

Retraction

Retracted: Construction of a Multimedia-Based University Ideological and Political Big Data Cloud Service Teaching Resource Sharing Model

Mathematical Problems in Engineering

Received 11 July 2023; Accepted 11 July 2023; Published 12 July 2023

Copyright © 2023 Mathematical Problems in Engineering. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

- (1) Discrepancies in scope
- (2) Discrepancies in the description of the research reported
- (3) Discrepancies between the availability of data and the research described
- (4) Inappropriate citations
- (5) Incoherent, meaningless and/or irrelevant content included in the article
- (6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their

agreement or disagreement to this retraction. We have kept a record of any response received.

References

- [1] J. Feng, W. Zhang, and S. Tsai, "Construction of a Multimedia-Based University Ideological and Political Big Data Cloud Service Teaching Resource Sharing Model," *Mathematical Problems in Engineering*, vol. 2021, Article ID 9907630, 12 pages, 2021.

Research Article

Construction of a Multimedia-Based University Ideological and Political Big Data Cloud Service Teaching Resource Sharing Model

Jian Feng ^{1,2}, Weiliang Zhang,¹ and Sang-Bing Tsai ³

¹School of Marxism, Central South University, Hunan 410083, Changsha, China

²College of Marxism, Sichuan University of Science & Engineering, Sichuan 643000, Zigong, China

³Regional Green Economy Development Research Center, School of Business, Wuyi University, Nanping, China

Correspondence should be addressed to Jian Feng; 2064307060@qq.com

Received 21 October 2021; Revised 9 November 2021; Accepted 23 November 2021; Published 29 December 2021

Academic Editor: Xiangtao Li

Copyright © 2021 Jian Feng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, the intelligent education cloud service platform is first constructed in view of the high cost consumption in the process of resource sharing scheduling in colleges and universities. Secondly, the hierarchical education resource sharing grid model is proposed. Specifically, according to the characteristics of the educational resource grid, the key factors affecting the performance of copy creation strategy are analysed, and a dynamic copy creation strategy is proposed. A multiresource equity distribution mechanism based on the concept of resource sharing equity is further proposed. The mechanism establishes a planning model according to the limited task resource demand and the amount of resources shared by the user at different times so that the global cumulative share vector of superior resources meets the dictionary order optimally. The simulation experiment shows that the grid sharing model proposed in this paper has better performance on the educational cloud service platform. The proposed resource allocation mechanism has achieved good results in ensuring the fair distribution of resources and ensuring high resource utilization when resource sharing users put forward multiple groups of time-changing resource demands.

1. Introduction

The multimedia-based education cloud service platform is a revolutionary driving force that promotes the sustainable evolution of high-quality educational resources, promotes the innovation of educational organization structure, promotes the transformation of education to service mode, and promotes the transformation of business processes [1–4]. Cooperate with the school's education and training plan to cultivate engineering and practical talents adapted to the society [5]. Online learning resources are the core and foundation of the online education process and have always been the focus of attention of experts, scholars, frontline teachers, and even learners [6]. In recent years, cloud computing is gradually penetrating the education field with its massive data storage and convenient network services. If cloud computing-related technologies and services are used to solve the problem of online learning resource sharing, a new idea can be explored to promote the effective sharing of online learning resources [7, 8].

Cloud computing is centralized computing and access on cluster servers in the network. Yao et al. [9] mainly studied how to achieve system integration and resource sharing and improve hardware utilization. However, there is a lack of research on massive data processing. Jiang et al. [10] proposed a hybrid architecture model to solve the problem of information islands in enterprises. But there is no introduction to resource integration and how to improve resource utilization. Zhou et al. [11] proposed a cloud-based virtual desktop system to study the use of virtual classrooms in the education field. Barreto et al. [12] proposed to integrate knowledge organization and knowledge management, and regard knowledge as a service. Martinez-Guijosa et al. [13] mainly studied the sharing and utilization of resources but did not solve the problem of unbalanced utilization of resources. Aldowah et al. [14] proposed to model the complex causal relationship in the system by constructing a system model and verify the result-factor relationship existing in the system model through model analysis. Liu [15] proposed a new teaching system that integrates web

services of courseware resources through the SOAP protocol. Al-Samarraie et al. [16] reviewed the transformation from traditional education information to education services under the cloud computing environment and the computing architecture of cloud education.

