2017 IEEE 14th International Scientific Conference on Informatics

INFORMATICS 2017

November 14-16, 2017, Poprad, Slovakia

PROCEEDINGS

Editors
Valerie Novitzká
Štefan Korečko
Anikó Szakál

Organized by

Affiliated branch of the Slovak Society for Applied Cybernetics and Informatics at Department of Computers and Informatics, FEEI TU of Košice

Faculty of Electrical Engineering and Informatics, Technical University of Košice

Association of Slovak Scientific and Technological Societes

IEEE SMCS Technical Committee on Computational Cybernetics

ISBN 978-1-5386-0888-3
IEEE catalog number CFP17E80-PRT

2017 IEEE 14th International Scientific Conference on Informatics

Copyright ©2017 by the Institute of Electrical and Electronics Engineers, Inc. All rights reserved.

Copyright and Reprint Permission
Abstracting is permitted with credit to the source. Libraries are permitted to photocopy beyond the limit of U.S. copyright law for private use of patrons those articles in this volume that carry a code at the bottom of the first page, provided the per-copy fee indicated in the code is paid through Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923. For reprint or republication permission, email to IEEE Copyrights Manager at pubs-permissions@ieee.org. All rights reserved. Copyright ©2017 by IEEE.

Other copying, reprint or reproduction requests should be addressed to IEEE Copyrights Manager, IEEE Service Center, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331.

IEEE Catalog Number CFP17E80-PRT
ISBN 978-1-5386-0888-3

Additional copies of this publication are available from
Curran Associates, Inc.
57 Morehouse Lane
Red Hook, NY 12571 USA
+1 845 758 0400
+1 845 758 2633 (FAX)
email: curran@proceedings.com

PREFACE

It is with pleasure that we hereby present the proceedings of the *2017 IEEE 14th International Scientific Conference on Informatics*. Its success builds upon our ever improving efforts to publish with higher standards in the various areas of informatics. As part of this, our conference becomes a significant international forum for presenting original research results, sharing experience, and exchanging new ideas. The topics of this conference cover theoretical and practical results, along with methods for transferring these research results into real-life domains, by scientist and experts working in computer science and informatics. The conference also provides an opportunity for young researchers to demonstrate their achievements and to discuss their results at an international scientific forum. The main topics of the conference are

- Computer Architectures,
- Computer Networks,
- Theoretical Informatics,
- Programming Paradigms and Languages,
- Software Engineering,
- Distributed Systems,
- Computer Graphics and Virtual Reality,
- Artificial Intelligence,
- Data and Knowledge Bases,
- Intelligent Information Systems,
- Applied Informatics and Simulation and
- Quality of Software and Services.

The proceedings start with keynote lectures from three eminent experts in their respective fields. Every paper in these proceedings has been peer-reviewed by two independent external referees. On behalf of the Programme and Organizing Committees, we would like to thank all the reviewers for their time and effort in reviewing the papers. Their contribution has ensured the high quality of publications in these proceedings. We would also like to extend our thanks to all the authors and keynote speakers, who contributed and guaranteed the high and professional standard of this conference.

The *2017 IEEE 14th International Scientific Conference on Informatics* has been organized by

- Faculty of Electrical Engineering and Informatics, Technical University of Košice,
- Slovak Society for Applied Cybernetics and Informatics at Department of Computers and Informatics,
- Association of Slovak Scientific and Technological Societies and
- IEEE SMCS Technical Committee on Computational Cybernetics.

We also thank the sponsors

- IEEE Hungary Section,
- IEEE SMC Chapter, Hungary,
- IEEE Joint IES/RAS Chapter, Hungary

and the technical co-sponsor

- IEEE SMC Society.

The conference is held in Poprad, a historical city nestled at the foot of the High Tatra Mountains, a region rich in culture and natural beauty. We are confident that inside these covers, you will find papers relevant to your field of interest. We also look forward to your participation at the next event of this conference.

