International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

Multi-Objective Materialized View Selection Using Discrete Genetic & Particle Swarm Optimization

Popuri Srinivasarao[1*] and Aravapalli Rama Satish[2]

Submitted: 27/04/2023 **Revised**: 28/06/2023 **Accepted**: 05/07/2023

Abstract: Using a data warehouse method, data from several heterogeneous and dispersed operating systems (OLTP) is retrieved, converted, and put into a centralized repository. It is primarily used to process queries and thoroughly analyze data that is important to decision-makers. Therefore, it is crucial to make this data available as soon as possible. Here we have the concept of the Materialize perspective. Data warehouses often store their given data as a collection of materialized perspectives. The most difficult part is deciding which views should be realized and quickly with less expensive functions. This paper presents, Multi-Objective Discrete Genetic Particle Swarm Optimization (DGPSO) based Materialized View Selection. Using DGPSO (discrete genetic operator based particle swarm optimization), the top-k views from a multidimensional lattice are chosen. Among the various objective functions in the proposed method, response costs, management costs, current query processing costs, and past query processing costs. The DGPSO-based mineralization view selection algorithm is able to choose views of higher quality for materialization, as demonstrated by a comparison between it and the basic view selection algorithm based on testing.

Keywords: *Data Warehouse, discrete genetic operator, particle swarm optimization, Materialized View Selection.*

1. Introduction

Businesses of today require the kind of information that could help in efficient decision-making. Since existing operational systems are distributed across several sources, may be inaccurate, and may only accurately represent the everyday activities of the organization, On-line Transaction Processing (OLTP) systems are usually unable to give analysts access to this sort of data.To solve this issue, as a possible solution, data warehousing was implemented [1]. With the use of several operational sources, this approach takes varying and in-consistant data, converts it, and then enters it into a big central repository to be used for archiving historical data for the organization. Online analytical processing (OLAP) technologies can query this data to determine the organization's business patterns. Businesses of today require the kind of information that could help in efficient decision-making [2]. On-line transaction processing (OLTP) systems are frequently unable to provide analysts with access to this kind of data as the organization's present operational systems are dispersed across several sources, might be inaccurate, and might only accurately reflect everyday operations. A potential solution to this issue, known as data warehousing, was put into operation [3]. After being transformed, this approach involves loading diverse and inconsistent data from many operational sources into a large central repository that maintains the historical data of the organization. OLAP tools may query this data to determine the organization's business trends In order to improve operations and make decisions more effectively, business operations, performance, and trend analysis are required. Analyses and queries are performed on the data in the data warehouse. A data warehouse is an analysis-focused federated store for data collected from an business systems many enterprise's. It is an exact copies of transaction data that has been formatted only for searching and analyzing. For data warehouses to access data more quickly and for analytical queries to be executed successfully, techniques for query processing and efficient access mechanisms are needed. In data warehouses, one technique used to increase the effectiveness of these queries is the usage of materialized views [4]. These views store data along with definition, in contrast to virtual views. Since these views are smaller in size, using them to respond to requests might speed up the process significantly.

Materialized view selection (MVS)'s primary goal is to select a suitable group of intermediate views to store in the displayed storage space [5]. With the use of a view representation framework, several MVS techniques investigate the issue space and choose the most effective views from the available information. These structures list all potential view sets for the task as well as their inherent connections. Using the edges to represent the correlations between the views and the nodes as the views, the structure is referred to as a graph. The data cube lattice's numerous

[1] School of Computer Science and Engineering, VIT-AP University, Amaravati, India
ORCID ID : 0000-0002-3810-496X
[2] School of Computer Science and Engineering, VIT-AP University, Amaravati, India
ORCID ID : 0000-0002-4323-8073
** Corresponding Author Email: popurisrinivas333@gmail.com*

properties are used to finish all viewpoints. In the lattice network, two views will be connected by an end if they have comparable properties. Additionally, the AND-OR (directed acyclic graph) DAG structure has been utilized in a number of research. A base relation or another perspective is applied to each relevant process in this graph-based structure as a view [6]. Each view in this method can be referred by either one of two sorts of nodes, such as an action node or an equivalence node. A relational algebra definition is represented by the operation node, and the process for creating the definition is explained by each of its identical linked nodes. The edges of the graph and the corresponding nodes are linked.