As the main application environment of the new generation of interconnection networks, the grid provides an effective means to solve the problem of distributed resource sharing [17]. In a grid system, creating a copy is an effective means to improve the quality of service of the system [18]. However, the current research on replica creation strategy is mainly concentrated in the grid system environment represented by the European data grid [19]. Compared with the European data grid, the educational resource grid has differences in the network environment such as raw data distribution, network bandwidth, and node storage capacity. First, this makes the existing copy creation strategy unable to be applied to the educational resource grid. Second, the resource-sharing environment with a single model can no longer meet the needs of various industries [20]. Resource balancing scheduling as an important part of network communication software technology has also become the focus of research in various fields [21–23]. However, the current shared resource balanced scheduling method generally has problems such as too long scheduling completion time and high cost consumption. Currently, Yimin et al. [24] proposed a cloud computing resource scheduling method based on the quantum particle swarm algorithm. This method analyses cloud computing resource scheduling. On this basis, a mathematical model of resource scheduling is constructed, the objective function is given, and the quantum particle swarm algorithm is used to solve it. This method can effectively improve the utilization of resource scheduling. However, it takes too long to complete in the scheduling process. Sun et al. [25] proposed a resource scheduling method based on resource weight and maximum resource utilization. This method analyses resources and obtains the resource weight ratio. Resource scheduling is performed according to the maximum resource utilization of the weight. This method can effectively improve the resource utilization rate of the entire system and quickly solve the load balancing problem. However, the cost required in the process of scheduling is relatively high.

In response to the above problems, we first built a smart education cloud service platform. Secondly, a hierarchical grid model of educational resource sharing is proposed. According to the characteristics of the educational resource grid, the key factors affecting the performance of the replica creation strategy are analysed, and a dynamic replica creation strategy is proposed. Finally, considering a cloud computing resource sharing system with multiple virtual machines, resource requirements for each task submitted by different users may be different. This paper designs a dynamic resource scheduling algorithm, a dynamic multiresource fair allocation algorithm in which users submit multiple sets of limited time-varying task requirements.

2. Overview of Cloud Services

2.1. SOA. Service-oriented architecture (SOA) is the methodology, software design model, and architecture model and the architecture of applications in a distributed environment [26]. The application function is service-oriented with a service-oriented design concept so that the application function is finally used in the form of a service. Services can communicate with each other, services can be reused, and services can operate independently of each other without affecting each other. Services can be reorganized and combined into new services to be used [27].

The core theme of SOA is service. Services are a bunch of processing methods. These processing operations are encapsulated to form a standard service, which is a service interface formed according to the definition of a service contract [28]. SOA is an extension of object-oriented design ideas. Component is an abstraction of the realization of specific functions in object-oriented design ideas. And service is to encapsulate components, and it is also an abstract concept for the realization of higher-level functions. Its specific functional implementation does not require waiting for the implementation processing state of other services to be acquired. It also does not need to rely on the response information context of other services, nor does it require additional information from service requesters. In this way, the service can achieve a high concurrency of request and response [29]. The SOA analysis process is shown in Figure 1.

2.2. Education Cloud. Education cloud [30–32] uses a new generation of basic equipment including large-scale servers and large-scale storage devices as the infrastructure. On this basis, a development and collaborative sharing educational knowledge resource management platform is established, and educational resources are pushed to the terminal through the platform. The education cloud service model can be divided into three levels: education cloud software as a service E-SaaS, education cloud platform as a service E-PaaS, and education cloud infrastructure as a service E-IaaS [33–35]. Education cloud is divided into three service types: IaaS, PaaS, and SaaS, as shown in Figure 2.

SaaS refers to providing educational application software as a service to end users. By deploying the service in the environment of the educational cloud service platform, educational cloud users can lease relevant educational services by using an educational cloud service purchase platform. They can also customize services to meet their needs from the cloud service platform without independent development and purchase.

PaaS refers to the use of software development platform as a service for purchasing users. The platform supports the development and use of a variety of programming languages, builds the running environment of the development software, and has runtime monitoring tools, business process development tools, secure and effective service middleware, and so on to support the safe and stable development and operation of the software.

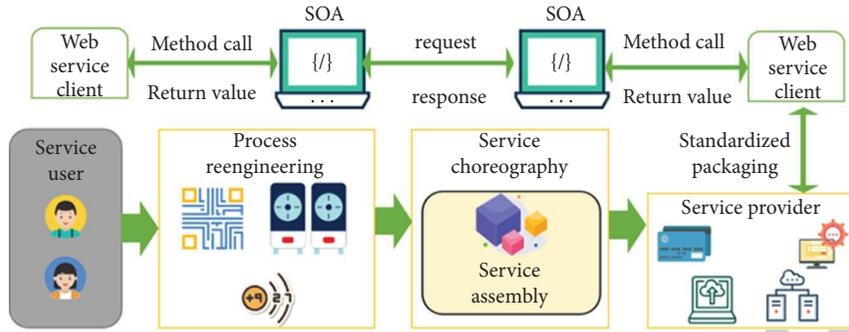


FIGURE 1: SOA analysis flowchart.

