Hindawi
Mobile Information Systems
Volume 2021, Article ID 9505249, 13 pages
https://doi.org/10.1155/2021/9505249

Research Article

Cloud Computing Storage Backup and Recovery Strategy Based on Secure IoT and Spark

Dajun Chang,[1,2] Li Li ,[1] Ying Chang,[3] and Zhangquan Qiao[1]

[1]*College of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022, Jilin, China*
[2]*School of Electrical Information, Changchun University of Architecture and Civil Engineering, Changchun 130604, Jilin, China*
[3]*School of Computer Science and Engineering, Jilin University of Architecture and Technology, Changchun 130114, Jilin, China*

Correspondence should be addressed to Li Li; ll@cust.edu.cn

Received 21 July 2021; Revised 30 September 2021; Accepted 29 October 2021; Published 23 November 2021

Academic Editor: Sang-Bing Tsai

Spatial data occupies a large proportion of the large amount of data that is constantly emerging, but a large amount of spatial data cannot be directly understood by people. Even a highly configured stand-alone computing device can hardly meet the needs of visualization processing. In order to protect the security of data and facilitate for users the search for data and recover by mistake, this paper conducts a research on cloud computing storage backup and recovery strategies based on the secure Internet of Things and Spark platform. In the method part, this article introduces the security Internet of Things, Spark, and cloud computing backup and recovery related content and proposes cluster analysis and Ullman two algorithms. In the experimental part, this article explains the experimental environment and experimental objects and designs an experiment for data recovery. In the analysis part, this article analyzes the challenge-response-verification framework, the number of data packets, the cost of calculation and communication, the choice of Spark method, the throughput of different platforms, and the iteration and cache analysis. The experimental results show that the loss rate of database 1 in the fourth node is 0.4%, 2.4%, 1.6%, and 3.2% and the loss rate of each node is less than 5%, indicating that the system can respond to applications.

1. Introduction

The Internet of Things is known as the third wave of information technology after computers and the Internet. It is currently one of the most popular scientific researches in the field of communication and information. With the emergence of IoT applications, information security issues also follow. Users who want to conduct information system security inspections and early warnings also need security services and software support and require a lot of investments in training professional teams to complete related tasks. On the other hand, there are more and more types of security products, and the product standards of suppliers are different. Users encounter difficulties when choosing security products. In such an environment, it is particularly important to provide a security architecture that focuses on the application layer.

With the development of the Internet and computer technology, the growth of network scale and application and economic demand has given birth to a new network computing model, cloud computing, which has increasingly appeared in people's field of vision, and people have quietly entered "Cloud Era." Cloud computing organically combines a variety of technologies such as virtualization, distributed computing and storage, and diverse terminals and integrates various traditional software resources, storage, and computing resources through the network to form a "supercomputer" with a huge resource pool, that is, the cloud. In the cloud, people can use the various software and hardware resources in the cloud as convenient as tap water, only need to drink like tap water, without the need for traditional storage or maintenance of this water, and only drink what they drink. Resources are paid on demand.

Based on the secure IoT and Spark cloud computing storage backup and recovery strategies, many scholars at home and abroad have conducted related research. Kumar believes that the Internet of Things is an emerging technology that can connect everyday objects to the Internet. The Internet of Things technology does provide an interface for different technologies. New applications can be realized with the help of embedded physical devices with intelligent thinking capabilities, playing an important role in connecting to the Internet. The IoT gateway must be smart enough to perform the collected operations based on their respective applications. The author proposed the gateway Pi, an IoT smart security gateway framework integrated with the Raspberry Pi board. This proposal does make the IoT gateway a smart thing, and it runs like a normal PC. In addition to native gateway functions, this article also emphasizes the security of IoT gateways. The author proposed three measures to provide security for the IoT gateway, making the gateway a firewall and using gateway Pi to implement a cost-effective and reliable IoT architecture for smart irrigation. In this study, the author conducted research on IoT gateways, aiming at improving the security of the Internet of Things, but the author did not draw the relevant framework diagram [1]. Mary A A believes that, in cloud computing, from the perspective of reliable storage of sensitive data and storage service quality, the storage of massive amounts of information is a very challenging task. Among different cloud security issues, data disaster tolerance is the most critical issue. The motivation of recovery technology is to help users collect data from any backup server when the server loses data and cannot provide data to the user. To achieve this goal, many types of research have developed different technologies. Therefore, the author proposed a data disaster tolerance process using the opposition group search optimizer (OGSO) algorithm, mainly to avoid disasters in the cloud. The proposed data recovery process consists of four modules: (1) file upload module, (2) copy generation module, (3) data backup module, and (4) disaster recovery module. First, the author split the data into multiple files and uploaded the files to the corresponding virtual machine using the OGSO algorithm. Then, a copy is generated based on the bandwidth of each file. The copy is mainly used for data backup strategy. Finally, files based on user queries are backed up and retrieved based on copies. Experimental results show that the proposed OGSO-based data disaster recovery process is better than other methods. The author conducted research on storage issues in cloud computing but did not discuss security issues [2]. Interlandi M debugging data processing logic in a data-intensive scalable computing (DISC) system is a difficult and time-consuming task. Today's DISC system provides very few tools for debugging programs, so programmers spend countless hours collecting evidence (e.g., from log files) and performing trial and error debugging. To help with this work, the author built Titian, a library that enables data source tracking data through transformations in Apache Spark. Data scientists who use the Titian Spark extension will be able to quickly identify the underlying cause of potential errors or abnormal results in the input data. Titian is directly built into the Spark platform and provides data source support at an interactive speed of several orders of magnitude faster than alternative solutions, while having minimal impact on Spark job performance; the observed overhead for capturing data lineage rarely exceeds that of the baseline job 30% of execution time. The author conducted program debugging for Spark but did not compare these platforms with other platforms [3].

This paper proposes cloud computing storage backup and recovery strategies based on the secure Internet of Things and Spark and conducts related research. In the method part, this article introduces the security Internet of Things, Spark, and cloud computing backup and recovery related content and proposes cluster analysis and Ullman two algorithms. In the experimental part, this article explains the experimental environment and experimental objects and designs an experiment for data recovery. In the analysis part, this article analyzes the challenge-response-verification framework, the number of data packets, the cost of calculation and communication, the choice of Spark method, the throughput of different platforms, and the iteration and cache analysis. The innovation of this article is to combine the secure Internet of Things with Spark and use these two technologies to study storage backup and recovery strategies based on cloud computing, so as to maximize the value of data.

2. Cloud Computing Storage Backup and Recovery Strategy Method Based on Secure IoT and Spark

2.1. Secure Internet of Things. As a value-added application in the information network, the Internet of Things is also an extension of the special application of the communication network. The development of the Internet of Things industry involves three main elements. The first is recognition, which is a basic premise, the second is communication, which is a very important support and platform, and the third is application, which is the main goal and the ultimate goal, which fully demonstrates the Internet of Things itself. In the development and application of the Internet of Things, the requirements for technology are very high, and solutions are also an important driving force for strengthening the Internet of Things to achieve leapfrog development [4, 5].

