Managing Multidimensional Historical Aggregate Data in Unstructured P2P Networks using Virtual Aggregate Cubes

Devika Rani. Kadari, Hari Krishna. Deevi, Dr. K. Rama Krishnaiah, G. Sudhir

1,2,3,4Dept. of CSE, NOVA College of Engineering & Tech., Jupudi, Ibrahimpatnam, Vijayawada, AP, India

Abstract
A P2P-based framework supporting the extraction of aggregates from historical multidimensional data is proposed, which provides efficient and robust query evaluation. When a data population is published, data are summarized in a synopsis, consisting of an index built on top of a set of subsynopses (storing compressed representations of distinct data portions). The index and the subsynopses are distributed across the network, and suitable replication mechanisms taking into account the query workload and network conditions are employed that provide the appropriate coverage for both the index and the subsynopses. In this paper we are introducing virtual Datacubes which holds historical aggregated data at different level of aggregations following the subsynopses.

Keywords
P2P Networks, Multidimensional Data Management, Data Compression

I. Introduction
PEER-TO-PEER (P2P) networks have become very popular in the last few years. Nowadays, they are the most widespread approach for exchanging data among large communities of users in the file sharing context. In this scenario, the success of P2P-based solutions is strictly related to the use of lossy data compression techniques (such as MPEG formats), which yield reasonable detail levels in representing large amounts of information and make data exchange feasible in practice by significantly reducing data transmission costs. However, the problem of suitably extending-data-compression-based solutions to application contexts other than file sharing has not been deeply investigated yet. Specifically, no P2P-based solution has imposed itself as an effective evolution of traditional distributed databases.

This is quite surprising, as the huge amount of resources provided by P2P networks (in terms of storage capacity, computing power, and data transmission capability) could effectively support data management. From this standpoint, one of the application contexts which are likely to benefit from the support of a P2P network is the analysis of multidimensional data. In this scenario, information is represented as points in a multidimensional space whose dimensions correspond to different perspectives over data: users explore data and retrieve aggregates by issuing range queries, i.e., queries specifying an aggregate operator and the range of the data domain from which the aggregate information should be retrieved. Specifically, we will consider the case of analytical applications dealing with historical data, which typically require huge computation and storage capabilities, due to the large amount of data which need to be accessed to evaluate queries. Although the multidimensional data model is substantially more complex than the representation paradigm adopted in the file sharing context (where data are organized according to hname; filei pairs), analytical applications dealing with historical multidimensional data and file-sharing applications share a fundamental aspect: they can rely on lossy data compression. In fact, analogously to tools for reproducing audio and/or video files, a lot of applications dealing with multidimensional data can effectively accomplish their tasks even in the case that only an approximate representation of data is available. For instance, in Decision Support Systems (DSSs) or statistical databases, users are often concerned with performing data exploration with the aim of discovering interesting trends rather than extracting fine-grained information. In this scenario, high accuracy in less relevant digits of query answers is not needed, as providing their order of magnitude suffices to locate the regions of the database containing relevant information. At the same time, fast answers to these preliminary queries allow users to focus their explorations quickly and effectively, thus saving large amounts of system resources.

Our aim is devising a P2P-based framework supporting the analysis of multidimensional historical data. Specifically, our efforts will be devoted to combining the amenities of P2P networks and data compression to provide a support for the evaluation of range queries, possibly trading off efficiency with accuracy of answers. The framework should enable members of an organization to cooperate by sharing their resources (both storage and computational) to host (compressed) data and perform aggregate queries on them, while preserving their autonomy. virtual data cubes are created for historical aggregations. A framework with these characteristics can be useful in different application contexts. For instance, consider the case of a worldwide virtual organization with users interested in geographical data, as well as the case of a real organization on an enterprise network. In both cases, even users who are not continuously interested in performing data analysis can make a part of their resources available for supporting analysis tasks needed by others, if their own capability of performing local tasks is preserved. This is analogous to the idea on which several popular applications for public resource computing are based. For instance, within the project SETI@home [39], members of a worldwide community offer their CPU, when it is idle, to analyze radio telescope readings in search of nonrandom patterns, such as spikes in power spectra.

In order to make participants really autonomous, they should be imposed no constraint on storage and computational resources to be shared, as well as on the reliability of their network connection. These requirements make traditional distributed frameworks unsuitable and suggest the adoption of a solution based on an unstructured P2P network, where peers are neither responsible of coordination tasks (such as super peers, which are called for a certain amount of resources and reliability), nor imposed to host specific pieces of data (as in DHT-based networks). We follow the strategies of previous model of compression, Indexing and replication.

