Final Project Report

Byung S. Lee
Department of Computer Science
University of Vermont
Burlington, Vermont 05405

Email: bslee@cems.uvm.edu

1. Project Information

Project title: Generating Cost Models of User-Defined Functions

Principal investigator: Byung S. Lee

Period of reported progress: January 1, 2002 through December 31, 2006

(including no cost extension from January 1, 2005 – December 31, 2006)

Recipient organization: University of Vermont, Burlington, Vermont 05405

DOE award number: DE-FG02-02ER45962

2. Summary of Accomplishments

1.1. Preface

The original project period was January 1, 2002 to December 31, 2004, but it was extended at no cost for another two years (i.e., until December 31, 2006). One undergraduate student and five graduate students were trained through the project at different stages. Additionally, a postdoctoral fellow was hired to play a key role in the project. The undergraduate student was David VanHorn. The graduate students were Vinod Kannoth, Li Chen, Jiangyan He, Songtao Jiang, and Dennis Fuchs. The postdoctoral fellow was Dr. Zhen He, a Ph.D. graduate from the Computer Science Department in the Australian National University.

There were four internal collaborators who participated at various stages of the project: Professor Jeff Buzas at Mathematics & Statistics Department, and Professors Robert Snapp, Sean Wang, and Alan Ling at Computer Science Department. There were two groups of external collaborators that participated in the project. Data Science Group at Lawrence Livermore National Laboratory (LLNL) was the main collaborator. Database Group at Korea Advanced Institute of Science and Technology also participated in topical researches at a later stage of the project.

1.2. Research activities

A detailed summary of the research activities and the accomplishments are described in this subsection, categorized by the technical area. For each category of activities, a brief description of the activities and the participants is given, and it is followed by the list of publications and their abstracts.

Query processing on simulation mesh data

These activities were a continuation of prior collaborative efforts with the Data Science Group at LLNL. The PI worked on technical issues of "approximate query processing of simulation mesh data" while a prototype system was built at LLNL. Professor Robert Snapp at the Department of Computer Science participated as well. Three journal papers and four conference papers have been published out of these activities.

Journals

- Byung Suk Lee and Ron Musick, "MeshSQL: the Query Language for Simulation Mesh Data," *Information Sciences*, Elsevier, Volume 158, Issues 1-4, January 2004, pp. 177-202.

  Abstract: Mesh data has been a common form of data produced and searched in scientific simulations, and has been growing rapidly in the size thanks to the increasing computing power. Today, there are visualization tools that assist scientists to explore and examine the data, but their query capabilities are limited to a small set of fixed visualization operations, which is far too short to meet the needs of most users. Thus, it is imperative to provide ad hoc query tools for them. In this paper, we propose an ad hoc query language MeshSQL, which has been extended from ANSI SQL99 to support the features unique to simulation mesh data, such as temporality, spatial regions, statistics, and similarity. After classifying MeshSQL queries based on three criteria related to efficient implementations of the queries, we present the syntax and semantics of MeshSQL, and support them with examples. We also discuss implementing MeshSQL queries in SQL99 in an object-relational database system that allows incorporating user-defined types and functions. To our knowledge, MeshSQL is the first and the only query language for simulation mesh data.

- Byung Suk Lee, Terence Critchlow, Ghaleb Abdulla, Chuck Baldwin, Roy Kamimura, Ron Musick, Robert Snapp, and Nu Ai Tang, "The Framework for Approximate Queries on Simulation Data," *Information Sciences*, Elsevier, Volume 157, Issues 1-4, December 2003, pp. 3-20.

  Abstract: AQSim is a system intended to enable scientists to query and analyze a large volume of scientific simulation data. The system uses the state of the art in approximate query processing techniques to build a novel framework for progressive data analysis. These techniques are used to define a multi-resolution index, where each node contains multiple models of the data. The benefits of these models are two-fold: 1) they have compact representations, reconstructing only the information relevant to the analysis, and 2) the variety of models capture different aspects of the data which may be of interest to the user but are not readily apparent in their raw form. To be able to deal with the data interactively, AQSim allows the scientist to make an informed tradeoff between query response accuracy and time. In this paper, we present the framework of AQSim with a focus on its architectural design. We also show the results from an initial proof-of-concept prototype developed at LLNL. The presented framework is generic enough to handle more than just simulation data.