Whenever there are limitations on disc capacity or maintenance costs, materialized perspectives can be selected [7]. The highest performance is achieved by materializing every view in a data warehouse, this is a reasonable alternative, but there are restrictions on storage space and maintenance costs. However, without the restrictions associated with the data warehouse, none of the views can be materialized. Selecting which subset of views will cost the least to process queries and maintain, all feasible views must be considered. On the other hand, the most efficient subset of views must be chosen, and this is found to be an NP-Hard issue with exponential time complexity [8]. As a result, the only remaining option is to choose a suitable subset of views that minimizes the costs associated with query processing and maintenance. Some systems are unable to accommodate the vast amount of disc space needed to create all materialized views. When the underlying data of the base table changes, the materialized views must be updated, exactly as the cache, which becomes inconsistant if the main memory contents are changed.

A swarm intelligence-based heuristic was utilized in this study to solve the problem of view selections. In order to handle combinatorial and continuous optimization issues, swarm intelligence (SI), an advanced distributed intelligent model, has been widely and successfully applied. The behaviour of a swarm of insects, such as ants, termites, bees, and wasps, serves as the foundation for SI. It has been shown that SI techniques are reliable and flexible. In this analysis, A discrete genetic operator-based particle swarm optimization (DGPSO) swarm intelligence technique has been used to resolve the view selection problem.

2. Literature Survey

Arun B, Kumar T. et al. [9] developed an optimization method known as an Artificial bee colony (ABC) to select top-k views from a standard method. The ABC technique is used to create an algorithm that solves the view selection problem. The newly created method operates by randomly inserting N points based on GBFS. At the processed bee phase and onlooker bee phase to find the process of local exploration the random N-point inclusion operation was set. This method assesses the global optimum solutions as well as the top k views, improving their quality. A complex query takes longer to execute since the information might be expanded across several sites.R Goswami, DK Bhattacharyya, Dutta M. et. al. [10] reduced the overall query processing time and introduced a differential evolutionary method. Through providing a solution to multiple objective optimization issues, the selection of the best viewpoint for materialization difficulties is the major goal of this research. Utilizing Hive 0.12.0 queries and log files generated by the execution of the HDFS-related data warehousing model on each node, a prototype of the view selection model was created. Kumar TV, Kumar S. et. al. [11] explore into the MVS randomized algorithms. In this case, in the first stage of the (view selection two phase optimization algorithm, or VS2POA), which processes views in two stages, the top k views that are appropriate are selected. The simulated annealing process is the topic of the next stage. The constructed model is contrasted with the traditional approach, proving that the new VS2POA offers better views for datasets with larger dimensions.

Hamdi Issam, Bouazizi Emna, Jamel Feki, et. al. [12] Describe the dynamic selection of materialized views (DynaSeV) algorithm, which chooses views from the outcomes of incoming queries while taking into consideration the system's limitations on both execution time and storage space. On big data sets, complicated queries with many joins and aggregation processes are used to do this analysis. Decision makers expect their requests to be handled quickly even though the complexity of their queries. Materialized views are one of the optimization approaches used to keep to such a time limitation. The goal of this study is to improve the management of materialized views and to address issues with view maintenance and selection in real-time data warehouses (RTDW). They propose a novel update method that, depending on the system workload and the frequency of access to the materialized views, dynamically decides the way to preserve those views.

Stephan M¨uller, Kai H¨owelmeyer, Lars Butzmann, Stefan lauck, Hasso Plattner et. al. [13] developed a unique view maintenance technique that takes into consideration columnar storage's main-delta framework and resulting merging procedure. This work introduced many maintenance techniques, such as the eager approach, which propagates any change to the impacted materialized view as soon as it occurs in the base table. The next method is being lazy, which keeps a materialized view while the view is requested. Ravindra N. Jogekar, Ashish Mohod, et. al. [14] A methodology for choosing the appropriate materialized view has been presented in order to effectively combine fast query execution, low-cost query processing, and economical view maintaining within the limits of available storage.

Xin Li, Junlin Jiang, Xu Qian, Ziqiang Wang, et. al. [15] In a data warehouse setting, one of the important strategies for accelerating query response is materialized view selection. In this research, this problem is effectively and an innovative SFL approach for selecting materialized views effectively solves the problem. According to experimental outcomes on TPC-D benchmark data sets, in terms of total handling costs and query response times, the recommended approach outperforms other well-known methods.

B.Ashadevi, R.Balasubramania, et. al. [16] The costs for storage and query processing are decreased by only materializing the views that are the most cost-effective. Base interaction, processing frequency, update frequency, query access cost, and view maintenance cost are some of the cost indicators for selecting materialized views, and system storage space limitations, are all taken into consideration in the suggested structure. The preservation of the current materialized perspective is also part of this work. This framework takes high access frequency queries into consider; it does not take high storage space or low access frequency queries into consideration.

