Skyline Query with Grid-Based and Angular Based Partition Method

Mitali Chauhan[1] Hitesh Patel[2]
[1]M.E. Student [2]Assistant Professor
[1,2]Department of Information Technology Engineering
[1,2]Kitrc-Kalol, Gandhinagar-382721

Abstract— During the last decades, management of Data and storage has become increasingly distributed. So requirement of the advanced query operators, such as skyline queries, are necessary in order to help users to handle the huge amount of available data by identifying a set of interesting data objects. Fast skyline selection of high-quality web services is of critically importance to upgrade e-commerce and various cloud applications. In this paper we are presenting the 5v's characteristics of the Big Data and also technology which is used to handle Big Data. Here we also discuss the concept of skyline which filters out the set of interesting points from the large set of data points from database. The skyline of a set of multi-dimensional points (tuples) consists of those points for which no clearly better point exists in the given set, using component-wise comparison on domains of interest. Angular partitioning divides the data space using angles, motivated by the observation that skyline tuples are located near the origin.
Key words: Big Data, Skyline query, Angular partition, Grid based partition

I. INTRODUCTION

Big Data is the collection of complex and large data sets, which are difficult to capture, process, store, search and the analysis of data/ information using conventional database management tools and traditional database management system [14].The main difficulty in handling such large amount of data is because that the volume is increasing rapidly in comparison to the computing resources [2]. Data (big data) is generated by machine, generated by humans and also generated by Mother Nature [5].The Big data term which is being used now a days is kind of misnomer as it points out only the size of the data not putting too much of attention to its other existing properties [2].Big data can neither be worked upon by using traditional SQL like queries nor can the relational database management system (RDBMS) be used for storage [5]. Hadoop an open source distributed data processing system is one of the prominent and well known solutions [5].
Big Data can be described by following characteristics

A. Volume:

It is the data at rest. Usually, Terabytes and Exabyte's of existing data to process for accurate analysis [11]. The social networking sites existing are themselves producing data in order of terabytes everyday and this amount of data is definitely difficult to be handled using the existing traditional systems [2].

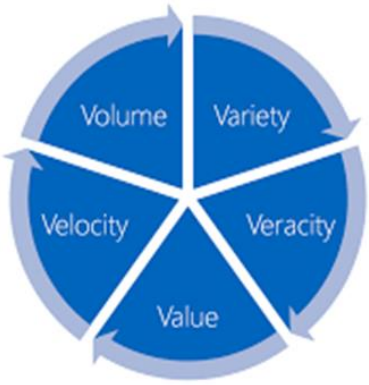

Fig. 1: 5v's of big data

B. Velocity:

It is the data in motion usually; streaming data has a few milliseconds to a few seconds for a response [14]. This characteristic is not being limited to the speed of incoming data but also speed at which the data flows [2]. Volume influences the velocity as the increase in data volume can reduce the rate at which the data is captured and transmitted [3].

C. Variety:

Variety corresponds to the different forms of data like web Pages, Web Log Files, social media sites, e-mail, documents, and sensor devices data both from active passive devices [2], comprising unstructured, semi-structured, structured, texts, logs, etc., with unstructured data forming the major portion of big data [3].

D. Veracity:

It is the data in doubt. Data is uncertain due to inconsistency, incompleteness, ambiguities, latency, deception and model approximations [11].

E. Value:

The value of data is determined by the various factors i.e.; quality of information, statistical information, survey data, data from reliable and well reputed sources [3]. It is all good and well having access to big data but unless we can turn it into value. It becomes very costly to implement IT infrastructure systems to store big data, and businesses are going to require a return on investment [5].

II. SKYLINE QUERY

Before the entry of skyline into database management there is a problem called maximum vector problem or Pareto optimum[1,10].skyline queries have been actively studied to support multi-criteria decision analysis[4] and also for decision making applications; skyline queries help users make intelligent decisions over complex data, where different and often conflicting criteria are considered[9].

For example, consider a database that contains information about hotels [1]. Each tuples of the database is represented as a point in a data space consists of numerous dimensions [1]. Assume a user is looking for a hotel that is cheap as possible and as close to the beach. To illustrate the idea of dominance relationships, Fig.1 gives hotel finding example in this example user is looking for a hotel based on two criteria, minimum price and minimum distance to the user standing location [1]. Fig.2 lists 9 hotel records and their values and Fig.1b depict the representation of the hotel in a 2D space [1]. Hotels p4, p7, p8 and p9 are all dominated by other points so skyline which return points that are not dominated by any other points. Consider the point, p7 which is dominated by p5 as it is more expensive than p5 but both have the same distance value [1].

