A Financial Transaction Methods Based on MapReduce Technology and Blockchain

Nianbin Wan
School of software Engineering
Xiamen University of Technology
Xiamen, China
546765869@qq.com

Yuliang Liu
Department of computer science and Engineering
Guangzhou College of Technology and Business
Guangzhou, China
l.y.shan615@163.com

Weidong Xiao
School of software Engineering
Xiamen University of Technology
Xiamen, China
xiaoweidong@xmut.edu.cn

2020 3rd International Conference on Smart BlockChain (SmartBlock) | 978-1-6654-4073-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/SMARTBLOCK52591.2020.00027

***Abstract*—With the widespread application of mobile internet 5G and blockchain technology, the research and application of blockchain big data analysis and mining has been paid more and more attention by enterprises, universities and research institutions. The use of MapReduce technology to analyze and explore financial transaction data such as banking, insurance, securities, funds, futures and foreign exchange has strong application requirements and academic research significance. This paper first analyzes the characteristics of MapReduce and technology and financial transaction, and then introduces the related work of MapReduce and financial transaction. Moreover, through analysis and comparison of MapReduce and financial transaction technologies, it discusses the financial data transaction method of MapReduce technology. Finally, from the perspective of new technologies, this paper also puts forward some valuable prospects and suggestions for the sustainable development of MapReduce technology in financial transaction methods.**

Keywords—MapReduce; financial transaction; blockchain; big data analysis; data mining

I. INTRODUCTION

Traditional data analysis technology has not satisfied the current requirements of intensive mass data processing. Cloud computing is a new service model that provides resources on demand rental and is a new type of internet data center business. According to the data in the computer and storage system that needs to be accessed, cloud computing can centralize the computing resources in the network, virtualize it into a resource pool, and use specific software to realize automation and intelligence, so that various computing source can work together. MapReduce provided by Google Labs analyzes the parallel computing model of big data sets, forming a "cluster revolution" in the industry. MapReduce is a cluster-based computing platform and a computing framework that simplifies distributed programming. It has the following characteristics:

(1) Easy to program. It simply realizes some interfaces to complete a distributed program. This distributed program can be distributed to a large number of inexpensive PC machine to run.

(2) Good scalability. It can expand the computing power through adding machines.

(3) High fault tolerance, MapReduce has high fault tolerance and can be deployed on cheap PC machines. When a machine crashes, MapReduce can transfer the computing task to another node to run. The entire process does not require human involvement and is completed by the system.

(4) Wide application range. It can satisfy the offline process of massive data and computing task in various fields.

In recent years, in the banking industry, various banks have determined the purchasing tendency and purchasing power of customer financing products, distinguished the repayment ability of credit users, assessed bankruptcy risks, and reduced bad debt rates through comprehensive grasp and analysis potential customer information [1]. In the insurance industry, a large number of data is used to analyze the probability of risk occurrence of policyholders; In the securities industry, big data technology is used to analyze the market conditions and trends of the stock. In the fund industry, fund companies use big data technology to mine economically valuable data and find suitable investment objects [2]. Blockchain has attracted great attention and attention in the academic and industrial areas, due to its characteristics of decentralization and high security, attracting the financial industry to invest and make research on blockchain. However, the wide application of MapReduce technology in the field of big data analysis and mining has also contributed the rapid development of financial transaction systems. The analysis and application of financial transaction data has become a research hotspot in the field of big data analysis and mining. In particular. Using MapReduce technology to analyze and mine financial transaction data such as banking, insurance, securities, fund, etc. has strong application requirements and research significance [3].

This paper is structured as follows. Section I introduces the background of MapReduce technology, and discusses the research status of MapReduce technology and financial transaction methods in Section II. The core technology of MapReduce are presented in Section III, Section IV introduces the financial transaction technology, and puts forward the financial transaction method based on MapReduce technology in Section V. Analyzes and compares the performance are presented in Section VI, and summarizes and prospects are presented in Section VII.