A. Proposal: The Framework in a Nutshell
Our proposal is a framework supporting the sharing and the analysis of compressed historical multidimensional data over an unstructured P2P network. From the user standpoint, two tasks are supported: data publication and data querying.
B. Main Contributions and Plan of the Paper

The main contributions of this paper and its organization may be summed up as follows:

1. A compression technique for building an indexed aggregate structure over a multidimensional data population, prone to be distributed, and accessed across a P2P network.
2. A storage model which employs additional data structures to support efficient and robust query answering over compressed data in an unstructured P2P network.
3. A dynamic replication scheme capable of maintaining appropriate levels of coverage w.r.t. the evolution of the query workload and the network conditions.
4. We are using a Virtual Aggregate cubes by following above methods.

The effectiveness of the proposed approaches is assessed through a thorough experimental analysis.

II. Related Work

Two topics are strictly related to our work: the compression of multidimensional data and the management of multidimensional data in P2P networks.

A. Compression of Multidimensional Data

The problem of effectively summarizing multidimensional data into lossy synopses has been investigated mainly in the contexts of query optimization [38] and exploratory OLAP analysis [6]. Some techniques are said to be parametric, as they rely on the assumption that data are distributed according to a mathematical (either statistical or polynomial) model. In particular, wavelet-based techniques (which are the most investigated in this group) work in two steps. First, a wavelet transform is applied to the data, yielding a set of coefficients. Then, these coefficients are suitably filtered and the “most relevant” ones are kept. Applying the inverse wavelet transform to these coefficients results in an approximate reconstruction of the data. Wavelet-based techniques differ mainly in the filtering strategy. In [31], coefficients with the “largest” values are kept, while in [13-14], a probabilistic thresholding scheme was introduced, yielding unbiased estimations for both equality and range queries, and in [15-16], a deterministic thresholding scheme was defined which can minimize the maximum relative/absolute error in the estimations. Although the most advanced wavelet-based compression techniques provide excellent performances for 1D data, their behavior in larger dimensionality settings has not been deeply studied yet Nonparametric techniques “let data speak for themselves” [19], that is, they do not assume that data fit any model. Histogram-based techniques are the most studied techniques in this class. Basically, a histogram over a data population D is built by partitioning D into a number of blocks (called buckets), and then, storing, for each bucket b, information summarizing the data contained in b’s range. Monodimensional histograms were first introduced in the context of approximate query answering in [29], where histograms built over single attributes were used to support query optimization, disregarding possible correlations among different attributes. Multidimensional histograms (which instead take into account attribute correlation, as they summarize the joint distribution of attributes) were first investigated in [33], where equiheight histograms were introduced, i.e., histograms consisting of nonoverlapping buckets containing about the same number of tuples. Indeed, building the “most effective” multidimensional histogram (called V-Optimal [24]) was shown to be an NPHard problem, even in the 2D case. Therefore, feasible approaches to the problem of constructing histograms providing reasonable accuracy for query estimates are based on greedy strategies. In particular, MHIST [36] and MinSkew [1] construct the histogram by means of hierarchical partitionings, driven by different heuristics. STHoles [5], GENHIST [20], and CHIST [10] construct histograms where bucket nesting and overlapping are allowed. In particular, STHoles uses query results feedback to refine bucket definition. GENHIST locates regions with nonhomogeneous data w.r.t. contiguous ones: it aggregates data according to progressively coarser grids and creates buckets corresponding to cells of these grids whose density is notably different from adjacent cells. The interested reader can find in [23] an “abridged,” though very accurate, survey about histogram-based compression techniques.