A Gong, Weijing Zhao, et. al. [17] provides a dynamic materialized view selection technique based on clustering. Prior to dynamically adjusting the materialized view set, it initially clusters the materialized views. The "jitter" that a dynamic materialized view selection method often exhibits is thus removed. The results of the experiment demonstrate that the approach not only enhances query response performance generally but also lowers the computational expense required while updating the materialized view[18][19].

3. Materialized View Selection Of Multi-Objective

instead of “It was observed that ...”). Remember to check spelling. If your native language is not English, please get a native English-speaking colleague to carefully proofread your paper.

The workflow of Multi-Objective Discrete Genetic Particle Swarm Optimization (DGPSO) based Materialized View Selection is represented in below Fig. 1.

In the design of data warehouse, the MVS issues are solved by using described DGPSO model. Described methodology is initiated by planning the Multiple View Processing Plan (MVPP) structure in which quires assisted plans are listed. Cost functions related multi-objective optimization problem from the materialization is formulated by described model. Current query processing costs, previous query processing costs, maintenance costs and response time cost are the proposed model objective functions. Declared multi-objective functions compromising gives materialization by selecting the top-k views in this model.

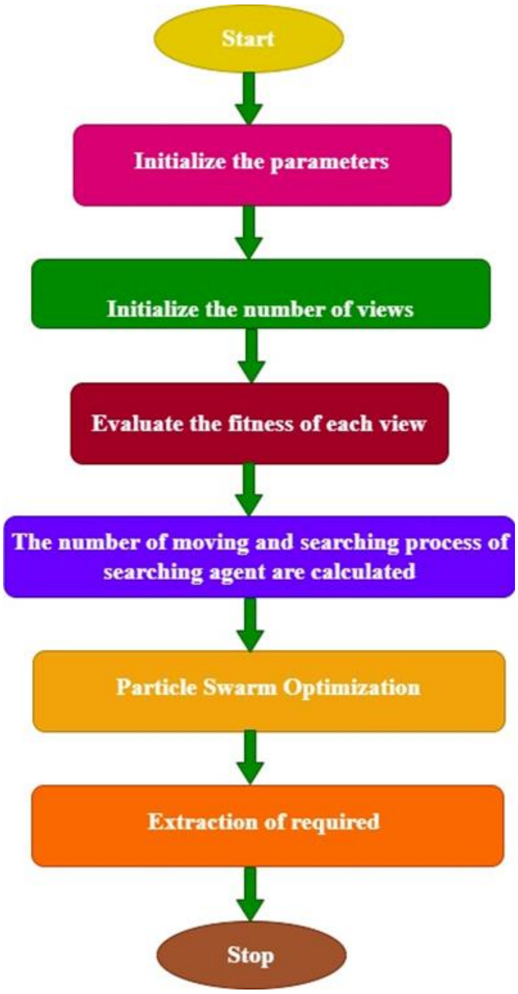

Fig. 1: Workflow of Discrete Genetic Materialized View Selection of Multi-Objective Using ‘Pso’

Different types of information are existed in data warehouse environment which generates some challenges. Because of the large data set, query answering response time is high. In order to decrease the query answering response time and access performance improvement, materializing the views are important. Materialized views selection (MVS) process too difficult, therefore it requires modern techniques in MVS process. A new DGPSOMVS model is developed for MVS with DGPSO in this model.

Described model starts from solutions (views) primary population. According to the constraint, group of views are generated by the initialization process. Group of views initiates the overall space and it is minimum than the size quantity of data warehouse’s space. A constant binary number is allotted to the every view as binary bit 0 or 1. As in the form of binary string, each MVPP view is encoded and in the structure of MVPP, amount of views are indicated by constant number. In MVPP structure, Not materialized views in the data warehouse are represented with bit ‘0’ and materialized views in the data warehouse are represented with bit ‘1’.

Definition of Fitness Function: maintenance cost and query numbers are minimized by mathematical model of some objectives in view selection. Cost function is defines the fitness function as follows:

$$\boldsymbol{Fitness}\ (\boldsymbol{G},\boldsymbol{M}) = \boldsymbol{\tau}(\boldsymbol{G},\boldsymbol{M}) \ldots\ldots (\mathbf{1})$$

Where, cost function is represents with $\boldsymbol{\tau}(\boldsymbol{G},\boldsymbol{M})$ and fitness function represents with Fitness $(\boldsymbol{G},\boldsymbol{M})$.