Figure 1 shows the basic framework of the Internet of Things, which includes a comprehensive application layer, a network construction layer, a management service layer, and a perception recognition layer. Perception recognition layer: the core technology of the Internet of Things is perception technology, which is the key center of communication with the physical world and the information world. The detection layer mainly includes automatic data acquisition equipment such as radio frequency (RFID) and wireless sensors, as well as various intelligent electronic products dedicated to manual information [6]. Network construction layer: the key task of this layer is to connect the analysis and identification equipment of the lower layer to the Internet so that the upper layer can access the application. The foundation of the Internet of Things is the

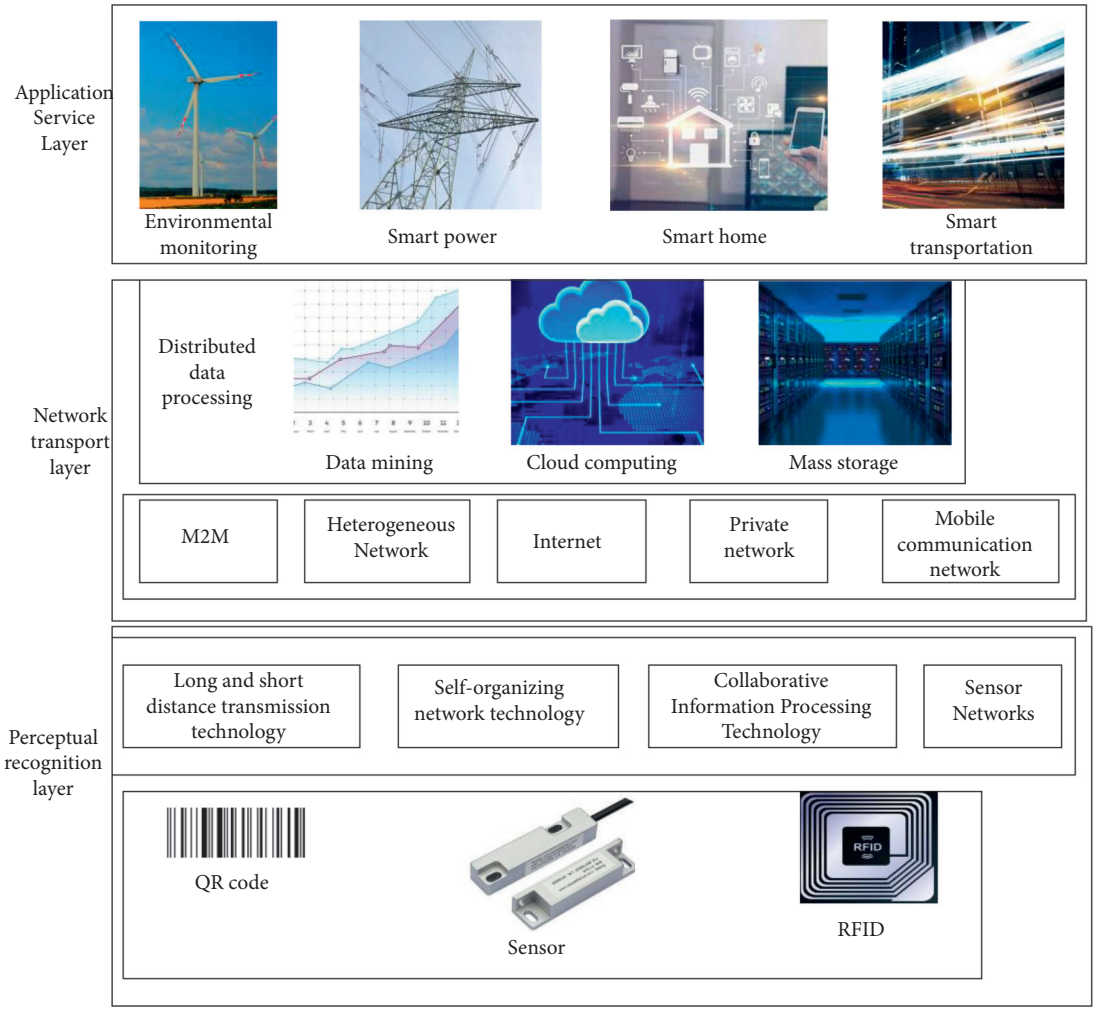

Figure 1: IoT system structure (pictures from Baidu picture).

Internet and the next-generation Internet. Various wireless networks have been providing Internet services, relying on powerful computers and mass storage technology to collect various amounts of data [7]. Comprehensive application layer: with the continuous advancement of computers, online applications are also undergoing earthquake-like changes. File transfer and e-mail are the keys to early data services. Since then, this kind of data service has become more widely used in user-centric network applications such as video images, online games, and social network [8].

At the same time, due to the large number of terminal nodes of the Internet of Things, the Internet of Things itself has some shortcomings, such as the interconnection and intercommunication of the Internet of Things systems and communication through the network. Despite any security measures, the system hardly provides any control and can trigger various network attacks. In addition, the sensing nodes in the Internet of Things also have low mobility [9]. Therefore, for these safety hazards, a secure Internet of Things is proposed. Figure 2 shows the safety supervision service system.

2.2. Spark. Spark is a fast and comprehensive large processing machine. In the case of sufficient memory, Spark runs 100 times faster than Hadoop and MapReduce. Even if the memory is insufficient, the flow to disk is 10 times faster. This is because Spark supports complex DAG drivers for circuit data flow and memory. Spark is implemented in Scala. It combines the language features of object-oriented and functional programming. It can operate distributed data sets as easily as local collection objects. It has the characteristics of fast running speed, simple operation, strong versatility, and good compatibility [10, 11].

Spark can realize the comprehensive and unified management of data sets such as text or graphs with different attributes and provides a computing architecture that can process real-time data streams like ordinary data. With the help of the Spark computing framework, the computing speed of cluster applications has been significantly improved. Before this, the computing speed of computer programs on the Hadoop platform in memory was only one percent of Spark, while the computing speed of programs on HDFS is only one-tenth [12, 13].

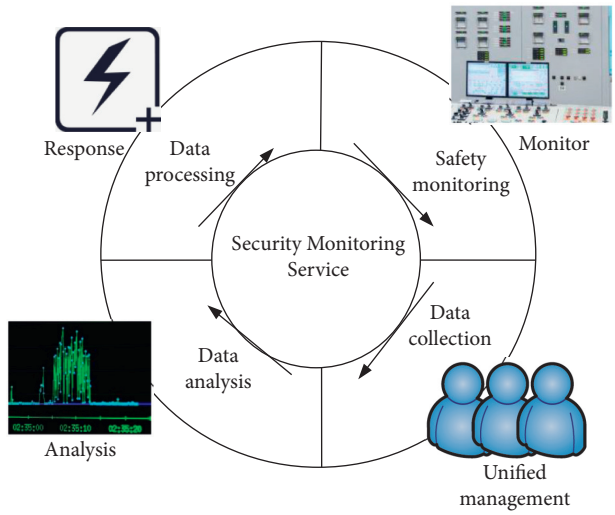

Figure 2: Security monitoring service (pictures from Baidu picture).