- Byung Suk Lee, Robert R. Snapp, Ron Musick, and Terence Critchlow, "Metadata Models for Ad Hoc Queries on Terabyte-Scale Scientific Simulations," *Journal of Brazilian Computer Society*, Brazilian Computer Society, Volume 8, Number 1, July 2002, pp 5-15.

Abstract: We present our approach to enabling approximation ad hoc queries on terabyte-scale mesh data generated from large scientific simulations through the extension and integration of database, statistical, and data mining techniques. There are several significant barriers to overcome in achieving this objective. First, large-scale simulation data is already at the multi-terabyte scale and growing quickly, thus rendering traditional forms of interactive data exploration and query processing untenable. Second, a priori knowledge of user queries is not available, making it impossible to tune special-purpose solutions. Third, the data has spatial and temporal aspects, as well as arbitrarily high dimensionality, which exacerbates the task of finding compact, accurate, and easy-to-compute data models. Our approach is to preprocess the mesh data to generate highly compressed, lossy models that are used in lieu of the original data to answer users' queries. This approach leads to interesting challenges. The model (equivalently, the content-oriented metadata) being generated must be smaller than the original data by at least an order of magnitude. Second, the metadata representation must contain enough information to support a broad class of queries. Finally, the accuracy and speed of the queries must be within the tolerances required by users. In this paper we give an overview of ongoing development efforts with an emphasis on extracting metadata and using it in query processing.

Conference Proceedings

- R. Kamimura, G. Abdulla, C. Baldwin, T. Critchlow, B. Lee, I. Lozares, R. Musick, and N. Tang, "Use of Numerical Models as Data Proxies for Approximate Ad-Hoc Query Processing," *Proceedings of the 7th Joint Conference on Information Sciences (JCIS)*, Research Triangle Park, NC, U.S.A.., September 26 – 30, 2003, pp. 588-593.

  Abstract: As datasets grow beyond the gigabyte scale, there is an increasing demand to develop techniques for dealing/interacting with them. To this end, the DataFoundry team at the Lawrence Livermore National Laboratory has developed a software prototype called Approximate Adhoc Query Engine for Simulation Data (AQSim). The goal of AQSim is to provide a framework that allows scientists to interactively perform adhoc queries over terabyte scale datasets using numerical models as proxies for the original data. The advantages of this system are several. The first is that by storing only the model parameters, each dataset occupies a smaller footprint compared to the original, increasing the shelf life of such datasets before they are sent to archival storage. Second, the models are geared towards approximate querying as they are built at different resolutions, allowing the user to make the tradeoff between model accuracy and query response time. This allows the user greater opportunities for exploratory data analysis. Lastly, several different models are allowed, each focusing on a different characteristic of the data thereby enhancing the interpretability of the data compared to the original. The focus of this paper is on the modeling aspects of the AQSim framework.

- Byung Suk Lee, Robert R. Snapp, Ron Musick, and T. Critchlow, "Ad hoc Query Support for Very Large Scientific Data: the Metadata Approach," *Proceedings of the 16th Brazilian Symposium on Databases (SBBD)*, Rio de Janeiro, Brazil, October 1-3, 2001, pp.199-212. (An extended version published in Journal of Brazilian Computer Society.)

Abstract: The same as the journal version above.

- Ghaleb Abdulla, Chuck Baldwin, Terence Critchlow, Roy Kamimura, Byung Suk Lee, Ida Lozares, Ron Musick, Robert Snapp, and Nu Ai Tang, "Approximate Ad-hoc Query Engine for Simulation Data," *Proceedings of the 1st ACM+IEEE Joint Conference on Digital Libraries (JCDL)*, Roanoke, VA, USA, June, 2001, pp. 255-256.

