:

As the data is distributed at different sites it is more challenging to compute efficient query plan in distributed environment. Following are the challenges in distributed database.

- Query needs to be broke into components that are isolated at different sites.
- Determine which site has the potential to yield the fewest qualified records as it affects the communication cost of the network. The least the data to transfer across the network, the less communication will be required.
- Transformation of result to another site where additional work is performed.
- For more than two sites, it requires even more complex analyses and more complicated heuristics to guide query processing.

- At each site compute cost using effective cost model

Following are the basic steps used by a distributed DBMS to develop a query processing plan:

1. Query decomposition In query decomposition, query is decomposed into simplified, structured, relational algebra form.

2. Data localization Here, the query is transformed from a query referencing data across the network as if the database were in one location into one or more fragments that each explicitly reference data at only one site.

3. Global optimization in this step, decisions are taken about the order in which to execute query fragments, which site is efficient to move the data, and where parts of the query will be executed.

4.Local Optimization When the fragmented query is sent to a particular site then that query will be executed and optimized locally

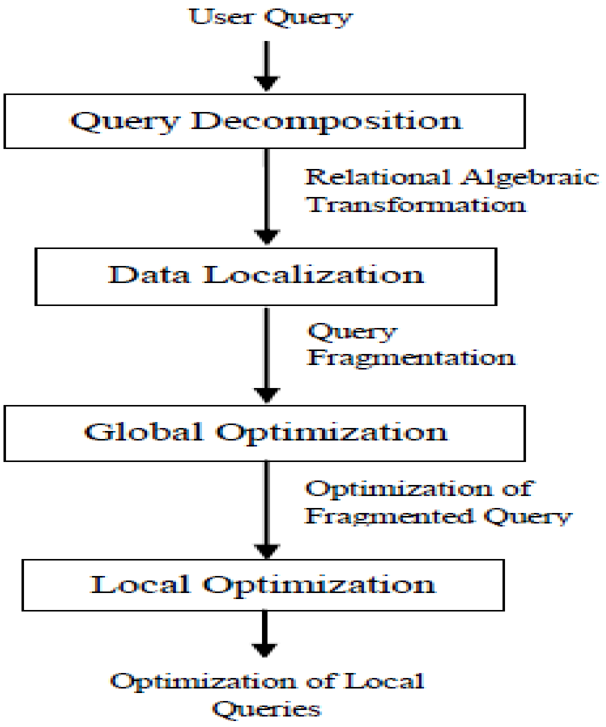


Fig 1: Query optimization in Distributed Database

The main components to optimize the query are query optimizer and query engine. The query engine produces the output by taking the input and operates on physical operators like SORT, NESTED LOOP JOIN, MERGE JOIN and INDEX SCAN etc. These physical operators construct a tree called parsed tree that depicts the flow of data from one operator to the others in the form of edges moving back and forth to the nodes. Query optimizer receives a parsed tree of the SQL query as an input from the execution engine and produces the best possible or close to optimal execution plan out of the possible execution plans for the given query based on the least resource consumption. For a given query, there are many logical algebraic representations and there are many choices of physical operators to implement these logical representations in addition to the variation of response time of each plan. Therefore, it is not an easy task for a query optimizer to generate an efficient query plan .

All printed material, including text, illustrations, and charts, must be kept within a print area of 6-1/2 inches (16.51 cm) wide by 8-7/8 inches (22.51 cm) high. Do not write or print anything outside the print area. All *text* must be in a two-column format. Columns are to be 3-1/16 inches (7.85 cm) wide, with a 3/8 inch (0.81 cm) space between them. Text must be fully justified.

A format sheet with the margins and placement guides is available as both Word and PDF files as <format.doc> and <format.pdf>. It contains lines and boxes showing the margins and print areas. If you hold it and your printed page up to the light, you can easily check your margins to see if your print area fits within the space allowed.

Hybridization of Ant Colony

Many researches [4][5][6][7][8] have been done to optimize queries in distributed database environment. But dynamic system is needed to overcome the issues of distributed database.

A combination of Ant Colony algorithm and Genetic algorithm can provide better execution plan for distributed database system.