For example, For 'u' views, maintenance cost represented by U(u, M) in the presence of materialized views M set and sinks set as L. Therefore, G denotes AND-OR graph and S denotes the storage size then set of views/nodes are selected by the view selection problem and M = {V1, V2,…, Vm}, that reduces $\boldsymbol{\tau}(\boldsymbol{G},\boldsymbol{M})$,where

$$\boldsymbol{\tau}(\boldsymbol{G},\boldsymbol{M}) = \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{k}} \boldsymbol{f}_{\boldsymbol{Q}_{\boldsymbol{i}}}.\boldsymbol{Q}(\boldsymbol{Q},\boldsymbol{M}) \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{m}} \boldsymbol{g}_{\boldsymbol{v}_{\boldsymbol{i}}}.\boldsymbol{U}(\boldsymbol{V}_{\boldsymbol{i}},\boldsymbol{M}) \ldots (\mathbf{2})$$

Swarm intelligence technique with population is also known as PSO which is stimulated by fish schooling and birds flocking as social behavior. Potential solutions initial population is known as particles in the classical PSO which are generated as random. In the swarm population every particle is known about its velocity $\boldsymbol{v}_{\boldsymbol{id}}(\boldsymbol{t})$position $\boldsymbol{x}_{\boldsymbol{id}}(\boldsymbol{t})$global best position $\boldsymbol{p}_{\boldsymbol{gd}}(\boldsymbol{t})$and personal best position $\boldsymbol{p}_{\boldsymbol{id}}(\boldsymbol{t})$. In the solution space, for promising positions search directions are made by collectively distributed information among individuals. "i" th particle position and velocity are denoted as follows:

$$\boldsymbol{x}_{\boldsymbol{id}}(\boldsymbol{t}+\mathbf{1}) = \boldsymbol{x}_{\boldsymbol{id}}(\boldsymbol{t}) + \boldsymbol{v}_{\boldsymbol{id}}(\boldsymbol{t}+\mathbf{1}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (\mathbf{3})$$

$$\boldsymbol{v}_{\boldsymbol{id}}(\boldsymbol{t}+\mathbf{1}) = \boldsymbol{v}_{\boldsymbol{id}}(\boldsymbol{t}) + \boldsymbol{c}_{\mathbf{1}}\boldsymbol{r}_{\mathbf{1}}\big(\boldsymbol{p}_{\boldsymbol{id}}(\boldsymbol{t}) - \boldsymbol{x}_{\boldsymbol{id}}(\boldsymbol{t})\big) + \boldsymbol{c}_{\mathbf{2}}\boldsymbol{r}_{\mathbf{2}}\left(\boldsymbol{p}_{\boldsymbol{gd}}(\boldsymbol{t}) - \boldsymbol{x}_{\boldsymbol{id}}(\boldsymbol{t})\right) \ldots\ldots\ldots (\mathbf{4})$$

Where, d=1,2,….n, i=1,2,….N, and t denotes the t th iteration, 0 and 1 are values taken by random variables r1 and r2, dth dimension is denoted as d, cognitive parameter is denoted as c1 and finally social parameter is denoted as c2. Different continuous optimization problems are solved by designing of classical PSO. Later it is applicable to solve the discrete optimization problems.

Insert mutation operator and two cut PTL crossover operators are included in DGPSO. But in some cases, Random mutation operator and single point cyclic crossover are also included in this scheme for solving the view selection problem. These new operators are represented in below Fig. 2 and Fig. 3.

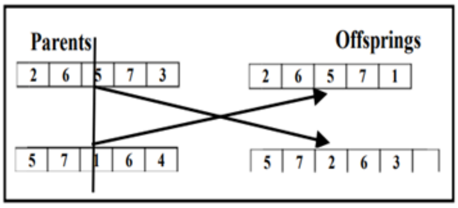

Fig. 2: SINGLE-POINT CYCLIC CROSSOVER

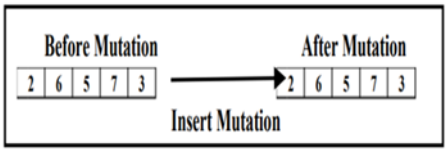

Fig. 3: RANDOM MUTATION

The view selection problems are addressed by adoption of these schemes in DGPSO model. Therefore, DGPSOMVS (DGPSO based Materialized views selection) model is described in which the Top-K views are selected from a multi-dimensional lattice. For every Top-K view, $\boldsymbol{xTKV}_{\boldsymbol{i}}(\boldsymbol{t}+\mathbf{1})$is updated position from $\boldsymbol{xTKV}_{\boldsymbol{id}}(\boldsymbol{t})$ Therefore this process is $\boldsymbol{\delta TKV}_{\boldsymbol{i}}(\boldsymbol{t}+\mathbf{1})$ is calculated after the calculation of $\boldsymbol{\lambda TKV}_{\boldsymbol{i}}(\boldsymbol{t}+\mathbf{1})$ and these calculations are required to compute the $\boldsymbol{xTKV}_{\boldsymbol{i}}(\boldsymbol{t}+\mathbf{1})$ Velocity Component: Particle velocity is denoted by$\boldsymbol{\lambda}_{\boldsymbol{i}}(\boldsymbol{t}+\mathbf{1}) = \boldsymbol{\omega} \otimes \boldsymbol{F}_{\mathbf{1}}(\boldsymbol{x}_{\boldsymbol{i}}(\boldsymbol{t}))$ in which mutation operator is represented with F1 with probability Z. 0 and 1 are values taken by random variables r1 and r2, and these are compared with Z. if Z is greater than r1 then perturbed position is generated by application of mutation operator F1 i.e$\boldsymbol{\lambda}_{\boldsymbol{i}}(\boldsymbol{t}+\mathbf{1}) = \boldsymbol{F}_{\mathbf{1}}(\boldsymbol{x}_{\boldsymbol{i}}(\boldsymbol{t}))$**.**