B. Managing Multidimensional Data in P2P Networks

Research has been devoting a great deal of attention to data dissemination and querying techniques in the P2P scenario, since the P2P paradigm has proven its effectiveness in providing high availability of computational resources, providing high availability of computational resources. One of the first works dealing with the problem of supporting range queries in a peer-to-peer network is [2], where data are ordered according to Hilbert curves, and then, distributed among the peers. Following the proposed dissemination strategy, as the number of peers gets larger, each peer tends to be associated with a smaller range and the number of peers to be accessed for evaluating query answers can increase dramatically with a negative impact on the network traffic. Since the majority of the OLAP applications focus on historical data, many recent works have been specifically targeted to this kind of data. The particular case of monodimensional range queries is investigated in [21], where order-preserving hash functions are exploited to locate the cached results of past queries whose ranges are similar to those of the queries to be answered. Thus, query answers that are evaluated by contacting the peers returned by a range lookup can be approximate, since their ranges may only partially overlap the query range. The idea of exploiting cached query results to answer range queries is also at the basis of [37], where a distributed range hashing mechanism is introduced which is able to locate peers containing results of past queries whose ranges contain the ranges to be answered. This technique was proposed for multidimensional data, even if its effectiveness was tested on monodimensional data only. The robustness of this approach strictly depends on the availability of the original data, which must be accessed in the case that range lookups do not return active peers. Limitations on the storage resources of the peers may limit the possibility of caching the results of queries defined on large ranges, and since no mechanism is provided for answering a query by partitioning its range and locating the answers of the so-obtained subqueries, this may overload the peer responsible for the original data. In [25], an efficient architecture is proposed where data are indexed and stored according to a balanced binary tree All the above cited works deal with structured P2P networks. The problem of estimating aggregates in the specific case of an unstructured P2P network has been studied in some recent works. Other works are specifically targeted to the dynamic maintenance of indexes, with the objective of balancing the load (meant as amount of data maintained) among the peers [11, 40-41, 45], or ensuring the consistency of a balanced tree [8]; in particular, in [40], a very simple partitioning scheme is adopted where dimensions are always split in two halves.
1. Compressing and Indexing Data
the strategy adopted for compressing the data to be distributed across the network. approach aims at satisfying the following requirements.
1. The compressed data must be prone to be fairly distributed across the network, in order to allow parallel evaluation of queries and suitable robustness.
2. The compressed data must support the efficient extraction of (approximate) aggregate information.
3. The representation of the compressed data must enable efficient location of the data portions involved in the queries.
In order to satisfy requirement 1, a compression technique yielding a synopsis, properly organized in subsynopses, which will be independently disseminated and replicated across the network. Requirement 2 is taken into account by building the subsynopses over nonoverlapping portions of the data domain (to limit the number of portions to be accessed to retrieve the compressed data), and through a compression technique that supports the estimation of aggregates.
Finally, for requirement 3, a suitable indexing mechanism will be devised, enabling an efficient location of the subsynopses to be accessed to evaluate queries. The synopsis is built in three steps—partitioning, compression, and indexing—which will be described in the following sections.

2. Disseminating Data and Index
The distribution process is started by a peer \( p \) that is willing to publish a data population, and works as follows:
First, for each subsynopsis \( h_j \) (respectively, leaf portion \( i_{inf} \)), \( p \) invokes \( \text{search}(C_{min}) \) to find \( C_{min} \) peers which can host a copy of \( h_j \) (respectively, \( i_{inf} \) along with \( i_{inf} \)). Then, for each \( i_{inf} \) and subsynopsis \( h_j \) referenced by \( i_{inf} \), location table \( \text{table}(i_{inf}) \) is filled with the IP addresses of the peers which will host \( h_j \). Correspondingly, each \( h_j \) is augmented with a reverse pointer to one of the peers which will host \( i_{inf} \). A similar process is performed to find \( C_{min} \) peers which will host \( Isup \) along with a location table, and to fill the table as well as the reverse pointers of leaf portions. In particular, as explained before, the location table of each peer that will host a copy of \( Isup \) is filled with the addresses of the other peers which will host \( i_{inf} \). After all of the location tables have been filled, the copies of \( s \)-blocks along with their associated location tables are sent to the appropriate peers.

It is worth noting that distributing the copies of the \( s \)-blocks randomly across the network well suits the search of data in our unstructured scenario, where search will be performed by randomly navigating across the network. At the same time, the information provided by the location tables allows, once an \( s \)-blocks related to a data population \( D \) is located, to quickly locate all the other \( s \)-blocks that are needed to answer queries over \( D \).
B. Distributed Querying

Peers participating to the network can issue two types of queries: explorative queries and range queries. In the following, we describe their semantics and evaluation process.

1. Explorative Queries

Explorative queries locate the data populations available over the network that match the interest of a user. Specifically, an explorative query Qexp issued by a peer p interested in a data population D specifies either the identifier of D or the name, or a list of keywords associated with D. The answer of Qexp is a set of peers (identified by their IP addresses) hosting the Isup portion of compressed data populations matching the criteria specified in Qexp. In current prototype, an explorative query Qexp issued by a peer p is processed by starting a random walker from p, which is propagated through the network until either the number of accessed peers satisfying Qexp reaches the maximum number of results specified by p, or the overall number of accessed peers reaches a given threshold. However, different query propagation models (such as flooding or expanding ring) could be used to support explorative queries. When the random walker reaches a peer p0 hosting an s-block of a population D matching Qexp, p0 sends the following information to p.

2. Range Queries

Range query asks for the evaluation of an aggregate operator on a data population D. It is sent from a peer p to a peer q which hosts D:Isup (q is randomly chosen among the peers associated with D in the shortcut table), and consists of a 4-tuple Q=(id(Q), id(D.Isup), rq, nrep) where Fig. 4. shows the sequence of messages exchanged by peers involved in evaluating a range query Q.

C. Dynamic Replication

Our dynamic replication scheme aims at both providing the appropriate coverage of s-blocks and balancing the load at the peers. To this aim, besides guaranteeing a minimum coverage for each s-block (so that published data remain accessible over time), our replication scheme provides adaptivity to the dynamic query workload by creating new replicas of an s-block each time it is queried and by removing less queried data through suitable aging policies.