The specific characteristics of Spark are as follows. (1) It is fast. Sparks and Hadoop applications can run on Hadoop. Sparks can run up to 100 times faster than memory. Even on the disc the running speed has been increased by 10 times. Intermediate data can be stored in the memory instead of the disk, which reduces the time of rereading from the disk and improves the computational efficiency [14]. (2) Running Spark supports multiple languages to build applications, mainly Java, Scala, and Python, including Spark written in Scala language, with more than 80 built-in high-level operators [15]. (3) Detailed analysis: it not only provides computer functions similar to Map and Reduce, but also provides functions such as drawing [16].

In addition to Spark, there are also two architectures, Hadoop and MapReduce. As a distributed system architecture, Hadoop can be used to store and process massive amounts of data. It enables large databases to be processed through computer clusters using a simple programming model. Designed to grow from a single server to thousands of servers, each server provides a local computer and storage. The platform can be understood as a computer cluster operating system, and Spark and MapReduce are the only programming languages supported by this operating system [17, 18]. HDFS is a derivative of the file system based on all computer file systems. Table 1 is a comparison between Hadoop and Google for cloud computing systems.

As shown in Figure 3, the process map of MapReduce is an important part of the Hadoop ecosystem. MapReduce is a high-performance parallel computing platform that forms a distributed and parallel computer cluster, containing tens, hundreds, or even thousands of nodes [19].

First, the MapReduce library of the user program splits the input file into multiple copies and then starts the multiple copies of the program in the cluster. The master node then selects inactive nodes from all working nodes and assigns or reduces mapping tasks for them. After assigning the work node to the job, it starts to read the contents of the corresponding input slice. The content of the input slice is divided into key/value pairs for each row. The intermediate key/value pairs generated by the Map function are stored in a buffer in memory. The data storage location on the local disk is returned to the master node, which is responsible for transmitting the location information to the job node that runs the mitigation job. Then, it reads the saved data from the node that will execute the map task and then sorts the data. When the master node determines that the Map and Reduce tasks of all nodes are completed, the master node starts the user program and calls MapReduce in the user code to return to the editing process.

2.3. Cloud Computing Storage Backup and Recovery. In cloud computing, data storage operations are provided in the form of services, which makes the data security of cloud computing have unique characteristics: (1) User data is stored in the cloud server, and both upload and download need to go through the network, which increases the transmission process, the risk of data leakage in the medium. (2) The data is stored in a semitrusted third party; (3) cloud computing is based on a distributed network, and the computer servers are noded, and the user's data is stored in a node in the network. Above, in theory, an attacker can access its surrounding nodes through a certain node through a certain method [20].

As an extension of cloud computing and its derivative technology, cloud storage has also aroused great interest in industry, academia, and even the government. Cloud storage is an emerging storage technology. Its core is to store and manage resources on a cloud platform, enabling people to access data through the Internet in real time. The world's IT giants Microsoft, Google, Amazon, and domestic companies such as Baidu, Ali, and Tencent have done a lot of research on cloud storage and provide corresponding cloud storage platforms. Architecture diagram of mobile phone backup system based on cloud storage is shown in Figure 4.

The redundancy of user data in the cloud storage system will increase the storage pressure of the cloud storage server, cause network transmission delays, and increase remote bandwidth pressure. In order to reduce the large amount of

Table 1: Comparison of Hadoop cloud computing system and Google cloud computing system.

Hadoop cloud computing system	Google cloud computing system
Hadoop HDFS	Google GFS
Hadoop MapReduce	Google MapReduce
Hadoop HBase	Google Bigtable
Hadoop ZooKeeper	Google Chubby
Hadoop Pig	Google Sawzall

redundant data in cloud storage servers and save storage space and network bandwidth to the utmost extent, data deduplication technology has gradually become a hot research topic in recent years. In addition, backup and data recovery of these existing data are also important. The cloud storage system puts pressure on it.

Compared with cloud computing, cloud storage's security issues are more focused on data issues. Data distribution in the cloud node transmission process may cause security risks. When internal attacks or illegal operations by employees take place, data leakage and loss may occur. When the system is under attack, user information may be leaked from it. Compared with traditional storage, the new features of cloud storage have brought many new security issues, especially the need to ensure the confidentiality and security of stored data, as well as the integrity and availability of data [21, 22].

2.4. Cluster Analysis. Clustering analysis is to divide a given data set into multiple classes or clusters. The goal is that objects in the same cluster have high similarity, and objects in different clusters have high dissimilarity [23].

Assuming that a data set$B = \{b_1, b_2, \ldots, b_n\}$, $b_m (m = 1, 2, \ldots, n)$is a data object, the data set is divided into l subsets according to the similarity between the data objects, and these subsets meet the following conditions:

$$\begin{cases} A_m \neq \varnothing, & m = 1, 2, \ldots, l, \\ \cup_{m=1}^{l} A_m = I, & m = 1, 2, \ldots, l, \\ A_m \cap A_n = \varnothing, & m \neq n,\ m,\ n = 1, 2, \ldots, l. \end{cases} \tag{1}$$

2.4.1. Data Structure. Clustering is a hierarchical cluster, that is, a collection of nested clusters similar to a tree structure, and a data matrix is obtained through structured data storage. The data matrix represents the attribute values of all data objects in the data set, such as

$$I_{t\times q} = \begin{bmatrix} I_{1a} & \cdots & I_{1b} & \cdots & I_{1c} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ I_{xa} & \cdots & I_{xb} & \cdots & I_{xc} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ I_{ta} & \cdots & I_{tb} & \cdots & I_{tc} \end{bmatrix}, \tag{2}$$

where t indicates that there are t data objects in the data set, and each object has q different attributes.

The value of each element in the dissimilarity matrix represents the difference between the two data objects, including

$$K_{t\times t} = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & \cdots \\ k(2,1) & 0 & \cdots & \cdots & \cdots \\ k(3,1) & k(3,2) & 0 & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ k(t,1) & k(t,2) & \cdots & \cdots & 0 \end{bmatrix}, \tag{3}$$

where $k(m, n)$ represents the quantized dissimilarity between objects m and n. Generally speaking, its value is a nonnegative number. The closer the two objects are to 0, the more similar the two objects are.

