UDC 004.75

M.V. TALAKH, YU.YA. TOMKA, YU.O. USHENKO, I.V. SOLTYS

# POSSIBILITIES OF USING HADOOP AND R TO ANALYZE LARGE ARRAYS OF GEOSPATIAL DATA

*Chernivtsi National University named after Yuriy Fedkovych, Chernivtsi,*
*2 Kotsjubynskyi Str. Chernivtsi, Ukraine, e-mail: m.talah@chnu.edu.ua*

**Анотація**. Проаналізовані основні проблеми пов'язані з обробкою Big Data, зокрема масивів, що містять геопросторові дані. Розглянуто платформу Hadoop, як один з базових підходів до аналізу великих масивів даних та можливості її інтеграції з середовищем R. Проаналізовано потенційні можливості використання платформи Hadoop для вирішення практичних задач в процесі аналізу геопросторових та просторово-часових даних.

**Ключові слова:** «великі дані», геопросторові дані, логічна операція; Hadoop, мова R

**Abstract**. The main problems associated with the processing of Big Data, in particular arrays containing geospatial data, are analyzed. The Hadoop platform is considered one of the basic approaches to the analysis of large data arrays and the possibility of its integration with the R environment. The potential possibilities of using the Hadoop platform for solving practical problems in the process of analyzing geospatial and spatiotemporal data are analyzed.

**Ключові слова**: Big Data, geospatial data, логічна операція; Hadoop, R language

**DOI: 10.31649/1681-7893-2022-44-2-49-54**

## INTRODUCTION

The last few decades can be safely called the era of big data. This is especially true for geographic or geospatial data, which is also associated with the rapid development of digital technology and its availability. There was a need to develop algorithms capable of processing large data arrays, the size of which is constantly increasing, and they, by their nature, can be both structured and poorly structured. In particular, this led to the development of new software tools and methods, as well as hardware tools for parallel computing and the search for new types of architectures.

## 1. GEOSPATIAL DATA AS BIG DATA

Spatial data describes special features of geographic orientation and spatial distribution in the real world. Geospatial data is typically multidimensional in spatial, spectral, and temporal terms. They include feature attributes, number, location, and relationship. They can be represented by point value, height, length, area, and pixel characteristics. Less commonly, it may be a place name string or self-image, spatial relationships, and other topologies. It should also be noted that they can appear in raster and vector formats and tabular forms [1, 2, 3].

One of the aspects that significantly affect the nature of the processing of geospatial data is associated with updating large arrays of this data. In general, Big Data are data sets whose size, structure and growth rate make them difficult to collect, manage, process or analyze with traditional technologies and tools [4, 5].

Moreover, it is spatial data that makes up about 80% of all Big Data [6,7].

## 2. HADOOP PLATFORM AND ITS STRUCTURE

The most common choice for working with large geospatial datasets is the Hadoop platform.

Hadoop is a free platform designed for distributed processing of large data sets using clusters that implement simple programming models [7, 19, 20]. Hadoop is the de facto standard in the field of data processing and storage and is a freely distributed set of utilities, libraries and frameworks, whose task is to manage a cluster of computers. Although it was developed in Java, other languages can also be used to work with Hadoop: R, Python or Ruby.

In general, the structure of Hadoop includes:

- distributed file system Hadoop (Hadoop Distributed File System – HDFS);
- Hadoop MapReduce is a system for the parallel processing of large data sets that implements the MapReduce distributed programming model [5];
- Hadoop YARN is a framework for task scheduling and cluster resource management.

The first two components are the main ones. HDFS provides file storage by distributing them across multiple nodes in a Hadoop cluster and unlimited scalability.

HDFS is based on a client/server architecture consisting of a single NameNode implemented on the main server managing the NameSpase file system and multiple DataNodes managing the storage attached to the nodes. Files are divided into blocks stored in a set of DataNodes. The NameNode is responsible for operations such as opening, closing, or renaming files, while the DataNodes is responsible for responding to read or write requests from clients.

Features of HDFS are that it has a block size of more than 64MB and is optimized for streaming data access and for using a large number of low-cost servers.

The second important component is the MapReduce programming and distributed computing platform, which eliminates the main problems of parallel and distributed architectures and allows you to develop applications. The main stages of the implementation of MapReduce are shown in Figure 1.

