DOI:10.1145/1562764.1562786

Technical Perspective
Relational Query Optimization—Data Management Meets Statistical Estimation

By Surajit Chaudhuri

RELATIONAL SYSTEMS HAVE made it possible to query large collections of data in a *declarative* style through languages such as SQL. The queries are translated into expressions consisting of relational operations but do not refer to any implementation details. There is a key component that is needed to support this declarative style of programming and that is the *query optimizer*. The optimizer takes the query expression as input and determines how best to execute that query. This amounts to a combinatorial optimization on a complex search space: finding a low-cost execution plan among all plans that are equivalent to the given query expression (considering possible ordering of operators, alternative implementations of logical operators, and different use of physical structures such as indexes). The success that relational databases enjoy today in supporting complex decision-support queries would not have been a reality without innovation in query optimization technology.

In trying to identify a good execution plan, the query optimizer must be aware of statistical properties of data over which the query is defined because these statistical properties strongly influence the cost of executing the query. Examples of such statistical properties are total number of rows in the relation, distribution of values of attributes of a relation, and the number of distinct values of an attribute. Because the optimizer needs to search among many alternative execution plans for the given query and tries to pick one with low cost, it needs such statistical estimation not only for the input relations, but also for many sub-expressions that it considers part of its combinatorial search. Indeed, statistical properties of the sub-expressions guide the exploration of alternatives considered by the query optimizer. Since access to a large data set can be costly, it is not feasible to determine statistical properties of these sub-expressions by executing each of them. It is also important to be able to maintain these statistics efficiently in the face of data updates. These requirements demand a judicious trade-off between quality of estimation and the overheads of doing this estimation.

The early commercial relational systems used estimation techniques using summary structures such as simple histograms on attributes of the relation. Each bucket of the histogram represented a range of values. The histogram captured the total number of rows and number of distinct values for each bucket of the histogram. For larger query expressions, histograms were derived from histograms on its sub-expressions in an adhoc manner. Since the mid-1990s, the unique challenges of statistical estimation in the context of query optimization attracted many researchers with backgrounds and interests in algorithms and statistics. We saw development of principled approaches to these statistical estimation problems that leveraged randomized algorithms such as probabilistic counting. Keeping with the long-standing tradition of close relation between database research and the database industry, some of these solutions have been adopted in commercial database products.

The following paper by Beyer et al. showcases recent progress in statistical estimations in the context of query optimization. It revisits the difficult problem of efficient estimation of the number of distinct values in an attribute and makes a number of contributions by building upon past work that leverages randomized algorithms. It suggests an unbiased estimator for distinct values that has a lower mean squared error than previously proposed estimators based on a single scan of data. The authors propose a summary structure (*synopsis*) for a relation such that the number of distinct values in a query using multiset union, multiset intersection, and multiset difference operations over a set of relations can be estimated from the synopses of base relations. Furthermore, if only one of the many partitions of a relation is updated, the synopsis for just that partition must be rebuilt to derive the distinct value estimations for the entire relation.

The authors showcase recent progress in statistical estimations in the context of query optimization.

It has been 30 years since the framework of query optimization was defined by System-R and relational query optimization has been a great success story commercially. Yet, statistical estimation problems for query expressions remain one area where significant advances are needed to take the next big leap in the state of the art for query optimization. Recently, researchers are trying to understand how additional knowledge on statistical properties of data and queries can best be gleaned from past executions to enhance the core statistical estimation abilities. Although I have highlighted query optimization, such statistical estimation techniques also have potential applications in other areas such as data profiling and approximate query processing. I invite you to read the following paper to sample a subfield that lies at the intersection of database management systems, statistics, and algorithms. C

Surajit Chaudhuri (surajitc@microsoft.com) is a principal researcher and research area manager at Microsoft. He is an ACM Fellow and the recipient of the 2004 ACM SIGMOD Contributions Award.