Cardinality Estimation of Distributed Join Queries

Faïza Najjar and Yahya Slimani
Dept. des Sciences de l'Informatique - Faculté des Sciences de Tunis
1060 Tunis - Tunisia - Fax: +216 (1) 885.190
Email : faiza.najjar@planet.tn, yahya.slimani@fst.rnu.tn

Abstract

Estimation of selectivities or intermediate result sizes are useful in query optimization, as a means of determining the feasibility of queries, and as a quick way of answering queries for which the size of the answer is of interest in its own right. In this paper we propose a statistical method for estimating the cardinality of the resulting relation after an equi-join and projection. Under various "uniformity" assumptions on the data, this method relies on statistical information maintained by the system. The result of some experiments on random generated queries and databases are shown. Our method achieve low relative errors, typically about 10%.

1 Introduction

Query optimization is still an active area of database research, not only for relational databases but also for object-oriented databases. A query optimizer aims to decide on the most efficient query execution plan, i.e. the least costly to be executed, among many possible plans for a given query will determine the execution sequence order of relational operators such as joins, selections and projections. Choosing an efficient execution plan relies on cost estimates derived from statistics on the database and intermediate query results. Estimating the size of results, be they intermediate or final results, has become even more important in today's systems of much larger database sizes, possibly distributed over a LAN or a WAN. In such systems, the query execution plans are expected to diverge much more in cost due to the database size and the volume data transmission. Therefore, good estimates for the cost of database operations are critical to the effective operation of query optimizer and ultimately of the database system that rely on them.

Most previous work can be classified into four methods [2, 5]: (1) Non-parametric methods which make no a-priori assumptions concerning the data distribution; (2) Parametric methods rely on assumptions about the underlying distributions of attributes; (3) Sampling methods execute the queries to be optimized on small samples of the real database, and use the results of these trials to determine cost estimates; and (4) Curve-fitting use a general polynomial function and apply the criterion of least-square-error to approximate attribute value distribution.

Sampling methods has been recently investigated for estimating the resulting sizes of queries. They usually give more accurate estimation than all other methods [5].

In this paper, we propose a statistical method to estimate the cardinality of derived relations, obtained by relational operators, most notably by a join or a projection, and to improve such cost estimation. This method requires statistical a-priori informations, available in the database profile, and employs the ideas of statistical sampling [5]. Probabilities can be applied to statistical informations and used to predict the likelihood of an outcome.

The rest of this paper is organized as follows. Section 2 defines some terminology related to the cost based optimization. Section 3 reviews some of the commonly used estimation method for database in the literature, and it then presents a statistical method estimation for join selectivities. Section 4 propose a sampling-based estimation of the number of distinct values of an attribute in a relation. Our experiment results are given in section 5. Section 6 summarizes our results and indicates directions for future work.

2 Notation and definitions

The quantitative properties that summarize the instances of a database are its *database profile [7]*. A database profile describes aspects such as the number of tuples, the number of distinct values, the distribution of values, and the correlation between attributes. It is assumed that the database profile is maintained in the system catalog, and all the statistics, about relations and attributes, are initialized when the database is loaded, and updated periodically by applying a database operation.

We use $|R|$ to denote the cardinality of a relation R, and $|A|$ to denote the cardinality of the domain of an attribute A in a given query. $R(A)$ denotes the set of distinct values for the attribute A appearing in R. The selectivity model for join and semijoin operations is employed to assess its cost. Under this model, it is assumed that there is a number σ_{ij}, called join selectivity, associated with each join $R_i \bowtie_A R_j$ such that $|R_i \bowtie_A R_j| = \sigma_{ij} * |R_i| * |R_j|$. In the other hand, the semijoin selectivity, ρ_{iA}, is associated with each semijoin $R_j \ltimes_A R_i$, such that $|R_j \ltimes_A R_i| = \rho_{ij}|R_j|$. Both σ_{ij} and ρ_{ij} are rational numbers ranging from 0 to 1.

3 Estimating the cardinality of a join

In query processing, the knowledge of resulting cardinalities (qualifying number of rows) of operations is typically very important to the generation of efficient query plans. The problem of estimation of selectivities to an acceptable degree of accuracy remains an open problem [8] even though it has been studied in the past [6]. In this section we consider the problem of estimating join selectivities. Since in general computing, a join is more expensive than computing a selection or projection, finding a good estimation for joins is critical. Some of the commonly used algorithms are given by Chen and Yu [1, 3].