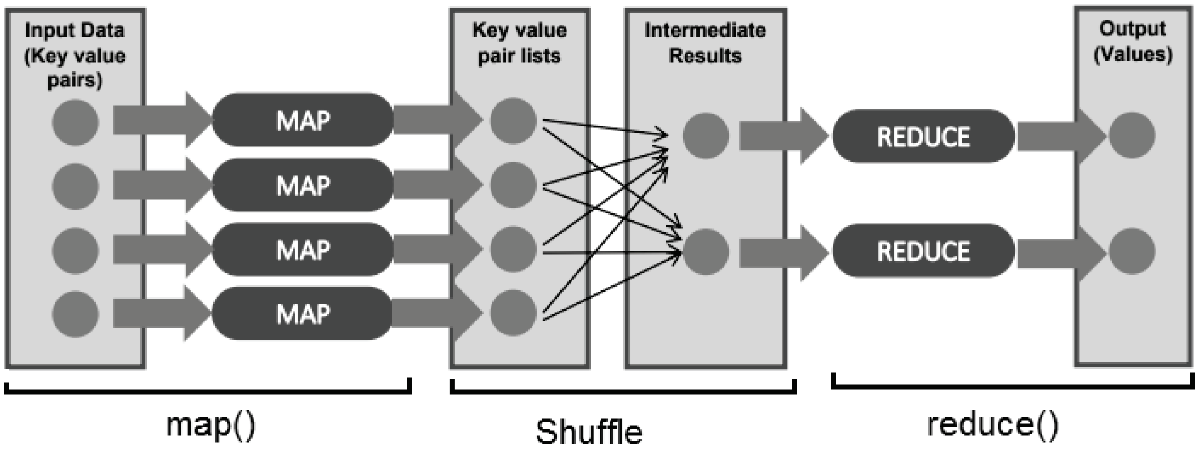

Figure 1 – How MapReduce works

1. Stage Map. At this stage, the data is preprocessed using the map function, which is defined by the user. The map function is applied to a single input record and returns multiple key-value pairs. What exactly will be contained in the key and the value is up to the user to decide. Subsequently, data with one key will fall into one instance of the reduce function.

2. Stage Shuffle. This is a step hidden from the user. On it, the outputs obtained in the previous stage are parsed into separate groups, where each group corresponds to one key.

3. Stage Reduce. Each group with values formed in the previous step is fed into the reduce function, which is specified by the user and calculates the final result for each group. The set of all values returned by reduce is the final output of MapReduce.

There are many new methods for supporting spatial queries on Hadoop, but most of them require internal modifications of the frameworks. In particular, they include a spatial index based on a hierarchical structure of spatial data stored in distributed file systems.

There are also specific extensions for Hadoop for working with geospatial data. The most commonly used for this purpose is GeoJinni (SpatialHadoop). It allows you to add geospatial features to various Hadoop layers and components to store, process, and index large geodata. At the same time, a new data type is added to the lowest level, which allows storing and processing of geodata as a key-value. Tools for loading and unloading different formats of geodata are also added. GeoJinni is installed as an extension to an already existing Hadoop cluster.

## 2. HADOOP PLATFORM AND ITS STRUCTURE INTEGRATION OF HADOOP AND R ENVIRONMENT

Hadoop can have different integration options with other programming languages. The most logical integration seems to be with the languages commonly used for data analysis, namely R and Python. The use of scripts and packages for data processing when working with Hadoop may involve rewriting them in Java or another language implemented by MapReduce. Therefore, the best option is to choose a way to connect Hadoop using already-written software. This feature is implemented for the R language [8, 9, 10].

Another reason for integrating with R is the way R works. In this case, data loaded into the main memory is being processed. Very large datasets cannot be loaded into RAM and for this data Hadoop integrated with R is one of the best solutions.

There are many approaches to integrating R and Hadoop, however, research has shown that the most commonly used approach for connecting R and Hadoop is Rhadoop [11, 21, 22], and it is also one of the easiest to implement.

RHadoop is an open-source project developed by Revolution Analytics that provides client-side integration between R and Hadoop and allows you to run the MapReduce function in R. The general structure of Rhadoop is shown in Figure 2.