  Abstract: In this paper, we describe AQSim, an ongoing effort to design and implement a system to manage terabytes of scientific simulation data. The goal of this project is to reduce data storage requirements and access times while permitting ad-hoc queries using statistical and mathematical models of the data. In order to facilitate data exchange between models based on different representations, we are evaluating using the ASCI common data model, which is comprised of several layers of increasing semantic complexity. To support queries over the spatial-temporal mesh structured data we are in the process of defining and implementing a grammar for MeshSQL

- Byung Suk Lee, Robert Snapp, and Ron Musick, "Toward a Query Language on Simulation Mesh Data: an Object-Oriented Approach," *Proceedings of the 7th International Conference on Database Systems for Advanced Applications (DASFAA)*, Hong Kong, April 18-20, 2001, pp. 242-249.

  Abstract: As simulation is gaining popularity as inexpensive means of experimentation in diverse fields of industry and government, the attention to the data generated by scientific simulation is also increasing. Scientific simulation generates mesh data, i.e., data configured in a grid structure, in a sequence of time steps. Its model is complex -- understanding it involves mathematical topology and geometry in addition to fields (in the relational sense). Moreover, there is no query language developed on mesh data at all. These are what we address in this paper. We develop a comprehensive model of mesh data in an object-oriented manner, propose a set of primitive algebraic operators, show their object-oriented implementation, and demonstrate that the well-known object query language ODMG OQL is powerful enough to express queries on mesh data, whether the queries are on mesh topology, geometry, fields, or combination of them. Finally, we discuss some physical implementation issues pertinent to executing queries efficiently.

User-defined function cost modeling

These activities constituted the main part of the project. There were two approaches taken one after another: regression-based static approach and self-tuning dynamic (and adaptive) approach.

Regression-based static approach: Three types of user-defined functions were selected, and their costs were modeled using regression techniques. The selected functions were (1) financial time series aggregation functions (e.g., moving window average of a group of time series), (2) text search functions (e.g., simple, Boolean, or proximity queries), and (3) spatial search functions (e.g., window, range, or k-nearest neighbor searches). These functions were those commonly supported in major commercial database management systems. A graduate student Vinod Kannoth worked on the research using financial time

series aggregation functions.   Professor Jeff Buzas at the Mathematics & Statistics Department also participated as an expert in regression techniques. An undergraduate student David VanHorn worked on the research using text search functions. Dr. Paul Thompson at Dartmouth College, as well as Professor Jeff Buzas, participated as co-advisors. A graduate student Songtao Jiang worked on the research using spatial search functions. Dr. Zhen He, the postdoctoral fellow, participated as a co-advisor. Two journal papers have been published out of these activities.

Journals:

- Vinod Kannoth, Byung Suk Lee, and Jeff Buzas, "Statistical Cost-Modeling of Financial Time Series Functions," *Journal of Computers and Applications*, ACTA Press, Volume 28, Number 3, 2006, pp. 181-188.

  Abstract: We present a statistical regression approach to building a cost model of an aggregate financial time series function. The cost model is needed by an object-relational DBMS query optimizer. This approach is much easier than the traditional analytical approach and yet achieves a highly precise model. Users only need to provide a set of variables influencing the costs. This requires only high-level understanding of how the function works. Experiments show that the cost models thus built are highly precise and that quadratic models are adequate.

- Songtao Jiang, Byung Suk Lee, and Zhen He, "The Cost Modeling of Spatial Operators Using Nonparametric Regression," *Information Sciences*, Elsevier, Volume 177, Issue 2, January 2007, pp. 607-631.

  Abstract: In an object-relational database management system, a query optimizer requires users to provide cost models of user-defined functions. The traditional approach is analytical, that is, it builds a cost model generated as a result of analyzing the query processing steps. This analytical approach is difficult, however, especially for spatial query operators because of the complexity of the processing steps. In this paper, a new approach that uses non-parametric regression is proposed. This significantly simplifies the process of building a cost model, while achieving highly accurate cost estimation. We demonstrate the simplicity and efficacy of this approach through experiments for three spatial operators -- the range query, the window query, and the k-nearest neighbor query -- commonly used in spatial databases, using both real and synthetic data sets.