The skyline query retrieves all hotels for which no other hotel exists that is cheaper and closer to beach. So the skyline result set which consist of {p2, p3, p1, p5, p6} [1].In database systems, queries specialized to search for the non-dominated data points are called skyline queries and their corresponding result set is known skyline set and Individual data points in a skyline result set are known as skyline Points[1].

Hotel	Price	Distance
P1	3	3
P2	1	6
P3	2	4
P4	3	7
P5	5	2
P6	7	1
P7	6	2
P8	4	4
P9	6	6

Table 1: Skyline query Dataset

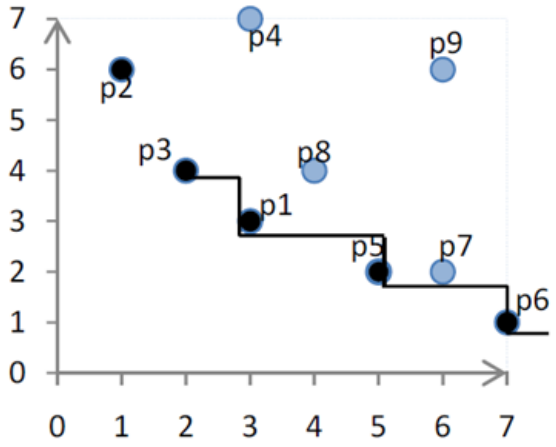

Fig. 2: An Example of Skyline Query

The database system at your travel agents' is unable to decide which hotel is best for you, but it can at least present you all interesting hotels and all hotels that are not worse than any other hotel in both dimensions so we can say this is set of interesting hotels the Skyline [16].

The same considerations also hold for a variety of applications (e.g., electronic marketing places or real-estate databases for houses), where the user is interested in mobiles, cars, houses, or other products likewise, a user who is interested in buying a car wants to find a good trade-off between minimum mileage, minimum age, and minimum price[9].With the growing number of applications that include uncertainty, e.g., sensor readings, human reading errors, and data imperfection, it became essential to support skyline queries over uncertain data[13]. As nowadays data is increasingly stored and processed in a distributed way, skyline processing over distributed data has attracted much attention recently [9].

A. Skyline Query:

Given a set of points, the skyline query proceeds a set of points (referred to as the skyline points), such that any point is not dominated by any other point in the dataset [15].

B. Skyline Query Processing:

Given a set of points, the skyline query is examined for the interesting point in the datasets is referred as Skyline Query Processing [12].

C. Range Based Skyline Query:

Given a set of points, the skyline query based on the range is examined for the interesting point in the datasets is referred as Range Based Skyline Query Processing [15]

Skyline query processing in distributed environments poses inherent challenges and requires non-traditional techniques due to the distribution of content and the lack of global knowledge. There are various different distributed systems with different requirements and unique characteristics that have to be exploited for efficient skyline processing [9].

A good distributed skyline algorithm should achieve the following goals:

1) Minimization of bandwidth consumption. We measure bandwidth in the total number of points transmitted over the network [12]. Precisely speaking, some extra bandwidth is needed for sending other synchronizing messages and the packet headers in order to enforce the underlying network protocol [12].
2) Progressiveness. That is, ideally, the algorithm should quickly output some early results soon after the beginning and produce a majority of the remaining results well before the end of execution [9].
3) Adaptability to user preferences. The algorithm should allow the flexibility of returning skyline points in different orders [12].

III. GRID BASED PARTITION

A grid-based approach for distributed skyline processing (AGiDS), assumes that each peer maintains a grid-based data summary structure for describing its data distribution. AGiDS assumes that all peers share common cell boundaries for the grid structure that leads to non-overlapping cells which increases the probability of domination between cells and enables efficient merging of local skyline set[6]. The set of cells of a peer that contain at least one data point and that are not dominated by other cells is called region-skyline set of the peer. Only these cells of the grid contain data that belong to the local skyline set. At query time, the query initiator first contacts all peers and gathers the region-skyline sets of all peers[6]. Then, the query initiator merges the collected cells into a new region-skyline set by discarding dominated cells. Finally, queries are forwarded only to peers

that correspond to at least one cell in the region-skyline set[6]. The query initiator requests only a subset of local skyline points, namely those that belong to the cells of the region-skyline set[6]. After having gathered all relevant points, the querying peer computes the global skyline set by testing only the necessary regions for dominance[6].