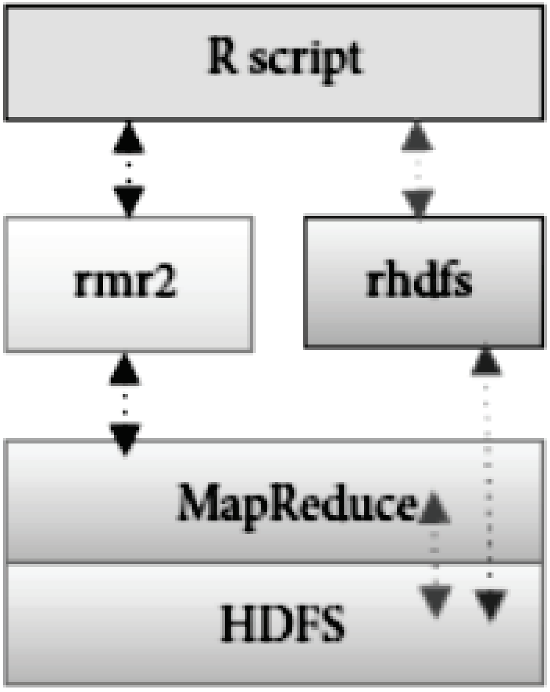

Figure 2 – Structure of RHadoop [13]

RHadoop consists of several packages. Below are the main ones:

- rhdfs - functions that provide management of HDFS files from R; used to read and write a function used for data in a cluster; It is installed only on the DataNode cluster;
- rmr (rmr2) - functions that provide the functionality of Hadoop MapReduce in R;
- rbbase are functions that provide database management for a distributed database within R.

Figure 3 shows the detailed architecture of Hadoop and Rhadoop [13].

The first hardware layer is usually represented by a cluster of computers. The middle tier manages the MapReduce system and HDFS. The next layer is the layer that provides the interface for data analysis. The interface layer can be integrated with other languages such as C, C++, Python and others.

Setting up RHadoop is not a difficult task, although RHadoop works with other R packages. Working with RHadoop involves installing the R and RHadoop packages on each DataNode of the Hadoop cluster. The general view of the script that Rhadoop uses to implement the MapReduce procedure can be represented at Listing 1.

First, the rmr2 library is loaded, and then the map function that you specify takes a (key, value) pair as input parameters. The reduce function is called with a key and a list of values as arguments for each unique map key. After that, mapreduce is launched for execution .

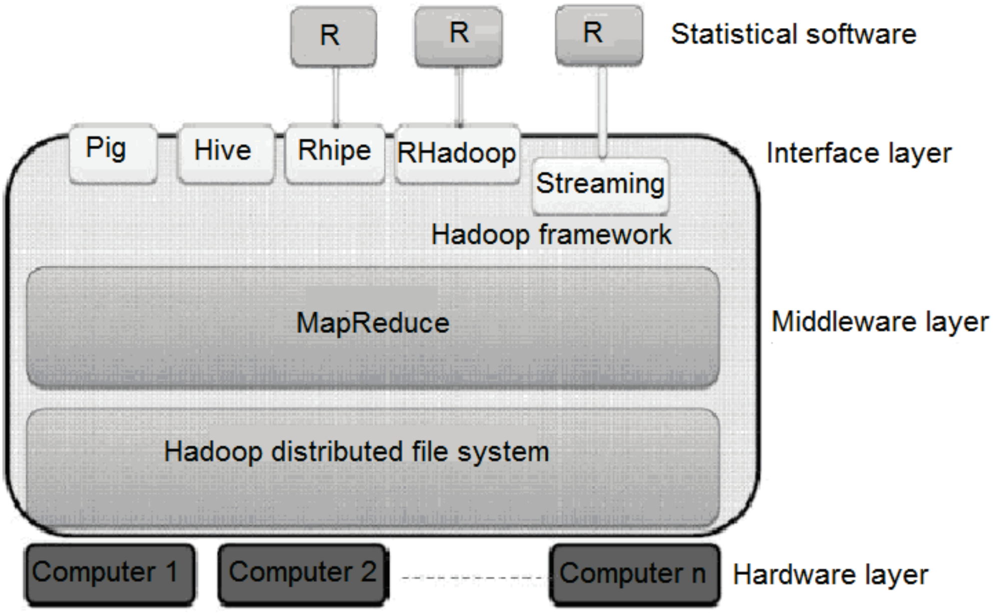

Figure 3 – Architecture of RHadoop and Hadoop[13]

Listing 1. Implementation of the MapReduce procedure in RHadoop

```
library(rmr2)
map<-function(k,v) { …}
reduce<-function(k,vv) { …}
result<- function (input, output=NULL) {
mapreduce(input = "data.txt", output = "output",
   input.format = inputformat , map = map, reduce=reduce)
```

## 4. POSSIBILITIES OF USING HADOOP FOR GEOSPATIAL DATA ANALYSIS

Hadoop can be used in many applications involving large amounts of geospatial data as well as spatiotemporal analysis.

