Retrieval of average sum of plans and degree coefficient between genes in distributed Query Processing

Sambit Kumar Mishra [1] , Dr.Srikanta Pattnaik [2]

[1] **Associate Professor , Department of Computer Sc.&Engg.**

Ajay Binay Institute of Technology , Cuttack , Orissa , India

[2] **Ex-Professor , U.C.E. , Burla**

Director , InterScience Institute of Management & Technology ,

Bhubaneswar , Orissa , India

Key words : Query optimization , Plan , Genetic algorithm , Gene , multicast optimization.

Abstract

Distributed query is one that selects data from databases located at multiple sites in a network and distributed processing performs computations on multiple CPUs to achieve a single result. Query processing is much more difficult in distributed environment than in centralized environment because a large number of parameters affect the performance of distributed queries .

The goal of distributed query processing is to execute queries to minimize the response time and to minimize the total communication costs associated with a query. In addition, when redundant data is maintained, one also achieves increased data reliability and improved response time. In this paper, the multi attribute based mechanism is proposed to meet the demand and the result is compared with some commonly used query optimization algorithms.

1. Introduction

The use of a relational query allows the user to specify a description of the data that is required without having to know where the data is physically located. In a relational database all information can be found in a series of tables. The most common queries are select-project-join queries .Minimizing the quantity of data transferred is a desirable optimization criterion.

The distributed query optimization has several problems related to the cost model, larger set of queries, optimization cost , and optimization interval.

The query optimization process can be defined as follows.

(i) The input query is fed to the search space and transformation rules are applied to it.
(ii) Equivalent query execution plan is generated and passed to the search strategy.
(iii) Formulation of various cost models are done.
(iv) Best query execution plan is obtained.

Considering new large scale database applications, it is necessary to be able to deal with larger size queries. The search complexity constantly increases and makes higher demand for better algorithm than traditional relational database queries.

There are number of query execution plans for distributed database such as row blocking, multicast optimization, multithread execution , joins with horizontal partitioning and Semi joins. An optimizer cost model includes cost functions to predict the cost operators, and formulae to evaluate the size of results. Cost functions can be expressed with respect to either the total time , or the response time. The total time is the sum of all times and the response time is the elapsed time from the initiation to the completion of the query.

In parallel transferring, response time is minimized by increasing the degree of parallel execution. This does not imply that the total time is also minimized. On contrary, it can increase the total time by having more parallel local processing and transmissions. Minimizing the total time implies that the utilization of the resources improves, thus increasing the system throughput. The main factor affecting the performance is the size of the intermediate relations that are produced during execution. When a subsequent operation is located at a different site, the intermediate relation must be transmitted over the network. It is of prime interest to estimate the size of data transfers. The estimation is based on statistical information about the base relations and formulae to predict the cardinality of the results of the relational operations.

The main factor affecting the performance is the size of the intermediate relations that are produced during the execution. When a subsequent operation is located at a different site, the intermediate relation must be transmitted over the network. It is of prime interest to estimate the size of the intermediate results in order to minimize the size of data transfers.

The estimation is based on statistical information about the base relations and formulae to predict the cardinality of the results of the relational operations.

In this work, we are concerned with the average sum of the plans and degree coefficient between the tasks within the plans in distributed query processing. The structure of the paper is as follows. Review of Literature has been mentioned in Section-2 , Problem analysis has been described in Section-3 , Need and necessity of genetic algorithm is discussed in Section-4 , Problem formulation has been discussed in section-5 , Experimental results, analysis and algorithm have been discussed in Section-6 , Tables and figures are mentioned in section-7, Discussion & Conclusion has been given in Section-8. and References have been furnished in Section-9.

2. Review of Literature

S.Babu et.al [8] have discussed in their paper that distributed database systems use a query

optimizer to identify the most efficient strategy called plan to execute declarative queries. For a query on a given database and system configuration, the optimizer's plan choice is primarily a function of the selectivities of the base relations participating in the query. Query optimizers often make poor decisions because their compile time cost models use inaccurate estimates of various parameters.

Stefan Berchtold et.al [9] have described in their paper that the problem of retrieving all objects satisfying a query which involves multiple attributes is a standard query processing problem prevalent in any database system. The problem especially occurs in the context of feature based retrieval in multi databases.