II. RESEARCH STATUS

A. MapReduce technology

In terms of processing TB and PB level data, MapReduce has become one of the most widely used parallel programming model [4-5]. The current research results of MapReduce mainly include the following aspects:

1) MapReduce parallel computing model

MapReduce parallel computing programming model can automatically execute large-scale computing task in parallel, hide the underlying implementation details, and reduce the difficulty of parallel programming through a well-defined interface and runtime support library [5]. MapReduce consists of two stages: Map/Reduce. It must transfer the intermediate data generated by Mapper to Reducer.

978-1-6654-4073-8/20/$31.00 ©2020 IEEE
DOI 10.1109/SmartBlock52591.2020.00027

Literature [6] proposed an incremental data distribution method to reduce the partition skew problem in Map and reduce. It divides the Map data into multiple smaller partitions, collects statistical information gradually in the Mapper process, divides the small partitions to Reducer, and then realizes data load balancing through small partitions scheduling and distribution, thereby improving overall parallel performance. Literature [7] proposed a method to verify the rationality of Map/Reduce process before executing it on the computer. This method constructs the basis for the rationality of Map/Reduce, automatically recognizes the standard of rationality, makes it impossible for unreasonable data to execute Map/Reduce, and improves parallel processing capabilities.

2) research on big data algorithm based on MapReduce parallel computing

At present, most of research no MapReduce technology focuses on algorithms, including big data analysis algorithms, data mining algorithm, machine learning algorithm and so on. Literature [8] aimed at the long execution time and poor clustering results of the original K-means method in MapReduce modeling, and proposed a divide-and-conquer K-means clustering method based on MapReduce, which splits the entire data set to be processed into smaller blocks and stores them in main memory of each machine. Through available machine propagation, this method clusters each block of the data set independently by the allocated machine, and uses the minimum weighted distance to determine the cluster to which the data point should be allocated, and then judges the convergence. Literature [9] proposed a massive graph structure clustering algorithm MRSCAN based on MapReduce, which is a MapReduce algorithm for computing core nodes and two merge clustering. Literature [10] researched the big data clustering algorithm based on MapReduce parallel computing, and proposed an improved big data clustering method of K-means parallel computing. Literature [11] researched the two-stage hybrid pipeline scheduling algorithm in the MapReduce system. This algorithm provides an approximate algorithm with a worst-case bound of 2-1/max{m1 ,m2 } for the uninterruptible situation of the map task. For the situation that the map task can be divided arbitrarily, the approximate algorithms H(2,J) and H(2,L) based no Johnson rule and LPT rule are respectively given, and the worst-case bounds of these two algorithms are respectively 2-1/m2 and 2.

3) Parallel machine scheduling optimization based on MapReduce model

In terms of MapReduce mode optimization, literature [12] researched parallel machine scheduling optimization based on MapReduce model with preparation time, and improved the performance by constructing a mixed integer programming model and designing a differential mutation strategy and an improved sine-cosine algorithm with a dimensional levy disturbance mechanism. Literature [13] studied parallel machine scheduling optimization with task partition based on MapReduce model, constructed a mixed integer programming model by optimizing the target, and designed an improved whale optimization algorithm solution model using differential mutation strategy and dimensional angle perturbation mechanism to improve the effect. Literature [14] proposed an iterative data equalization partition strategy, which subdivides the data blocks to be processed by each Mapper node and processed iteratively. Then, according to the micro-partition allocation result of the iterated round, the micro-partition allocation scheme of the current iteration is determined, and the data is gradually balanced and partitioned, and the overall performance of MapReduce is improved. By continuously adjusting the data skew generated by previous iterations, the data is gradually balanced and partitioned, and the overall performance of MapReduce is improved.

