EXTENDIBLE MULTI-DIMENSIONAL SPARSE ARRAY REPRESENTATIONS

Catherine Ann Honegger

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering.

Johannesburg, July 2017

The financial assistance of the National Research Foundation (NRF) towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the author and are not necessarily to be attributed to the NRF.

Declaration

I declare that this dissertation is my own, unaided work, except where otherwise acknowledged. It is being submitted for the degree of Master of Science in Engineering to the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination to any other University.

Signed this ____ day of _____ 20__

_____

Catherine A. Honegger

Abstract

Acknowledgements

I would like to thank my research supervisor, Professor Ekow Otoo, for taking me on as a masters student. My special thanks also go to Professor Ken Nixon for his help and support throughout the project.

I am grateful to the University of the Witwatersrand for awarding me a Postgraduate Merit Award as financial assistance during my studies.

I am also grateful to the National Research Foundation (NRF) South Africa for their financial support for my study.

Contents

Declaration **i**

Abstract **ii**

Acknowledgements **iii**

Contents **iv**

1 Introduction **1**

1.1 Problem Motivation 1

1.2 Problem Statement 2

1.3 Significance 2

1.4 Application 3

1.5 Early Results from Literature 3

1.6 Contribution to Field 3

1.7 Challenges 4

1.7.1 "this is not research, this is what any competent engineer knows/can do/can find out." 4

1.7.2 "this is not an important topic to be working on" 4

1.8 Organisation of the Proposal 5

2 Background **6**

2.1 Data Warehousing 6

2.2 Early Results 6

2.3 Result Differentiation 6

2.4 Methodologies 6

2.5 Sparse Array Representations 7

2.6 Dynamic Sparse Arrays 7

2.7 Big Data Warehouse 8

2.8 Big Data Representations 8

2.9 TileDB storage manager 9

2.10 FGPAs 10

2.11 Impact 10

2.12 New Results 10

3 Methodology **11**

3.1 Extendible Multi-Dimensional Sparse Arrays 11

3.1.1 Multi-Dimensional Sparse Array Representation 11

3.1.2 Extendible Multi-Dimensional Array Representation 12

3.1.3 Extendible Multi-Dimensional Sparse Array Representation . 12

4 Experimental Setup **14**

4.1 Computational Environment 14

4.1.1 Experimental Environment 14

4.1.2 Computational Model 14

4.1.3 Software 14

4.2 Experimental Data 14

4.3 Software Engineering Practice 14

5 Preliminary Results **15**

6 Future Work **16**

7 Validation of Results **17**

8 Risk Management **18**

9 Schedule and Time-Line for Completion **19**

10 Summary **20**

Chapter 1

Introduction

Over recent years society has been able to store and collect more data, leading to the emergence of huge databases with quintillion bytes of data being produced everyday. By combining different data sets, useful information can be retrieved from the stored data. Information resources are incredibly important and valuable in business management and decision-making [1, 2]. Due to their significance, these data assets must be stored properly and easy to access [1]. Various applications and technologies for managing big-data and high volume data-streams in data intensive computing is becoming essential to further big-data, data-mining and machine learning in several scientific domains. Thus the field of Big Data research has materialised from the need to store such large quantities of information. Data analysis in all societal domains involves the storage, retrieval, processing and visualisation of huge databases [3].

1.1 Problem Motivation

Big data is a rapidly growing global field however there is a bottleneck of technology that is required to extend the capabilities of the field. Several big data resources are required that currently don't exist, including: architectures, algorithms, and techniques [4].

1.2 Problem Statement

Question:
How can multidimensional sparse arrays be extendible?
Investigation:
An investigation into the extendibility of multidimensional sparse arrays.
Hypothesis:
The extendibility of multidimensional sparse arrays can be created using new storage techniques.
Statement:
The effects of storage techniques of the extendibility of multidimensional sparse arrays.
A title:
The extendibility of multidimensional sparse arrays: The effects of new storage techniques.

1.3 Significance

Data-Warehousing is a method used to combine multiple varied datasets into one complete and easily manipulated database. On-Line Analytical Processing (OLAP) can then be performed on these databases by an organisation in order to analyse the data, complete data mining and determine trends. The organisation is then able to build business intelligent decisions by utilising the analysed data. Data-warehousing has many applications in medical informatics and public-health, smart cities, mining, energy, physics and financial systems to name but a few.

The storage of the datasets influences the accuracy, speed and performance of the data analysis and thus it is crucial to assess how this information is stored and accessed. Several advancements have been made in multi-dimensional arrays in order to ensure that the datasets are efficient and inexpensive. Multi-dimensional arrays are used as their selection, aggregation, summation and other range queries are more efficient than their counter part SQL huge database queries. These datasets are continuously increasing by appending new data to the dataset as new information is added [3].