Poprad, November 2017

On the behalf of the Programme and Organizing Committees
Valerie Novitzká

COMMITTEES

General Chair

prof. Ing. Liberios Vokorokos, PhD., *Dean of Faculty of Electrical Engineering and Informatics, Technical University of Košice, SK*

Honorary Chair

akad. prof. Ing. Ivan Plander, DrSc., *Slovak Society for Applied Cybernetics and Informatics, SK*

Program Chair

Valerie Novitzká, *Technical University of Košice, SK*

Program Committee

Miklós Bartha, *Memorial University of Newfoundland, CA*
Andreas Bollin, *University of Klagenfurt, AT*
Dmitriy B. Buy, *National University of Taras Shevchenko, Kyiv, UA*
Jan Čapek, *University of Pardubice, CZ*
Svetlana Cicmil, *University of the West England, Bristol, UK*
Zbigniew Domański, *Częstochowa University of Technology, PL*
Erik Duval, *Katholieke Universiteit Leuven, BE*
Dimitar Filev, *Ohio State University, US*
Zoltán Fülöp, *University of Szeged, HU*
Gianina Gábor, *University of Oradea, RO*
Ján Genči, *Technical University of Košice, SK*
Andrzej Grzybowski, *Częstochowa University of Technology, PL*
Klaus Haenssgen, *Leipzig University of Applied Sciences, DE*
Tamás Haidegger, *Óbuda University, Budapest, HU*
Aboul Ella Hassanien, *Cairo University, Giza, EG*
Zdeněk Havlice, *Technical University of Košice, SK*
Pedro Rangel Henriques, *University of Minho, Braga, PT*
Pavel Herout, *University of West Bohemia, Pilsen, CZ*
Ladislav Hluchý, *Slovak Academy of Sciences, Bratislava, SK*
Elke Hochmüller, *Carinthia University of Applied Sciences, AT*
László Horváth, *Óbuda University, Budapest, HU*
Zoltán Horváth, *Lóránd Eötvös University, Budapest, HU*
Péter Kádár, *Óbuda University, Budapest, HU*
Waldemar W. Koczkodaj, *Laurentian University, CA*
Levente Kovács, *Óbuda University, Budapest, HU*
Jiří Kunovský, *Brno University of Technology, CZ*
Sandra Lovrenčić, *University of Zagreb, HR*
Mária Lucká, *Slovak University of Technology, Bratislava, SK*
Ivan Luković, *University of Novi Sad, RS*
Dragan Mašulović, *University of Novi Sad, RS*
Karol Matiaško, *University of Žilina, SK*
Marjan Mernik, *University of Maribor, SI*
Jurij Mihelič, *University of Ljubljana, SI*
Hanspeter Mössenböck, *Johannes Kepler University Linz, AT*
Günter Müller, *Albert-Ludwigs-Universität Freiburg, DE*
Hiroshi Nakano, *Kumamoto University, JP*
Mykola S. Nikitchenko, *National University of Taras Shevchenko, Kyiv, UA*
Lucia Pomello, *University of Milano-Bicocca, IT*
Herbert Prähoffer, *Johannes Kepler University Linz, AT*
Horia F. Pop, *Babes Bolyai University, Cluj, RO*
Stanislav Racek, *University of West Bohemia, Pilsen, CZ*
Mihály Réger, *Óbuda University, Budapest, HU*
Sonja Ristić, *University of Novi Sad, RS*

Imre J. Rudas, *Óbuda University, Budapest, HU*
Gábor Sági, *Hungarian Academy of Sciences, Budapest, HU*
Abdel-Badeeh M. Salem, *Ain Shams University, Cairo, EG*
Wolfgang Schreiner, *RISC, Johannes Kepler University Linz, AT*
Elena Somova, *University of Plovdiv, BG*
William Steingartner, *Technical University of Košice, SK*
Jiří Šafařík, *University of West Bohemia, Pilsen, CZ*
Petr Šaloun, *VŠB - Technical University of Ostrava, CZ*
Jarmila Škrinárová, *Matej Bel University, Banská Bystrica, SK*
Michal Štepanovský, *Czech Technical University in Prague, CZ*
József K. Tar, *Óbuda University, Budapest, HU*
Katarína Teplická, *Technical University of Košice, SK*
Renáta Tkáčová, *Technical University of Košice, SK*
Tsuyoshi Usagawa, *Kumamoto University, JP*
Valentino Vranić, *Slovak University of Technology, Bratislava, SK*
Neven Vrček, *University of Zagreb, HR*
František Zbořil, *Brno University of Technology, CZ*
Jaroslav Zendulka, *Brno University of Technology, CZ*
Doina Zmaranda, *University of Oradea, RO*

Organizing Committee

Milan Šujanský, *Technical University of Košice, SK* (chair)
Anikó Szakál, *Óbuda University, Budapest, HU* (financial chair)
William Steingartner, *Technical University of Košice, SK* (general manager)
Martina Dragošeková, *Technical University of Košice, SK*
Sergej Chodarev, *Technical University of Košice, SK*
Štefan Korečko, *Technical University of Košice, SK*