Cognition Component: particle cognition part is represented by declaring the second component as $\delta_i(t+1) = c_1 \otimes F_2\big(\lambda_i(t+1), p_i(t)\big)$ Cognition part of particle is also known as particle personal thinking. Here, crossover operator is represented with F2 with probability c1, $p_i(t)$ and $\lambda_i(t+1)$ are parent particles.

Social Component: particle social part is represented with$x_i(t+1) = c_2 \otimes F_3\left(\delta_i(t+1), p_g(t)\right)$ and these are association among the particles. crossover operator is represented with F3 with probability c2, $p_g(t)$ and $\delta_i(t+1)$ are parent particles.

Optimal top-k views are selected in described DGPSOMVS model for materialization. Some of the objective functions are also satisfied with this model and these are previous query processing cost, response time and maintenance cost. Materialized views are selected effectively with the utilization of multi objective based PSO. Therefore described model decreases the total cost and query response time.

4. Iv. Result Analysis

Described DGPSOMVS (DGPSO based Materialized views selection) model is experimentally verified in this section and comparative results are also represented. TPC-H (Transaction Processing Council Ad-hoc/decision support benchmark) named dataset is used for evaluation of described model efficiency. This data consists of business-related sequences of ad hoc queries and simultaneous data renovations. Decision support systems are also described by this dataset which are used in analysis of large dataset, high level complexity related query processing and gives solutions for this complex business questions.

Described model main objective is to find the group of views in such a way that these views are must be materialized, total cost of these materialized set is must be low in all possible materialized view set. Maintenance cost and query cost are included in MVPP total cost. Therefore these costs are analyzed and compared with other models.

Query cost comparison according to different data sizes are represented in Fig. 4. For comparative analysis previous models GTMVS (Game theory based MVS) and EGTMVS (Evolutionary game theory based MVS) are considered along with described DGPSOMVS model. Lower query costs are attained by DGPSOMVS model for each data size than other methods. Therefore, DGPSOMVS model achieves efficient performance than other methods.

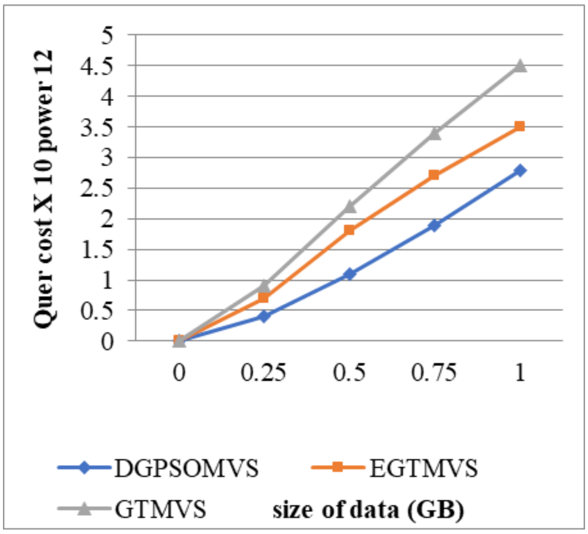

Fig. 4: Query Cost By Varying The Size Of The Data

Maintenance cost of views according to different data sizes of different models are illustrated in Fig. 5. For comparative analysis previous models GTMVS (Game theory based MVS) and EGTMVS (Evolutionary game theory based MVS) are considered along with described DGPSOMVS model. Therefore, DGPSOMVS model achieves reduction in maintenance costs than other methods.