2.4.2. Similarity Measurement. The distance between data objects is commonly used to evaluate the similarity between objects. The higher the similarity between the data objects in the cluster and the greater the difference between the data objects and between the clusters, the better the j clustering result. The difference between data objects is usually measured by the distance between data objects. The shorter the distance, the higher the similarity. Typical similarity measures are as follows:

Euclidean distance refers to the true distance between two points in m-dimensional space [24]. The calculation formula is

$$k(m,n) = \sqrt{\left(i_{x1} - i_{y1}\right)^2 + \left(i_{x2} - i_{y2}\right)^2 + \cdots + \left(i_{xt} - i_{yt}\right)^2}, \tag{4}$$

where i_x and i_y represent two t-dimensional data objects, and a weight can be added to the attributes of each dimension. The calculation formula is

$$k(m,n) = \sqrt{V_1\left(i_{x1} - i_{y1}\right)^2 + V_2\left(i_{x2} - i_{y2}\right)^2 + \cdots + V_n\left(i_{xt} - i_{yt}\right)^2}, \tag{5}$$

where V_1, V_2, andV_n are the weight of each dimension attribute.

Manhattan distance is used to describe the average difference of objects in each dimension in a multidimensional space [25], and its calculation formula is

$$k(m,n) = \left|i_{x1} - i_{y1}\right| + \left|i_{x2} - i_{y2}\right| + \cdots + \left|i_{xt} - i_{yt}\right|. \tag{6}$$

Minkowski is a generalization of Euclidean distance, and Euclidean distance is a special case of Minkowski distance.

$$k(m,n) = \left(\left|i_{x1} - i_{y1}\right|^{\mathrm{q}} + \left|i_{x2} - i_{y2}\right|^{q} + \cdots + \left|i_{\mathrm{xt}} - i_{yt}\right|^{q}\right)^{(1/q)}, \tag{7}$$

where q is a positive integer, and when the parameter q is 1, Min's distance is converted to Manhattan distance, and when q is 2, it is converted to Euclidean distance.

In addition to using distance for similarity measurement, the similarity coefficient can also be used as the unit of

measurement. At present, the commonly used similarity coefficient is the angle cosine similarity. For two vectors s and t, the calculation formula is

$$\operatorname{sim}(s,t) = \frac{s \cdot t}{\|s\| \cdot \|t\|} = \frac{\sum_{x=1}^{n} s_x \times t_x}{\sqrt{\sum_{x=1}^{n} (s_x)^2} \times \sqrt{\sum_{x=1}^{n} (t_x)^2}}. \quad (8)$$

2.4.3. Objective Function. Clustering objective function is often used to evaluate the quality of clustering results. The input process is composed after each node in the tree search. The result of this process is generally to reduce the number of subsequent nodes that must be searched, which reduces the total computer time required to determine the isomorphism, which can be used for data recovery. It can reflect the similarity of objects within a class and the difference of objects between classes. Generally, it includes two objective functions: error sum of squares criterion and absolute error criterion.

The most widely used cluster objective function in cluster analysis is the error sum of squares criterion. Its specific definition is as follows:

$$\sigma = \sum_{x=1}^{r} \sum_{i_y \in B_x} \left\| i_y - b_x \right\|^2, \quad (9)$$

where σ is the sum of squared errors of all objects in the data set.

The absolute error criterion is to select a representative object in each cluster as a reference point. This change can reduce the influence of outliers on the clustering algorithm in some cases. Its specific definition is as follows:

$$\sigma = \sum_{x=1}^{r} \sum_{i_y \in B_x} \left\| i_y - c_x \right\|^2. \quad (10)$$

2.5. Ullman Algorithm. Ullman's algorithm is based on the branch and bound search of the search space, derived from the improvement of brute force enumeration. The core of Ullman algorithm is mainly divided into three parts: mapping matrix, compatibility matrix, and conditional judgment of isomorphism.

2.5.1. Mapping Matrix Q. The subgraph isomorphism can be described as a mapping g, and $R_m \longrightarrow R_n$ is described by a two-dimensional matrix M of order $q_m \times q_n$, that is, the mapping relationship g between two vertices. When a vertex $s \in R_m$ is mapped to a vertex $t \in R_n$ through g, the corresponding value in the matrix can be expressed as

$$q_{st} = \begin{cases} 1, & g(s) = t, \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

Matrix Q represents a mapping from vertex R_m to vertexR_n. In this process, each row of matrix Q has one and only one 1, and each column can only have one 1 at most:

$$\forall s \in R_m: \sum_{t \in R_n} q_{st} = 1 \text{ and } \forall t \in R_n: \sum_{s \in R_m} q_{st} \le 1. \quad (12)$$

2.5.2. Compatibility Matrix. The initial compatibility matrix isQ^0; if the degree of the vertex $s \in R_m$ is not greater than the degree of the vertex $t \in R_n$, then the vertex t is a candidate node of the vertex s, and the formula is

$$q_{st}^0 = \begin{cases} 1, & \vartheta_{Dm}(s) = \vartheta_{Dn}(t), \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Ullman's algorithm adds this judgment condition on the basis of brute force enumeration and achieves the purpose of simplifying the compatibility matrix. It gradually generates matrix Q from the compatibility matrix based on the backtracking method.

$$F_b(s) = \left\{ t \in R_m | q_{st}^b = 1 \right\}. \quad (14)$$

This is a set of candidate sets of vertices s. The matrix generated by filtering and simplification operations follows

$$q_{st}^b = 0 \Rightarrow q_{st}^{b+1} = 0. \quad (15)$$

2.5.3. Isomorphism Judgment. Ullman defines the matrix F in the method, which represents the adjacency matrix of the data graph. The matrix Q has the following conditions for the matrix F:

$$(\forall s \forall t)(c_{st} = 1) \Rightarrow (b_{st} = 1)(1 \le s \le T_m, 1 \le t \le T_n). \quad (16)$$

3. Cloud Computing Storage Backup and Recovery Strategy Experiment Based on Secure IoT and Spark

3.1. Experimental Environment. The cluster required for the experiment consists of 6 nodes, and the node configuration is shown in Table 2.

The Spark cluster of this system is built with 9 Red Hat Enterprise Server x64 virtual machines, and each virtual machine is configured with 500 G hard disk and 4 G memory. The software configuration information is shown in Table 3.

3.2. System Use Object. The user behavior analysis system involves two main user roles: system administrator responsible for platform development and daily management and maintenance, analyzing and verifying the correctness of the mining results, and business administrator responsible for the mining result data generated by the system calculation evaluation and management, optimizing and adjusting the corresponding business [26].

