Chapter 6

Conclusion and Future Direction

This dissertation makes four important contributions to the body of knowledge on materializing views in data warehouses by selecting optimum set of views with respect to over all query efficiency, materialized view maintenance and memory space constraint of data warehouses. In this chapter, we summarize the main contributions made in this dissertation and provide directions for future works.

6.1 Conclusions

Following conclusions are drawn from the contributions in this dissertation.

- In Chapter 2, a comprehensive report on the approaches introduced to select views for materializing in data warehouses have been presented with associated issues and challenges that have been identified in the study. It has been observed that the scalability due to exponential explosion of solution space with dimension of data warehouses is a big issue with deterministic and heuristic algorithm based solution search methods applied in view selection. Evolutionary and stochastic methods like Genetic Algorithm (GA) and Simulated Annealing (SA) algorithms search solutions in a multi-dimensional fashion and can provide effective search performance in finding an optimum set of views for materializing near the global optimum. But in these approaches, the solution quality depends on different parameters and values that are specified. Soft-computing approaches in the view selection problem use clustering and associative rule mining on a (frequent) query versus view matrix. The quality of the quasi-optimum solutions discovered by these techniques depends on the pre-defined clustering parameters. Analysis of large number of complex queries for finding frequent significant sub-queries, aggregation functions and views that may be defined as candidate solution set of views is also a big issue. Defining generalized cost function representation of this optimization problem is another issue. In most of the existing approaches in materialized view selection, all the associated costs that are

to be minimized are summed up as a single cost for minimizing, ignoring the trade-offs between them. The existing models define views as some derived functions or relations on some normalized relational model based tables or relations. These models do not support semi-structured or un-structured databases with very little indexing capabilities as used in Big data framework based data warehousing.

- When an optimization problem with multiple non-dominating objectives is converted into single objective, it ignores that different solutions may offer trade-offs between the objectives. In Chapter 3, the view selection problem is defined as multi-objective optimization problem for minimizing total analytical query processing cost of data warehouse by selecting a set of views for materializing within limited available memory space with minimized maintenance cost of the materialized views. Multi-objective Differential Evolution (MODE) algorithm has been patched up for binary encoded solution representation of the problem for utilizing conventional multiple view processing plan as input. NSGA-II also has been applied with equivalent parameters in this problem and it has been observed that the solutions yielded by both NSGA-II and multi-objective DE algorithms are distributed in similar curve in the objective function space. But it has been observed that the solution quality of solutions obtained by this approach in view selection for materialization are somewhat better than that of NSGA-II with respect to convergence property and total cost function values.

- In Big data framework, frequent sub-queries or views may be materialized for speeding up MapReduce computing paradigm based query processing. Materializing frequent sub-queries and views means that the views reside in the memory of one or more nodes in the cluster of commodity hardware save MapReduce costs by reducing repetitions of submission and scheduling cost of Distributed File System jobs for query processing. In Chapter 4, materialized views are defined as resultant data of frequent sub-queries and aggregation functions of a set of Big data warehousing queries. The problem is defined as a multi-objective optimization problem for minimizing the total query processing MapReduce cost, MapReduce cost for maintaining the materialized views and the number of views selected for materializing with maximized total size of the views selected while selecting views for materializing. The patched-up Multi-Objective DE used in Chapter 3 is modified and applied here. The NSGA-II has been implemented to study comparative performances for developing a recommendation system for selecting views for materializing in Big data warehousing. It is observed that the diversity of solutions generated by MODE-BE in solution space is more than that of NSGA-II generated solutions. The diversity in solution vector space is preferred because diversity preservation on objective function values may lead to loss of some significantly distinct solutions on the basis of constituent selected views in them. In our experimentation it is observed that MODE-BE generates 37.04% more number of solutions than NSGA-II based system. More number of solutions may be useful for selecting most appropriate solutions. But by applying Mann-Whitney U test on both MODE-BE and NSGA-II generated solutions at 5% level of significance, it cannot be

rejected the null hypothesis that the solution vectors generated by both the systems are from the same population.