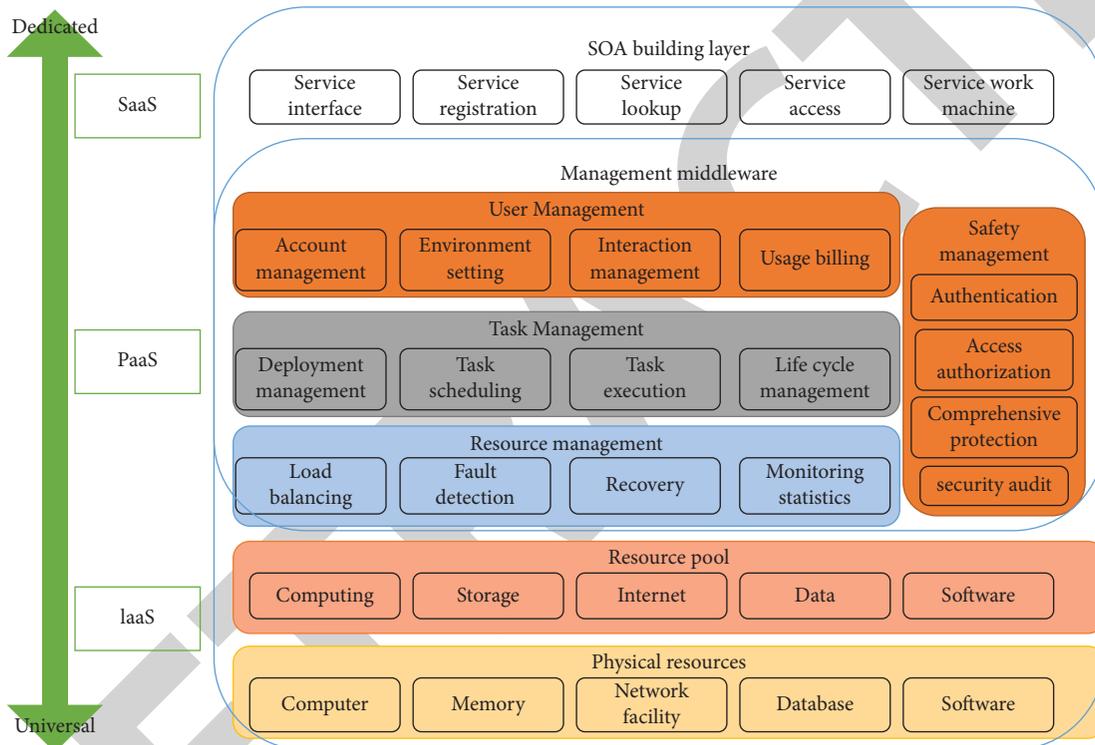


FIGURE 2: Architecture of educational cloud computing.

IaaS refers to the use of virtualization technology to virtualize servers, storage devices, networks, and other hardware devices into a computable and distributable pool of virtual resources. Users can use the resource pool service provided by the education cloud service platform by time, traffic, and other billing types. Because it is leased on demand, users can adjust their needs according to their own needs, and the resource pool resource management system will flexibly adjust resource allocation according to user needs.

3. Construction of Teaching Resource Sharing Model

3.1. *Cloud Service Platform Based on Multimedia.* This article establishes a low-cost, multitenant-oriented, scalable, and digital education cloud service platform. Utilizing the

resource sharing characteristics of the education cloud service platform, the platform forms a collaborative and innovative digital education alliance based on the accumulation of massive resources and educational organizations. The platform provides decision-making services and collaborative cooperation services for the education alliance.

The layers of the platform are interrelated and independent of each other. The upper layer has implementation dependence characteristics on the lower layer, but the result of the lower layer needs to be delivered to the user by the upper layer. The service characteristics of each layer are different. IaaS provides virtualized IT resources as services to users, PaaS provides virtualized development platforms as services to users, and SaaS provides various teaching functions, teaching office functions, business functions, and decision-making functions as services to users, and the smart

education platform develops, deploys, and manages cloud services. The architecture of the multimedia-based cloud service platform is shown in Figure 3.

