Essentials for Modern Data Analysis Systems[*]

Mehrdad Jahangiri, Cyrus Shahabi
University of Southern California
Los Angeles, CA 90089-0781
{jahangir, shahabi}@usc.edu

Abstract

Earth scientists need to perform complex statistical queries as well as mining queries such as outlier/pattern detection on very large multidimensional datasets produced by AIRS instrument. On top of that, the desired accuracy varies per application, user and/or dataset and it can well be traded-off for faster response time. Towards this end, we have designed and developed a data storage and retrieval system which deploys wavelet transform and provides fast approximate answers with progressively increasing accuracy in support of the scientific queries. We employ a standard web-service infrastructure to assist NASA scientists to interact with AIRS dataset.

1 Introduction

The Atmospheric Infrared Sounder (AIRS) instrument collects the Earths atmospheric water vapor and temperature profiles at a very high rate. This potentially massive dataset supports Earth scientists to detect the global climate change. Data mining algorithms require to perform complex statistical queries on large multidimensional datasets. This is only feasible if an efficient data analysis system is employed.

We have designed and developed a system which utilizes the wavelet decomposition of multidimensional data to achieve progressiveness by retrieving data based on the significance of both query and data values. The use of the wavelet decomposition is justified by the well-known fact that the query cost is reduced from being a function of the query size to a function of the logarithm of the data size, which is a major benefit especially for large range queries. Wavelet multi-resolution property elevates this by re-distributing the energy of data and retrieving based on the data significance for progressive query processing. Our system, as described in this paper, is effectively built on our previous work in query estimation using wavelets [2, 3, 4, 5, 6] where wavelets are used as a tool for approximate or progressive query evaluation. Using these techniques, we support any high order polynomial range-aggregate query (e.g., variance, covariance, correlation, etc).

The remainder of this paper is organized as follows. We begin our discussion, in Section 2, with presenting an overview of discrete wavelet transform and our work. In Section 3, we explain the necessary factors in building a modern data analysis system to deal with AIRS datasets. We present our experimental studies in Section 4 and conclude our discussion in Section 5.

[*] This research has been funded in part by NSF grants EEC-9529152 (IMSC ERC) and IIS-0238560 (PECASE), unrestricted cash gifts from Microsoft, an on-going collaboration under NASA's GENESIS-II REASON project and partly funded by the Center of Excellence for Research and Academic Training on Interactive Smart Oilfield Technologies (CiSoft); CiSoft is a joint University of Southern California - ChevronTexaco initiative.

2 Research Background

The Discrete Wavelet Transformation provides a multi-scale decomposition of data by creating “rough” and “smooth” views of the data at different resolutions. In the simple case of the Haar wavelets, the “smooth” view is comprised of averages, whereas the “rough” view is comprised of details, or differences. At each resolution, termed level of decomposition or scale, the averages and details are constructed by pairwise averaging and differencing of the averages of the previous level. We refer the reader to [7] for further information about Wavelets.

Wavelets are often thought of as a data approximation tool, and have been used this way for approximate range query answering. The efficacy of this approach is highly data dependent; it only works when the data have a concise wavelet approximation. Furthermore the wavelet approximation is difficult to maintain. To avoid these problems, we use wavelets to approximate incoming queries rather than the underlying data. Note that the data set is still transformed using wavelet; however, it is not approximated since we keep all the coefficients. We are able to obtain accurate, data-independent query approximations after a small number of I/Os by using the largest query wavelet coefficients first. This approach naturally leads to a progressive algorithm. The details of our techniques can be found in [4, 5, 6]. Moreover, we propose the most efficient wavelet data maintenance methods in [2] to make practical use of wavelet in large scale applications. We implemented all these techniques using web-services to build an enterprise system for scientific data analysis (see [3]).

3 Essentials

AIRS dataset is very large and multidimensional. This dataset continuously grows and needs to be updated. The queries performed by Earth scientists are complex aggregate-queries. The query results can be fast approximate and/or progressively become exact. These characteristics lead us to believe that wavelet transform will become a likely tool for future database query processing. We address seven requirements needed by an expert user in this section and we show how wavelet transform fulfills these needs.