Technical staff

Ján Perháč, *Technical University of Košice, SK*
Jana Šťastná, *Technical University of Košice, SK*

Darko Galinec, William Steingartner:
Combining Cybersecurity and Cyber Defence to achieve Cyber Resilience 87

Kornél Gyöngyösi, Péter János Varga, Zsolt Illési:
WLAN heat mapping in hybrid network 94

Peter Halaš, Marek Lóderer, Viera Rozinajová:
Prediction of Electricity Consumption using Biologically Inspired Algorithms 98

Median Hilal, Christoph G. Schütz, Michael Schrefl:
Superimposed Multidimensional Schemas for RDF Data Analysis 104

Dana Horváthová, Vladimir Siladi:
Creation of interactive panoramic video for phobia treatment 111

Marián Hudák, Štefan Korečko, Branislav Sobota:
Peripheral Devices Support for LIRKIS CAVE 117

Sergej Chodarev, Michaela Bačíková:
Development of Oberon-0 using YAJCo 122

Judith Jakob, Kordula Kugele, József Tick:
Defining Camera-Based Traffic Scenarios and Use Cases for the Visually Impaired by means of Expert Interviews 128

Milan Jančár, Jaroslav Porubän:
Examining Truthfulness of Informal Diagrams More Than a Year After Their Creation 134

Tomáš Jarábek, Peter Laurinec, Mária Lucká:
Energy Load Forecast Using S2S Deep Neural Networks with k-Shape Clustering 140

Milan Jičínský, Jaroslav Marek:
Clustering analysis of phonetic and text feature vectors 146

Ján Juhár, Liberios Vokorokos:
Towards a Uniform Code Annotation Approach with Configurable Annotation Granularity 152

Ondrej Kainz, Frantisek Jakab, Miroslav Michalko:
Proposal of Human Body Description Format XML Schema and Validation of Anthropometric Parameters .. 158

Maksat N. Kalimoldayev, Vladimir Siladi, Maxat N. Satymbekov, Lyazat Naizabayeva:
Solving Mean-Shift Clustering Using MapReduce Hadoop 164

Ammar Ahmed Khan, Mika Ahistus, Tero Liukko, Joni Lumela, Otto Sassi, Seppo J. Ovaska:
aaltOS for Energy Harvesting Applications: Effects of Clock Frequency and System Tick on Power and Energy Consumption 168

Pawel Kossecki, Łukasz Korc, Stefan Kossecki:
Valuation of Intellectual Property – Income Approach and Scenario Analysis. Software on Early Stage of Implementation 173

Rafał Kozik, Michał Choraś, Damian Puchalski, Rafał Renk:
Data Analysis Tool Supporting Software Development Process 179

Solving Mean-Shift Clustering Using MapReduce Hadoop

Maksat N. Kalimoldayev
Institute of Information and Computational Technologies
Almaty, Kazakhstan
mnk@ipic.kz

Vladimir Siladi
Faculty of Natural Sciences
Matej Bel University
Banska Bystrica, Slovakia
vladimir.siladi@umb.sk

Maksat N. Satymbekov
Institute of Information and Computational Technologies
Almaty, Kazakhstan
m.n.satymbekov@gmail.com

Lyazat Naizabayeva
Institute of Information and Computational Technologies
Almaty, Kazakhstan
naizabayeva@gmail.com

***Abstract*—Paper presents results of a practical experiment that was conducted in order to pursuit main objective - design and implement novel iterative MapReduce framework based on Hadoop technology. To study the effects of implementation parallel scientific applications, we deployed. Despite the fact that they did not show better performance than the same algorithm implemented in MPI, we conducted experiments to solve problems iterative computations using the Hadoop architecture. As a result of the experiments, the fail-safe behavior of Hadoop technology was revealed in the solutions of complex hadacies.**