Economides et al presented a new geospatial database management system based on the Hadoop framework for the efficient processing of complex spatial queries with high performance. For fast query processing for a typical database, a spatial index, in particular R-Tree, is used. The tree splits the space into many nested rectangles. Each object has its bounding rectangle (MBR, minimum bounding rectangle). In this case, the MBR of a node is a rectangle describing the MBR of child nodes/objects [14].

Cary and co-authors proposed a variant of using the MapReduce procedure for processing digital aerial photographs, conducting further research, and storing image quality characteristics as metadata. The main result of this work was a demonstration of the use of MapReduce for the parallel processing of a large amount of raster data [15].

Spatiotemporal data is one type of spatial data that can be very large, causing problems in analysis. Such a data structure is often encountered, for example, in mapping and predicting natural disasters [16].

There are many examples of applications that operate on rich geospatial data. For example, a biomass monitoring system involves working with time series of high-resolution satellite images. Data from sensors (in particular, MODIS) are organized into tiles of 4800×4800 pixels, and Landsat space images have an average dimension of 8000×8000 pixels. In all these examples, the use of Hadoop for data processing has fully justified itself [17].

Another example would be looking for specific patterns in geospatial data. Such methods work well for the thematic classification of images from medium to very high resolution (pixel size 5 meters or less). Such image accuracy and the use of geospatial data analysis methods together with parallel computing algorithms make it possible to solve many practical problems [18].

## CONCLUSIONS

The main problems associated with the processing of Big Data, in particular arrays containing geospatial data, are analyzed.

The Hadoop platform is considered one of the basic approaches to the analysis of large data arrays, the features of its architecture and the principle of operation of the MapReduce algorithm.

The possibilities of integrating Hadoop with other programming languages are described. In particular, the interaction of Hadoop with R and the general approach to the implementation of the MapReduce procedure in the R environment is characterized.

The potential possibilities of using the Hadoop platform for solving practical problems in the process of analyzing geospatial and spatiotemporal data are analyzed.

Currently, the main task of using parallel processing of large arrays of geospatial data is to speed up work with a large amount of data of different formats, which will involve processing the entire image instead of extracting a "region of interest", as is usually the case. In particular, such a task can be the generation of interactive thematic maps on the fly based on constantly updated input data.

