



University of HUDDERSFIELD

University of Huddersfield Repository

Samson, Grace, Lu, Joan and Xu, Qiang

Large spatial datasets: Present Challenges, future opportunities

Original Citation

Samson, Grace, Lu, Joan and Xu, Qiang (2016) Large spatial datasets: Present Challenges, future opportunities. In: International Conference on Change, Innovation, Informatics and Disruptive Technology ICCIIDT' 16, London- U.K, October 11,12 2016, 11th to 12th October 2016, London, UK.

This version is available at <http://eprints.hud.ac.uk/30955/>

The University Repository is a digital collection of the research output of the University, available on Open Access. Copyright and Moral Rights for the items on this site are retained by the individual author and/or other copyright owners. Users may access full items free of charge; copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational or not-for-profit purposes without prior permission or charge, provided:

- The authors, title and full bibliographic details is credited in any copy;
- A hyperlink and/or URL is included for the original metadata page; and
- The content is not changed in any way.

For more information, including our policy and submission procedure, please contact the Repository Team at: E.mailbox@hud.ac.uk.

<http://eprints.hud.ac.uk/>

Large spatial datasets: Present Challenges, future opportunities

Grace L. Samson,^a Joan Lu,^b Qiang Xu^c

^a Department of Computer Science, School of Computing and Engineering,
University of Huddersfield; UK

zenhev2@gmail.com

^{b,c} Department of Computer Science, School of Computing and Engineering,
University of Huddersfield; UK

j.lu@hud.ac.uk, Q.Xu2@hud.ac.uk

Abstract. The key advantages of a well-designed multidimensional database is its ability to allow as many users as possible across an organisation to simultaneously gain access and view of the same data. Large spatial datasets evolve from scientific activities (from recent days) that tends to generate large databases which always come in a scale nearing terabyte of data size and in most cases are multidimensional. In this paper, we look at the issues pertaining to large spatial datasets; its feature (for example views), architecture, access methods and most importantly design technologies. We also looked at some ways of possibly improving the performance of some of the existing algorithms for managing large spatial datasets. The study reveals that the major challenges militating against effective management of large spatial datasets is storage utilization and computational complexity (both of which are characterised by the size of spatial big data which now tends to exceeds the capacity of commonly used spatial computing systems owing to their volume, variety and velocity). These problems fortunately can be combated by employing functional programming method or parallelization techniques.

Keywords: Spatial Database, Large Datasets, Hadoop, Cloud, Map-Reduce, Bulk-Loading,

1 INTRODUCTION:

Attention to location, spatial interaction, spatial structure and spatial processes lies at the heart of activities in several disciplines today and as such demands the urgent development of tools capable of analysing and managing such data which typically can only be represented by means of geometric features [1]. According to [2] the large volumes of spatial data available on a continuous or periodic basis in many fields, offers a potential for researchers to discover useful information and knowledge that has not been seen before. Spatial or geometric data structures organises objects based on space or geometry. Scientific activities these days (including surveys and exploration of Satellite and aerial images of the Earth) generate large spatial databases which always come in a scale nearing terabyte of data size. In most cases these databases are multidimensional (i.e. possessing varying attributes or dimension) and yet must be studied, visualised, interpreted and mined so as to extract qualitative, meaningful and useful information and new relationship. Unfortunately, because the huge sizes of the datasets hardly fits into main memory, constant improvement for disk based algorithm is frequently required in order to manage the versatility and the highly non-uniformed distribution that comes with the large datasets. According to [3], since most science data (including coordinates, time etc.) are multidimensional and continuous in nature, then the models or the equations built on them employ several of the variables where quantities (entities) are points in the multidimensional space and theories and model are hyper surfaces that establishes relationships between the variables. Multidimensional databases (*see figure 1*) enables

flexible, high performance access and analysis of large volumes of complex and interrelated data. The term multidimensional according to [4] is used to describe the characteristics of databases with refer to how large datasets are characterised and viewed. It could also mean the idea of using a data cube according to [5] to represent the dimensions of data available to a user; for example an object (taken as measure attribute) of the data cube could be viewed in different dimension (taken as feature attribute). One of the major challenges of large spatial datasets (for example the spatial data from databases like telephone calls, Census demographics, card payment, environmental records etc.) is the ability to accurately and optimally analyse the relationship within these enormous amount of datasets (which are associated with certain geographic locations) in a the shortest period of time possible [6]. In this study, we have looked into the various types of challenges faced by large spatial datasets and we have also suggested ideas (based on several reviews) on how handle or tackle this challenges.