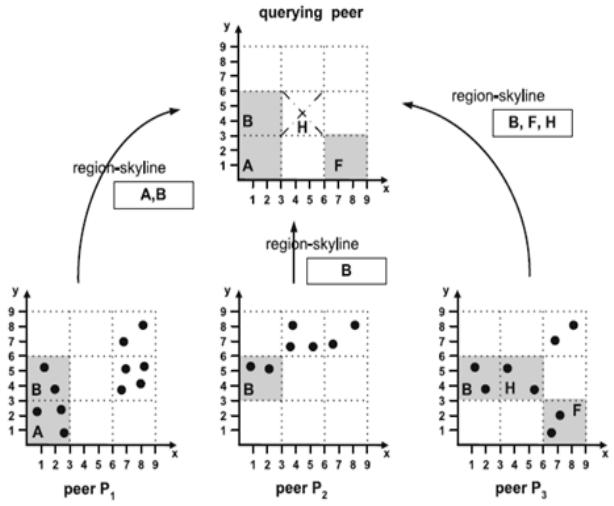

Fig. 3: Grid Partition

IV. ANGULAR PARTITION

Angular partitioning divides the data space using angles, motivated by the observation that skyline tuples are located near the origin [8]. Angular partitioning is based on making partitions by dividing the data space up using angles [7]. So by dividing the data space up using angles, skyline tuples should be distributed into several partitions while non-skyline tuples should be grouped with skyline tuples that dominates them [7].

Angle-based space partitioning scheme is proposed to use the hyper spherical coordinates of the data points to alleviate most of the problems found in traditional random and grid partitioning techniques [17]. It first maps the entire dataset from the Cartesian coordinate space into a hyper spherical space, in which the data space is partitioned based on the angular coordinates [17].

Apparently, this angle-based partitioning reduces many redundant computations and balances the workload, because each subdivided data block involves both high-quality and low-quality data points in the data space[11].For instance, each partitioned block (an angular sector) involves some global skyline services: fs1, s2g,fs3g,fs4,s5g, and fs6, s7g[11]. The angular partitioning process contains two steps: (1) Mapping the Cartesian coordinate space into a hyper spherical space and (2) Dividing the data space into N sectors according to the angular coordinates[11].

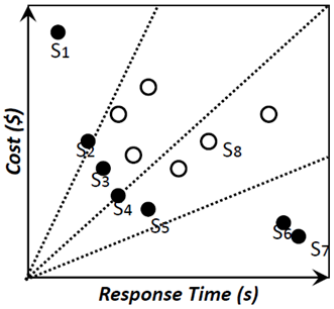

Fig. 4: Angular Partitions

Consider the dataset is partitioned over four servers, in Figure 5 using the angle-based partitioning technique, while in Figure 6 using grid partitioning. The black points are the global skyline points returned to the user[18]. For the angle-partitioning, each partition retrieves only few local skyline points and the majority of them are also skyline points of the entire dataset, while for the grid partitioning some of the partitions (such as the upper right) do not contribute to the skyline result set at all[18]. In this example, the angle-based partitioning scheme gathers 6 local skyline points and 5 of them are the global skyline points, while the grid approach results 11 local skyline points[18]. Thus, the total number of local skyline points, that need to be processed by the coordinator node in order to compute the 5 global skyline points, is much higher in the case of the grid partitioning. Therefore, the post-processing cost is significantly lower for the angle-based partitioning [18]. Another important feature of partitioning scheme is that the average pruning power of a local data point is much higher, compared to the case of grid partitioning [18].