## REFERENCES

1. Li DR Theory and Application of Spatial Data Mining (fisrt edition) / DR Li, SLWang, DY Li. - Beijing: Science Press, 2006. - 344 p.
2. Roberts, D.R.; Bahn, V.; Ciuti, S.; Boyce, M.S.; Elith, J.; Guillera-Arroita, G.; Hauenstein, S.; Lahoz-Monfort, J.J.; Schröder, B.; Thuiller, W.; et al. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. Ecography 2017, 40, 913–929.
3. Jhummarwala A. Parallel and Distributed GIS for Processing Geo-data: An Overview / A. Jhummarwala, MB Potdar, P. Chauhan // International Journal of Computer Applications. - 2014. - Vol. 106.–No.16. - R. 9-16.
4. Guhaniyogi R, Banerjee S. Multivariate spatial meta kriging. Stat Probab Lett. 2019;144:3–8.
5. Kousar H, Babu BP. Multi-Agent based MapReduce Model for Efficient Utilization of System Resources. Indones JElectr Eng Comput sci. 2018;11(2):504–514.
6. Grossner K. Defining a digital earth system. / K. Grossner, M. Goodchild, K. Clarke // Transactions in GIS. - 2008. - Vol. 12. - No 1. - R. 145-160.
7. Zhang L, Datta A, Banerjee S. Practical Bayesian modeling and inference for massive spatial data sets on modest computing environments. Stat Anal Data Min. 2019;12(3):197–209.
8. Lee XJ, Hainy M, McKeone JP, Drovandi CC, Pettitt AN. ABC model selection for spatial extremes models applied to South Australian maximum temperature data. Comput Stat Data Anal. 2018;128:128–144.
9. Izbicki R, Lee AB, Pospisil T. ABC–CDE: Toward Approximate Bayesian Computation With Complex HighDimensional Data and Limited Simulations. J Comput Graph Stat. 2019;p. 1–20.
10. White T. Hadoop: Definitive Guide. – 3nd edition. - Sebastopol: O'Reilly Media, 2012. - 688 p.
11. Holmes A. Hadoop in practice 2nd edition / A. Holmes. - New Jersey: Manning Publications, 2014. - 512 p.
12. Prajapati V. Big data analysis with R and Hadoop / V. Prajapati. – Birmingham: Pakt Publishing. - 2013. - 238
13. Oancea B. Integrative R and Hadoop for Great Data Analysis / B. Oancea, RM Dragoescu // Romanian Statistical Review. - 2014. - Vol. 2, no. 2 - R. 83-94.
14. Mazin A. Geo-book Big Data Mining Techniques / A. Mazin, A. Jhummarwala, MB Potdar // International Journal of Computer Applications. - 2016. - Vol. 135th – No.16. - R. 9-16.
15. CaryA. Cary, Z. Sun, V. Christidis, N. Rishe // Scientific and statistical database management Conference. A. Experiences on processing patial data with mapreduce. - 2009. - R. 302-319.
16. Vatsavai RR . Daily transit time in the era of big short-term data: algorithms and applications / RR Vatsavai, A. Ganguly, V. Chandola, A. Stefanidis, S. Klasky, S. Shekhar // 1st ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data. - 2012. - R. 1-10.
17. Yin H.-M. Modeling for geospatial database of national fundamental geographic information / H.-M. Yin, S.-W. Su // Geoscience and Remote Sensing Symposium. - 2006. - R. 865 - 868.
18. Chang BR Development of multiple big data analytics Platforms with Rapid Response / B.R. Chang, Y.-D. Lee, Liao P.-N. // Hindawi Scientific Programming - 2017. - Vol. 1, no. 12 - P.143-155.

19. Olexander N. Romanyuk, and etc. "A function-based approach to real-time visualization using graphics processing units", Proc. SPIE 11581, Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments 2020, 115810E (14 October 2020).
20. L.I. Timchenko, N.I. Kokriatskaia, S.V. Pavlov, and etc. "Q-processors for real-time image processing", Proc. SPIE 11581, Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments 2020, 115810F (14 October 2020).
21. Intellectual Technologies in Medical Diagnosis, Treatment and Rehabilitation: monograph / [S. In Pavlov, O.G. Avrunin, S.M. Zlepko, E.V. Bodyanskyi, etc.]; edited by S. Pavlov, O. Avrunin. - Vinnytsia: PP "TD "Edelveiss and K", 2019. -260 p. ISBN 978-617-7237-59-3
22. Intelligent Technologies of Computer Planning and Modeling in Medical Diagnosis, Treatment and Rehabilitation: monograph // edited by S.V. Pavlov, O.G. Avrunin, O.V. Hrushko - Zhytomyr: "Euro-Volyn" PE, 2021. - 202 p. ISBN 978-617-7992-15-7.

Надійшла до редакції 5.12.2022р.

**TALAKH MARIA –** Ph.D., assistant professor of Computer Science Department, Yuriy Fedkovich Chernivtsi National University, Chernivtsi, Ukraine, ***e-mail: m.talah@chnu.edu.ua***

**TOMKA YURIY –** Ph.D., assistant professor of Computer Science Department, Yuriy Fedkovich Chernivtsi National University, Chernivtsi, Ukraine, ***e-mail: y.tomka@chnu.edu.ua***

**USHENKO YURIY -**D.Sc., Professor of Computer Science Department, Yuriy Fedkovich Chernivtsi National University, Chernivtsi, Ukraine, ***e-mail: y.ushenko@chnu.edu.ua***

**SOLTYS IRYNA** – Ph.D., Assistant Professor of Optics and Publishing Department, Yuriy Fedkovich Chernivtsi National University, Chernivtsi, Ukraine, ***e-mail: i.soltys@chnu.edu.ua***