Recent breakthroughs in extendible multi-dimensional arrays have led to a new area of study in data-warehousing, which requires further research and development.

Efficient dynamic storage schemes for storing dense, extendible, multi-dimensional arrays by chunks have been determined [5]. However in data-warehousing there are often scenarios where sparse arrays occur due to new branches of information being created without any aligned historic data. Currently no storage or mapping techniques for extendible multi-dimensional sparse arrays exist. With computation and storage costs being the most expensive part of data-warehousing and data-mining, being able to utilize extendible multi-dimensional sparse arrays will extend the capabilities and domains for big-data analysis.

1.4 Application

There are several problems that currently exist in the world that would utilize Big Data. A major example would be the frequency and quality of demographic statistics produced by countries as well as demographic statistics deficits.

- There are 46 African countries operating without a complete birth and death registration system.
- Population censuses have not been conducted since 2010 in several African countries.
- Several African countries have not conducted an agricultural census in the last ten years. South Africa in particular held their last agricultural census in 2007.

Having up-to-date, reliable demographic statistics enables people within the population to partake in highly important activities. These include gaining an education, formal employment, voting in elections, access to financial services, obtaining passports and obtaining IDs to name but a few. Reliable data also enables governments to budget better and improve the delivery of public goods and services.

1.5 Early Results from Literature

1.6 Contribution to Field

The aim of the proposed study is to improve the modelling and representation of extendible multi-dimensional sparse arrays so that useful compression and storage

techniques can be determined. The representation of the extendible multi-dimensional sparse arrays must include the characteristics of an array format, such that the array can take on any designed multiple hyper-plane dimensions. The characteristics of the array allow for easy data access. In addition, this allows for both drill-in and roll-out queries. Where drill-in queries allow for detailed data access and roll-out queries allow for summary data access.

The modelling of the extendible multidimensional sparse arrays will be able to contribute to developing algorithms that can process data at a higher speeds, use less physical computational resources and improve the usage of computational power.

1.7 Challenges

1.7.1 "this is not research, this is what any competent engineer knows/can do/can find out."

Conclusion: Currently no storage or mapping techniques for extendible multidimensional sparse arrays exist.

Multidimensional sparse arrays require architectures, algorithms, and techniques that currently don't exist [2].
Architectures, algorithms, and techniques include storage and mapping techniques.
Storage and mapping techniques allow for the extendibility of multidimensional arrays.
Currently no storage or mapping techniques for extendible multidimensional sparse arrays exist.

1.7.2 "this is not an important topic to be working on"

Conclusion: Extendible multidimensional sparse arrays are important.

Multidimensional arrays are continuously increasing by appending new data to the dataset as new information is added [1].
In data-warehousing there are often scenarios where sparse arrays occur due to new branches of information being created without any aligned historic data.
Extendible multidimensional sparse arrays are important.

1.8 Organisation of the Proposal

Chapter 2

Background

Give chapter summary

2.1 Data Warehousing

Information is incredibly important [1]

2.2 Early Results

2.3 Result Differentiation

2.4 Methodologies

In order to develop a model for extendible multi-dimensional sparse arrays, the current representations of extendible multi-dimensional arrays must be analysed. The study of the current representations will give a clear indication of how to integrate the representation of extendible multi-dimensional sparse arrays, so that the model can easily be integrated into the current system, preventing unnecessary additional costs. Once a model has been developed, compression of data as well as computing speed will be assessed.

2.5 Sparse Array Representations

Hairong Wang [2]

- Space linear in the number of non-zero elements
- Acceleration of operations using General Purpose Computing on Graphics Processing Unit
- 2-dimensional arrays
  - Compressed Row (or alternatively Column) Storage (CRS/CCS)
- multi-dimensional sparse arrays
  - Bit-Encoded Sparse Storage (BESS)
  - Extended the CRS/CCS known as xCRS and xCCS
  - PATRICIA Trie Compressed Storage (PTCS)
- New storage techniques
  - Bit-Encoded Compressed Row (or Column) storage (BxCRS/BxCCS)
  - Hybrid approach (Hybrid) that combines BESS with xCRS

Matrix operations include:

- Get, Insert, Delete, Update
- Addition, Multiplication, Subtraction, Division

2D sparse array can be represented as a linked list or as triplicate

2.6 Dynamic Sparse Arrays

Increasing Density

- Bounds and structure remain the same
- new elements are slot into a chunk where there was a free space