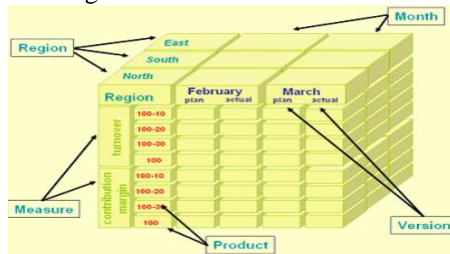


Figure 1: Example of a multidimensional dataset [4]

2 THEORETICAL FRAMEWORK/ RELATED WORKS:

2.0 Large spatial dataset description:

Large spatial data sets can be seen as a result of accumulating samples or readings of phenomena in the real world while moving along two dimensions in space [6]. For a typical example of systems that could be referred to as a large spatial datasets, how they are generated, their various application areas and all the existing literature, see [7]. Spatial dataset can be described according to [8], as a data that has some connection with coordinates (in a 2-dimensional, 3-dimensional or even a higher dimensional space) as a property. Some examples include components on printed circuit boards, roads and houses on maps etc. [7] describes a large spatial dataset using four major paradigm including language, indexing, query processing, and visualization; according to the authors, an efficient system for managing large spatial datasets is expected to support these four components. Spatial datasets are large in quantity (because many spatial databases of real-world interest are very large, with sizes ranging from tens of thousands to millions objects) and are always complex in structures and relationships [8], these database can be seen as a collection of data objects over a particular multi-dimensional space. [2] stated that a spatial dataset is said to be large if it contains many attributes for each object, this according to him poses the problem of finding similarity among object that frequently co-locate with each other in such large databases. The concept of “bigness” or “largeness” in spatial data comes as a result of the fact that the contents of spatial dataset are richer in depth and breadth with varied types, complex structures and increasing data volumes that comes as a result of growing spatial data features, this size problem poses the challenge of inefficient interpretations and increased time complexity [9]. The notion of big spatial data is an evidence of the fact that the conventional tools, techniques and hardware existing about a decade ago have met with the limitations in handling the size and volume of the data that comes from geometric information systems and devices [10]. The idea of Big Spatial Data describes a set of data whose size (*volume*), complexity (*variability*), and rate of growth (*velocity*) according to [11], make them complex

to be collected, managed, processed or analysed by current technologies and tools. In the study of the analysis of large datasets, two (2) terms has been carefully defined by [35].

a. Large: defined for the size of a dataset as the combination of two separate characteristics; *the absolute number of data elements within a single data frame*, and *the number of data frames that make up the dataset*

b. Multidimensional: the term that describes a data element as depending on two separate characteristics; *the number of different attributes or dimensions embedded in the element*, and *the number of unique values each attribute can represent (e.g., a binary, multivalued, or continuous attribute)*

2.1 Related Works:

Though a great number of works has been done relating to the topic of this article ([12]; [13]; [14]; [15]; [16]), they have majorly focused on the non-spatially related big data databases. Notwithstanding, a number of research has focused on the various challenges and opportunity facing spatial data as a “big data” element, and suggestions has also been put forward as to how these menace can be eradicated. In [17], A synthesises of the problems of big spatial data, major issues and challenges with current developments as well as recommendations for what needs to be developed further in the near future was emphasized. [18] explored several case studies with which they demonstrated the importance and benefits of the analytics of geospatial big data, including fuel and time saving, revenue increase, urban planning, and health care, they also introduced a new emerging platforms for sharing collected geospatial big data and for tracking human mobility via mobile devices. Some other authors who worked on this a similar topic include ([19]; [20]; [21]; [22]; [23]; [24]; [25]; [26]). As a little deviation, [25] provides an overview to spatial cyber infrastructure and also presents some potential future directions of spatial cyber infrastructure.

2.2 Architecture:

The present challenges facing the management of large spatial datasets /database as identified by ([7]; [9]; [10]; [27]) has necessitated inventing new software tools and techniques as well as parallel computing hardware architectures to meet the requirement of timely and efficient handling of the big data. With the increase in the volume of readily available spatial data from several location based applications, there has been a surge in the storage and the processing capabilities needs, this has led to the emergence the concept of parallel computing and several architectures have been proposed and frameworks have also been developed to take advantage of recent developments of hardware. Among these architectures and frameworks (of which distributed data model wherein it will be possible to access many storing and processing units is used) include: Network Computing, Hadoop Framework for distributed computing, Cloud computing platforms, CUDA Computing using the array of processors in GPUs manufactured by NVIDIA, and recently developed OpenMP programming model based on Intel Xeon Phi co-processors [10]. The Hadoop framework distributed computing has proven very useful in areas such as Biomedical Imagery analysis, Spatio-temporal data analysis, Geospatial data analysis and even many other applications involving large volumes of data. A description in [27] offers a new Spatial Database Design which is a Hadoop based framework for handling efficiently complex spatial queries with high performance, this framework employs the R-tree index structure for spatial databases applies the Hadoop framework to further improve the tree. Actually, there are two main part that constitutes the distributed paradigm (architecture) for managing large volume datasets, these include; a distributed file systems (of connected nodes that forms a cluster, for data storage – example Hadoop) and a programming model to process the data, which two main functions include mapping and reducing – example MapReduce. The mapper in this case receives an input key/value pairs from an input reader,

processes the pairs and outputs a transitional key/value pairs to the reducer which through the partitioner collects the results of the sub-problems grouped based on their transitional key and combines them using the programmer definition to form an output solution for the original problem. Several versions of this distributed framework for large volume spatial database management has been proposed including: [28]; [29]; [30]; [31]; [32]; [33]).

2.3 Design Technologies and Access Methods:

The database structure for relational databases differ enormously from that of spatial databases. According to [9], spatial database structures are more complex than the table in normal classical databases and the presence of both raster images and vector graphic makes them extremely computationally complicated because in the most cases, the spatial object's attributes are not explicitly stored in the database. Further ways to highlight the characteristics of large spatial datasets include heterogeneity, uncertainty, and sensitivity of time-space; these characteristics put together presents a system whereby the data for a single spatial object stored in various distributed location, built upon different structural framework and standard yet must be viewed. Spatial data bases are built up of spatial data types, which in general can come in form of Rasters or Vectors.