Self-tuning dynamic (and adaptive) approach: In the first stage of this approach, Li Chen, an MS graduate from the Computer Science Department, joined the team as a research assistant and worked with another graduate student Vinod Kannoth to build a research prototype using the Oracle database management system. Professor Jeff Buzas also participated with his expertise in regression techniques. In this second stage, Dr. Zhen He joined the team as a postdoctoral fellow and worked to expand the scope of the work and extend the approaches. Additionally, two graduate students worked on different parts of the project, supervised by Dr. Zhen He and myself. Professor Robert Snapp at the Computer Science Department also participated as an expert in statistical data modeling. Two journal papers and one conference paper have been published out of these activities.

Journals

- Byung Suk Lee, Li Chen, Jeff Buzas, and Vinod Kannoth, "Regression-Based Self-Tuning Modeling of Smooth User-Defined Function Costs for an Object-Relational Database Management System Query Optimizer," *The Computer Journal*, Oxford University Press, Volume 47, Number 6, November 2004, pp. 673-693.

Abstract: We present a new approach to modeling the execution costs of user-defined functions (UDFs) for the query optimizer of an object-relational DBMS (ORDBMS). Our approach self-tunes a cost model incrementally based on the costs of the recent executions of a UDF. The approach is centered on a feedback loop in which the feedback information comprises individual UDF execution records. Each execution record contains the cost variable values used in the execution and the resulting CPU and disk I/O costs. This feedback information is saved in the execution log and used in a batch to update the cost model. Furthermore, our approach handles nominal cost variables by maintaining separate cost models for recently used values of the variables. We have built a framework that implements the feedback loop in a commercial ORDBMS. Then, we have performed experiments using common database UDFs with smooth cost variations and incrementally modeling the data using multiple regression. The experimental results demonstrate the adaptive accuracy that makes the cost model stabilize quickly while incurring small errors in cost estimations. Our approach has the advantages of incurring little overhead while tuning the cost model continuously throughout the UDF executions.

- Zhen He, Byung Suk Lee, and Robert Snapp, "Self-Tuning Cost Modeling of User-Defined Functions in an Object-Relational DBMS," *ACM Transactions on Database Systems*, ACM Publications, Volume 30, Issue 3, September 2005, pp. 812-853.

Abstract: Query optimizers in object-relational database management systems typically require users to provide the execution cost models of user-defined functions (UDFs). Despite this need, however, there has been little work done to provide such a model. The existing approaches are static in that they require users to train the model a-priori with pre-generated UDF execution cost data. Static approaches can not adapt to changing UDF execution patterns and thus degrade in accuracy when the UDF executions used for generating training data do not reflect the patterns of those performed during operation. This paper proposes a new approach based on the recent trend of self-tuning DBMS, by which the cost model is maintained dynamically and incrementally as UDFs are being executed online. In the context of UDF cost modeling, our approach faces a number of challenges, that is, it should work with limited memory, work with limited computation time, and adjust to the fluctuations in the execution costs (e.g., caching effect). In this paper we first provide a set of guidelines for developing techniques that meet these challenges while achieving accurate and fast cost prediction with small overheads. Then, we present two concrete techniques developed under the guidelines. One is an instance-based technique based on the conventional k-nearest neighbor (KNN) technique which uses a multi-dimensional index like the R*-tree. The other is a summary-based technique which uses the quadtree to store summary values at multiple resolutions. We have performed extensive performance evaluations comparing these two techniques against existing histogram-based techniques and the KNN technique, using both real and

synthetic UDFs/data sets. The results show our techniques provide better performance in most situations considered.

Conference proceedings

- Zhen He, Byung Suk Lee, and Robert R. Snapp, "Self-tuning UDF Cost Modeling Using the Memory Limited Quadtree," *Proceedings of the 9th International Conference on Extending Database Technology (EDBT)*, Heraklion, Crete, Greece, March 14-18, 2004, pp 513-531.