Increasing Bound

2.7 Big Data Warehouse

- Bill Inmon Defines: A data warehouse is a subject-oriented, non-volatile, integrated, time variant collection of data created for the purpose of managements decision making.
  - A big data solution is a technology and data warehousing is an architecture.
- Ralph Kimball defines: A data warehouse is a copy of transaction data specifically structured for query and analysis.
- Volume - Scalability, storage to grow - investigate using a parallel file system
- Velocity - Cope with speed, rate at which data comes in
- Variety - Data types
- Veracity - validity
- Value
- Fault tolerance must be researched

2.8 Big Data Representations

- Capable of holding very large amounts of data.
- Hold the data in inexpensive storage devices.
- Processing is done by the Roman census method.
- Data is stored in an unstructured format.
- Hierarchical Data Format (HDF5)
  - Data model, library, and file format for storing and managing data
  - Supports an unlimited variety of datatypes, and is designed for flexible and efficient I/O and for high volume and complex data.
  - Portable and extensible, allowing applications to evolve in their use of HDF5.
  - Tools and applications for managing, manipulating, viewing, and analyzing data in the HDF5 format.
  - Organized in a hierarchical structure, with two primary structures: groups and datasets.

    * group: a grouping structure containing instances of zero or more groups or datasets, together with supporting metadata.
    * dataset: a multidimensional array of data elements, together with supporting metadata.

- Hadoop
  - Distributed storage and processing of very large data sets.
  - Consists of computer clusters built from commodity hardware.
  - Assumption that hardware failures are common occurrences and should be automatically handled by the framework
  - Hadoop can run parallel queries over flat files. This allows it do basic operational reporting on data in its original form.
  - Hadoop splits files into large blocks and distributes them across nodes in a cluster. It then transfers packaged code into nodes to process the data in parallel
  - Hadoop framework includes following four modules:
    * Hadoop Common: These libraries provides filesystem and OS level abstractions and contains the necessary Java files and scripts required to start Hadoop.
    * Hadoop YARN: This is a framework for job scheduling and cluster resource management.
    * Hadoop Distributed File System (HDFS): A distributed file system that provides high-throughput access to application data.
    * Hadoop MapReduce: This is YARN-based system for parallel processing of large data sets.

- Network Common Data Form (NetCDF)
  - Set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.

2.9 TileDB storage manager

- Used for data which is represented as multi-dimensional arrays.
- Sparse arrays significantly impact storage and application performance

- when storing data cells that are accessed together should be co-located - global cell orders in data tiles in sparse arrays include row-major vs column-major
- bookkeeping is very important for locating the correct data as exact distribution of the non-empty cells is unknown
- Still working on understanding the internals and basic operations in more depth
- Downloaded some files on reading/writing to sparse arrays (C) - DBexamples on Github - Intel HCS compiler - link between FPGA and GPU

2.10 FGPAs

Integrated circuit consisting of an array of identical logic blocks with programmable interconnections. -I do not feel that FGPAs would be the best suited for the research. [6]

2.11 Impact

2.12 New Results

Chapter 3

Methodology

Based on the literature review in Chapter 2, it is evident that the extendibility of multidimensional sparse arrays is an incredibly important topic that has been overlooked, specifically with regards to data warehousing and OLAP. In order to make a contribution to this field, an extendible multidimensional sparse array model must be developed, implemented and evaluated. This chapter presents the proposed model design and experimental procedures for the research study.

3.1 Extendible Multi-Dimensional Sparse Arrays

In order to represent a data warehouse in a multidimensional space a formula needs to be defined to increase the dimensionality of the storage scheme without changing the mapping function f(). As sparse data warehouses consist of large quantities of information with few non-zero elements, an efficient storage scheme that represents these non-zero elements must be chosen. Furthermore a method for extendibility of the multi-dimensional sparse array representation must be chosen.

3.1.1 Multi-Dimensional Sparse Array Representation

In order to improve the storage efficiency of multi-dimensional sparse arrays, a good storage scheme must be chosen. A comparison of the different sparse array storage techniques detailed in Section 2 is carried out.

CRS and CCS are both only used for 2-dimensional arrays. In the work of Goil et al. [7] it is clear that BESS out performs Offset-Value pair. In the work of Wang [2] it is evident that although PTCS has a better storage ratio and xCRS/xCCS and BxCRS have faster element retrieval times, BESS has a faster construction time and a faster multi-dimensional aggregation time than all of the other storage schemes.

BESS is a very simple and efficient multi-dimensional sparse array representation as it maps a multi-dimensional sparse array space to a one-dimensional array space, resulting in a bit encoded key index [2].