A. Representation of spatial data

Through the two (Raster and vector) basic models in *figure 2*, spatial data can be saved on a computer system for processing. While the raster model represents data (discrete and continuous entities) as a regular grid, where the value in each grid cell corresponds to the characteristics of a spatial property at that specific location, the vector model assumes data types like points, lines, polygons and regions as a means of representing varying levels of detail of a real world object. Using the vector, one can endlessly enlarge an image without any pixelation effect; in additions the vector data model is capable of effectively representing two dimensional spatial features such as geological, medical and other scientific data [34]). In general large spatial datasets are multidimensional data sets of information through which the real world features can represented. This can be achieved using the raster or vector spatial models that can represent natural or fabricated entities or even phenomena in a manner that computer systems can interpret ([34]; [28]).

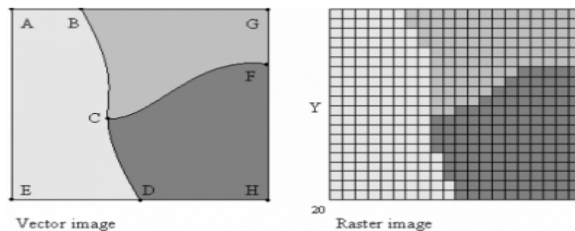


Figure 2: spatial data representation

B. Multidimensional database management

Generally speaking, multidimensional data elements are elements with two or more dimensions [35]. We can see a spatial database as a dataset that describes the spatial/temporal properties of a real world phenomena. Spatial databases are majorly implemented in a geographical information system environment or in a spatial engine accessed through an application programming interface (API), and sitting on top of a DBMS or in a universal

(object-relational) server with spatial extension or even in a web server with spatial viewer etc., all of which can be designed according to [3], using either a flat file, hierarchical, network, relational, object-oriented, multidimensional or hybrid structures. Furthermore, they can be organised in very diverse architectures such as stand-alone GIS, client-server solutions, intranets or spatial data warehouses. Gone are the days of a spatial database implemented solely on a stand-alone GIS [36]. The data in the spatial data warehouse (containing geographic data - satellite images, aerial photographs and also non-spatial data –example census dataset, highway traffic data etc.) are often modelled as a multidimensional space to facilitate Online Analytical Processing (*OLAP*) in the query engines, where queries typically combine data across many dimensions in order to detect trends and anomalies [37].

Relational table:

Classical relational database management systems (*see figure 3*) store data as numbers, alphabets, alphanumeric or even symbols, they can be used to link location to an attribute table describing what is found there. Nevertheless, Relational database technology is inadequate for managing large spatial data [38], and it is quite tedious to build a spatial database directly using a classical database model without an extension to handle spatial data types (or data models), spatial functions, and spatial indexes. This extension could either be done by extending existing relational models with object-oriented features or by adding a special row and table based data types into object-oriented databases [39]. Nevertheless according to [40], extended RDBMs like PostgreSQL’s POGIS and similar methods like the *multidimensional OLAP cubes* are not generally good enough for higher dimensional data analysis.

C. Multidimensional data Structures:

Multidimensional arrays are the most effective and efficient ways of presenting large multidimensional dataset than the relational table but unfortunately, the performance advantages of storing multidimensional data in a multidimensional array (in certain cases) only increases as the size of the dataset increases but the relative performance disadvantages of storing non-multidimensional data in a multidimensional array increase as the size of the dataset increases.

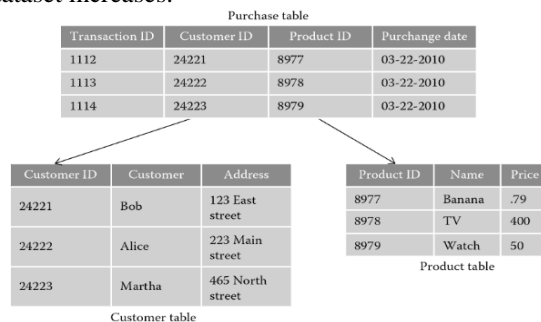


Figure 3: example of relational table

Using the multidimensional array, the organizational structure of the database makes arithmetic operations that are of interest to users naturally easier to generate, also this makes the system more efficient to perform. In general, a properly implemented multidimensional database always defines a hierarchy that manages the intricate relationship that exist among large multidimensional dataset. Managing multidimensional spatial data deals with a total consideration of the *view, rotation, range, query, architecture and accessibility of the*

underlying spatial database .The information content in the multidimensional spatial, temporal or spectral domains of a data object represents real world features. Multidimensional database allow for the efficient and convenient storage and retrieval of large volumes of highly related data which can be stored, viewed and analysed from different viewpoint known as dimensions. In a multidimensional database, data can be viewed from different angles [41], this gives a broader perspective of a problem unlike other models.

C. Major Design issues

As we have mentioned earlier the four (4) major aspects to be considered in the analysis of large spatial datasets include: language, indexing, query processing, and visualization.