***Index Terms*—Hadoop, MapReduce, Mean-Shift, MPI, HDFS**

I. Introduction

Paper presents results of a practical experiment that was conducted in order to pursuit main objective - design and implement novel iterative MapReduce framework [1] based on Hadoop technology [2], [3]. To study the effects of implementation parallel scientific applications, we deployed algorithm of mean shift clustering on Hadoop platform. The reminder of the article is structured as follows. The first section presents introduction. In second section we present matematical model mean shift clustering. In third section we describe Hadoop, MapReduce and HDFS. In fourth section we comprised parallel algorithm for solving the mean shift clustering. In fifth section "Implementation and evaluation" presents MapReduce implementations performance. Finally sections discussion and conclusions. Our work is based upon modifying the programming model and using Hadoop standard features. Cluster is a collection of data members having similar characteristics. The process of establishing a relation or deriving information from raw data by performing some operations on the data set like clustering is known as data mining. Data collected in practical scenarios is more often than not completely random and unstructured. Hence, there is always a need for analysis of unstructured data sets to derive meaningful information. This is where unsupervised algorithms come in to picture to process unstructured or even semi structured data sets by resultant. Mean Shift Clustering is one such technique used to provide a structure to unstructured data so that valuable information can be extracted. This paper discusses the implementation of the Mean Shift Clustering Algorithm over a distributed environment using ApacheTM Hadoop. Research result is the comparison of the result with different cluster settings and MPI model.

II. Mean shift clustering

The mean shift algorithm is a nonparametric clustering technique which does not require prior knowledge of the number of clusters, and does not constrain the shape of the clusters. Given n data points $x_i, i = 1, 2, ..., n$, on a d-dimensional space R_d, the multivariate kernel density estimate obtained with kernel $K(x)$ and window radius h is

$$f(x) = \frac{1}{nh^d} + K(\frac{x - x_i}{h}) \tag{1}$$

For radially symmetric kernels, it suffices to define the profile of the kernel $k(x)$ satisfying

$$K(x) = c_{k,d}k(||x||^2) \tag{2}$$

Where $c_{k,d}$ is a normalization constant which assures $K(x)$ integrates to 1. The modes of the density function are located at the zeros of the gradient function $\nabla f(x) = 0$.

The gradient of the density estimator (1) is

$$\nabla f(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}(x_i x)g(||\frac{x - x_i}{h}||^2) \tag{3}$$

where $g(s) = -k^1(s)$. The first term is proportional to the density estimate at x computed with kernel $G(x) = c_{g,d}g([||x||^2])$ and the second term

$$m_h(x) = \frac{\sum\limits_{i-1}^{n} x_i g(||\frac{x-x_i}{h}||^2)}{\sum\limits_{i-1}^{n} g(||\frac{x-x_i}{h}||^2)} - x \tag{4}$$

978-1-5386-0889-0/17/$31.00 ©2017 IEEE

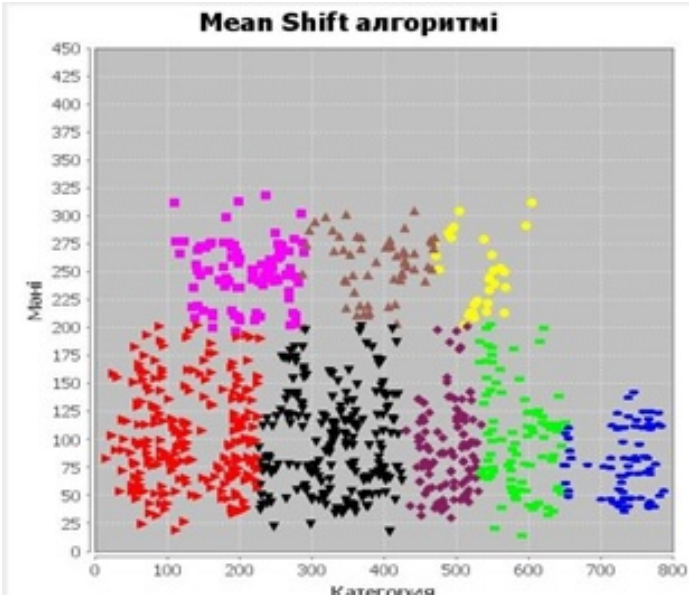

Fig. 1. Mean Shift.

is the mean shift. The mean shift vector always points toward the direction of the maximum increase in the density. The mean shift procedure, obtained by successive

- computation of the mean shift vector $m_h(x^t)$
- translation of the window $x^{t+1} = x^t + m_h(x^t)$

is guaranteed to converge to a point where the gradient of density function is zero. Mean shift mode finding process is illustrated in Figure 1. The mean shift clustering algorithm is a practical application of the mode finding procedure:

III. Hadoop, MapReduce and HDFS: A Developers Perspective

Hadoop is an open-source project overseen by the Apache Software Foundation Hadoop is mostly used for reliable, scalable and distributed computing, but can also be used as a general purpose file store capable of hosting petabytes of data. Many companies use Hadoop for both research and production purposes. Initially, Hadoop was primarily a tool for storing data and launching MapReduce tasks, but now Hadoop is a large stack of technologies that are somehow related to processing large data (not just with MapReduce). The core components of Hadoop are:

-Hadoop Distributed File System (HDFS) is a distributed file system that allows you to store information of almost unlimited volume.