B. Research status of financial transaction methods

In any market, all financial systems will conduct transactions, which realizes the transfer of funds from the surplus to the insufficient, so that the principal or interest of financial assets can be exchanged. This exchange can be conducted directly by both parties or indirectly through an intermediary. The process of financial transaction can product massive amounts of financial data, and we can use big data platforms to support research on financial transaction services, such as the study of financial transaction mechanisms in the commodity supply chain, the research on financial transaction methods based on deep reinforcement learning algorithms, and the study of blockchain-based financial transaction methods. Literature [15] solved the problem of commercial secret protection in financial transaction under big data by expansive interpretations of ‘’secrecy’’, learning from copyright protection methods, changing the distribution of burden of proof, and increasing penalties. Literature [1] studied the relationship between Internet information and transaction volume based on the perspective of financial big data, and developed quantitative transaction based on financial big data by analyzing the dynamic relationship between individual stocks' daily transaction volume, daily return rate and Baidu media index during the sample period. Literature [3, 4] studied financial transaction systems based on reinforcement learning, and summarized many research results in adaptive algorithms, transaction strategies, transaction systems, etc. Literature [16] focuses on the characteristics of two types of abnormal transactions, such as suspension of favorable insider trading and continuous tracking of "rat warehouse", and constructs an analysis model of abnormal transactions to identify suspicious accounts. Gai et al. [17] proposed a method based on user privacy protection and mining the relationship between energy trading volume and other information in smart grid. This method effectively selects the energy sales distribution of sellers, and then improves the efficiency of energy sales. Experiments show that this method is effective. Pan et al. [18] proposed a design scheme of asset securitization business system based on blockchain technology, which has been proved to have good performance by experiments.

III. MAPREDUCE CORE TECHNOLOGY ANALYSIS

MapReduce is a cluster-based computing platform, and a computing framework that simplifies distributed programming. It uses both Map and Reduce function to implement basic parallel computing task, and provides abstract operations and parallel programming interfaces to easily complete programming and computing process of large-scale data. MapReduce technology process is divided into two parts: Map task processing and Reduce task processing. In the process of Map task processing, the system first reads the HDFS input file, parses each line into a key-value pair, such as <Key1,Value1>. Then system calls the Map function for each key-value pair to convert it into a new

key-value pair, Data in different partitions can be sorted and grouped by Key value. Different values of the same key will be placed in the same set, such as <Key2,{Value2,…}>, and the grouped data can also be regulated. In the process of Reduce task processing, the system first copies the output key-value pairs processed by multiple Map tasks to different Reduce Node through the network according to different partitions, and then calls Reduce() function to merge and sort, and finally converts it into a new key-value pair, such as <Key3,Value3>. The results is output to the output file, as shown in Fig.1.

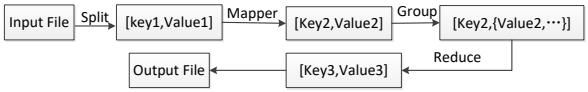

Fig. 1. Working principle of MapReduce.

According to the working principle described above, we can analyze the specific data processing process of MapReduce through its internal logic, as shown in Fig. 2. Because the data in the HDFS input file is stored in block form, each block is converted into corresponding split according to the InputFormat. A split can correspond to one block or multiple blocks, such as split 0,split 1,…,split n. At this time, the data of each split is read through InputFormat, and the data is parsed into several key-value pairs in the form of <key, value>. Then the Mapper function processes these data, and each word reads can get a key-value pair (such as <a,1>). When a split is read, a set of key-value pairs (such as{<a,1>,<c,1>,…}) can be obtained. Then the key-value pair collection is partitioned by Partitioner, and the data after partitioning by Partitioner is stored in the local disk, and then it will be handed over to the system to select which Reducer for processing.

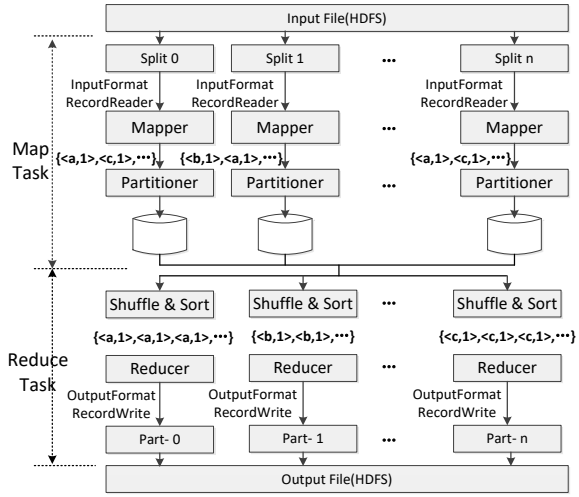

Fig.2. MapReduce data processing process.