2.3.1. Language

The language aspect deals with complexities of spatial database management. According to [7], the underlying language for building a spatial database (which should contain basic spatial support including standard data types and functions), frequently employed for designing a spatial database provides nontechnical users access to system functionality without worrying about its implementation details. Fortunately, majority of recent systems for big non-spatial data analysis are designed with these portable simple high level language. For examples Hadoop uses Pig Latin [42], Hive uses HiveQL [43], etc.

2.2.2. Indexing

Spatial indexes according to [44] play a crucial role in spatial databases for the efficient execution of queries involving spatial constraints. A spatial database contains a huge collection of objects that are located in multidimensional space, as such, using an (appropriate data structure (spatial index) as a technology for management of spatial database; one is able to improve data retrieval efficiency by constructing the relationship between spatial position and spatial object [45]. Unlike traditional systems where data are not spatially organized [7], the spatial attributes of a spatial datasets must be taken into consideration in order to decide where to store each record so as to avoid sub-performance for spatial applications. Spatial databases support operators such as *intersect and contain* which are more computational complex compared to the *conventional operators join or selection* as such, efficient processing of queries that manipulating these spatial relationships relies upon auxiliary indexing structures [8]. Large spatial data are complex in structures and relationships therefore, in order to achieve fast query processing a typical spatial database needs an index. More so, because construction time and performance gain in query processing are very important in a large spatial databases, experiments has shown that efficient implementation of indexes in multidimensional data spaces is important [46]. Several numbers of index structures have been proposed to speed up spatial query processing; example of these structures include; *Grid File, and Quad Tree, R-tree, X-tree etc*. However migrating these indexes to other systems for big data is challenging given the different architectures used in each one [7], therefore generalised methods must be employed to support these indexes in other environments. Indexing in a spatial database (SD) deals with multi-dimensional objects which are associated with spatial coordinates, where search queries are based on the spatial properties of objects and not on the attribute values. A spatial (two-dimensional) index is typically meant to be optimised for spatial characteristics (size and x- and y-coordinates) of the object [47]. In [48] a description of the extension of the Generalized Index search tree [GIST] framework for efficient OLAP queries on a Spatial Data warehouse was given. The GIST provides 2 (predicate and gist) interfaces to extend.

2.2.3. Visualization

The visualization approach for large spatial datasets depends on their spatial dimension or extent, therefore in most cases, spatial attributes are directly and easily mapped to the two physical screen dimensions, whereby the resulting visualization describes the underlying phenomena (dimension and extent) and objects [6]. Some of the problems of visualizing large spatial dataset includes: plotting (which may present the problem of inappropriate conclusion through misleading visualizations if wrongly plotted). In general, there mainly three (3) spatial phenomena to visualize namely: point, lines and regions (areas).

Points: There are several examples of spatial datasets that can easily be viewed as point data e.g. oil wells, census demographics etc.

Lines: Examples are large telecommunication networks, internet, and boundaries between countries. **Regions: Examples** are lakes, and political units such as states or counties. Critically speaking according to [37], the major difference between conventional and spatial data warehouses lies in the visualization of the results, whereas conventional data warehouse **OLAP** results are often shown as summary tables or spread sheets of text and numbers, the result of spatial data warehouses may be *albums of maps*, moreso, it is not easy to convert the alpha-numeric output of a data cube on spatial data warehouses into an organized collection of maps. Multidimensional data visualization according to [35] involves representation of multidimensional data elements in a low dimensional environment, such as a computer screen or printed media, existing traditional visualization techniques are not well suited to solving this kind of problem.

2.2.4. Query Processing in Large Spatial Dataset:

An efficient and fast query response is one of the fundamental requirements of large spatial systems. Though SQL has become the best standard for data querying and can be used to express most spatial queries, it is often very difficult to write efficient SQL based queries due to structural differences in relational and spatial data programming models [49]. The fundamental functioning component to support spatial applications is spatial query processing according to [50], however, advanced techniques of spatial query processing are facing significant challenges as the data expand and user accesses increase. The need to constantly analyse large volumes of spatial data stored in a spatial data warehouse demands querying the warehouse/data store using spatial online analytical processing (SOLAP) systems. The diagram in **figure 4** is an example spatial related query that could be operated on a spatial data warehouse. [48] proposed an efficient search algorithm which uses the predicate “Consistent” to find all the leaf nodes (actual data objects as referenced by an index) which are consistent with query predicate. In addition, a new state was also introduced for the “consistent” predicate called “Partial true” which work with a new search algorithm for efficient results during an online analytical processing (OLAP) query involving a spatial data warehouse.

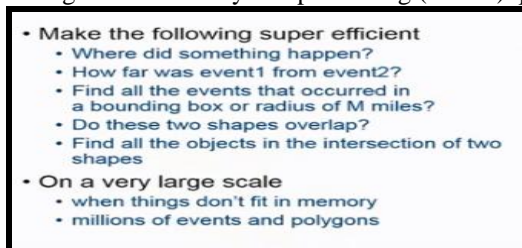
- 
- Make the following super efficient
 - Where did something happen?
 - How far was event1 from event2?
 - Find all the events that occurred in a bounding box or radius of M miles?
 - Do these two shapes overlap?
 - Find all the objects in the intersection of two shapes
 - On a very large scale
 - when things don't fit in memory
 - millions of events and polygons

Figure 4: Example of a spatial queries on a large dataset

Spatial query predicates are complex and typical spatial queries are based not only on the value of alphanumeric attributes but also on the spatial location, extent and measurements of spatial objects in a reference system. As such, spatial query processing over big spatial data requires intensive disk I/O accesses and spatial computation [50]. In order to approach the concern of efficient query processing, researchers nowadays rather adopt expressive query languages (in order to avoid worries over query translation, optimization and execution) instead of implementing programming APIs [49].