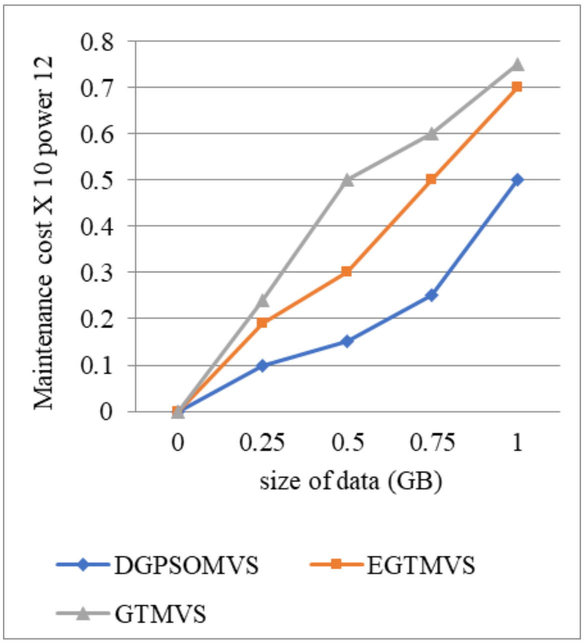

Fig. 5: Maintenance Cost Of Varying The Size Of The Data

PSO optimization technique for MVS is preferred by the described model. Cost functions are decreased and achieve high performance using this method in the data warehouse environment. Total optimal cost is used to measure the described model efficiency and performance comparative analysis is used to analyze the described model effectiveness than other models. Fig. 6 shows the comparative performance of different models in terms of total cost and shows the better performance attained for only DGPSOMVS model than other models.

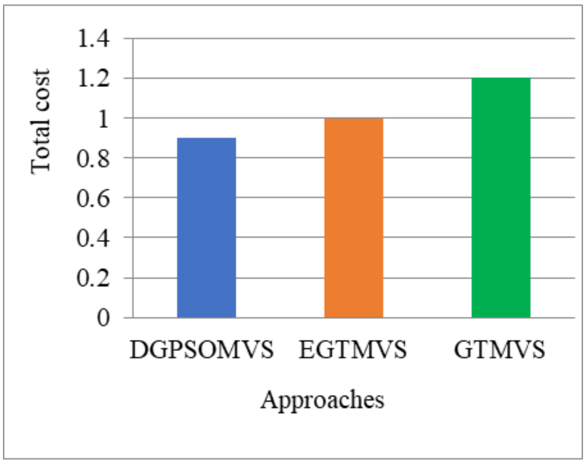

Fig. 6: Comparison Of The Total Cost Of Different Approaches

There is a difficulty in analytical queries, and it is important for large amount of data for improving the response time. In a data warehouse environment, one of the important issues is MVS which is used to increase the speed of query execution. With the selection of materializing suitable group from all views keeps the query response time decrement which is done by the MVS.

The response time is reduced by the described PSO optimization method without affecting the performance of model. Fig. 7 shows the response time comparison analysis for described model with other models.

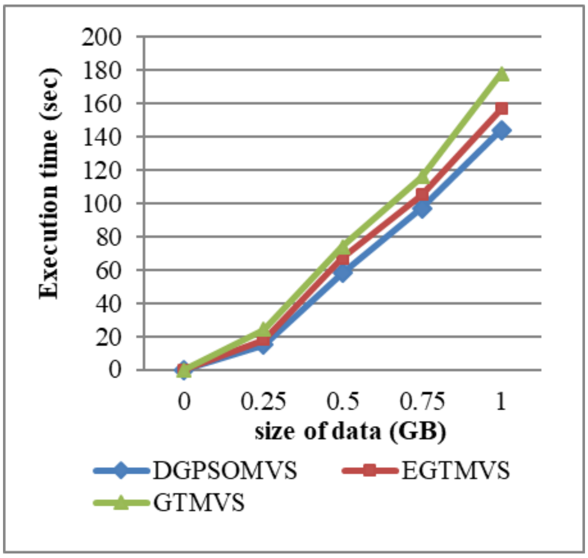

Fig. 7: Response Time Comparison

Result of Fig. 7 shows that, effective performance is achieved by the described DGPSOMVS model than other models. 0 sec as response time is recorded by DGPSOMVS model when 0 GB as data size. 15 sec as response time is recorded by DGPSOMVS model when 0.25 GB as data size. 58 sec as response time is recorded by DGPSOMVS model when 0.5 GB as data size. 97 sec as response time is recorded by DGPSOMVS model when 0.75 GB as data size and it takes 144 sec for 1 GB.

The comparisons of the previous models like GTMVS (Game theory based MVS) and EGTMVS (Evolutionary game theory based MVS) with DGPSOMVS model were carried out in terms of the Total View Evaluation Cost (TVEC) of the Top- {5, 6, 7, 8, 9, 10} views are calculated for 100 iterations. Fig.8, Fig.9 and Fig.10 shows the performed experiments graphical representation for 5, 7 and 10 respectively.