- Hadoop YARN - a framework for managing cluster resources and task management, including the MapReduce framework.

Many of the features and configurations supported by Hadoop provide usability, power and wide-ranging application possibilities for this platform. For example, Yahoo! and countless other organizations are using Hadoop to analyze tons of bits and bytes of information. On the other hand, Hadoop

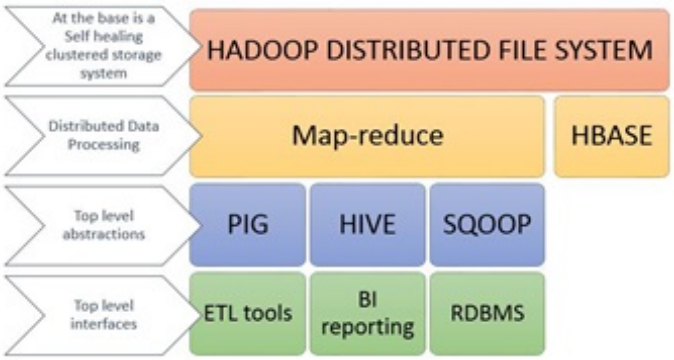

Fig. 2. Hadoop Architecture.

can be easily launched on one node. All you need for this is the ability to program in Java (including generics).

In the Hadoop infrastructure, the map and reduce functions are implemented as an extension of the base classes. Specific implementations of functions are linked into a single module in accordance with the selected configuration, which, in addition, defines the input and output formats. Hadoop is well suited for processing huge files containing structured data. One of the main advantages of this platform is that it organizes the initial parsing of the input file, so you just have to process each line. Thus, the task of the map function is to determine which data is to be extracted from the incoming text line.

Hadoop handles large number of data from different system like Images, videos, Audios, Folders, Files, Software, Sensor Records, Communication data, Structured Query, unstructured data, Email conversations, and anything which we can't think in any format[5]. All these resources can be stored in a Hadoop cluster without any schema representation instead of collecting from different systems. There are many components involved in Hadoop like Avro, Chukwa, Flume, HBase, Hive, Lucene, Oozie, Pig, Sqoop and Zookeeper. The Hadoop Package also provides Documentation, source code, location awareness, Work scheduling. A Hadoop cluster contains one Master node and Many Slave nodes. The master node consists of Data node, Name node, Job Tracker and Task Tracker where slave node acts as both a Task Tracker and Data node which holds compute only and data only worker node. The Job Tracker manages the job scheduling. Basically Hadoop consists of two Parts. They are Hadoop Distributed File system (HDFS) and Map Reduce[5].HDFS provides Storage of data and Map Reduce provides Analysis of data in clustered environment. The Architecture of Hadoop is represented in figure 2

HDFS is the primary storage system used by Hadoop applications. HDFS creates multiple replicas of data blocks and distributes them on compute sites throughout a cluster to enable reliable, extremely rapid computations:

- Data is distributed across many machines at load time.
- HDFS is optimized for large, streaming reads of files rather than for irregular random reads.

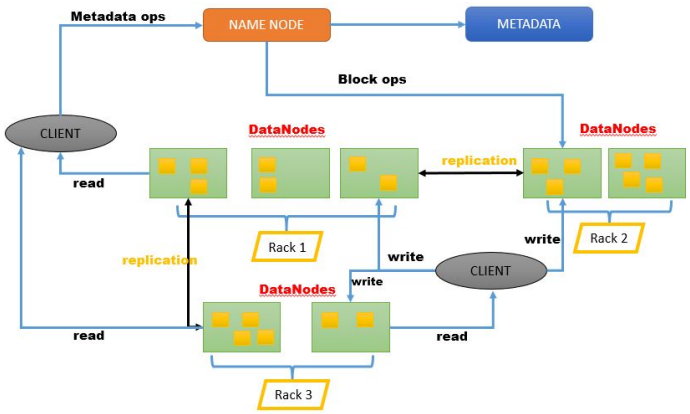

Fig. 3. HDFS Architecture.