3 CHALLENGES OF LARGE (MULTIDIMENSIONAL) SPATIAL DATASETS MANAGEMENT

Large spatial datasets according to [27] are capable of accumulating highly detailed information, including multi-variable data sets, resulting in enormous storage and processing demands. The major challenge faced by large spatial dataset is the problem of storage utilization and computational complexity. Unfortunately, recent literature ([7]; [9]; [10]; [51]; [27]; [11]; [52]) has revealed that the urgent need to manage and analyse big spatial data is hindered by the lack of dedicated systems, techniques, and algorithms to support such data. Fighting the challenges associated with large spatial systems means managing a flood of structured, semi-structured and unstructured data in incongruent databases, both on-premises and in the cloud. [9] also added that the main challenge of processing large amounts of spatial data is compounded by the continuous expansion of spatial data in terms of *depth, scope and scale* all of which has rendered existing systems for processing spatially referenced application inadequate. [10] has pointed out that huge volumes of data acquired in different formats, having large complexity and non-stop generation have posed an insurmountable challenge in scientific and business world alike, as such, the conventional tools, techniques and hardware existing about a decade ago have met with the limitations in handling such data. [51] identified a number of the challenges facing large spatial datasets management in the concept of big data management;

- *Time complexity in making useful data available for analysis*
- *The discovery of spatial and temporal correlations between different spatial data points (objects)*
- *Time complexity for data loading in order to make data available for use*
- *Creating appropriate spatial indexes to aid efficiency of processing*
- *Leveraging the code from spatial database applications developed over the years*
- *Ability to develop predictive analytics for various applications*

The challenge of Big Spatial Data majorly emanates from the *size (volume), complexity (variability)*, and *rate of growth (velocity)* according to [11], which makes them complex to be collected, managed, processed or analysed by current technologies and tools. In essence, [52] reported that combating the challenge facing Big Spatial Data analysis require ingenious steps for data analysis, characterized by its three main components: *variety, velocity and volume*". [49] gave a brief summary of the major challenges facing large spatial datasets in recent times including: *Large volumes of multi-dimensional data, high computational complexity and complex spatial queries*

4 OPPORTUNITIES for LARGE SPATIAL DATASETS MANAGEMENT

According to [53] Big Data are datasets that are too large for traditional data-processing systems, that requires new technologies, like Hadoop, Hbase, MapReduce, MongoDB or Couch-DB. Despite all the formidable challenges militating against the success of spatial database management, it is interesting to know that evolution of the Hadoop [54] one of the

implementations of MapReduce [55] has provided us with the opportunity to efficiently process large scale data sets by exploiting parallelization. Latest technology trends for Big Data Technology include – Hadoop, MapReduce, Hadoop File System (HDFS), Apache SPARK (spatial spark). While some recent research ([29];[24]; [56]; [57]) has indicated that the importance of parallel and distributed programming for handling big data sets in the general context or even in the geospatial context is highly significant, some others researchers have suggested that functional programming concepts or languages such as Haskell, ML, Closure, Scheme, Lisp etc. ([58]; [59]; [60]; [17]), and other frameworks should be more appropriate. Functional programming (FP) according to [17] proves more efficient for handling geospatial big data streams as the concept of data race (which relates to the notion of concurrency) is eliminated and handled by strictly controlling the simultaneous access to mutable data. More so, [58], added that **FP** is a major breakthrough in the analysis of big spatial data due to its support for parallel computing and concurrency and its high performance.

Cloud Computing:

Cloud computing is a necessity for big spatial data management and the efficiency of spatial indexing for huge datasets at cloud computing environment cannot be over emphasized [45]. The goal of implementing the cloud based platform according to [61], is to solve the issues faced by traditional geospatial information platform, such as data-intensive, computing-intensive, and concurrent-intensive problems, this would in turn enhance the implementation of big geo-data analytics and management, provide geospatial information services for multi-departments of government, and facilitate information sharing. Cloud computing according to [62], is the use of resources that are delivered as a service over a network and due to the flexibility and scalability in cloud computing, now cloud computing plays an important role to handle a large-scale data analysis.

Spark technology

The **spark technology** [63] designed to exploit large main memory capacities, is built on the notion of Resilient Distributed Dataset and implemented using Scala, it utilizes built-in data parallel functions for vectors/collections (such as map, sort and reduce), which not only makes the programs more concise but also makes them parallelization friendly. [64] Proposed the SpatialSpark which supports indexed spatial joins based on point-in-polygon test and point-to-polyline distance computation and has been designed for large-scale spatial join query processing in Cloud.

Bulk loading

Another way forward for managing large spatial dataset is by the use of bulk loading methods. Since most spatial applications are based on write once read many access model according to [65], the *large amounts of spatial data could be quickly imported into storage systems* for rapid deployment of spatial information services. However, bulk-loading of spatial data is time-consuming and cannot satisfy the desire of the applications dealing with massive spatial data as such, the *parallel technique of bulk loading* is proposed by [66], it is designed to accelerate the processing of spatial data bulk loading for building tree-based in parallel. Bulk-loading spatial data using the popular *MapReduce framework* is intended overcome the problems associated with parallel bulk-loading of tree-based indexes which has

the disadvantage that the quality of produced spatial index decrease considerably as the parallelism increases [67].