Shuffle, Sort, and Reduce are mainly responsible for processing in the Reduce task phase. Each Reduce reads the corresponding data through Shuffle, and then sorts it by key. The sorting process is based on the mixed mode of memory and disk. It takes multiple Merge to complete the sorting, and finally get <key,(list of values)> or {<key,value>,<key,value>,...}, that is List of key names and values. <key,(list of values)> is sent to the reduce() function of the Reducer to perform calculations. Each <key,(list of values)> calls the reduce() function once, and the result is written into HDFS according to the output type specified by the user, and an output file in the form of part-* is generated.

IV. Financial Transaction Technical Analysis

Financial transaction technical analysis refers to the market behavior as the research object, to judge the financial market trend and follow the cyclical changes of the trend to make the stock and all financial derivatives transaction decision-making methods. It is a rational analysis and the conclusion is relatively objective. The various signals reflected in the charts and technical indexes cannot be changed by people's subjective wishes, and are applicable to various financial transaction media such as stocks, futures, and exchange rates. Financial transaction technical analysis includes graphical analysis and technical index analysis.

A. Graphical analysis

Graphical analysis is a technological analysis method that predicts price trends by analyzing various graphics that reflect changes in the price of financial transaction media (such as stock prices). In graphical analysis, time is used as the horizontal axis and price or index is used to as the vertical axis. The basic graphics include line chart, column chart, bar chart, point and figure chart, K-line chart, etc. Among them, the K-line chart is often used, which focuses on revealing the contrast between the strengths and weaknesses of multiple forces. The K-line chart uses Yin, Yang, upper and lower shadow lines to indicate the resistance, support, and intensive transaction areas of price fluctuations, and shows different market conditions and predicts different trends through a variety of combinations (e.g. bottom combination, top combination, relay combination) and different forms (such as yang line K line, yin line K line, "cross" line, "T-word" and "inverted T-word" line, etc.).

The transaction volume chart is a graph that expresses the transaction volume in the form of a bar chart, such as the number of shares transaction volume chart and the amount of transaction volume chart. The number of shares transaction volume chart can reflect the supply and demand status of the market stocks. The amount of transaction volume chart reflects the capital supply status and capital flow of the market, and is mainly used for market analysis.

B. Technical index analysis

Technical index analysis observes and analyzes various transaction data, and obtains the judgment of future trends through mathematical models. Technical indexes include trending indexes, overbought and oversold indexes, popularity indexes, pressure support indexes, etc. Trend indexes, such as Moving Average Index (MA), Moving Average Convergence and Divergence (MACD), Directional Movement Index (DMI), etc. send an operational signal based on the upward or downward movement of at least two trend lines or down and the intersection of the trend line. MA is a moving average index that uses statistical processing to arithmetic average the stock prices of several days, and then connect the arithmetic average prices obtained daily with lines. MA reflects a technical index of the current average holding cost. MACD eliminates abnormal violent fluctuations through smooth calculations, and uses the signs of aggregation and dispersion of short-term (fast) moving averages and long-term (slow) moving averages to determine the timing of buying and selling. Based on the kinetic energy of the highest or lowest price newly generated during the rise

and fall of the transaction price, DMI analyzes the power changes of the long and short parties to find the equilibrium point of the power of the buyer and the seller and predict the future direction of the price.