Abstract: Query optimizers in object-relational database management systems require users to provide the execution cost models of user-defined functions (UDFs). Despite this need, however, there has been little work done to provide such a model. Furthermore, none of the existing work is self-tuning and, therefore, cannot adapt to changing UDF execution patterns. This paper addresses this problem by introducing a self-tuning cost modeling approach based on the quadtree. The quadtree has the inherent desirable properties to (1) perform fast retrievals, (2) allow for fast incremental updates (without storing individual data points), and (3) store information at different resolutions. We take advantage of these properties of the quadtree and add the following in order to make the quadtree useful for UDF cost modeling: the abilities to (1) adapt to changing UDF execution patterns and (2) use limited memory. To this end, we have developed a novel technique we call the memory-limited quadtree (MLQ). In MLQ, each instance of UDF execution is mapped to a query point in a multi-dimensional space. Then, a prediction is made at the query point, and the actual value at the point is inserted as a new data point. The quadtree is then used to store summary information of the data points at different resolutions based on the distribution of the data points. This information is used to make predictions, guide the insertion of new data points, and guide the compression of the quadtree when the memory limit is reached. We have conducted extensive performance evaluations comparing MLQ with the existing (static) approach.

Database query selectivity estimation

These activities were initiated to complement the user-defined function cost modeling efforts, because selectivity modeling is needed in addition to cost modeling for a query optimizer to select an efficient query execution plan. A graduate student Dennis Fuchs worked on extending a state of the art histogram technique called "STHoles", co-advised by Dr. Zhen He and myself. Dr. He and I also worked with Professor Sean Wang at the Computer Science Department to develop a histogram-based technique combining the traditional reactive approach and the new proactive approach. One journal paper has been published to date, and another journal paper is under the second review after revision and resubmission.

Journals
- Dennis Fuchs, Zhen He, and Byung Suk Lee, "Compressed Histograms with Arbitrary Bucket Layouts for Selectivity Estimation," *Information Sciences*, Elsevier, Volume 177, Issue 3, February 2007, pp. 680-702.

Abstract: Selectivity estimation is an important step of query optimization in a database management system, and multidimensional histogram techniques have

proved promising for selectivity estimation. Recent multidimensional histogram techniques such as GenHist and STHoles use an arbitrary bucket layout. This layout has the advantage of requiring a smaller number of buckets to model tuple densities than those required by the traditional grid or recursive layouts. However, the arbitrary bucket layout brings an inherent disadvantage of requiring more memory to store the location information of each bucket. This diminishes the advantage of requiring fewer buckets and, therefore, has an adverse effect on the resulting selectivity estimation accuracy. To our knowledge, however, no existing histogram-based technique with arbitrary layout addresses this issue. In this paper, we introduce the idea of bucket location compression and then demonstrate its effectiveness for improving selectivity estimation accuracy by proposing the STHoles+ technique. STHoles+ extends STHoles by quantizing each coordinate of a bucket relative to the coordinate of the smallest enclosing bucket. This quantization increases the number of histogram buckets that can be stored in the histogram. Our quantization scheme allows STHoles+ to trade precision of histogram bucket locations for storing more buckets. Experimental results show that STHoles+ outperforms STHoles on various data distributions, query distributions, and other factors such as available memory size, quantization resolution, and dimensionality of the data space.

- Zhen He, Byung Suk Lee, X. Sean Wang, "Proactive and Reactive Multi-dimensional Histogram Maintenance for Selectivity Estimation," submitted for publication

  Abstract: Many state-of-the-art selectivity estimation methods use query feedback to maintain histogram buckets, thereby using the limited memory efficiently. However, they are 'reactive' in nature, that is, they update the histogram based on queries that have come to the system in the past for evaluation. In some applications, future occurrences of certain queries may be predicted and a 'proactive' approach can bring much needed performance gain, especially when combined with the reactive approach. For these applications, this paper provides a method that builds customized proactive histograms based on query prediction and mergers them into reactive histograms when the predicted future arrives. Thus, the method is called the Proactive and Reactive Histogram (PRHist). Two factors affect the usefulness of the proactive histograms and are dealt with during the merge process: the first is the predictability of queries and the second is the extent of data updates. PRHist adjusts itself to be more reactive or more proactive depending on these two factors. Through extensive experiments using both real and synthetic data and query sets, this paper shows that in most cases, PRHist outperforms STHoles, the state-of-the-art reactive method, even when only a small portion of the queries are predictable and a significant portion of data is updated.