Ant colony algorithm: Ant colony algorithms are becoming popular approaches for solving combinatorial optimization problems in the literature.The basic idea of ant heuristics is based on the behaviour of natural ants that succeed in finding the shortest paths from their nest to food sources by communicating via a collective memory that consists of pheromone trails. As ants have weak global insight of its environment, an ant moves at random when no pheromone is available. However, it tents to follow a path with a high pheromone level when many ants move in a common area that leads to an autocatalytic process. The ant does not choose its direction based on the level of pheromone exclusively, but also considers the neighbourhood of the nest and of the food place, respectively, into account. This allows the discovery of new and potentially shorter paths.

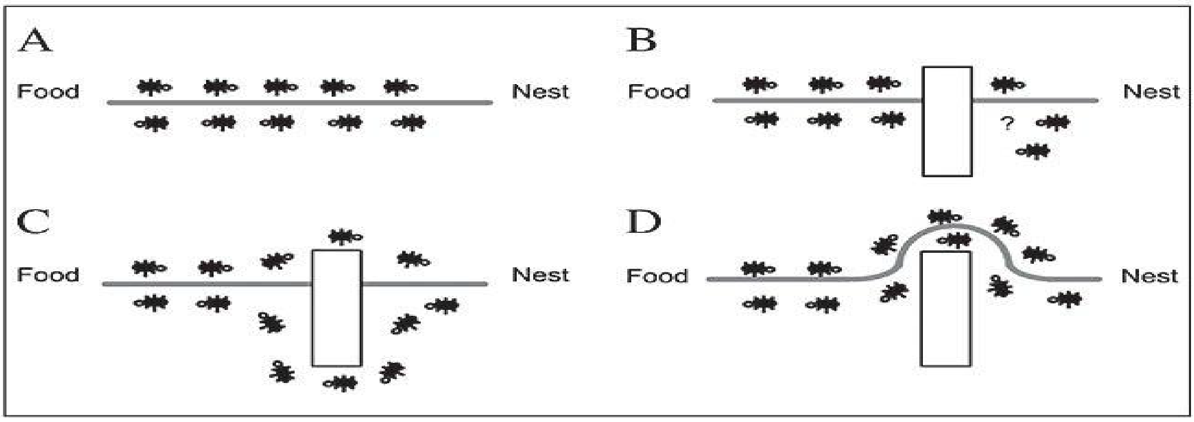
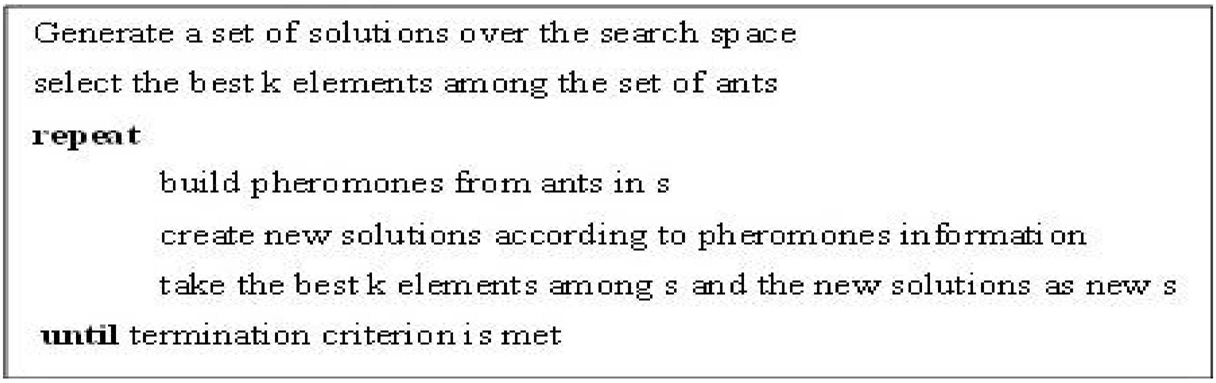


Fig 2: Ants Searching their paths



```
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
```