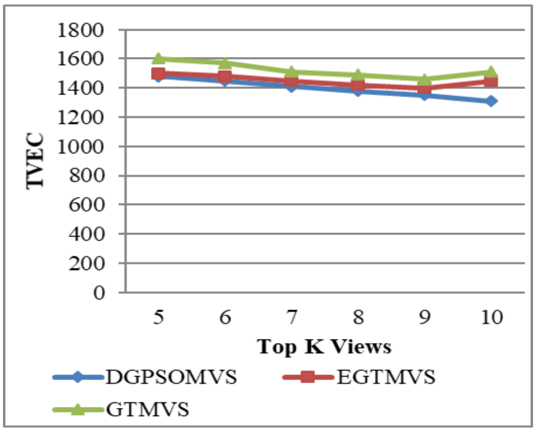

Fig. 8: Tvec For Dimension=5

From results, Top-K views selection is efficient in described DGPSOMVS model and it attains low values of TVEC than other models. Therefore computed views using DGPSOMVS is materialized and decreases the response time and find solutions for analytical queries which make the effective performance.

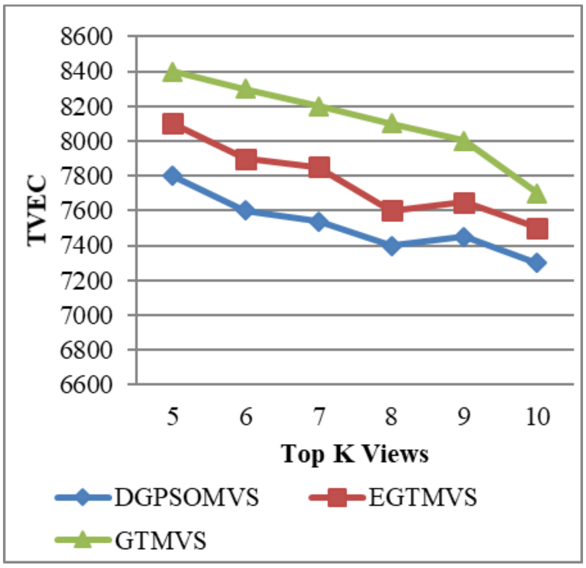

Fig. 9: Tvec For Dimension=7

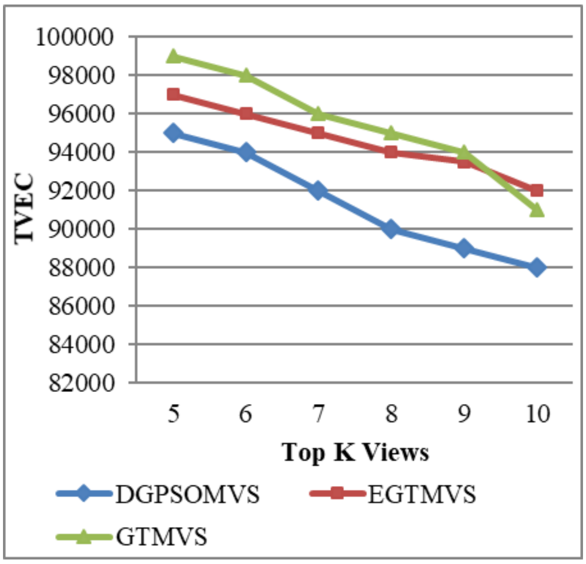

Fig. 10: TVEC FOR DIMENSION=10

5. Conclusion

In this paper, Multi-Objective Discrete Genetic Particle Swarm Optimization (DGPSO) based Materialized View Selection is described. Data warehouse contains high amount of data, so there is need have to improve the query processing with MVS in the data warehouse environment. Therefore DGPSOMVS model is described in this paper for solving the problems of large datasets. According to different costs with the result of materialization, the issue of multi-objective optimization is evaluated by described model. A multidimensional lattice is used to select the Top-K views with the described DGPSOMVS model according to view selection algorithm of DGPSO. DGPSO algorithm hybrid characteristics are used in this model which includes genetic and PSO operators as mutation and crossover. This model evaluates the process of selecting the Top-K views from every iteration. From results it is clear that, TPC-H dataset based described model is efficient than other models for MVS. It is also states that, described DGPSOMVS model effectively computes the materializing views and reduces the analytical queries response time. In the future, it will be possible to improve the materialized view selection by utilizing additional processes that have been enhanced, and it will also be possible to conduct an analysis of it utilizing a variety of benchmark datasets that contain a significant number of records. Both things will be possible in the future.

Author contributions

Popuri SrinivasaRao: Conceptualization, Methodology, analyzing, software Improvements, Writing—Original Draft Preparation, Supervision, Investigation, Resources, Data Curation.