3.3. Experimental Design. The network data backup system is designed based on a hierarchical structure and includes three components: customer layer, service layer, and management layer. The client layer is the resource of the client computer that

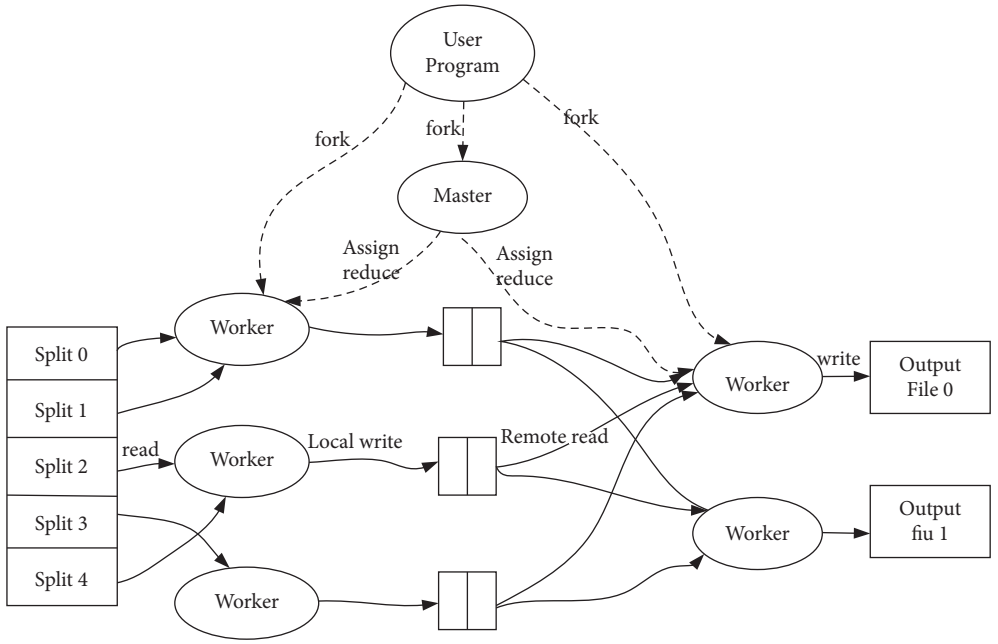

Figure 3: MapReduce flowchart.

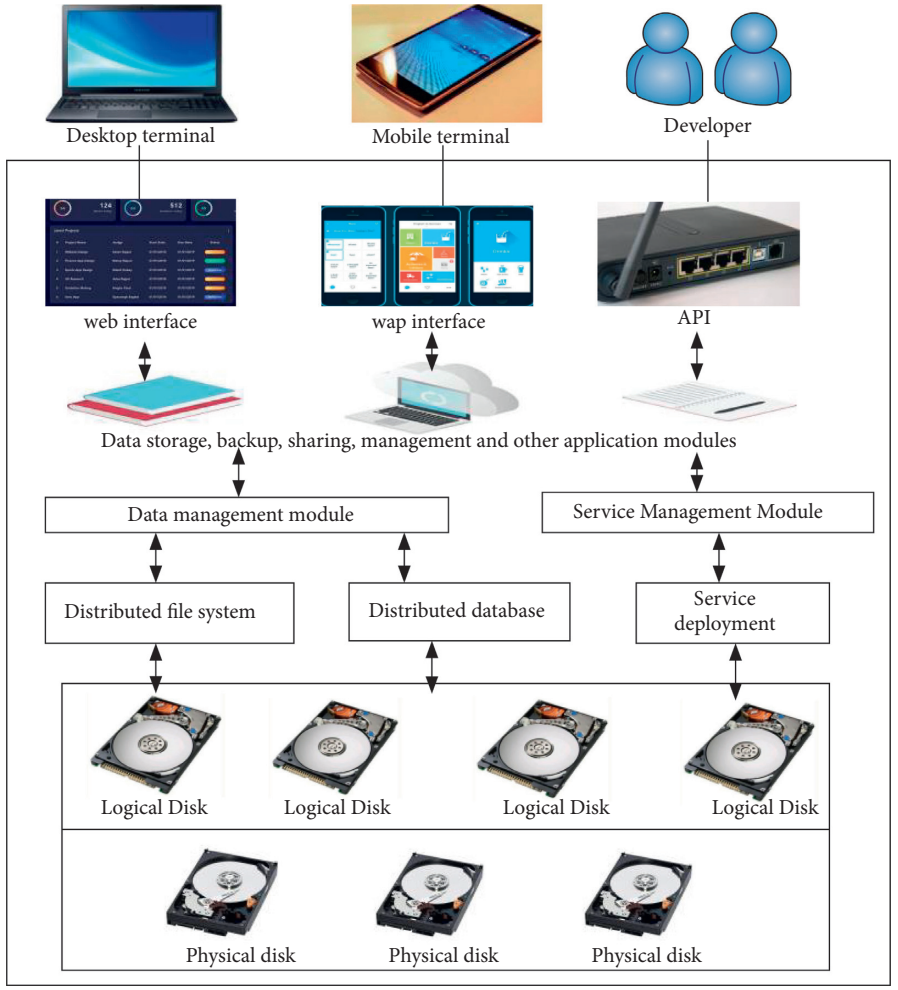

Figure 4: Architecture diagram of mobile phone backup system based on cloud storage (pictures from Baidu picture).

Table 2: Experimental environment configuration.

Parameter type	Parameter value
CPU information	Intel® Pentium® CPU G3260@3.30GHz
Number of CPU cores	16
RAM	4 G
Hard disk	500 G
Internet	1000 Mbps LAN
Operating system	CentOS 7.0 64 bit

the system needs to manage; the management layer is the core of the system, and the management layer includes backup contacts, backup computers, and authentication centers; the service layer represents server resources that provide storage. Among the three practical layers of the backup system, the management layer is the core of the business management logic [27]. The client software with backup statements distributed online is the client layer. Clients generally perform data backup operations, including backup plans, backup and test settings, and environmental settings recovery; backup and recovery security measures, including data encryption and decryption, user certificate application, management and use; and cancellation of the recovery process. The service layer can be implemented with one or more duplicate servers. The storage device 21 connected to the server provides a backup of the storage space. The archive backup system management service is distributed on the server side, responds to backup and recovery requests, and performs backup and recovery organizations, saves, and ends [28].

The data recovery process includes three stages: First, there is the data recovery application review process. The backup customer submits the recovery application to the console through the backup adapter, and the console authenticates the application through the CA. The second is the process of establishing a data recovery channel. The data transmission channel is established between the backup client and the backup service under the scheduling of the console. The third is the completion of the data recovery process. When the data recovery is over, the connection between the client and the backup server is closed, and the recovery result is notified to the console [29].

4. Cloud Computing Storage Backup and Recovery Strategy Analysis Based on Secure IoT and Spark

4.1. Challenge-Response-Verification Framework. It can be seen from Figure 5 that as the number of challenge data blocks increases, the time overhead of challenge-response-verification is gradually increasing, and the increase in the overhead of the verification process is small, which is difficult to see; under normal circumstances, the challenge is generated. The time cost is much smaller than the response cost, and the cost gradually increases with the increase in the number of challenge data blocks, but when the number of data blocks is very large, for example, when it reaches 1500, the time cost of the challenge increases sharply, and it becomes related to the verification process. The cost is comparable.