Overbought and oversold indexes are based on a certain value as the criterion for judging overbought and oversold, and send buying and selling signals based on the value of the moving position of the trend line, such as KDJ, RSI, W%R, etc. KDJ is a stochastic indicator, which mainly studies the relationship between high-low prices and closing prices. The K-line and D-line reflect the strength of price changes and the phenomenon of oversold and oversold at different rates. RSI is the Relative Strength Index, which shows the strength of the market by comparing the statistical results of the average closing increase and decrease over a period of time. At the same time, it analyzes the strength and intentions of buyers and sellers in the market to make predictions about future market trends. W%R is the William Overbought/Oversold Index. It studies the relationship between high price, low price and closing price over a period of time, and applies the swing principle to determine whether it is at a high or low point and whether it is overbought or oversold.

V. FINANCIAL TRANSACTION METHOD BASED ON MAPREDUCE TECHNOLOGY

MapReduce can organize clusters to process large-scale data, which requires processes such as data preprocessing, association rule mining, classification, clustering, anomaly detection, evolution analysis, specific group analysis, etc. MapReduce provides a policy-driven workload scheduler for enterprise-level financial transaction applications, provides resource priority for preemptive jobs, and fairly schedules Map jobs and Reducer jobs. In the MapReduce process, the type of the input data source file system can be different from the output data source file system, and it supports extraction, conversion, and loading of workflow logic.

A. Data Preprocessing

Before performing the main processing of Map/Reduce on the data, a series of data preprocessing work such as necessary cleaning, integration, conversion, discretization, specification, feature selection and extraction are performed on the original data. MapReduce technology needs to sample data before processing financial transactions. M samples can be randomly selected from the total number of known samples n. These samples are generally stream data. If there are missing values in the data, we can use methods such as ignoring tuples and manually filling in missing values for processing, and then use deviation detection and deviation correction methods to clean the data and store it in the data warehouse. If the data sources come from different databases or file systems, it is also necessary to integrate multiple data sources and eliminate redundant data. After the above operations, the data source files are generated.

B. Map Stage

MapReduce takes the data source file as input data and cuts it into several *InputSplits* to achieve parallelization [20]. This process requires the use of the *InputFormat* class, and *getSplits* and *createRecordReader* methods. The *getSplits*() method returns a collection of *InputSplit* objects. It needs to verify whether the input file is splittable, the size of the block when the file is stored, and the size of the file. An *InputSplit* will be assigned to an independent MapTask. The *createRecordReader*() method returns the RecordReader object of <key, value>. The *getSplits*() method of the implementation class of the subclass *FileInputFormat* of the *InputFormat* class is responsible for splitting the input file and generating several input fragments *FileSplit*. Then call the *createRecordReader*() method of the *FileInputFormat* subclass to get the *lineRecordReader object*, which parses the key-value pair <key, value> from the *InputSplit* shard, and calls map() for each <key, value> for processing. Then each <key, value> calls map() for processing. To sum it up, this stage is to filter, read, slice, read out the key-value pair <key, value> of the input file, and then hand it over to the Mapper class for processing. The pseudo code of the Map phase of MapReduce technology processing financial transaction data is as follows:

```
long minSize=Math.max(getFormatMinSplitSize(),
                      getMinSplitSize(jobTask));
long maxSize=getMaxSplitSize(jobTask);
InputSplit  inputSplit=new FileSplit(file,start,length,hostblock[]);
 // The splits collection is used to store fragmentation results
List<InputSplit> splits=new ArrayList<InputSplit>();
List<FileStatus> files=listStatus(jobTask);
for(FileStatus file:files){
     path path=file.getPath();//Gets file path
     FileSystem fs=path.getFileSystem(jobTask.getConfiguration());
     long  fileLen=file.getLen();
 // Gets all the block information collection of the file
//{hostname,offset,length}
BlockLocation[] blockLocation=fs.getFileBlockLocations(file,0,fileLen);
// Loop fragmentation
while(remaining/splitSize>split_slop){
     Reads data into the splits collection; }
while(context.nextKeyValue()){
     map(context.getCurrentKey(),context.getCurrentValue(),context); }
```