- Finally Chapter 5 discusses how multi-objective Simulated Annealing based techniques may be applied in selecting sub-query results or views in MapReduce based query processing framework for materializing. A comparative performance analysis of this technique and common EA based techniques in view selection problem in this paradigm is presented. The original AMOSA algorithm of total run time complexity $O((Total\ iterations) \times (N \times logN))$ has been customized for materialized view selection application with overall run time complexity of $O((Total\ iterations) \times (N + logN))$. The customized algorithm produced acceptable convergence measure γ despite measuring it with respect to large number of non-dominated solutions obtained from MODE-BE and NSGA-II applied in this problem. But overall it has been observed that the MODE-BE algorithm converges the best empirically in this application. In this version of AMOSA, termed as AMOSA-MVS, while maintaining diversity in solution space in intermediate generations, distance based measure is used for filtering solutions in stead of the Single-linkage clustering used in the original version. Solutions yielded by this technique is found to be of comparable quality with respect to other similar randomized algorithms despite its much lower computational complexity.

6.2 Future Directions

Few of the possible directions for future research in this area may be outlined as below.

- From our humble survey on approaches in view selection for materializing in data warehouse for efficient query response, we found that a technique applicable for large high dimensional realistic data warehouses, independent of its schema, as well as applicable for *Big data* framework [76] with reasonable run time and space complexity is needed to be designed. A cost effective method to input queries from large query workload based data warehouse and a generalized data structure for storing them, are also to be developed. Designing a flawless test-bed with unprejudiced (benchmark) databases to evaluate different approaches is yet to be taken up for handling this NP-hard problem. The future focus should be on developing an analytical model for big and complex view processing environment.

- Presently the view materialization problem has been defined from the perspective of homogeneous data warehousing system where the data warehousing is considered either as a conventional RDBMS based data warehouse or a Big data system. But with the advent new computing technologies like mobile computing to Big data distributed computing at heterogeneous cluster of commodity hardware, the query execution performance improvement by view materialization will involve optimization of many other parameters

with large number of different types of resource constraints. Therefore, for optimized materialization of views in this scenario, objective functions are to be defined for distributed data warehouse, spread over heterogeneous data nodes with heterogeneous data organization, resources and constraints.

- Extraction of candidate views from query workload on data warehouses is presently done by offline syntactic analysis of the query execution log files in the system. The basic assumption in most of the works on this problem is that the frequent analytical queries and the intermediate views generated on a data warehouse in a specific period of time will reoccur as frequent queries on the data warehouse in future. But, in present business scenario of complex strategic decision making process in very highly competitive business world, it is not always true. Therefore, integrated view selection and materialization system with on-line analysis of triggered queries is to be designed for dynamic and incremental updating of pool of candidate views for materializing.

- Theme based partitioning of analytical processing on data warehouse for enhanced query performance is another potential area of research. Works are to be done for analysis of queries for finding different themes of analytical processing. Data mining techniques specifically frequent item-set mining techniques may be useful for partitioning or dynamic partitioning of the candidate views for materializing on different identified themes. Finding candidate views and finally finding the solution set of views for materializing by user specified criteria or theme and configuration is another research direction.

- The existing data warehousing system in Apache Hadoop Distributed File System (HDFS) converts user's SQL (i.e, HiveQL) query statement into *abstract syntax tree* (AST) which is then converted to physical operator tree for execution by syntactic analysis. Query optimization is done on this physical operator tree. On the other hand query processing plans are generated by semantic information obtained from semantic analysis of queries. Down stream query performance optimizations like in case of selecting optimum set of views for materializing is mainly done by looking at these semantic information based query execution plans. Due to difference in semantic information based query execution plan and physical query tree, the query performance optimization may not be achieved at times. This is another major issue to be addressed in this area of research.