4.2. Number of Data Packets. After starting the test, you can compare the number of data packets received by each node of the remote monitoring software. The test results are shown in Table 4. The packet loss rate is equal to the number of packets sent minus the number of packets received divided by the number of packets sent.

From the data in Table 4, it can be seen that the data packet received by the monitoring software is smaller than the data packet sent by the monitoring node, indicating that this is a packet loss phenomenon in the data transmission process. This phenomenon is largely due to the large irregularities in the communication unit of the low-power unit. The results in Table 1 show that the loss rate of database 1 at the fourth node is 0.4%, 2.4%, 1.6%, and 3.2% and the loss rate of each node is less than 5%, indicating that the system can respond to applications.

4.3. Calculation and Communication Costs. The size of the file stored here is limited to 20 kb to 20mb, the number of elements is from 20 to 200, and the sample ratios are 10%, 20%, 30%, 40%, and 50%. The experimental results are shown in Figure 6. When the sample ratio is 50%, as the file size increases, the cost of calculation and communication rises from 0 to 193. These results indicate that the cost of calculation and communication increases with the file size and sample ratio.

The cost of response and inquiry is similar, and the cost of the answer and verification process is also similar. Figure 7 shows the experimental results of different query ratios. In order to confirm these verification results, the significant sample ratio in this paper is 10%–50%, the file size is limited to 10 MB, and each block has 200 elements. In the hybrid cloud $P = \{P1, P2, P3\}$, the data block is verified. The ratios are 0.1, 0.2, 0.3, 0.4, and 0.5, respectively. It can be seen from the figure that the cost of calculation and communication of inquiry and response changes slightly with the change of the sample rate, but the cost of answering and verification increases with the increase of the sample ratio. Here, the challenge and response are divided into two subprocesses: response 1 and response 2. Furthermore, the proportion of data blocks in each CSP largely affects the calculation and communication costs of queries and responses.

4.4. Spark Method Selection. For data storage and backup on the spark platform, algorithms need to be used to build a special processing system. Here, the APCA segmentation, ratio R, difference D, and durationik T methods are selected for error research.

Table 3: Software configuration.

Name of software	Version
Java	8u131-x64
Hive	2.3.2
My SQL	5.7.21
Tomcat	7.0.52
Hadoop	2.7.3
Spark SQL MLlib	2.1.1

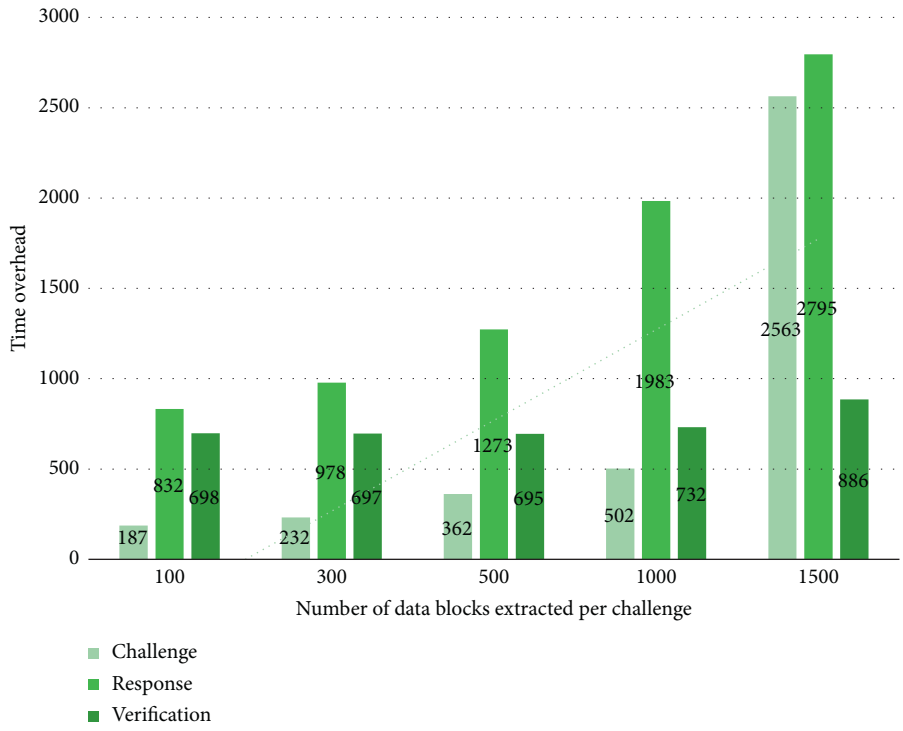

Figure 5: Challenge-response-authentication process time overhead.

As can be seen from Figure 8, compared with APCA segmentation, the errors of the other three two-stage representation methods are relatively small. The experiment found that the average error selected by the ratio R method is the smallest and these two methods are better than selecting important points by controlling the duration. Therefore, comparing the results of running on Spark, it is best to use the ratio R to select important points.

4.5. Throughput of Different Platforms. Figure 9 shows the comparison result with the native system throughput. After the optimization of the index mechanism, the throughput of S-TSQS is significantly higher than that of SparkDS and SparkSQL. Because of the better optimization strategy of SparkSQL, the efficiency of similarity query is slightly higher than that of SparkDS. Experimental data shows that the query efficiency of S-TSQS is about 3–6 times that of SparkSQL and SparkDS.

4.6. Iteration and Caching. It can be seen from Figure 10 that when the number of iterations is 1, the processing time within the range of 1–7 packets is not much different and the difference is almost negligible, but when the number of packets is 10, the impact of buffering begins to manifest. It turns out that the cached one needs less processing time, 104 hours, while the uncached one needs 133 hours. However, when the number of iterations is 3, there is basically no significant change in whether the data packet changes from 1 to 13, and it is basically the same throughout the whole process.

The default noncaching strategy means that no caching is performed, and the default caching means that no cost evaluation and optimization processing are performed on nodes with caching value, and the caching is set directly. Four representative queries were selected from the experimental results for analysis, and they were referred to as query 1, query 2, query 3, and query 4. The result data is shown in Table 5 (the value unit is seconds).

Table 4: Data packet experiment results.