3.1 Chen's formula

To estimate the expected number of tuples in a relation resulting from an arbitrary number of joins, specified by a join query graph, Chen and Yu [1, 3] propose the following proposition, under the traditional assumptions (uniform distribution of values and independence of attributes within the same relation):

Proposition 1 *Let $G = (V, E)$ be a connected join graph, where $R_1, R_2, ..., R_p$ are the relations corresponding to vertices in V, $A_1, A_2, ..., A_q$ are the distinct attributes associated with edges in E. Let m_i be the number of different vertices (relations) that edges with attribute A_i are incident to. Suppose R_M is the relation resulting from all join operations between relations in G, then the expected number of tuples in R_M, $N_T(G)$, is given by*

$$N_T(G) = \frac{\prod_{i=1}^{p} |R_i|}{\prod_{i=1}^{q} |A_i|^{m_i - 1}}$$

Note that $|A|$ denotes the cardinality of the domain of an attribute A in a given query. In fact, $|A|$ is defined as the union over all relations referenced by the query of all known possible values of attribute A in these relations before the query is processed. For example, if there are three relations, R_1, R_2, and R_3, with attribute A, and $R_1$and R_2 are referenced by the query, the domain of A, for this query is the union of all known possible values in attribute A of R_1 and R_2.

3.2 Estimating one-join

Under the selectivity model, we have for each join $R \bowtie_A S$, the cardinality of derived relation given by $|R \bowtie_A S| = \sigma_{ij} * |R| * |S|$. The probability that a pair of tuples (t_R, t_S), where $t_R.A \in R(A)$ and $t_S.A \in S(A)$ will be in the resulting relation is : $|R(A) \cap S(A)| / |R(A)||S(A)|$. Thus, in order to estimate the join selectivity, we have to sufficiently approximate the expected number of distinct values in common between the two relations. Let X, a random variable with integer values, be the expected number of distinct values in common between the two relations. Then, as pointed out in [9], the expectation of X by elementary computation is expressed in the following proposition :

Proposition 2 *Let $R \bowtie_A S$ be a join of two relations R and S on an attribute A. p, k and m denote respectively $|R(A)|$, $|S(A)|$ and $|A|$. Letting j be the number of distinct values in common between R and S, i.e. $X = j$, such as $max(0, p + k - m) \leq j \leq \min(p, k)$. The expectation of X is thus given by:*

$$E(X) = \sum_j j * \left[\binom{k}{j} \binom{m-k}{p-j} / \binom{m}{p} \right]$$

The proof is given in [9]. However, the cardinality of a relation resulting from a join $R \bowtie_A S$ is estimated by

$$|R \bowtie_A S| = \frac{E(|R(A) \cap S(A)|)}{|R(A)||S(A)|} |R||S|$$

In the case of multi-joins, and when the resulting relation of an intermediate join will be used in the next join, we may estimate the number of distinct values of the later join attributes. While if two joins are uncorrelated, i.e. $R_1 \bowtie_A R_2$ and $R_3 \bowtie_B R_4$, a good approximation of the selectivity is $\sigma = \sigma_{12} * \sigma_{34}$.

4 Expected number of distinct values of an attribute

In this section, we consider a very challenging problem, which is the (sampling-based) estimation of the number of distinct values of an attribute in a derived relation, resulting from a relational operation. This problem is much more difficult, for example, than estimation of the selectivity of a join. Perhaps for this reason, distinct-value estimation has received less attention [4] than other database sampling problems, in spite of its importance in query optimization.

Throughout, we consider a fixed relation R consisting of n tuples and a fixed attribute A of this relation, having m distinct values, i.e. $m = |R(A)|$. For $1 \leq i \leq m$, let n_i to be the number of tuples in R with attribute value i, so that $\sum_{i=1}^{m} n_i = n$. We further assume that the values of the attribute A are arbitrary (non uniformly) distributed on the relation. Consider a sample of k tuples selected randomly and uniformly from R, without replacement, so as to minimize the estimation errors. The problem of estimating the number of distinct values of A in R, for a number k of tuples, can be described by the following combinatorial problem: "There are n balls with m different colors. Each ball has one color and let n_i be the number of balls with color i, $(1 \leq i \leq m)$. Find the expected number of colors if k balls are selected from the n balls." Let X, a random variables with integer values, be the expected number of colors of the k selected balls. Then, as pointed out in [9], the expectation of X by elementary computation is expressed as follows:

$$E(X) = m - \frac{1}{\binom{n}{k}} \sum_{i=1}^{m} \binom{n-n_i}{k}$$

Proof: We define a random variable Z to be the number of balls with different colors from a color i. The probability, p_i, that the color i does not appear in the k selected balls is equal to the hypergeometric probability:

$$p_i = P(Z = k) = \binom{n-n_i}{k} / \binom{n}{k}$$

Let $Y_i = 1$ if the color i does not appear in the selected balls and $Y_i = 0$ otherwise. Observe that:

$E(Y_i) = P(Y_i = 1) = p_i$

Letting X, a random variable, denotes the expected number of colors of the k selected balls from the n balls. Thus, we have $X = m - \sum_{i=1}^{m} Y_i$, we find that $E(Y) = E(m - \sum_{i=1}^{m} Y_i) = m - \sum_{i=1}^{m} E(Y_i) = m - \sum_{i=1}^{m} p_i$ ■

In particular, if $n_1 = ... = n_m = n/m$, i.e. the attribute-value distribution is perfectly uniform, we have $E(X) = m(1 - \binom{n-n_1}{k}/\binom{n}{k})$. After expansion and simplification, we obtain:

$E(X) = m(1 - \prod_{i=1}^{k} (\frac{n(m-1)}{m} - i + 1)/(n - i + 1))$.

In order to find the quality of the estimated number of distinct values of an attribute, we suggest to compute the variance, denoted V. Lower variance indicates more accurate estimates.

$$V(X) = \sum_{i=1}^{m} p_i(1-p_i) + 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} (p_{ij} - p_i p_j).$$

where $p_{ij} = P(Y_i = 1, Y_j = 1) = \binom{n-n_i-n_j}{k}/\binom{n}{k}$

In particular, if n is very large and $n = m_1 m$, it can be verified [9] that :

$\lim_{n\to\infty} V(X) = m(1-\frac{1}{m})^k(1-m(1-\frac{1}{m})^k) + m(m-1)(1-\frac{2}{m})^k$

It is easy to note that this limit is very weak ($\ll 1$). In other words, if we consider an attribute A in a relation R, then for a number k of tuples, the number of distinct values of this attribute are concentrated around this average, found by the expectation E. This affects the accuracy of the estimated joins, which are calculated using estimated number of distinct values of later join attributes in a multi-join query.

5 Experimental results

In this section, we study the performance of our statistical method for estimating the cardinality of derived relations. Throughout this section, we give numerical results allowing to judge the quality of the statistical method for estimating joins and projection. We present in addition a comparative study with exact values obtained by the execution of the different joins, as well as a comparison with the method proposed by Chen and Yu [1]. To compare our estimation method with others, we have considered the following parameters:

- *Database-dependent parameters*:
  - Cardinality of each relation R_i, $|R_i|$.
  - Cardinality of the domain of each attribute A_i, $|A_i|$.
  - Whether or not the distribution of attribute values is uniform.
  - Attribute values independence assumption.

- *Query-dependent parameters*:
  - n : Number of relations involved.
  - m : Number of joins in the join query graph.
- *Relative estimation error* $= \frac{|S-S^e|}{S} * 100$, where S and S^e are the actual and estimated result sizes.

To construct a join query graph, we used a random-number generator to produce edges between nodes. In the continuation, we show the influence of the different parameters on the quality of our proposed method of estimation, by varying one from parameters considered. First of all, we present a comparative study of our join estimation method with that of Chen and Yu, and exact values found in the practice, by a perusal total of relations to be joined. We suppose that values of an attribute are uniformly distributed on all the tuples in a relation. We denote respectively by JTk, Mk, and $Realk$ the value given by our estimation, that one obtained by Chen's method, and the exact value found by exhaustive processing; k indicates the number of joins in a join query graph. The table 1 allows to study the effect of the increase of the number of tuples in relations involved in a given join query on the cardinality of the resulting relation. Note that the cardinality of the domain of each attribute in a join is fixed initially to 1000. Each entry in the table is the average of random selection of data distributions and in each experiment we ran the join query generated 10 times. We observe that the three methods of estimation JTk, Mk and $Realk$ have the same behavior, independently of the number of tuples in each relation. This explains the fact that we are in the uniform case.

$\|R_i(A)\|$=1000 $\|A\|$=1000						
$\|R_i\|$	JT3	Real3	M3	JT4	Real4	M4
1500	3375	3375	3375	5062	5062	5062
2000	8008	8002	8000	16001	16002	16000
2500	15624	15627	15625	39062	39062	39062
3000	27000	27000	27000	81000	81001	81000
3500	42875	42875	42875	150062	150062	150062
4000	64001	64000	64000	256000	256001	256000
4500	91125	91126	91125	410065	410065	410062
5000	125000	125000	125000	625000	625000	625000

Table 1: Variation of the cardinality of relations resulting from joins

We are going now to compare our method with that Chen, on the criterion of value uniformity of attributes on all the database. To make this, we vary the cardinality of the domain of each join attribute, while the

	Relative errors(%)			
	Uniform Dist.		Non Uniform Dist.	
$\|R_i\|$	JT_2, M_2	JT_3, M_3	JT_2, M_2	JT_3, M_3
1000	5.8	8.3	0.8	1.3
2000	4.7	5.6	1.3	2.2
3000	7.2	6.7	7.9	1.3
4000	4.6	6.9	2.9	2.5
5000	2.2	9.5	3.6	1.5

Table 2: Relative errors in various data distribution

number of distinct values of these attributes remain fixed. This allows to translate the passage of an uniform distribution of attribute values on the database to a non uniform distribution.

