Materialized View Selection Using Self-Adaptive Perturbation Operator-Based Particle Swarm Optimization

Amit Kumar, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India
T. V. Vijay Kumar, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India

ABSTRACT

A data warehouse is a central repository of time-variant and non-volatile data integrated from disparate data sources with the purpose of transforming data to information to support data analysis. Decision support applications access data warehouses to derive information using online analytical processing. The response time of analytical queries against speedily growing size of the data warehouse is substantially large. View materialization is an effective approach to decrease the response time for analytical queries and expedite the decision-making process in relational implementations of data warehouses. Selecting a suitable subset of views that deceases the response time of analytical queries and also fit within available storage space for materialization is a crucial research concern in the context of a data warehouse design. This problem, referred to as view selection, is shown to be NP-Hard. Swarm intelligence have been widely and successfully used to solve such problems. In this paper, a discrete variant of particle swarm optimization algorithm, i.e. self-adaptive perturbation operator based particle swarm optimization (SPOPSO), has been adapted to solve the view selection problem. Accordingly, SPOPSO-based view selection algorithm (SPOPSOVSA) is proposed. SPOPSOVSA selects the Top-K views in a multidimensional lattice framework. Further, the proposed algorithm is shown to perform better than the view selection algorithm HRUA.

KEYWORDS

Analytical Queries, Data Warehouse, Materialized Views, Particle Swarm Optimization, View Selection

1. INTRODUCTION

Enterprises throughout the world use business intelligence processes, technologies and tools to transform enormous amount of data from operational systems into existing, actionable and decision ready information. This information supports the organization’s decision-making process and helps in making future strategies that drives lucrative commercial action (Sauter, 2010; Turban et al., 2005). Different business processes gather and handle data in loosely coupled or independent form and thus, the data becomes inconsistent and the information obtained using it could be misleading (Immon, 2003). The best possible solution to this problem is a well-designed data warehouse, which forms a central repository of integrated data and is used for transforming data into information (Kimball
and Ross, 2002; Mohania et al. 1998). Data warehouse, in addition to being integrated, contains subject-specific and time variant non-volatile data with the goal to support business decision making (Inmon, 2003). Thus, data warehouse is key to business intelligence. Decision support applications access data warehouse using online analytical processing (OLAP) tools to extract information. The analytical queries involves summarization and aggregation of data from the continuously growing data warehouse. These analytical queries, in order extract relevant and required information from data in data warehouse, take long time for processing (Chaudhuri and Dayal, 1997; Gupta et al., 1997, Theodoratos and Sellis, 1997). Several query optimization techniques exist that minimize the query response time in order to expedite the decision making process. Amongst these, view materialization and indexing has been found to be effective (Harinarayan et al., 1996) and is also the focus of this paper. In order to aid and support analytical processing, Star schema storage representation is used by a data warehouse to store data. Star schema comprises a single fact table, which stores business measures, surrounded by several dimension tables, which describe the measures and are related to the fact table through a primary-foreign key relationship. The result of analytical queries, which join facts and some of the dimensions is limited by restrictions enforced in the dimensions. Since the size of fact table is enormous, joining tables is a very time-consuming operation (Chan et al., 1999; Chirkova and Yang, 2011). Join operations can be optimized by implementing table views, which are created by joining dimension tables in accordance with the OLAP query. However, it is not an efficient approach in the context of a Decision Support System (Chan et al., 1999; Chirkova and Yang, 2011), as enormous quantities of data is processed and alike queries are recurrently repeated in warehousing applications. A virtual view does not store data but fetches data in accordance to the query posed on it. Therefore, for efficient data analysis, the views are pre-aggregated and pre-computed on the relevant dimensions and stored in the data warehouse. Such views are referred to as materialized views. Materialized views drastically improve the query performance by offering fast lookups on the precomputed data. Answering queries using materialized views takes less response time, as they are lesser in size in comparison to a data warehouse (Harinarayan et al., 1996). Selecting a suitable set of views that minimize the total query response time and the maintenance cost while conforming to the limited storage space is key to efficient and effective data warehouse design (Agrawal, 2000; Chirkova et al., 2001; Nadeau and Teorey, 2002). This is termed view selection, which is discussed next.

1.1 View Selection

View selection, which is concerned with selecting a suitable subset of views while conforming to the resource constraint in order to minimize the response time for analytical queries, is one of the most challenging problem in data warehouse design. Selecting optimal subset of views is a complex problem and is shown to be an NP-Hard problem (Gupta, 1997). Therefore, good set of views, if not optimal, that can minimize the response time for analytical queries is required to be selected. Alternately, views can be selected empirically or heuristically. View selection problem has been addressed for different frameworks namely lattice framework (Harinarayan et al., 1996), Cube framework (Shukla et al., 1998, 2000), AND-OR view graph (Gupta, 1997; Sun and Wang, 2009), Multi-View Processing Plan (Yang et al., 1997; Zhang et al., 1999). This paper addresses the view selection problem for a lattice framework.

A multidimensional lattice of views includes nodes signifying the views. It exhibits the partial order relationship between views. A path from a view \( v_1 \) to a view \( v_2 \) in the lattice signifies that grouping attributes on \( v_2 \) can be computed using grouping attributes on \( v_1 \). The root view of a lattice, which represents the base fact table, comprises grouping on all attributes whereas, None represents an empty set of group-by attributes. Consider a four dimensional lattice with dimensions \( A, B, C \) and \( D \) shown in Figure 1. The indices are given alongside the name of the views, in square brackets \('[\cdot] \)”, and the sizes are given alongside the name of the views, in circle brackets ‘(•)’. The number of nodes in the lattice is \( 2^4 = 16 \) views. Queries posed on any of the views in the lattice can be answered by the root view \( ABCD \). Further, due to partial ordering amongst views at different level, query on view
Related Content

Financial Trading Systems: Is Recurrent Reinforcement Learning the Way?
www.igi-global.com/chapter/financial-trading-systems/28382?camid=4v1a

Retaining Knowledge in Smaller Building and Construction Firms
www.igi-global.com/article/retaining-knowledge-in-smaller-building-and-construction-firms/182666?camid=4v1a
Engineering and Reengineering of Technology Enhanced Learning Scenarios Using Context Awareness Processes
www.igi-global.com/chapter/engineering-reengineering-technology-enhanced-learning/56071?camid=4v1a

Cooperation and Free Riding in Cyber Security Information-Sharing Programs
www.igi-global.com/article/cooperation-and-free-riding-in-cyber-security-information-sharing-programs/176383?camid=4v1a