Node number	Sent data packet	Receive packet	Packet loss rate (%)
Node 1	500	498	0.4
Node 2	500	488	2.4
Node 3	500	492	1.6
Node 4	500	484	3.2

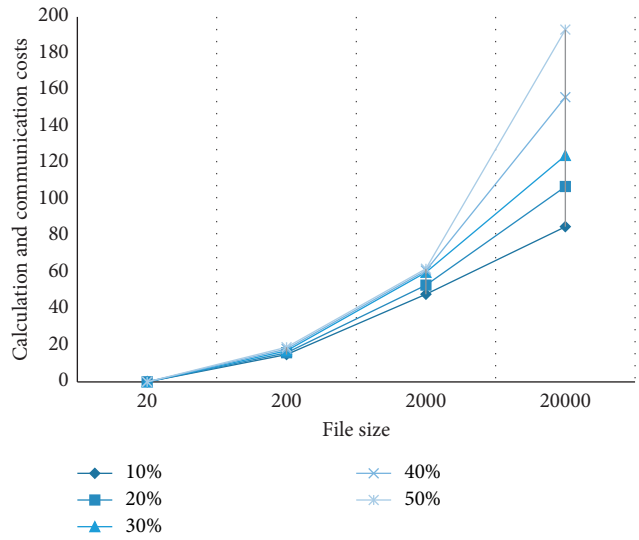

Figure 6: Use different ratios of experimental results.

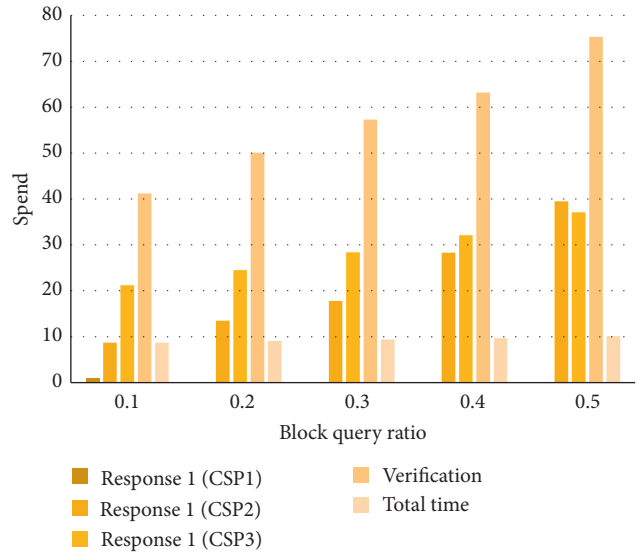

Figure 7: Experimental results of different query ratios.

5. Conclusion

With the continuous development and rapid growth of spatial big data, the demand for visualization will become more evident in real time. Under the Spark platform, it is worth continuing to be studied from the perspective of streaming data processing to achieve real-time data visualization operations. Since its emergence, cloud computing has been attracting attention and has been developing at an extremely fast speed. However, due to potential security

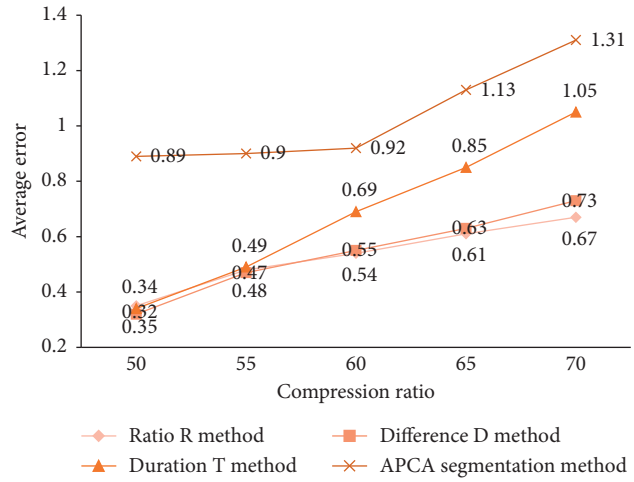

Figure 8: Reference sequence segmentation error comparison.

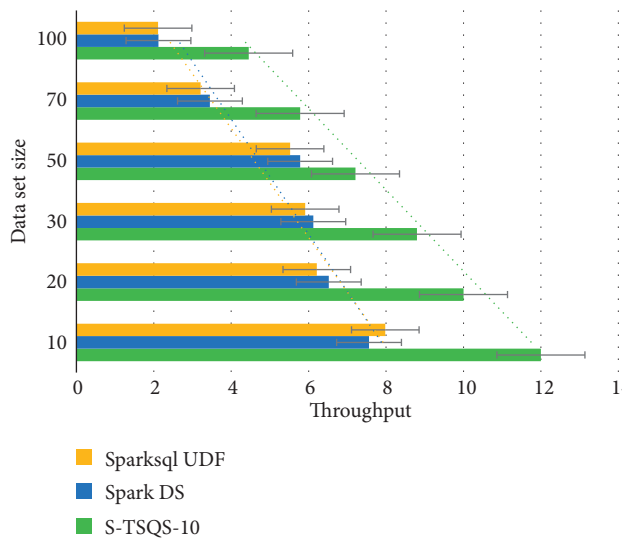

Figure 9: Comparison with native system throughput.

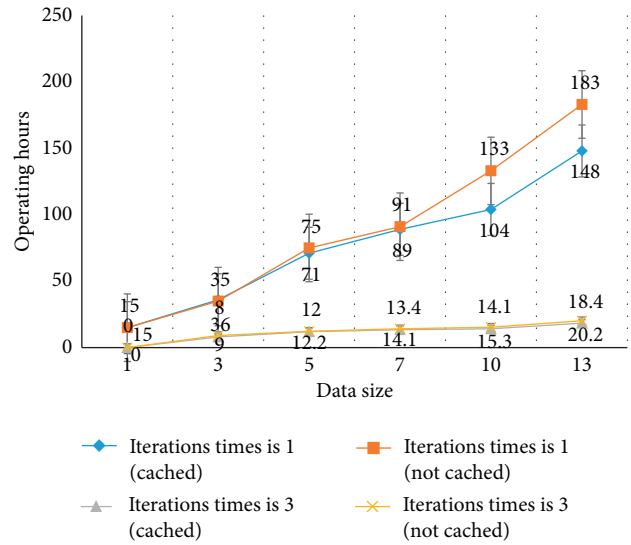

Figure 10: Experimental results of the algorithm in this paper.

Table 5: Cache strategy experiment result data.

Query strategy	Not cached by default	Default cache	Adaptive caching	Adaptive caching strategy
Query 1	81.2	79.3	68.6	Union push down + cache
Query 2	94.3	129.4	99.4	Do not set cache
Query 3	249.4	289.6	258.5	Do not set cache
Query 4	365.6	221.9	186.0	Union push down + cache

issues, many people and companies have been holding a wait-and-see attitude towards cloud computing. Among them, cloud storage security is the core that people pay most attention to. This paper conducts research from two perspectives of data integrity and data privacy protection research. The shortcoming of this article is that the projects in the article are all independent projects; that is, the work is divided into multiple independent computer projects for parallel computer processing. In a real cloud environment, many projects are not independent projects, but interdependent and important. Future work can explore how to improve the data center and reduce the completion time of dependent projects. In the design of the backup plan algorithm, only one indicator of the backup cost is considered. In the actual cloud environment, there are many factors that affect project organization, such as the storage space of resources. When designing project backups, future work may have a greater impact on all aspects.

Data Availability

No data were used to support this study.

Conflicts of Interest

There are no potential conflicts of interest in this study.