C. Reduce Stage

The processing process of the Reduce phase is based on the result of the Map phase, aggregating the equivalent classes with the same key value to form a values set, and outputting a new key-value pair <key, M> after completing the corresponding operation required by the user [18,19]. This stage requires the use of the abstract class Reducer, which defines cleanup(), reduce(), run(), setup() and other methods. The reduce() method prototype is:

Protected void reduce(KEYIN key, Iterable <VALUEIN> values, Context context) ;

The function of this method is to merge the input <key, value>, which is to merge different values under the same primary key into a list set to generate a new key-value pair <key, M> set. Then analyze and mine the new key-value pair <key, M>. The main algorithms include classification algorithms, association algorithms, clustering algorithms, prediction sequence algorithms, estimation and prediction algorithms, description and visualization algorithms, etc. Classification algorithms include decision tree classification algorithm, Bayesian classification algorithm, neural network algorithm, K-Nearest neighbor algorithm, support vector machine algorithm, etc. Clustering algorithms include K-means clustering algorithm, K-center point clustering algorithm, self-organizing neural network clustering algorithm, density-based spatial clustering of applications with noise algorithm, etc. Common prediction sequence algorithms include moving average algorithm, exponential smoothing algorithm, linear regression algorithm, grey prediction algorithm, etc. Big data analysis of financial transactions mainly uses classification, clustering, and

prediction algorithms. According to different fields of data and different data analysis and mining requirements, a suitable data model can be established and then a suitable algorithm can be selected.

VI. Performance Comparison and Analysis

The underlying platform used by the MapReduce-based financial transaction method system is built on the Linux operating system. The core functions of the platform are implemented in Java language and can be developed and deployed across platforms. The performance comparison between MapReduce and distributed database management system is shown in TABLE I.

TABLE I. The performance comparison between MapReduce and distributed database management system

System Index	Configuration parameter	Scalability	Effectiveness	Execution performance
MapReduce	Concise	Better	Higher	Better
Distributed database management system	Complex	General	Slow	Poor

As can be seen from the above table, MapReduce can automatically optimize program parameters, and adopt dynamic configuration and late binding, which is relatively concise. Distributed database management systems need to process relational databases, and there are a large number of complex SQL commands. The execution of complex queries needs to be optimized, and the execution efficiency of testing massive data is low. This paper compares the performance of 10 nodes, 50 nodes, 100 nodes, 200 nodes, 500 nodes, and 1000 nodes in a virtual simulation environment to process 100MB, 500MB, 1GB, 10GB, and 100GB data in a cluster environment. This paper compares the performance of 10 nodes, 50 nodes, 100 nodes, 200 nodes, 500 nodes and 1000 nodes in a cluster environment to process 100MB, 500MB, 1GB, 10GB, and 100GB data. Experimental results show that MapReduce can achieve node load balancing and distributed storage of data, and improve the data processing performance of MapReduce programs.

VII. Summary and Outlook

It is a brave exploration and innovation to use MapReduce technology to analyze and mine massive data in the field of financial transactions. The advantages of MapReduce technology can promote the development of financial transactions. This paper analyzes the core technology of MapReduce and financial transaction technology, and proposes a data financial transaction method based on MapReduce. And through the MapReduce parallel programming model, the large-scale underlying data infrastructure is designed. In the next step, we will study the clustering algorithm for processing cloud blockchain financial transactions based on MapReduce and the algorithm for mining cloud blockchain financial transactions based on MapReduce.

Acknowledgment

This work is supported by Planning subject for the 13th five-year plan of Educational Science in Fujian Province (Grant No. 2020CG0851).

References

[1] K. Gai, Y. Wu, L. Zhu, Z. Zhang, M. Qiu, "Differential Privacy-Based Blockchain for Industrial Internet-of-Things". IEEE Trans. Ind. Informatics 16(6): 4156-4165, 2020.

[2] H. Nie, N. Li: Protection of Trade Secrets in Financial Transactions under Big Data Background: Dilemma and Countermeasures. Science Technology and Law, (1): 31-37, 2020.