The figure 1 shows that the curves with $JT3$ and $Real3$ have the same behavior. While $M3$ is off by orders of magnitude when the uniformity assumption on all the database is violated. This justify the fact that our estimation method is general, in the sense where it does not make hypotheses on the cardinality of the domain of a join attribute. Note that $JT3$ over-estimates the exact value of the cardinality of derived relation from join.

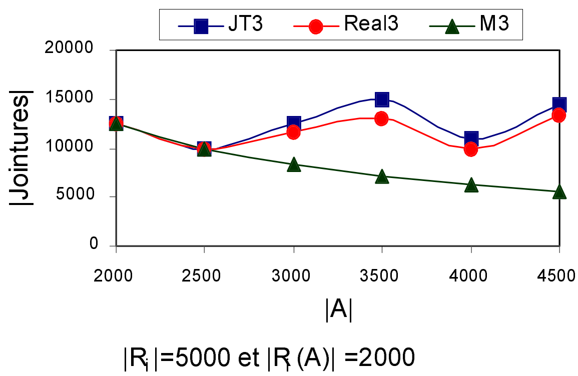

Figure 1: Effect of the cardinality of domain of join attributes on cardinality of joins.

Table 2 shows the overall relative errors, to gauge an average error between the estimated result sizes and their actual result size in different data distribution.

A clear trend that one can perceive is that the average errors via our statistical method reduces from total uniform data distribution (an average of 6.15%) to non uniformity over all the database (about 2.5%). Hence, we can conclude that our proposed estimation method can indeed assist in more accurate.

6 Conclusion

Identification of improved estimation of join queries is difficult. Ideally, a search for a good estimation would proceed by comparing analytical expressions for the accuracy of the estimators under certain considerations, and choosing the estimators that work best over a wide range of distributions. In this paper, we have achieved the following:

- We have proposed a statistical method to solve the problem of cardinality estimation for derived relations from joins. Our experiment results show that its performance is generally acceptable under various uniformity assumptions on database.
- We have used sample-based estimation of the number of distinct values of an attribute in a relation, after applying a relational operation, most notably a join or a semi-join.

In future work, we plan to investigate sampling-based estimation for query size estimation that overcomes the uniformity assumptions. We also need to develop hybrid estimators to deal with more complex join queries, resulting in further improvements in performance.

References

[1] M-S Chen and P.S Yu. Using Combination of Join and Semijoin Operations for Distributed Query Processing. *Proc. 10^{th} Int. Conf Dist. Comp. Syst.,* May 1990, 328-335.

[2] C.M. Chen and N. Roussopoulos. Adaptative Selectivity Estimation Using Query Feedback. *ACM-SIGMOD, Int. Conf. on Management of Data*, 1994.

[3] M-S. Chen. P.S. Yu. Optimization of parallel execution for multi-join queries. *IEEE Trans. on Know. and Data Eng.,* vol. 8(3), Jun. 1996, 416-428.

[4] P. Haas, J. F. Naughton, S. Seshadri, and L. Stokes. Sampling-based estimation of the number of distinct values of an attribute, *Proc. of the 21st VLDB Conference, Zurich (Swizerland)*, 1995, 311-321.

[5] B. Harangsri, J. Shepherd, and A. Ngu. Selectivity Estimation for Joins Using Systematic Sampling. TR, University of New South Wales, Sydney 2052, Australia, 1997.

[6] R.J. Lipton and J.F. Naughton. Query size estimation by adaptative sampling. *In Proc. of the 14^{th} Int. Conf. on VLDB*, Los Angeles (California), Aug. 1990, 240-251.

[7] M. Mannino, P. Chu and T. Sager. Statistical profile estimation in database systems, *ACM Computing Surveys,* vol. 20(3), Sep. 1988.

[8] A. Bellussi and C. Faloutsos. Self-Spacial Join Selectivity Estimation Using Fractal Concepts. *ACM TIS*, vol. 16(2), Apr. 98, 161-201.

[9] F. Najjar. Optimizing Join Queries in Distributed Database Systems. PhD Thesis, Faculté des Sciences de Tunis, 1999.