References

[1] P. C. P. Kumar and G. Geetha, "Gateway pi-design and implementation of smart and secure internet of things gateway integrating with Raspberry Pi," *Journal of Computational and Theoretical Nanoscience*, vol. 14, no. 9, pp. 4448–4453, 2017.

[2] A. Arul Mary and K. Chitra, "OGSO-DR: oppositional group search optimizer based efficient disaster recovery in a cloud environment," *Journal of ambient intelligence and humanized computing*, vol. 10, no. 5, pp. 1885–1895, 2019.

[3] M. Interlandi, K. Shah, and S. D. Tetali, "Titian: data provenance support in spark," *Proceedings of the VLDB Endowment*, vol. 9, no. 3, pp. 216–227, 2016.

[4] J. Y. Kim, W. Hu, H. Shafagh, and S. Jha, "SEDA: secure over-the-air code dissemination protocol for the internet of things," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 6, pp. 1041–1054, 2018.

[5] L. Zhe, X. Huang, and H. Zhi, "On emerging family of elliptic curves to secure internet of things: ECC comes of age," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 3, pp. 237–248, 2017.

[6] S. Roy, S. Chatterjee, and G. Mahapatra, "An efficient biometric based remote user authentication scheme for secure internet of things environment," *Journal of Intelligent and Fuzzy Systems*, vol. 34, no. 3, pp. 1403–1410, 2018.

[7] S. Davidson, "Engineering secure internet of things systems," *IEEE Design & Test*, vol. 34, no. 5, pp. 97-98, 2017.

[8] B. Kim and S.-B. Cho, "3D TSV-based inductor design for a secure internet of things," *International Symposium on Microelectronics*, vol. 2016, no. 1, pp. 364–367, 2016.

[9] M. Zaharia, R. S. Xin, P. Wendell et al., "Apache spark," *Communications of the ACM*, vol. 59, no. 11, pp. 56–65, 2016.

[10] I. V. Lukiyanchuk, V. S. Rudnev, and V. G. Kuryavyi, "Surface morphology, composition and thermal behavior of tungsten-containing anodic spark coatings on aluminium alloy," *Thin Solid Films*, vol. 446, no. 1, pp. 54–60, 2016.

[11] J. Merlin, B. A. Evans, N. Dehvari, M. Sato, T. Bengtsson, and D. S. Hutchinson, "Could burning fat start with a brite spark? Pharmacological and nutritional ways to promote thermogenesis," *Molecular Nutrition & Food Research*, vol. 60, no. 1, pp. 18–42, 2016.

[12] G. Yang, Y. Yao, J. Fang, T. Gan, Q. Li, and L. Lu, "Large-eddy simulation of shock-wave/turbulent boundary layer interaction with and without SparkJet control," *Chinese Journal of Aeronautics*, vol. 29, no. 3, pp. 617–629, 2016.

[13] M. Penchal Reddy, R. A. Shakoor, A. M. A. Mohamed, M. Gupta, and Q. Huang, "Effect of sintering temperature on the structural and magnetic properties of MgFe2O4 ceramics prepared by spark plasma sintering," *Ceramics International*, vol. 42, no. 3, pp. 4221–4227, 2016.

[14] N. V. Patil, C. R. Krishna, and K. Kumar, "Apache spark based real-time DDoS detection system," *Journal of Intelligent and Fuzzy Systems*, vol. 38, no. 3, pp. 1–9, 2020.

[15] S. H. Popkin, B. Z. Cybyk, C. H. Foster, and F. S. Alvi, "Experimental estimation of SparkJet efficiency," *AIAA Journal*, vol. 54, no. 6, pp. 1831–1845, 2016.

[16] M. Charles and Ugras, "Identification of the norton-green compaction model for the prediction of the Ti-6Al-4V densification during the spark plasma sintering process," *Advanced Engineering Materials*, vol. 18, no. 10, pp. 1720–1727, 2016.

[17] A. H. Sebayang, H. H. Masjuki, and H. C. Ong, "A perspective on bioethanol production from biomass as alternative fuel for spark ignition engine," *RSC Advances*, vol. 6, no. 13, pp. 14964–14992, 2016.

[18] J. Chen, K. Li, Z. Tang et al., "A parallel random forest algorithm for big data in a spark cloud computing environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 4, pp. 919–933, 2017.

[19] L. Zeng, S. Xu, and Y. Wang, "VMBackup: an efficient framework for online virtual machine image backup and recovery," *Concurrency and Computation: Practice and Experience*, vol. 28, no. 9, pp. 2630–2643, 2016.

[20] S. Gokulakrishnan and J. M. Gnanasekar, "Data integrity and recovery management under peer to peer convoluted fault recognition cloud systems," *Journal of Computational and Theoretical Nanoscience*, vol. 17, no. 5, pp. 2147–2150, 2020.

[21] F. Deng, L. Dong, and C. Zhe, "Control strategy of wind turbine based on permanent magnet synchronous generator and energy storage for stand-alone systems," *Chinese Journal of Electrical Engineering*, vol. 3, no. 1, pp. 51–62, 2017.

[22] W. Wei, X. Fan, and H. Song, "Imperfect information dynamic stackelberg game based resource allocation using hidden markov for cloud computing," *IEEE Transactions on Services Computing*, vol. 11, no. 99, pp. 78–89, 2018.

[23] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation offloading and resource allocation in mixed fog/cloud computing systems with min-max fairness guarantee," *IEEE Transactions on Communications*, vol. 66, no. 4, pp. 1594–1608, 2018.

[24] A. Kaur and S. Sharma, "An analysis of task scheduling in cloud computing using evolutionary and swarm-based algorithms," *International Journal of Computer Application*, vol. 89, no. 2, pp. 11–18, 2018.

[25] R. Jegadeesan, N. Amulya, and K. Lavanya, "Towards security protecting substance based picture recovery in cloud," *International Journal of Innovative Research in Science Engineering and Technology*, vol. 6, no. 3, pp. 340–350, 2019.

[26] A. M. Al-Momani, M. A. Mahmoud, and M. S. Ahmad, "Factors that influence the acceptance of internet of things services by customers of telecommunication companies in Jordan," *Journal of Organizational and End User Computing*, vol. 30, no. 4, pp. 51–63, 2018.

[27] L. Z. Zhang, M. Mouritsen, and J. R. Miller, "Role of perceived value in acceptance of "bring your own device" policy," *Journal of Organizational and End User Computing*, vol. 31, no. 2, pp. 65–82, 2019.

[28] L. Fabisiak, "Web service usability analysis based on user preferences," *Journal of Organizational and End User Computing*, vol. 30, no. 4, pp. 1–13, 2018.

[29] S. Namasudra and P. Roy, "Ppbac: popularity based access control model for cloud computing," *Journal of Organizational and End User Computing*, vol. 30, no. 4, pp. 14–31, 2018.