3.1 Multi-resolution View

Due to the explosive size of data accumulated by AIRS instrument, NASA scientists define three different levels of abstractions, e.g. Level 1, Level 2, and Level 3, to reduce the size of data and consequently address the challenge of storage and retrieval of this data. However, working with higher levels of abstractions results in lower accuracy in spite of its simplicity. Using wavelets, not only we access the finest resolution of the data with no extra cost but also we are able to summarize the data at various levels of abstractions (e.g. daily, weekly, or monthly) on-the-fly.

3.2 Progressive Query

The desired accuracy varies per application, user and/or dataset and it can well be traded-off for faster response time. By using the most important query wavelet coefficients first, we provide excellent approximate results that progressively become exact. In most of our experiments, we achieve 90% accuracy when only 10% of the query is completed.

3.3 Batch of Queries

We have observed that most scientists typically submit batches of range-sum queries simultaneously rather than issuing individual, unrelated queries. For example, one may partition the data using latitude and longitude grids and ask for weekly average temperature per grid cell. Therefore, we have enhanced our system by employing a wavelet-based technique[4] that exploits I/O sharing across a query batch to evaluate a group of queries progressively and efficiently. We ensure that the structure of error across cells remains fixed as the result set progresses. The user can observe the result per grid cell as it progressively converges to the accurate answer; the user and/or the system can prune the non-desired cells to guide the query answering.

3.4 System architecture

In many real database systems, most of the computations occur at the application side increasing both the data transfer and the client-side processing, while not exploiting process sharing and result caching at the server side. We designed and developed our system based on a 3-tier architecture. The bottom tier is the data storage tier, which can optionally be implemented either as a DBMS (for easy data management) or as a custom file-system (to achieve efficiency). The middle-tier is a query processing module comprising a set of web-services developed in the Microsoft .NET framework. Finally, the top tier consists of a web-accessible Graphical User Interface (GUI) implemented in C#. We refer the reader to [1, 3] for further information.

3.5 Polynomial Queries

In addition to many predefined statistical range-aggregate queries such as count, sum, and average, Earth scientists need to run more sophisticated queries such as variance, covariance, and correlation. Moreover, their desired query can not always be predicted by a database designer; thus, we need a facility to enable the scientists to form their own new queries. We introduced a novel technique that supports any polynomial range-sum query in [5]. With this technique, we treat all dimensions, including measure dimensions, symmetrically and support range-aggregate queries where the measure can be any polynomial in the data dimensions. We implemented this technique by developing two web-services, *PushTerm* and *PushOperator*, using which we can form any polynomial query and submit its post-order form to the system (see Example 1 and 2).

Example 1: Consider the case where we need to re-implement the variance function using *PushTerm* and *PushOperator*. Variance is defined as following:

$$Var(x) = \frac{\sum x_i^2 - (\sum x_i)^2}{n}$$

The post order representation of this function is $\sum x_i^2$, $\sum x_i$, ^{2}, $-$, n and $/$. Therefore, we can submit our polynomial query with 7 calls as shown in Figure 1.

Example 2: Consider the case where we need to implement regression coefficient using the same operators. We can form this function with 4 calls as shown in Figure 1, knowing the fact that $Var()$ and $Cov()$ functions have already been formed by other users in the system.

```
// Variance
// Assume dimId(x)=1
    PushTerm(1,2);           // $\sum x_i^2$
    PushTerm(1,1);           // $\sum x_i$
    PushOperator('**');      // $^2$
    PushOperator('-');       // $-$
    PushTerm(1,0);           // $n$
    PushOperator('/');       // $/$
    SubmitQuery();           // submit
```

```
// Regression Coefficient
// Assume dimId(x)=1 & dimId(y)=2
    Cov(1,2);                // $cov(x,y)$
    Var(1);                  // $var(x)$
    PushOperator('//');      // $\sqrt{}$
    PushOperator('/');       // $/$
    SubmitQuery();           // submit
```

Figure 1. Forming Polynomial Queries

3.6 Large Multidimensional Datasets

NASA datasets are massive and multidimensional. Any precalculation or transformation of data needs to be efficient enough to be performed on this huge data. We transform very large datasets into wavelets efficiently using our SHIFT-SPLIT technique (see [2]). We also transform queries into wavelets very fast by using our lazy wavelet transform (see [5] for more details).