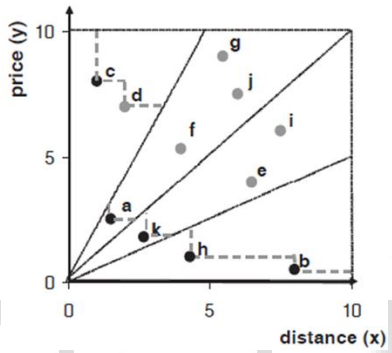

Fig. 5: Angle based Partition

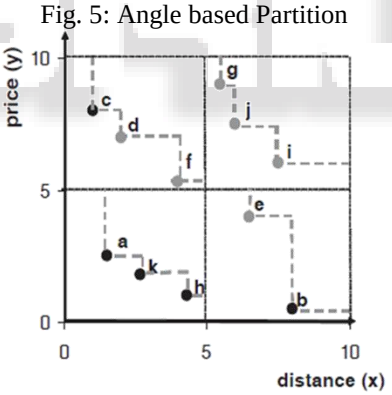

Fig. 6: Grid based Partition

V. CONCLUSION

Skyline queries retrieve the non-dominated points from a large database system based on the user preference so it can be used in preference based applications. Angular partitioning method to apply the MapReduce model for fast Skyline selection of optimal web services.It successfully eliminates all the dominated points by using Angular partition and another is the grid partition efficient technique. Capitalizing on hyper spherical coordinates, partitioning scheme alleviates most of the problems of traditional grid partitioning techniques, thus managing to reduce the response time and share the computational workload more fairly.

REFERENCES

[1] Angel C Bency, S Deepa Kanmani," A SURVEY OF SKYLINE PROCESSING IN VARIOUS ENVIRONMENT", Indian Journal of Computer Science and Engineering (IJCSE), ISSN : 0976-5166 Vol. 5 No.1 Feb-Mar 2014
[2] Avita Katal Mohammad Wazid R H Goudar," Big Data: Issues, Challenges, Tools and Good Practices", IEEE, 2013
[3] Cameron Seay, Rajeev Agrawal, Anirudh Kadadi, Yannick Barel," Using Hadoop on the Mainframe:A Big Solution for the Challenges of Big Data", 12th International Conference on Information Technology - New Generations, IEEE, 2015
[4] Hyountaek Yong, Jongwuk Lee, Jinha Kim, Seung-won Hwang *,"Skyline ranking for uncertain databases *", Elsevier Inc, 2014
[5] Ishwarappa , Anuradha J"A brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology", Published by Elsevier, 2015
[6] Kamalpreet Singh, Ravinder Kaur," Hadoop: Addressing Challenges of Big Data", IEEE, 2014
[7] Kasper Mullesgaard Jens Laurits Pedersen "Efficient Skyline Computation for Large Volume Data in MapReduce Utilising Multiple Reducers "
[8] Kasper Mullesgaardy, Jens Laurits Pederseny, Hua Luy, Yongluan Zhou "Efficient Skyline Computation in MapReduce" March 24-28,2014
[9] Katja Hose, Akrivi Vlachou," Distributed Skyline Processing: a Trend in Database Research Still Going Strong",ACM,March 26–30, 2012,
[10] Katja Hose • Akrivi Vlachou," A survey of skyline processing in highly distributed environments", Springer-Verlag , The VLDB Journal (2012) 21:359–384
[11] Liang Chen, Kai Hwang, Jian Wu "MapReduce Skyline Query Processing with A New Angular Partitioning Approach" IEEE,2012
[12] Lin Zhu, Yufei Tao, and Shuigeng Zhou," Distributed Skyline Retrieval with Low Bandwidth Consumption", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 21, NO. 3, MARCH 2009
[13] Mohamed E. Khalefa, Mohamed F. Mokbel, Justin J. Levandoski," Skyline Query Processing for Uncertain Data *", ACM, October 26–30,2010
[14] Pradeep Adluru, Srikari Sindhoori Datla, Xiaowen Zhang*, "Hadoop Eco System for Big Data Security and Privacy", IEEE,2015
[15] R.Prema Steffi 1, S.Sundaramoorthy 2," Survey on Skyline Queries with Its Algorithms and Operators", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Vol. 2 Issue 11, November – 2013
[16] Stephan Borzsonyil Donald Kossmann2 Konrad Stocker'," The Skyline Operator*",IEEE, 2001
[17] Xiaofang Zhou Henning Köhler1 Jing Yang2 "Efficient Parallel Skyline Processing using Hyperplane Projections " ACM, 2011
[18] Yannis Kotidis Akrivi Vlachou, Christos Doulkeridis, "Angle-based Space Partitioning for Efficient Parallel Skyline Computation" ACM June 9–12, 2008