Fig 3: Pseudo code for Ant Colony Algorithm

Genetic Algorithm: The basic concept of GA is designed to simulate processes[10] in natural systems necessary for evolution. They represent an intelligent exploitation of a random search within a defined search space to solve the given problem. GAs uses the idea of the survival of the fittest individuals within a given population. GA creates and maintains a population of strings (solutions to a specified problem). GA then iteratively creates new populations from the old by ranking the strings, then choose the fittest for interbreeding to create new strings that are hopefully will be closer to the optimum solution for the problem. Genetic algorithm processes can be used to generate new and better generations. Genetic algorithm operations include:

A. The selection of the fittest individuals using the fitness function; this is called Reproduction.

B. The exchange of genes between two individual chromosome; this is called Crossover.

C. The process of randomly altering the genes in a particular chromosome is called Mutation.

Hybridized Ant Colony System:

A system has been implemented using the combination of Ant Colony algorithm and Genetic algorithm. GA[11] has strong flexibility, quick global searching ability with higher population scattering ability for extensive amplitude of answers. On other hand ACO has low population scattering ability, parallel processing, high convergence speed, and global

searching ability with a positive feedback mechanism. By using the combination of both algorithms, strong features of both can improve the performance while overcoming the drawn backs of each. The performance of GA depends on the size of population and operator used in algorithm. If population size is less, then it converges fast.

To execute a single query there may be several execution plans available. Each execution plan will be executed parallelly using Ant Colony algorithm. And from these execution plans, the optimum one will be selected and will be used in future query execution using Genetic Algorithm. A flow of this technique is shown in the figure

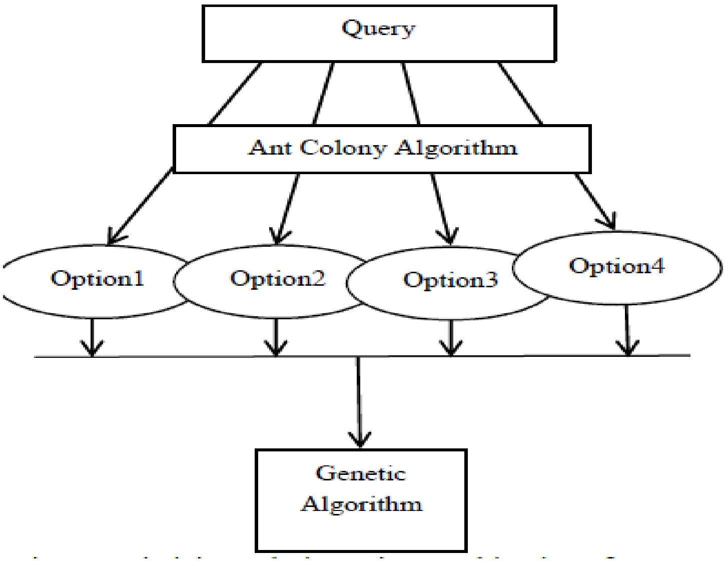


Fig 4: Optimizing solution using combination of Ant Colony Algorithm & Genetic Algorithm

In this way the population available for genetic algorithm will be reduced and ant colony algorithm will converge fast. And this solution will be dynamic one for the size of data over the server.

Conduct of Experimentation

The Hybridized Ant Colony system has been implemented on a distributed database environment containing three sites. Each site is having 100 to 200 records in tables. Experiment has been performed with operating system Windows service pack1, with Intel Pentium IV on web server IIS Server 7.0. Database on each site is maintained on SQL Server 2008. In this experiment information is retrieved from three different relations available at three different sites There are many possible ways to retrieve this information from three relations. Nine possible combinations of queries are provided to ant colony algorithm, which returns the query execution time of all nine queries. To obtain more query execution plans these nine queries are provided as input to genetic algorithm, that cross overs these nine queries and generates eighty-one queries with query execution time of each. Now to retrieve the same information, there are eighty-one options available. Among these eighty-one options the query with least time is most efficient query

Analysis

To retrieve the same information from three relations in distributed database environment, if only ant colony algorithm is used then there are limited query execution plans available and not possible to get query with the least time.

If only genetic algorithm is used then it takes time to generate population and cross over. And still there are less option compared to hybridized ant colony algorithm.