3.7 Archive and Synopsis

AIRS instrument gathers data continuously and this data needs to be appended to the existing transformed data. Appending is fundamentally different from updating in that it results in the increase of the domain of one or more dimensions. As a result, the wavelet decomposed dimensions also grow, new levels of transformation are introduced and therefore the transform itself changes. We would like to perform appending directly in the wavelet domain, preserving as much of the transformed data as possible and avoiding reconstruction of the original data. The SHIFT-SPLIT operations [2] helps us achieve this goal.

We can also maintain a wavelet approximation of this vast multidimensional data in the case of limited resources when new data are coming. The intention here is to construct a space and time efficient algorithm for maintaining the best K-term synopsis. We show that we can maintain a K-term synopsis effectively if certain conditions are met (see [2]).

4 Experiments

We evaluate out system with two particular datasets, namely GPS level2 data and AIRS level2 data provided by NASA/JPL. We form two four-dimensional datacubes using these datasets. GPS datacube includes latitude, longitude, altitude, and time as dimension attributes, and temperature, refractivity, and water vapor pressure as measure attributes. We obtained this data for a 9 month period and its size is 2 GB. AIRS datacube includes latitude, longitude, pressure level, and time as dimension attributes, and temperature, water vapor mass mixing ratio, and ozone volume mixing ratio as measure attributes. This data is gathered over a year and has a size of 320 GB.

In addition to the web-service implementation, we also developed a number of customized GUIs to provide a friendly application-specific interface for our NASA/JPL users (see [1]). The user can graphically specify some dimension ranges and query any of the measure attributes. In addition to the well-known statistical queries, e.g. average, covariance, variance, and correlation, the user can form any polynomial query and track the result as it gets updated.

5 Conclusion and Future work

We have employed wavelets to support exact, approximate and progressive statistical range-aggregate queries on large multidimensional datasets. We support any arbitrary polynomial query and provide progressive query answering. We have pushed most of the application side processing to our storage system to enhance the overall efficiency. However, there is a need to investigate the usefulness of these technique for very sparse datasets and to enable performing queries across different datacubes (for example, correlation of temperature between GPS and AIRS datasets).

6 Acknowledgement

We would like to thank Brian Wilson at the Jet Propulsion Laboratory for motivating a scientific application of progressive OLAP and providing the atmospheric datasets.

References

[1] ProDA: http://infolab.usc.edu/projects/proda/.

[2] M. Jahangiri, D. Sacharidis, and C. Shahabi. SHIFTSPLIT: I/O Efficient Maintenance of Wavelet-Transformed Multidimensional Data. In *Proceedings of ACM SIGMOD*, 2005.

[3] M. Jahangiri and C. Shahabi. ProDA: A Suite of WebServices for Progressive Data Analysis. In *Proceedings of ACM SIGMOD (demonstration)*, 2005.

[4] R. Schmidt and C. Shahabi. How to evaluate multiple range-sum queries progressively. In *Proceedings of ACM PODS*, pages 3–5.

[5] R. Schmidt and C. Shahabi. Propolyne: A fast wavelet-based technique for progressive evaluation of polynomial range-sum queries. In *Proceedings of EDBT*, 2002.

[6] C. Shahabi, M. Jahangiri, and D. Sacharidis. Hybrid Query and Data Ordering for Fast and Progressive Range-Aggregate Query Answering. *International Journal of Data Warehousing and Mining*, 1(2):49–69, April-June 2005.

[7] B. Vidokovic. *Statistical Modeling by Wavelets*. Wiley Inter-Sience, 1999.