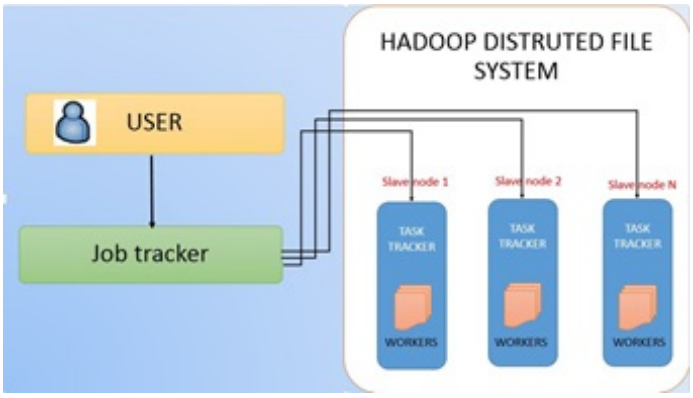

Fig. 4. Mapreduce Architecture.

- Files in HDFS are written once and no random writes to files are allowed.
- Applications can read and write HDFS files directly via the Java API.

The communication between the nodes occurs through Remote Procedure calls. Name node stores metadata like the name, replicas, file attributes, locations of each block address and the fast lookup of metadata is stored in Random Access Memory by Metadata. It also reduces the data loss and prevents corruption of the file system. Name node only monitors the number of blocks in data node and if any block lost or failed in the replica of a datanode, the name node creates another replica of the same block. Each block in the data node is maintained with timestamp to identify the current status. If any failure occurs in the node, it need not be repair immediately it can be repaired periodically. The architecture of HDFS is shown in Figure 3. Security Issues in HDFS The HDFS is the base layer of Hadoop Architecture contains different classifications of data and it is more sensitive to security issues.

MapReduce is a programming model and a framework for writing applications of large-scale computations on the basis of a large number of computations:

- Provides automatic parallelization and distribution of tasks.
- has built-in mechanisms for maintaining stability and operability in case of failure of individual elements.
- Gives status and monitoring tools.
- Provides a clean abstraction layer for programmers.

As an infrastructure designed to handle large data sets, the MapReduce model is optimized for distributed work across multiple computers. As the name implies, this infrastructure includes two main functions. The task of the map function is to split the incoming data stream into smaller sets and transfer these data sets to other processes for further processing. The reduce function analyzes the results of processing individual sets, obtained by the map function, and passes them to the output of the program.

The programmer designs a Map function that uses a (key,value) pair for computation. The Map function results in the creation of another set of data in form of (key,value) pair which is known as the intermediate data set. The programmer also designs a Reduce function that combines value elements of the (key,value) paired intermediate data set having the same intermediate key. [5] Map and Reduce steps are separate and distinct and complete freedom is given to the programmer to design them. Each of the Map and Reduce steps are performed in parallel on pairs of (key,value) data members. Thereby the program is segmented into two distinct and well defined stages namely Map and Reduce. The Map stage involves execution of a function on a given data set in the form of (key,value) and generates the intermediate data set. The generated intermediate data set is then organized for the implementation of the Reduce operation. Data transfer takes place between the Map and Reduce functions. The Reduce function compiles all the data sets bearing the particular key and this process is repeated for all the various key values. The final out put produced by the Reduce call is also a dataset of (key,value) pairs. An important thing to note is that the execution of the Reduce function is possible only after the Mapping process is complete. Each MapReduce Framework has a solo Job Tracker and multiple task trackers. Each node connected to the network has the right to behave as a slave Task Tracker. The issues like division of data to various nodes , task scheduling, node failures, task failure management, communication of nodes, monitoring the task progress is all taken care by the master node. The data used as input and output data is stored in the file-system.

IV. Parallel algorithm for solving the Mean-Shift Clustering

Algorithm 1 Mapper design for Mean-Shift Clusreing

1. **procedure** MeanShiftDesign
2. Load Cluster file
3. fp=Mapcluster file
4. *Create two list*
5. *listnew=listold*
6. *CALL read(Mapcluster file)*
7. *newfp = MapCluster()*

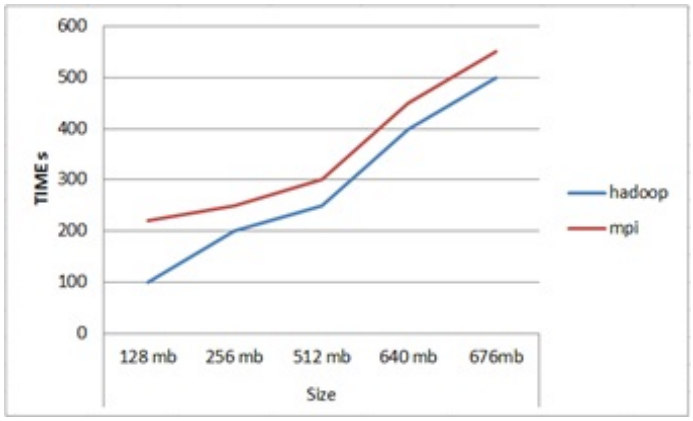

Fig. 5. Mapreduce Architecture.