Data mining / data aggregation

These activities were taken somewhat opportunistically as we found interesting and challenging technical problems while pursuing the main parts of the project. We worked with two groups of collaborators. The first group was Professors Sean Wang and Alan Ling at the Computer Science Department in the University of Vermont, and the second group was Professor Kyu-Young Whang and his graduate students at the Computer Science Department in the Korea Advanced Institute of Science and Technology. Three journal papers have been published or accepted for publication out of these activities.

Journals

- Zhen He, X. Sean Wang, Byung Suk Lee, and Alan C.H. Ling, "Mining Partial Periodic Correlations in Time Series," *Knowledge and Information Systems*, Springer (accepted for publication).

Abstract: Recently, periodic pattern mining from time series data has been studied extensively. However, an interesting type of periodic pattern, called partial periodic (PP) correlation in this paper, has not been investigated. An example of PP correlation is that power consumption is high either on Monday or Tuesday but not on both days. In general, a PP correlation is a set of offsets within a particular period such that the data at these offsets are correlated with a certain user-desired strength. In the above example, the period is a week (seven days), and each day of the week is an offset of the period. PP correlations can provide insightful knowledge about the time series and can be used for predicting future values. This paper introduces an algorithm to mine time series for PP correlations based on the principal component analysis (PCA) method. Specifically, given a period, the algorithm maps the time series data to data points in a multi-dimensional space, where the dimensions correspond to the offsets within the period. A PP correlation is then equivalent to correlation of data when projected to a subset of the dimensions. The algorithm discovers, with one sequential scan of data, all those PP correlations (called minimum PP correlations) that are not unions of some other PP correlations. Experiments using both real and synthetic data sets show that the PCA-based algorithm is highly efficient and effective in finding the minimum PP correlations.

- Joon-Ho Woo, Byung Suk Lee, Min-Jae Lee, Kyu-Young Whang, and Woong-Kee Loh, "Temporal Aggregation Using a Multi-dimensional Index," *Journal of Database Management*, Idea Group Inc., Volume 18, Issue 2, April-June 2007, pp. 62-80.

Abstract: We present a new method for computing temporal aggregation that uses a multi-dimensional index. The novelty of our method lies in mapping the start time and end time of a temporal tuple to a data point in a two-dimensional space to be stored in a two-dimensional index and in calculating the temporal aggregates through a temporal join between the data in the index and the base intervals (defined as the intervals delimited by the start times or end times of the tuples). To enhance the performance, this method calculates the aggregates by incrementally modifying the aggregates from that of the previous base interval without re-reading all tuples for the current base interval. We have compared our method with the SB-tree, which is the state-of-the-art method for temporal aggregation. The results show that our method is an order of magnitude more efficient than the SB-tree method in an environment with frequent updates while increasingly comparable in a read-only environment as the number of aggregates calculated in a query increases.

- Won-Young Kim, Byung Suk Lee, and Kyu-Young Whang, "Partial Rollback in Object-Oriented/Object-Relational Database Management Systems with Dual Buffer," *Journal of Information and Software Technology*, Elsevier, Volume 48, Issue 2, February 2006, pp. 121-132.

Abstract: Partial rollback mechanism has been widely supported by many database management systems (DBMS). It allows a transaction to be rolled back partially, that

is, only back to a certain savepoint set by the user. A partial rollback, however, makes the DBMS buffer management complicated because it requires the DBMS to restore the state of not only the database but also the buffers. There are several literatures addressing such a partial rollback in a relational DBMS (RDBMS) which has page buffer only. However, to our knowledge, there exists no literature addressing it in an object-oriented/relational DBMS (OO/ORDBMS). The RDBMS partial rollback scheme cannot be applied to OO/ORDBMSs directly. The reason is that, unlike RDBMSs, many OO/ORDBMSs use dual buffer which consists of object buffer and page buffer. In this paper we thoroughly study the partial rollback schemes for OO/ORDBMSs with dual buffer. For this, we propose four different partial rollback schemes which are based on (single) page buffer, (single) object buffer, dual buffer using a soft log, and dual buffer using shadows, respectively. The schemes proposed are practical enough to be implemented in a real OO/ORDBMS. The results of performance evaluations show that the dual buffer-based scheme using shadows achieves the best performance.