A multi-dimensional sparse array data model using a BESS array representation scheme will be analysed.

3.1.2 Extendible Multi-Dimensional Array Representation

In order to improve the performance of extendible multi-dimensional arrays, chunking techniques are used.

3.1.3 Extendible Multi-Dimensional Sparse Array Representation

By making use of chunking techniques on a BESS array representation scheme. The proposed extendibility model can increase the dimensionality of the storage scheme by either increasing the density of the array or increasing the dimension of the indices of the array. Two algorithms will need to be developed and implemented in software in order to determine which algorithm performs better.

To assess the performance of the extendible multi-dimensional models, a measure of the storage efficiency and order of retrieval of queries shall be conducted using basic construction, low level retrieval and partial match query array operations.

Figure 3.1 shows the main steps in the proposed methodology.

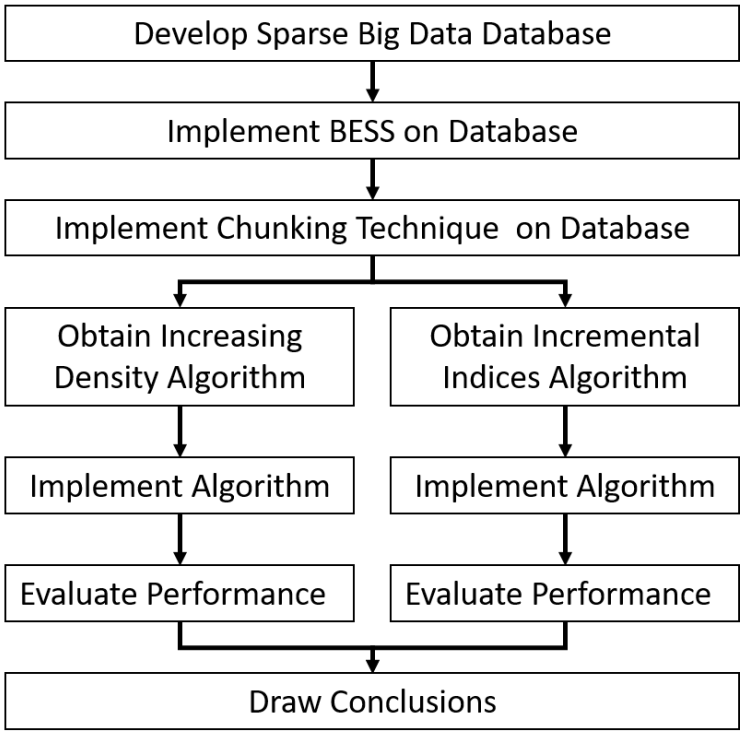

Figure 3.1: Proposed Methodology

Chapter 4

Experimental Setup

Give chapter summary

4.1 Computational Environment

4.1.1 Experimental Environment

4.1.2 Computational Model

4.1.3 Software

4.2 Experimental Data

4.3 Software Engineering Practice

Chapter 5

Preliminary Results

Give chapter summary

Chapter 6

Future Work

Give chapter summary

Chapter 7

Validation of Results

Give chapter summary

Chapter 8

Risk Management

Give chapter summary

Chapter 9

Schedule and Time-Line for Completion

Give chapter summary

Chapter 10

Summary

Give chapter summary

Bibliography

[1] M. Golfarelli and S. Rizzi. *Data Warehouse Design: Modern Principles and Methodologies*, chap. 1, pp. 1–42. India: McGraw-Hill Professional, first ed., Jan. 2009.

[2] H. Wang. *Sparse Array Representations And Some Selected Array Operations On GPUs*. MSc Dissertation, University of the Witwatersrand, Johannesburg, South Africa, Jul. 2014.

[3] E. J. Otoo and D. Rotem. "Efcient Storage Allocation of Large-Scale Extendible Multi-dimensional Scientic Datasets." In *18th International Conference on Scientific and Statistical Database Management*, pp. 179–183. IEEE, 2006.

[4] B. Twala. "Big Data for Africa." presentation, Feb. 2017.

[5] G. Nimako, E. J. Otoo, and D. Ohene-Kwofie. "Chunked Extendible Dense Arrays for Scientic Data Storage." In *41st International Conference on Parallel Processing Workshops*, pp. 38–47. 2012.

[6] C. H. Roth and L. John. United States of America: Cengage Learning, third ed., 2016.

[7] S. Goil and A. Choudhary. "BESS: A Sparse Storage Structure of Multidimensional Data for OLAP and Data Mining." Tech. Rep. CPDC-TR-9801-005, Centre for Parallel and Distributed Computing, Northwestern University, 1997.