5 DISCUSSION and CONCLUSION:

5.1 Discussion

Some benefits of improving multidimensional data structures:

Spatial data are used in almost all research field that needs to illustrate multidimensional data. And recent studies has shown that processing large spatial data requires the efficacy and knowledge of parallel computing. Parallel computing is more beneficial as against its sequential counterpart that is why parallelization promises a better solution to a variety of tough computational problems

Some benefits that comes with constantly improving the performance of multidimensional data structures include:

- a) *Increased speed of informational retrieval*
- b) *Improved accuracy and precision*
- c) *Ease of data manipulation and maintenance*
- d) *Improved efficiency*

5.2 Conclusion

This research has focused on the nature and characteristics of big/large spatial entities/objects with respect to their spatial attribute and non-spatial attribute. The work also described the current modelling, querying and implementing techniques for large spatial datasets using advanced and state of the art technologies. We outlined the current challenges facing the management of large spatial dataset and also presented latest technologies and future prospect for handling or analysing large spatial datasets. In addition, we have reviewed a selection of spatial data theory and methods as an extension of the methods for handling big classical data. Big spatial data can be seen as a set of structured or unstructured datasets with enormous data volumes that cannot be easily captured, stored, manipulated, analyse and managed. Based on these, we have suggested some possible ways of improving the management of large spatial datasets.

- o Design new spatial indexing and algorithms to handle the complex nature of real-time analytics for geographical data.
- o Design methods that explore spontaneous and explanatory relationships.
- o Build efficient methods for data visualization
- o Develop approaches for error propagation

Reference:

- [1] Samson, G. L., Lu, J., & Showole, A. A. (2014). Mining Complex Spatial Patterns: Issues and Techniques. *Journal of Information & Knowledge*
- [2] Al-Naymat, G. (2013, May). *GCG: Mining maximal complete graph patterns from large spatial data*. In Computer Systems and Applications (AICCSA), 2013 ACS International Conference on (pp. 1-8). IEEE.
- [3] Csabai, I., Trencseni, M., Herczegh, G., Dobos, L., Józsa, P., Purger, N., ... & Szalay, A. (2012). *Spatial indexing of large multidimensional databases*. arXiv preprint arXiv:1209. 6490.
- [4] Coveney, M. (2003) *Cooperate Performance Management: Selecting the Right Technologies*. <ONLINE> [Available at <http://www.businessforum.com/Comshare04B.html>]

- [5] Rouse M. (2016) *multidimensional database (MDB)* <online> available at <http://searchoracle.techtarget.com/definition/multidimensional-database>
- [6] Keim, D. A., Panse, C., & Sips, M. (2003, September). "Visual data mining of large spatial data sets". In *International Workshop on Databases in Networked Information Systems* (pp. 201-215). Springer Berlin Heidelberg.
- [7] Eldawy, A., & Mokbel, M. F. (2015, April). The era of big spatial data. In *Data Engineering Workshops (ICDEW), 2015 31st IEEE International Conference on* (pp. 42-49). IEEE.
- [8] Owen Owen B., Sacks-Davis, R., & Han, J. (1993). Indexing in spatial databases. *Unpublished/Technical Papers*.
- [9] Li, Deren, Li Deyi, and Shuliang Wang (2015) *Spatial Data Mining: Theory and Application*. Springer-Verlag Berlin Heidelberg
- [10] Alkathiri, M., Abdul, J. and Potdar, M. B. (2016). Geo-spatial Big Data Mining Techniques. *International Journal of Computer Applications* 135(11):28-36.
- [11] Bhosale, H. S., & Gadekar, D. P. (2014). A REVIEW PAPER ON BIG DATA AND HADOOP. *International Journal of Scientific and Research Publications*, 756.
- [12] Dinov, I. D. (2016). Methodological challenges and analytic opportunities for modeling and interpreting Big Healthcare Data. *Gigascience*, 5(1), 1.
- [13] Li, B. (2013). Survey of Recent Research Progress and Issues in Big Data. *December10*.
- [14] Khan, N., Yaqoob, I., Hashem, I. A. T., Inayat, Z., Mahmoud Ali, W. K., Alam, M., ... & Gani, A. (2014). Big data: survey, technologies, opportunities, and challenges. *The Scientific World Journal*, 2014.
- [15] Huang, T., Lan, L., Fang, X., An, P., Min, J., & Wang, F. (2015). Promises and challenges of big data computing in health sciences. *Big Data Research*, 2(1), 2-11.
- [16] Raghupathi, W., & Raghupathi, V. (2014). Big data analytics in healthcare: promise and potential. *Health Information Science and Systems*, 2(1), 1.
- [17] Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., ... & Cheng, T. (2016). Geospatial big data handling theory and methods: A review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115, 119-133
- [18] Lee, J. G., & Kang, M. (2015). Geospatial big data: challenges and opportunities. *Big Data Research*, 2(2), 74-81.
- [19] Global Geospatial Information Management (2016). *Big Data challenge & Opportunity Alain Kabamba, Hexagon Geospatial*. <ONLINE> [Available at <https://www.google.co.uk/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=ggim/>]
- [20] Eldawy, A., & Mokbel, M. F. (2015, June). The Era of Big Spatial Data: Challenges and Opportunities. In *2015 16th IEEE International Conference on Mobile Data Management* (Vol. 2, pp. 7-10). IEEE.
- [21] Conklin, J. (2015) *Big Spatial Data – Your day has come*. <ONLINE> [Available at <http://www.ccri.com/2015/07/19/big-spatial-data-your-day-has-come/>]
- [22] Kitchin, R. (2013). Big data and human geography Opportunities, challenges and risks. *Dialogues in human geography*, 3(3), 262-267.
- [23] Graham, M., & Shelton, T. (2013). Geography and the future of big data, big data and the future of geography. *Dialogues in Human Geography*, 3(3), 255-261.
- [24] Shekhar, S., Gunturi, V., Evans, M. R., & Yang, K. (2012). Spatial Big-Data Challenges Intersecting Mobility and Cloud Computing. *Proceedings of the 11th ACM International Workshop on Data Engineering for Wireless and Mobile Access - MobiDE '12*. doi: 10.1145/2258056.2258058