Falout C. Barber et.al [1] have mentioned in their paper that the cost function in task allocation is sum of inter processor communication and processing cost that are actually different in measurement unit.

Hong Chen et.al [3] have discussed in their paper that the multi query processing takes several queries as input, optimizes them as a whole and generates a multi query execution strategy.

Cristina Lopez et.al [5] have defined in their paper that population of individuals known as chromosomes , represent the possible solutions to the problem. These are randomly generated , although if there is some knowledge available concerning the said problem , it can be used to create part of the initial set of potential solutions.

3. Problem Analysis

The query is submitted by user to the query distributor agent and then it will be distributed .After receiving the user query , the query distributor agent sends sub queries to responsible local optimizer agents. The query distributor agent can also create search agents if needed. The local optimizer agents apply a genetic algorithm based sub query optimization and return a result table size to the global optimizer agent. The global optimizer agent has the responsibility to find best join order via network. The global optimizer agent receives resultant table size information from local optimizer agents. Using an evolutionary method, it finds a semi optimal join order. However, this time the genetic algorithm fitness function is based on minimizing communication rate among different sites.

4. Need / necessity of Genetic algorithm

In distributed query processing environment , a single query may have a single plan or multi plans. Similarly a single plan may have a single task or multi tasks. In the same manner the multi query may have multi plans with multi tasks per plan. In this work our aim is to find average sum of the plans and degree coefficient between the tasks within the plans. Since it is a NP complete problem we need Genetic algorithm to solve the problem.

5. Problem Formulation

The first step to represent this problem as a genetic algorithm problem is determining the chromosome, genetic algorithm operators and fitness function. For the crossover , one point in the selected chromosome would be selected along with a corresponding point in another chromosome and then the tails would be exchanged. Mutation processes causes some bits to invert and produces some new information. The only problem of mutation is that it may cause some useful information to be

corrupted. Therefore the best individual is used to proceed forward to the next generation without undergoing any change to keep the best information.

Defining fitness function is one of the most important steps in designing a genetic algorithm based method, which can guide the search toward the best solution. After calculating the fitness function value for each parent chromosome , the algorithm will generate **n** number of children. The lower a parent chromosome's fitness function value, the higher probability it has to contribute one or more offsprings to the next generation. After performing operations , some chromosomes might not satisfy the fitness and as a result the algorithm discards this process and gets **q** (q<=n) children chromosomes. The algorithm then selects **n** chromosomes with the lower fitness value from the **q+n** chromosomes (**q** children and **n** parents) to be parent of the next generations. This process will be repeated until a certain number of generations are processed, after which the best chromosome is chosen.

6. Experimental results, analysis and algorithm

Maximum generations=100

Numberofqueries=100

Number of relations=100

Size of Chromosome (Plan in a query)=5

Population=round(rand(numberofqueries, sizeof chromosome))

Probability for crossover operation=0.06

Probability for mutation operation=0.001

Crossover point= round(1+rand*(size of chromosome-1))

6.1. Algorithm

Sum=0;

to=initial CPUtime;

t1=CPUtime after mutation and crossover operation

CPUtime , t2=t1-t0;

for i=1 : number of queries

planselect(i)=x(i)/(numberofqueries*planquery)

real_cost(i)=planselect(i)/numberofqueries +t2;

est_cost(i)=real_cost(i)/number of queries;

sum=sum+est_cost(i);

end

avgsum=sum/number of queries;

for i=1 : number of queries

if(avgsum>est_cost(i))

geneval(i)=(avgsum-real_cost(i))+est_cost(i);

else

geneval(i)=(real_cost(i)-avgsum)+est_cost(i);

end

end

x(i) represents number of chromosomes.

Crossover point, cp=2

Size of chromosomes=5

There is no doubt that dynamic programming methods always give us optimal solution. However , since the time and space complexity of the genetic algorithm base optimization is much less, it is not a practical approach for high amount of nested joins.

An evolutionary query optimization mechanism in distributed heterogeneous systems has been proposed using genetic algorithm approach. Genetic and randomized algorithms do not generally produce an optimal access plan.