1. Guaranteeing the Minimum Coverage: In our framework, location tables encode links among s-blocks spread over the network. Thus, they are kept updated w.r.t. events causing data unavailability by deleting the addresses of the peers that no longer host these data. Our approach is independent of the way the unavailability of data is identified; in practice, this can be done through periodic pinging (as in our prototype) or notification protocols.

2. Query-Based Replication: This strategy may be seen as a direct adaptation of the path replication strategy to the case of hierarchically organized data, like our three-level structure consisting of the inner portion of the index, its leaf portions, and the subsynopses.

3. Storage Space Management: We now detail our data replacement strategy for managing the local storage space in a peer in response to a request to host new data. This strategy allows the framework to take into account the recent interest in data portions exhibited by users when choosing the s-blocks to be removed in order to free storage space. Our approach reflects the storage space management.

IV. Experimental Analysis

We performed several experiments to assess the effectiveness of our approach. Specifically, we studied the accuracy of query estimates and the performance of our replica management strategies in terms of generated network traffic, data reachability, and query performances. In the experiments, we used synthetic data similar to those in [10]. Each population was generated by creating an empty d-dimensional array D of size nd, then by populating r regions of D, each containing a portion of the total sum T. The size of the regions along each dimension was randomly chosen between 10 and 40, and the regions were uniformly distributed in the multidimensional array. T was divided across the r regions according to a random distribution. To populate each region, we first generated a Zipf distribution whose parameter is randomly chosen between 0.5 and 2. Then, we associated these values with the cells in such a way that the closer a cell to the center of the region, the larger its value. Outside dense regions, some isolated nonzero values were randomly assigned to the array cells. As explained in [10], data sets generated this way well represent many classes of real-life populations.

The proposed method enhances the performances of the previous system by comparing Overall comparison table over Query estimation, explorative queries and range queries and replication traffic performance.
V. Conclusions

We proposed a framework for sharing and performing analytical queries on historical multidimensional data in unstructured peer-to-peer networks. In our approach, participants make their resources (and possibly their data in a suitable compressed format) available for the other peers in exchange for the possibility of accessing and posing range queries against the data published by others. Our solution is based on suitable data summarization and indexing techniques, and on mechanisms for data distribution and replication that properly take into account the need of preserving the autonomy of peers as well as the interest exhibited by the users in the data to support an efficient query evaluation. The experimental results showed the effectiveness of our approach in providing fast and accurate query answers, and ensuring the robustness that is mandatory in peer-to-peer settings.

Future work will be devoted to considering data updates. On the one hand, updates along the temporal dimension can be managed relatively easily: new data could, in fact, be considered as a new data set to be partitioned, compressed, indexed, and distributed independently from the old data; queries over a time range that involves synopses referring to different time intervals could just be split into subqueries to be processed separately. On the other hand, removing the assumption of consolidated/historical data makes the problem much more complex, as updates can affect the homogeneity of data, making the results of both the partitioning and the compression steps obsolete. The crucial objective is, therefore, that of avoiding the computation of the partitioning and the construction of the subsynopses from scratch by possibly detecting the regions of data whose features are not significantly affected by the update. This would limit the computational load for computing the up-to-date synopsis, as well as the network traffic for replacing old data. The absence of a centralized coordination in our setting poses further challenges, as it makes it necessary to devise a nontrivial mechanism for distinguishing among old and new data during query evaluation.

References


Devika Rani Kadari completed B.Tech in CSE from R.V.R. & J.C. College of Engineering in the year 2007. She trained at reputed company over managerial and service oriented skills as a knowledge seeker. She attended workshop in JNTU Kakinada to develop into professional instructor. She is pursuing M.Tech in NOVA College of Engineering & Technology for Women. Her research interest comprises in “Managing Multidimensional Historical Aggregate Data in Unstructured P2P Networks Using Virtual Aggregate Cubes”.

Mr. D. Hari Krishna is a qualified person in M.Tech and experienced in presumed engineering colleges which contributed his service and skills to guide the students and their activities. He published in various National journals and workshops with his incredible work to gain the knowledge for feature errands.

Dr. K. Rama Krishnaiah is a highly qualified person holding M.Tech and Ph.D degree in CSE, an efficient and eminent academician. He is an outstanding administrator, a prolific researcher, who has published 21 research papers in various international journals and a forward looking educationist. He worked in prestigious K.L. University for 13 years and he contributed his service for NBA accreditation in May 2004, Aug 2007 with ‘record rating’, ISO 9001:2000 in 2004, Autonomous status in 2006, NAAC accreditation of UGC in 2008 and University status in 2009.