- [25] Wright, D. J., & Wang, S. (2011). The emergence of spatial cyberinfrastructure. *Proceedings of the National Academy of Sciences*, 108 (14), 5488-5491.
- [26] VanWey, L. K., Rindfuss, R. R., Gutmann, M. P., Entwisle, B., & Balk, D. L. (2005). Confidentiality and spatially explicit data: Concerns and challenges. *Proceedings of the National Academy of Sciences*, 102(43), 15337-15342.
- [27] Economides, G., Piskas, G. and Siozos-Drosos, S. (2013). *Spatial Data and Hadoop Utilization*.
- [28] Cary, A., Sun, Z., Hristidis, V., & Rish, N. (2009, June). Experiences on processing spatial data with mapreduce. In *International Conference on Scientific and Statistical Database Management* (pp. 302-319). Springer Berlin Heidelberg.
- [29] Lee, K., Ganti, R. K., Srivatsa, M., & Liu, L. (2014). Efficient Spatial Query Processing for Big Data. *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - SIGSPATIAL '14*, 469-472. doi: 10.1145/2666310.2666481
- [30] Moens S., Aksehirli E., Goethals B., (2013). Frequent Itemset Mining for Big Data, *IEEE Int. Conf. on Big Data, IEEE*, 2013, 111-118.
- [31] Liao, J., Zhao, Y., & Long, S. (2014). MRPrePost—A parallel algorithm adapted for mining big data. *Paper presented at the Electronics, Computer and Applications, 2014 IEEE Workshop*.
- [32] Owen, S., Anil, R., Dunning, T., & Friedman, E. (2011). *Mahout in action: Manning Shelter Island*.
- [33] Eldawy, A. (2014). Spatial Hadoop: towards flexible and scalable spatial processing using mapreduce. *Paper presented at the Proceedings of the 2014 SIGMOD PhD symposium*.
- [34] Sagar, A. and Bellur, U. (2011). Distributed Computation on Spatial Data on Hadoop Cluster, *Department of Computer Science and Engineering Indian Institute of Technology, Bombay Mumbai-400076*.
- [35] Healey, C. G. (1996). Effective visualization of large multidimensional datasets (Doctoral dissertation, University of British Columbia).
- [36] Bédard, Y. (1999). Principles of spatial database analysis and design. *Geographical Information Systems*, 1, 413-424.
- [37] Shekhar, S., Lu, C., Zhang, P., & Liu, R. (2002). Data mining for selective visualization of large spatial datasets. *Paper presented at the 41-48*. doi:10.1109/TAI.2002.1180786
- [38] Mamoulis, N. (2012) *Spatial data management (1st ed.)*. US: Morgan & Claypool Publishers
- [39] Stonebraker, M., Brown, P., Zhang, D. and Becla, J. (2013). “SciDB: A Database Management System for Applications with Complex Analytics,” *Computing in Science and Engineering*, vol. 15, no. 3, pp. 54–62.
- [40] Szalay, A., Gray, J., Fekete, G., Kunszt, P., Kukol, P. & Thakar, A. (2005). “Indexing the Sphere with the Hierarchical Triangular Mesh”, *Microsoft Research Technical Report (123)*
- [41] Williams, C., Garza, V.R., Tucker, S., & Marcus, A.M. (1994, January 24). Multidimensional models boost viewing options. *InfoWorld*, 16 (4)
- [42] Olston, C., Reed, B., Srivastava, U., Kumar, R. & Tomkins, A. (2008) “Pig Latin: A Not-so-foreign Language for Data Processing,” in *SIGMOD*, pp. 1099–1110.
- [43] Thusoo, A., Sen, J. S., Jain, N., Shao, Z., Chakka, P., Anthony, S., Liu, H., Wyckoff, P. & Murthy, R. (2009) “Hive: A Warehousing Solution over a Map-Reduce Framework,” *PVLDB*, pp. 1626–1629.