Aravapalli Rama Satish: Conceptualization, Methodology, Software, Validation, Formal Analysis, Review & Editing, Visualization, Supervision, Project Administration.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Munawar, "Extract Transform Loading (ETL) Based data Quality for Data Warehouse Development", 2021 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI), Year: 2021

[2] Xiao Meng, Güneş Aluç, "Exploratory Data Analysis in SAP IQ Using Query-Time Sampling", 2021 IEEE 37th International Conference on Data Engineering (ICDE), Year: 2021

[3] Abhishek Gupta, Arun Sahayadhas, "Proposed Techniques to Optimize the DW and ETL Query for Enhancing Data Warehouse efficiency", 2020 5th International Conference on Computing, Communication and Security (ICCCS), Year: 2020

[4] Rahul Sawarkar, M.M. Baig, "Performance Tuning of Queries in Distributed System using Secure materialized views Approach", 2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC), Year: 2020

[5] Wesal A. Abdullah, Naji M. Sahib, Jamal M. Abass, "Creation of Optimal materialized views Using Bitmap Index and Firefly Algorithm in Data Warehouse", 2019 2nd Scientific Conference of Computer Sciences (SCCS), Year: 2019

[6] Nabila Berkani, Ladje Bellatreche, Carlos Ordonez, "ETL-aware materialized view selection in semantic data stream warehouses", 2018 12th International Conference on Research Challenges in Information Science (RCIS), Year: 2018

[7] ABBASSI Kamel, Tahar Ezzedine, "Creation of materialized views Based on Neural Network Algorithm", 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), Year: 2018

[8] Yuxin Liu, Chao Gao, Zili Zhang, Yuxiao Lu, Shi Chen, Mingxin Liang, Li Tao, "Solving NP-Hard problems with Physarum-Based Ant Colony System", IEEE/ACM Transactions on Computational Biology and Bioinformatics, Volume: 14, Issue: 1, Year: 2017

[9] Arun B, Kumar T. "Materialized view selection using artificial bee colony optimization",. Int J Intell Info Technol. 2017;13:26-49

[10] Goswami R, Bhattacharyya DK, Dutta M. "Materialized view selection using evolutionary algorithm for speeding up big data query processing", J Intell Inform Syst. 2017;49(3):407-433.

[11] Kumar TV, Kumar S. "Materialised view selection using randomised algorithms",. Int J Business Info Syst. 2015;19(2):224-240

[12] Issam Hamdi, Emna Bouazizi, Jamel Feki, "Dynamic management of materialized views in real-time data warehouses", 2014 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR), Year: 2014

[13] Stephan M¨uller, Lars Butzmann, Kai H¨owelmeyer, Stefan Klauck, Hasso Plattner, "Efficient View Maintenance for Enterprise Applications in Columnar In-Memory Databases", 17th IEEE International Enterprise Distributed Object Computing Conference, 2013

[14] Ravindra N. Jogekar, Ashish Mohod,"Design and Implementation of Algorithms for Materialized View Selection and Maintenance in Data Warehousing Environment", ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 9, September 2013

[15] Xin Li, Xu Qian, Junlin Jiang, Ziqiang Wang, "Shuffled Frog Leaping Algorithm for Materialized Views Selection", 2010 Second International Workshop on Education Technology and Computer Science, Year: 2010

[16] B.Ashadevi, R.Balasubramania, "Cost Effective App roach for Materialized Views Selection in Data Warehousing Environment", IJCSNS International

Journal of Computer Science and Network Security, VOL.8 No.10, October 2008

[17] An Gong, Weijing Zhao, "Clustering-Based Dynamic Materialized View Selection Algorithm", 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery, Year: 2008

[18] Srinivasarao, P., & Satish, A. R. (2023). Multi-objective materialized view selection using flamingo search optimization algorithm.Software: Practice and Experience, 53(4), 988-1012.

[19] P. Srinivasarao, A. R. Satish and K. L. Revathi, "An Expert Uncertainty in Healthcare Using Materialized View through Big-Query," 2023 Third International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 2023, pp. 1-5.

[20] Naidu k, P. ., Rao, V. L. ., Gunturu, C. S. ., Niharika, A. ., Anupama, C. R. ., & Srivalli, G. . (2023). Crop Yield Prediction Using Gradient Boosting Neural Network Regression Model . International Journal on Recent and Innovation Trends in Computing and Communication, 11(3), 206–214. https://doi.org/10.17762/ijritcc.v11i3.6338

[21] Ms. Elena Rosemaro. (2014). An Experimental Analysis Of Dependency On Automation And Management Skills. International Journal of New Practices in Management and Engineering, 3(01), 01 - 06. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/25