International Journal of Distributed Artificial Intelligence
Volume 13 • Issue 1 • January-June 2021

Materialized View Selection Using Swap Operator Based Particle Swarm Optimization

Amit Kumar, Jawaharlal Nehru University, India
https://orcid.org/0000-0002-8594-4515

T. V. Vijay Kumar, Jawaharlal Nehru University, India

ABSTRACT

The data warehouse is a key data repository of any business enterprise that stores enormous historical data meant for answering analytical queries. These queries need to be processed efficiently in order to make efficient and timely decisions. One way to achieve this is by materializing views over a data warehouse. An n-dimensional star schema can be mapped into an n-dimensional lattice from which Top-K views can be selected for materialization. Selection of such Top-K views is an NP-Hard problem. Several metaheuristic algorithms have been used to address this view selection problem. In this paper, a swap operator-based particle swarm optimization technique has been adapted to address such a view selection problem.

KEYWORDS

Analytical Queries, Data Warehouse, Decision Making, Materialized View Selection, Particle Swarm Optimization, Swarm Intelligence

1. INTRODUCTION

Businesses predominantly use technology that results in the generation of enormous quantity of data. This raw data provides useful information about customers. In order to become successful and competitive, a company must utilize this data effectively and efficiently. Further, Business Intelligence is increasingly become a widely accepted tool for companies to gain a competitive advantage in the market space (Ranjan, 2005). The data warehouse accomplishes this objective of business intelligence by working with all the business related data, along with the enterprise's historic data, gathered from disparate data sources (Gupta & Singh, 2014; Kimball, 2008). The data warehouse converts this data into a multidimensional data model, which is thereafter used for cost-effective querying, analysis and decision making (Inmon, 2005). Data warehouses are subject-oriented and voluminous data repository is for answering complex analytical queries. The processing time of these queries is usually high. This time needs to be reduced for efficient decision making. Materialized views can be used to reduce this processing time. Materialized views comprise pre-computed aggregate data, which is stored on a disk

DOI: 10.4018/ijdai.2021010103

and is refreshed with changes in the data in the underlying data sources. The key issue associated with materialized views is that they become outdated if they are not constantly updated with the underlying data (Ross et.al, 1996). In a relational data warehouse, the information is stored using the star schema in which there is one centralized fact table with one or more dimension tables linked to it. In (Gray et. al, 1996), a data cube operator, which computes the aggregates over all subsets of the dimension specified in the operation, was proposed. In (Baralis et. al, 1997; Harinarayan et.al, 1996), a multidimensional lattice framework was used to indicate this relation between the aggregate views. Aggregates are reflected by the vertices of an n-dimensional lattice. In a lattice representation, the most beneficial views can be computed immediately from the schema of the data warehouse. There is no need to consider log files of the queries and/or their access frequency. For a star schema having one fact table and n dimension tables, the number of possible views would be . In literature, several view selection techniques exist that select appropriate subsets of views. In (Harinarayan et.al, 1996), a greedy view selection algorithm, referred to as HRUA (Haider & Vijay Kumar, 2011, 2017; Vijay Kumar & Haider, 2015), is proposed that selects Top-K views from a lattice of views. Selecting such Top-K views is an NP-Hard problem (Harinarayan et al., 1996) and therefore, several randomized stochastic optimization methods have been proposed in literature to address this problem. These can be classified as randomized (Vijay Kumar & Kumar, 2015), evolutionary (Vijay Kumar & Kumar, 2014; Kumar & Vijay Kumar, 2018), swarm (Sun & Wang, 2009; Arun & Vijay Kumar, 2015a, 2015b, 2017a, 2017b; Vijay Kumar & Arun, 2016, 2017; Kumar & Vijay Kumar,, 2017a, 2017b, 2017c, 2018, 2020). Also, multi-objective evolutionary algorithms VEGA (Prakash & Vijay Kumar, 2019a), MOGA (Prakash & Vijay Kumar, 2020a), SPEA-2 (Prakash & Vijay Kumar, 2019b) and NSGA-II (Prakash & Vijay Kumar, 2020b) have also been used to solve the materialized view selection (MVS) problem.

In this paper, a new swap operator based particle swarm optimization (NSOPSO), given in (El-Ashmawi et al., 2018, 2020), has been used to select subsets of views from a multi-dimensional lattice. Accordingly, a NSOPSO based MVS algorithm is proposed that selects the Top-K views in the perspective of a lattice framework.

The paper is organized as follows: Particle Swarm Optimization (PSO) is briefly discussed in section 2 followed by View selection using NSOPSO in Section 3. An example in section 4 illustrates the steps involved in the proposed MVS algorithm. Section 5 shows the experimental results followed by conclusion in step 6.

2. PSO

Particle Swarm Optimization (PSO) algorithm is a population-based swarm algorithm introduced in (Kennedy and Eberhart, 1995). It is modelled upon the movement aesthetics of a flock of birds. This algorithm uses a swarm of particles that are flown through a hyper dimension search space. A solution, or particle is moved according to its own local best known position, called pbest, and also by the best position of the swarm called gbest. These pbest and gbest values are kept up to date, as improved positions are unearthed by other particles. In PSO, at each time step, a particle is accelerated toward its pbest and gbest locations with random weighted accelerations (Kennedy and Eberhart, 1995):

$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)}, i = \left\{1, \ldots, SW\right\} \tag{1}$$

$$v_i^{(t+1)} = v_i^{(t)} + c_1 r_{i1} \times \left(p_{best_i}^{(t)} - x_i^{(t)}\right) + c_2 r_{i2} \times \left(g_{best} - x_i^{(t)}\right) \tag{2}$$