8. *dv = 0*
9. *Assign correct centeroid*
10. *read(div)*
11. *calculate centeroid*
12. *div = minCenter()*
13. *CALL MeanSHIFTReduce()*
14. ***end procedure** = 0*

Algorithm 2 Reducer design for Mean Shift Clustering
1. **procedure** MeanShiftReduceClustering
2.NEW listofclusters
3. COMBINE result clusters from MAPCLASS.
4. **if** cluster size too high or too low **then**
RESIZE the cluster
5. C_{Max} = findMaxSize(ListofClusters)
6. C_{MIN} = findMinSize(ListofClusters)
7. **if** $C_{Max} > \frac{1}{20}$ totalSize then Resize(Cluster)
8. WRITE cluster FILE to output DIRECTORY

Algorithm 3 Implementing MeanShift Function
1. **Procedure** MeanShift Function
2. **If** Initial Iteration **then** LOAD cluster file from DORECTORY
else Read cluster file from previous iteration
3. Create new JOB
4. SET MAPPER to map class Defined
5. SET REDUCER to reduce class define
6. Paths for output directory
7. SUBMIT JOB

V. Implementation and evaluation

In order to test the application we set up the infrastructure that included 8 Core i-5 Processor PCs with 16Gb RAM, HP Blade servers with 4 Core Intel Xeon Processors and Gigabit network connection Switch. All computing machines got 64-bit Ubuntu 12.10 operating system and Hadoop 1.2.1 platform installed.

VI. CONCLUSION

In this paper we studied the adaptation of scientific computing tasks to a cloud environment, such as MapReduce. The presented study presents a new iterative processing platform for Hadoop. Despite the fact that they did not show better performance than the same algorithm implemented in MPI, we conducted experiments to solve problems iterative computations using the Hadoop architecture. As a result of the experiments, the High-Availability behaviour of Hadoop technology was revealed in the solutions of complex problem.

References

[1] J. Dean and S. Ghemawat, “Mapreduce: Simplified data processing on large clusters,” Communications of the ACM, vol. 51, no. 1, pp. 107–113, 2008.
[2] C. Lam, Hadoop in action. Manning Publications Co., 2010.
[3] T. White, Hadoop: The Definitive Guide. O’Reilly Press, 2009.
[4] Lammel, R.: Google’s MapReduce Programming Model - Revisited. Science of Computer Programming 70, 1–30 (2008).
[5] A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu,P. Wyckoff, R. Murthy, “Hive – A Warehousing Solution Over a Map-Reduce Framework,” In Proc. of Very Large Data Bases, vol. 2 no. 2,August 2009, pp. 1626-1629
[6] M. Fashing and C. Tomasi. Mean shift is a bound optimization. IEEE Trans. Pattern Analysis and Machine Intelligence, 27(3):471–474, Mar. 2005.
[7] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Analysis and Machine Intelligence, 24(5):603–619, May 2002.
[8] Apache Hadoop. http://hadoop.apache.org/
[9] Y. Wong. Clustering data by melting. Neural Computation, 5(1):89–104, Jan. 1993
[10] X. Yuan, B.-G. Hu, and R. He. Agglomerative mean-shift clustering. IEEE Trans. Knowledge and Data Engineering, 24(2):209–219, Feb. 2010.