- [44] Roumelis, G. (2016) *an alternative algorithm for bulk-loading xbr+-trees*: <Online available at> <http://enh2016.uth.gr/session/an-alternative-algorithm-for-bulk-loading-xbr-trees/>.
- [45] El-Sayed, L. S., Abdul-Kader, H. M., & El-Sayed, S. M. (2015). Performance Analysis of Spatial Indexing in the Cloud. *International Journal of Computer Applications*, 118(4).
- [46] Lee, T., & Lee, S. (2003, June). OMT: Overlap Minimizing Top-down Bulk Loading Algorithm for R-tree. In *CAiSE Short Paper Proceedings* (Vol. 74).
- [47] Theodoridis, Y., Vazirgiannis, M., & Sellis, T. (1996, June). Spatio-temporal indexing for large multimedia applications. In *Multimedia Computing and Systems, 1996., Proceedings of the Third IEEE International Conference on* (pp. 441-448). IEEE.
- [48] Rao, F., Zhang, L., Yu, X. L., Li, Y., & Chen, Y. (2003, November). Spatial hierarchy and OLAP-favored search in spatial data warehouse. In *Proceedings of the 6th ACM international workshop on Data warehousing and OLAP* (pp. 48-55).
- [49] Ajiand, A. & Wang, F. (2016). *Big Data: Storage, Sharing, and Security*. Eds: Fei Hu. Taylor & Francis LLC, CRC Press
- [50] Zhong, Y., Han, J., Zhang, T., Li, Z., Fang, J., & Chen, G. (2012, May). Towards parallel spatial query processing for big spatial data. In *Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2012 IEEE 26th International* (pp. 2085-2094). IEEE.
- [51] Ravada, S. (2014). *Trends and Research Opportunities in Spatial Big Data Analytics and Cloud Computing: NCSU GeoSpatial Forum*. [Online: <https://cnr.ncsu.edu/geospatial/wp-content/uploads/sites/6/2016/04/Spatial-Cloud-Ravada.pdf>]
- [52] Sagiroglu, S., & Sinanc, D. (2013). Big data: A review. Paper presented at the *Collaboration Technologies and Systems (CTS), 2013 International Conference on*.
- [53] Provost, F., & Fawcett, T. (2013). *Data Science for Business: What you need to know about data mining and data-analytic thinking*. " O'Reilly Media, Inc."
- [54] Apache Hadoop (2016). <Online> [Available at <http://hadoop.apache.org/>: Accessed 23/08/2016]
- [55] Lammel R. (2008) Google's MapReduce programming model – Revisited, *Science of Computer Programming*. (70) pp. 1-30.
- [56] Shekhar, S., Evans, M. R., Gunturi, V., Yang, K., & Cugler, D. C. (2014). *Benchmarking Spatial Big Data*. In T. Rabl, M. Poess, C. Baru & H.-A. Jacobsen (Eds.), *Specifying Big Data Benchmarks* (pp. 81-93): Springer Berlin Heidelberg.
- [57] Wang, S., Ding, G., & Zhong, M. (2013). On Spatial Data Mining Under Big Data. *Journal of China Academy of Electronics and Information Technology*, 8(1), 8-17.
- [58] Mintchev, S. (2014). *User-Defined Rules Made Simple with Functional Programming*. In W. Abramowicz & A. Kokkinaki (Eds.), *Business Information Systems* (pp. 229-240): Springer International Publishing.
- [59] Maitrey, S., & Jha, C. K. (2015). Handling Big Data Efficiently by Using Map Reduce Technique. *Paper presented at the The 2015 IEEE International Conference on Computational Intelligence & Communication Technology (CICIT)*, Ghaziabad, IN.
- [60] Mohammed, E. A., Far, B. H., & Naugler, C. (2014). Applications of the MapReduce Programming Framework to Clinical Big Data Analysis: Current Landscape and Future Trends. *BioData Mining*, 7(1), 22. doi: 10.1186/1756-0381-7-22
- [61] Song, W. W., Jin, B. X., Li, S. H., Wei, X. Y., Li, D., & Hu, F. (2015). Building Spatiotemporal Cloud Platform for Supporting GIS Application. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(4), 55.

- [62] Wang Y., Wang S., & Zhou D., (2009) Retrieving and Indexing Spatial Data in the Cloud Computing Environment, *in the First International Conference on Cloud Computing, Springer-Verlag.*
- [63] Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). Spark: cluster computing with working sets. *HotCloud*, 10, 10-10.
- [64] You, S., Zhang, J., & Gruenwald, L. (2015, April). Large-scale spatial join query processing in cloud. *In Data Engineering Workshops (ICDEW), 2015 31st IEEE International Conference on* (pp. 34-41). IEEE.
- [65] Liu, X., Han, J., Zhong, Y., Han, C., & He, X. (2009, August). Implementing WebGIS on Hadoop: A case study of improving small file I/O performance on HDFS. *In 2009 IEEE International Conference on Cluster Computing and Workshops* (pp. 1-8). IEEE.
- [66] Qina. Z., Ershuna, Z. and Yaohuanc, H. (2008) "Research on Parallel Bulk-Loading R-Trees Based on Partition Technology of Database" *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.* (37). Part B4.
- [67] Liu, Y., Jing, N., Chen, L. et al. Wuhan Univ. J. Nat. Sci. (2011) "Parallel bulk-loading of spatial data with MapReduce: An R-tree case". *Wuhan University Journal of natural science.*16: 513. doi:10.1007/s11859-011-0790-3