7. Tables, Figures

7.1. Table -1

Plan	Population	X(Plan)	Est_cost	Gene value
1	11111	31	0.015575	0.020025
2	11010	11	0.011575	0.015975
3	11100	07	0.010775	0.023175
4	00000	00	0.009375	0.035775
5	11010	11	0.011575	0.015975
6	11011	27	0.014775	0.012825
7	01010	10	0.011375	0.017775
8	00000	16	0.012575	0.006975
9	00000	00	0.009375	0.035775
10	11001	19	0.013175	0.001575

As shown above in the table, we have 10 different plans of various sizes.

For example, the estimated cost of 5^{th} plan is 0.011575.The value of the plan is 11. The corresponding cost of Gene value in this case is 0.015975.

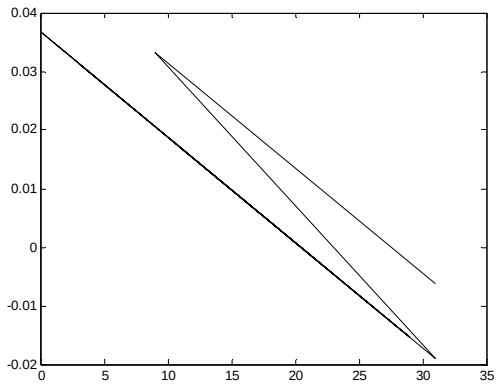
The estimated cost of 10^{th} plan is 0.013175. The value of the plan is 19. The corresponding cost of Gene value in this case is 0.001575.

The CPU time recorded after mutation and crossover operation is 0.09375.

The average sum of plans is 0.12015.

The average association degree coefficient between genes is 0.1358.

7.2. Figure –1



Plan

(Plan Vs Cost of Gene value)

8. Discussion and Conclusion

Data allocation defines the type of data stored while operation allocation states where accessing and processing of operations i.e. Select , Project , Join etc. are taken place.

The problem of retrieving all objects satisfying a query which involves multiple attributes is a standard query processing problem prevalent in any database system. Task allocation is an essential phase in distributed database system. To find the solution to task allocation problem, complete knowledge about the tasks and processors should be accumulated .

9. Reference

[1] Falout C. Barber , R.Flickner M, Efficient and effective query , Journal of Intelligent Information Systems-2000 .

[2] Guohua , Shuzhi Zhang , Dongming Zhang , The College Of Information Science and Engineering , Yanshan University , QinHuangdao , China International Conference in Computational Intelligence for modeling , IEEE transaction , 2006.

[3] Hong Chen , Sheng Zhou , Shan Wang , School of Information , Remin University China , International Conference in Data and knowledge Engineering , 2005.

[4] Lynda Tamine , Claude Chrisment , Mohand Boughanem , University Of Toulouse, France , IPM-2003.

[5] CristinaLopez, Vicente P.Guemero-Bote,University of Extremadura ,Badajoz , Spain ”Proc.of ACM Sigmod Intl.conf of Management of Data” , June 2005.

[6] W.T.Balke and V.Guntzer ,” Multi objective query processing for database “ , VLDB , 2004.

[7] K.Gajos and D.S.Weld ,” Preference elicitation for interface optimization”, UIST,2005.

[8] S. Babu, P.Bizarro, D.Dewitt , Proactive Reoptimization “ Proc. Of ACM SIGMOD Intl.conf. of Management of Data “ , June 2005

[9] Stefan Berchtold , Cristian Bohm, University of Munich , Oettinge Str , Germany 2001.

First Author: Sambit Kumar Mishra

1. Passed B.E. in Computer Sc.&Engg. From Amravati University , Maharastra in 1991.
2. Passed M.Tech. in Comp.Sc. from Indian School of Mines , Dhanbad.
3. Continuing ph.D.(Comp.Sc.)in Optimal Query Processing Techniques using soft computing Tools under Prof.Dr. Srikanta Pattnaik , Ex-Professor , U.C.E. , Burla.
4. Total Teaching Experience : 16 Years in various Engineering Colleges , Orissa , India.
5. Life Member of Indian society for Technical Education.
6. Member of International Association of Engineers.
7. Participated and Published 07 nos. of Conference papers in National and International Conferences.

Second Author: Dr.Srikanta Pattnaik

1. Passed B.E. from U.C.E. , Burla.
2. Completed M.E. and ph.D. from Jadavpur University , West Bengal.
3. Guided more than 07 ph.D. students.
4. Editor and Editor in chief of many journals.