AUTHOR INDEX

Ahistus, M. 168
Akagic, A. 45
Anderla, A. 13
Andrešič, D. 394
Arsenovic, M. 13
Baban, G. 18
Bačíková, M. 24, 122
Bajkó, D. 238
Bilanová, Z. 420
Boar, R. 31
Bobik, P. 425
Boháčik, J. 35, 40
Buza, E. 45
Ciocarlie, H. 31
Cojocar, G. S. 51
Cosariu, C. 18
Čegan, L. 431
Čibej, U. 57, 265
Dankovičová, Z. 63
Derda, T. 68
Dobravec, T. 74
Domański, Z. 68
Dostál, J. 359
Drąg, P. 80
Drotár, P. 63
Dudas, A. 400
Feciľák, P. 238
Filip, P. 431
Galinec, D. 87
Galko, L. 24
Gazda, J. 63
Genči, J. 425
Grzybowski, A. Z. 1, 221, 227
Guran, A. 51
Guzan, M. 344
Gyöngyösi, K. 94
Hajdu, B. 216
Halaš, P. 98
Hasic, H. 45
Hilal, M. 104
Hluchý, L. 185
Horváth, R. 306
Horváthová, D. 111
Hudák, M. 117
Hulič, M. 259
Hurná, S. 412
Hurtuk, J. 238
Hvizdová, E. 24
Chodarev, S. 122
Choraś, M. 179
Chovancová, E. 238
Illési, Z. 94
Iovanovici, A. 18, 31
Ivaniga, P. 289
Ivaniga, T. 289
Jakab, F. 158
Jakob, J. 128
Jančár, M. 134
Jarábek, T. 140
Jičínský, M. 146
Juhár, J. 152
Kainz, O. 158
Kalimoldayev, M. N. 164
Khan, A. A. 168
Kollár, J. 322, 353
Korc, Ł. 173
Korečko, Š. 117
Kossecki, P. 173
Kossecki, S. 173
Kotlińska, M. 377
Kozik, R. 179
Krammer, P. 185
Kugele, K. 128
Kunovský, J. 271
Kurachka, K. S. 194
Kurdel, P. 318
Kurkowski, M. 388
Kvassay, M. 185
Kvet, M. 197, 253
Lakatos, I. 203
Lalic, B. 13
Laurinec, P. 140, 210
Lazányi, K. 216
Liukko, T. 168
Lóderer, M. 98
Lucká, M. 140, 210
Lumela, J. 168
M. Eldojali, M. A. 359
Madarász, N. 383
Madeja, M. 232
Madoš, B. 238
Malčík, M. 394
Marek, J. 146, 276
Márton, G. 243
Márton, M. 249
Maťaš, L. 293
Matiaško, K. 197, 253
Melicherčík, M. 416
Mihaľov, J. 259
Mihályi, D. 293, 420
Mihelič, J. 57, 265
Michalko, M. 158
Mrekaj, B. 318
Naizabayeva, L. 164
Nečasová, G. 271
Nedvědová, M. 276
Nespěšný, D. 394
Omanovic, S. 45
Oravec, J. 284

Ovaska, S. J. 168
Ovseník, Ľ. 249, 289, 349
Perháč, J. 293
Péter, T. 203
Plander, I. 299
Porkoláb, Z. 243
Porubän, J. 134, 232, 372
Povinský, M. 327, 416
Prodan, L. 18
Puchała, P. 221
Puchalski, D. 179
Radaković, D. 359
Renk, R. 179
Ristić, S. 6
Rozinajová, V. 98
Rudas, I. J. 12
Sági, G. 306
Sassi, O. 168
Satymbekov, M. N. 164, 327
Savaş, S. 313
Sedláčková, A. N. 318
Schrefl, M. 104
Schütz, C. G. 104
Sičák, M. 322
Siedlecka-Lamch, O. 388
Siladi, V. 111, 164, 327
Silváši, F. 332
Skupa, J. 339
Sladojevic, S. 13
Sobota, B. 117, 344
Solus, D. 349
Spišiak, M. 353
Starczewski, T. 227
Stefanovic, D. 13
Steingartner, W. 87, 359
Styczeń, K. 80
Subecz, Z. 365
Sulír, M. 372
Świerczyńska-Kaczor, U. 377
Szekeres, I. 243
Szikora, P. 383
Szymoniak, S. 388
Šafařík, J. 339
Šaloun, P. 394
Šátek, V. 271
Škrinárová, J. 400
Špes, M. 249
Šťastná, J. 406
Štepanovský, M. 299
Teplická, K. 412
Tick, J. 128
Titrik, Á. 203
Tomášek, M. 332, 406
Topaloğlu, N. 313
Trajteľ, Ľ. 327
Tsalka, I. M. 194
Turán, J. 284, 289, 349
Vagač, M. 416
Varga, P. J. 94
Veigend, P. 271
Vesel, E. 400
Vokorokos, L. 63, 152, 420
Vrábel, M. 425
Zábovský, M. 35, 40
Zoričák, O. 372