Materialized View Selection in a Data Warehouse Using Evolutionary Algorithms

Jorng-Tzong Horng
Department of Computer Science and Information Engineering National Central University, Taiwan
horng@db.csie.ncu.edu.tw

Yu-Jan Chang
Department of Computer Science and Information Engineering National Central University, Taiwan

Baw-Jhiune Liu
Department of Computer Science and Information Engineering, Yuan-Ze University, Taiwan

Cheng-Yuan Kao
Department of Computer Science and Information Engineering, National Taiwan University, Taiwan

Abstract

A data warehouse stores lots of materialized views to provide an efficient decision-support or OLAP queries. The view-selection problem addresses to select a fittest set of materialized views from a variety of MVPPs (Yang, 1997) forms a challenge in data warehouse research. In this paper, we present genetic algorithm to choose materialized views. We also use experiments to demonstrate the power of our approach.

1 INTRODUCTION

A data warehouse system is a repository of integrated information, which can be provided for query or analysis. Data warehouse systems can collect and maintain information form multiple distributed, autonomous, or heterogeneous information sources, related data retrieved from the information source can be processed and transformed into internal types available for data warehouse systems.

The problem solved in this paper can be described as follows. Select a set of materialized views in a data warehouse to minimize total query response time and view maintenance cost of the selected views. This problem was denoted as view-selection problem (VSP) in (Gupta, 1997). Our approach is based on the concept of Multiple View Processing Plan (MVPP) (Yang, 1997).

2 OUR APPROACH

Each chromosome is represented by two parts. The former part, we call ***query-plan string,*** records which query plan is chosen to answer a query. In database query processing. A query may have many execution plans. Each execution plan will use many intermediate operations. The latter part, we call ***view string***, records which view will be materialized in a data warehouse.

We then use the basic strategies including selection, crossover, and mutation in our genetic approach.

3 EXPERIMENTS AND RESULTS

In Our experiments, the problem size is represented as*xQxxn*. The number before the character Q indicates the number of queries and that after Q is the number of views involved in such queries.

	GLS	YKL97	YKL force
7Q 46n	2.56	1	13
8Q 56n	7.7	2	166
9Q 66n	10.2	8	3306
10Q 94n	33.9	69	NA

Table 1: The Comparison of Execution Time of Different Problem size

Table 1 shows the comparison of our approach with other approaches in the execution time of the different problem sizes. The YKL_force approach on Table 1 means it first applies YKL97 (Yang, 1997) to find the best MVPP and then applies brute-force to select a set of views such that the total cost including query cost and maintenance cost is minimum. The Symbol "NA" in Table 1 indicates it is impossible to test all the solutions because the search space is too large. We find our approach can save much time than YKL97 in large problem size from Table 1. As to the solution quality, Table 2 demonstrates the minimal cost of each approach found for different queries.

	GLS	YKL97
4 query	958.65	2543.42
5 query	3478.69	5063.46
6 query	3483.19	5396.46
7 query	1264473.19	1266488.46
8 query	1714923.31	1721380.53
9 query	2165145.19	2167608.66
10 query	2176432.88	2392646.35

Table 2: Execution quality of Different Problem size

References

H. Gupta. (1997) Selection of views to Materialize in a Data Warehouse. Proceedings of the 23rd VLDB conference, Athens, Greece, pages 156-165.

J. Yang, K. Karlapalem, and Q. Li (1997). Algorithms for Materialized View Design in Data Warehousing Environment. VLDB page 136-145