[3] T. Liang, X. Yang, L. Wang, Z. Han: Review on financial trading system based on reinforcement learning. Journal of Software,30(3): 845-864, 2019.

[4] W. Liang, W. Huang, J. Long, K. Zhang, K.-C. Li and D. Zhang: Deep Reinforcement Learning for Resource Protection and Real-time Detection in IoT Environment, in IEEE Internet of Things Journal, 7(7): 6392 - 6401, 2020.

[5] Li, J., Cui, J., Wang, D., Yan, L., Huang, Y.: Survey of MapReduce Parallel Programming Model. electronic journal, 39(11): 2635-2642, 2011.

[6] Z. Wang, Q. Chen, B. Suo, W. Pan: Reducing partition skew on MapReduce: an incremental allocation approach. Frontiers of Computer Science, (5): 960-975, 2019.

[7] Z. Ding, D. Guo, X. Chen, J. Chen: MapReduce rationality verification based on object Petri net. Journal of Systems Engineering and Electronics, 30(5): 861-874, 2019.

[8] Y. Zang, Y. Xi: Divide and conquer k-means clustering method based on MapReduce. Computer Engineering and Design, 41(5): 1345-1351, 2020.

[9] W. Zhang, Z. Li, Z. Li, R. Li, Y Liu.: MapReduce-Based Graph Structural Clustering Algorithm. Journal of Software, 29(3): 627-641, 2018.

[10] W. Zhang, L. Jiang: Parallel computation algorithm for big data clustering based on MapReduce. Application Research of Computers, 37(1): 53-56, 2020.

[11] Q. Wei, Y. Wu, Y. Jian, Y. Zhao: Algorithms for two-stage hybrid flow shop in MapReduce systems. Systems Engineering-Theory & Practice, (5): 1255-1265, 2020.

[12] J. Huang, F. Zheng, Y. Xu, M. Liu: Parallel machine scheduling with setup time in the MapReduce system. Systems Engineering-Theory & Practice, 39(1): 174-182, 2019.

[13] J. Huang, F. Zheng, Y. Xu, M. Liu: Parallel machine scheduling with splitting jobs in MapReduce system. Control and Decision, 34(7): 1514-1520, 2019.

[14] Y. Zhang, J. Jiang, J. Lu, J. Xu, G. Xiao: An Iterative Data Partitioning Strategy for MapReduce. Chinese Journal of Computers,42(8): 1873-1885, 2019.

[15] W. Liang, K.-C. Li, J. Long, X. Kui, Zomaya: An Industrial Net-work Intrusion Detection Algorithm based on Multi-Characteristic Data Clustering Optimization Model. IEEE Transactions on Industrial Informatics. 16(3): 2063-2071, 2020.

[16] S. Zhao: Discussion on Models of Abnormal Trading Models in Securities Market Based on Big Data Analytics in Financial System Auditing. Auditing Research, (5):17-21, 2018.

[17] K. Gai, Y. Wu, L. Zhu, M. Qiu, and M. Shen, "Privacy-preserving Energy Trading Using Consortium Blockchain in Smart Grid," IEEE Transactions on Industrial Informatics 15(6): 3548-3558, 2019

[18] W. Pan and M. Qiu, "Application of Blockchain in Asset-Backed Securitization," 2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), Baltimore, MD, USA, pp. 71-76,2020.

[19] J. Zhang, X. Xiong, X. Jin: Internet Information and Trading Volume: An Financial Big Data Perspective. Systems Engineering, 34(09): 1-8, 2016.

[20] H. Xu, W. Zhao, L. Meng: An Improved Text Input Method for Parallel Word Segmentation Based on Mapreduce. Computer Knowledge and Technology,2016, 12(22):171-175.

[21] Xu, H., Zhao, W., Meng, L.: An Improved Text Input Method for Parallel Word Segmentation Based on Mapreduce. Computer Knowledge and Technology,2016, 12(22):171-175.