For the same information the query with least execution time is different in all the three algorithms with different time, but the least time is obtained through hybridized ant colony algorithm as shown in the following graph.

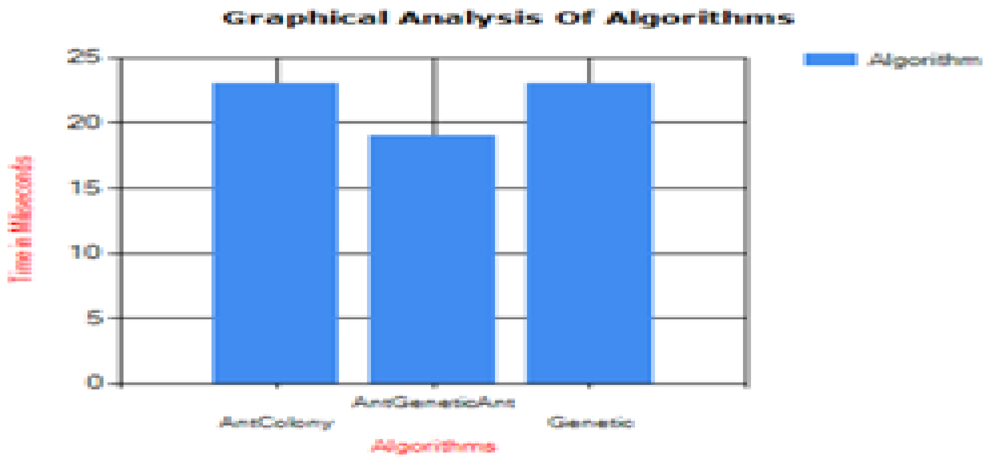


Fig 5: Analysis of Ant Colony algorithm, Genetic algorithm and Hybrid Ant Colony algorithm

Conclusion :

From Hybridized Ant colony system, more possible query execution plans have been obtained and the query with least execution time is obtained with combination of ant colony and genetic algorithm

As the experiment is done on three relations and it is static, this hybridized ant colony algorithm can be extended to optimized queries dynamically in distributed database environment. Ant the same hybridized at colony algorithm can also be extended to heterogeneous databases.

References

[1] C.T. Yu and C.C. Chang, "Distributed Query Processing" ACM Computational Surveys, vol. 16, no.4, pp. 399-433, Dec. 1984.

[2] P.A. Bernstein, N.Goodman, E.Wong, C. Reeve and J.B. Rothnie, " Query Processing in a System for Distributed Database(SDD-1)", ACM Trans. Database Sys, Vol.6, no. 4, pp 602-625, Dec. 1981.

[3] Doshi P. and Raisinghani V., "Review of Dynamic Optimization Strategies in Distributed Database", Electronics Computer Technology (ICECT), 3rd International Conference, April 2011

[4] SurajitChaudhuri, "An Overview of Query Optimization in Relational Systems", Microsoft Research.

[5] Fan and Xifeng, "Distributed Database System Query Optimization Algorithm Research", IEEE2010.

[6] MS Chen, PS Yu, "Interleaving Join Sequence with Semijoins in Distributed Query Processing", IEEE Transactions, Issue 5, Vol. 3, August 2005

[7] Donald Kossmann, Konrad Stocker, "Iterative Dynamic Programming: A New Class of Query Optimization Algorithms".

[8] XUE Lin, "Query Optimization Strategies and Implementation Based on Distributed Database", IEEE 2009.

[9] Adel AlinezhadKolaei and MarziehAhmadzadeh, "The Optimization Of Running Queries In Relational Databases Using Ant-colony Algorithm", IJDMS Vol.5, No.5, October 2013.

[10] Eman Al Mashagba, Feras Al Mashagba, Mohammad Othman Nassar, " Query Optimization Using Genetic Algorithms in the Vector Space Model", IJCSI, Vol. 8, Issue 5, no. 3, September 2011.

[11] Preeti Tiwari, Swati Chande, "Optimization of Distributed Database Queries Using Hybrids of Ant Colony Optimization Algorithm", IJARCSSE, Vol. 3, Issue 6, June 2013.