3.2. Hierarchical Educational Resource Sharing Grid Model.

As a key technology in cloud computing and a new generation of Internet technology, grid has a huge advantage in realizing resource sharing. Resource sharing in this technological environment is a new product of the organic combination of traditional resources, network technology, and computer technology. Its essence is a series of technologies that are generated to meet the internal resource sharing and problem solving of virtual organizations.

The “five-layer hourglass structure” in the grid technology disperses the operation, management, and use functions of shared resources in five different levels. From bottom to top, they are Fabric, Connectivity, Resource, Collective, and Application. Based on this, combined with the three service forms of cloud computing, a layered architecture is adopted to design the network learning resource-sharing system model from the three levels of resource layer, management layer, and application layer, as shown in Figure 4.

The basic operations of educational resource grid users mainly include uploading, retrieving, and downloading data files. After introducing the copy mechanism in the educational resource grid, creating a reasonable copy can effectively increase the speed of user resource retrieval and download and shorten the response time for users to request resources.

In the educational resource grid, a data file can have multiple copies, and each copy has a physical file name (PF) and a logical file name (LF). Each copy corresponds to a unique physical file name, and the logical file name of the same copy is the same. That is, one logical file name corresponds to multiple physical file names. We denote the set of all physical file names corresponding to a logical file name as $|PF|_s$.

When a user sends an access request to a data file, the copy mechanism selects the copy with the shortest response time for the user from multiple copies of the data file. Therefore, the response time of the copy to a specific request directly affects the performance of the copy mechanism. The response time is the sum of the time of the requesting node, the waiting time before transmission, the time of reading and writing files, and so on. The response time (RT) of the duplicate physical file name to a specific request can be calculated by the following formula:

$$RT(PF) = FT(PF) + FA(PF). \quad (1)$$

Among them, FT represents the time to transfer the physical file name of the file from its storage node to the requesting node. FA means the time it takes to access the file’s physical file name, including the waiting time before transmission and the time to read and write the file.

$$FT(PF) = \frac{PF.size}{\alpha}. \quad (2)$$

Among them, PF.size represents the size of the copy and α is the minimum bandwidth of the transmission link between the copy storage node and the requesting node (see Figure 4).

Therefore, the network bandwidth between the replica storage node and the requesting node, and the size of the replica directly affects the response time of the replica. The factors that affect the FA include the physical file name, the load of the storage node, the read efficiency of the storage node, and the write efficiency of the requesting node. We have generally considered

$$FT(PF) \gg FA(PF). \quad (3)$$

The educational resource grid is composed of several nodes, the nodes form a hierarchical structure, and the nodes are denoted as N . N can contain multiple CUs and SUs. Among them, CU is a computing unit, used to process user’s job requests; SU is a storage unit, used to store the original resources and copy resources of the system. Each node maintains a historical access record, and each item in the historical record is a triplet. (SN, PF, f) is used to represent the frequency with which the replica PF in the SU is accessed. (CN, LF, f) is used to represent the frequency of CN requesting data file LF.

When a user sends an access request to a data file, the copy mechanism selects a copy with the shortest response time for the user from multiple copies of the data file.

The file size of the copy PF is the physical file name’s size, and the current access frequency is LF.f. The node is N_1 , and the node with the same LF as PF after PF is deleted is N_2 ; then,

$$PF.c = PF.size * \frac{PF.f}{BW(N_1, N_2)}. \quad (4)$$

Among them, BW is the minimum bandwidth of the transmission link between nodes N_1 and N_2 .

Assume that the size of the logical file LF is LF.size. The current request frequency is LF.f. The node where the newly created LF copy is located is N_1 . The node of LF before the new copy is N_2 . So we can define

$$LF.r = LF.size * \frac{LF.f}{BW(N_1, N_2)}. \quad (5)$$

The existing replica creation strategies are either designed for the special grid environment and cannot be applied to the grid system in the ordinary network environment, or consider few network parameters, which cannot well reflect the actual network environment, resulting in frequent establishment and deletion of replicas, and the implementation of some strategies is also limited. Aiming at the problem of sharing small and medium-sized educational resources in the general network environment, this paper puts forward a hierarchical educational resource grid model and defines the function of each node. Then, based on this model, a dynamic replica creation strategy is proposed. This strategy not only fully considers the factors such as network bandwidth, file size, and request frequency when creating replicas but also solves the problem of frequent establishment and deletion of replicas caused by the small storage capacity of grid nodes.

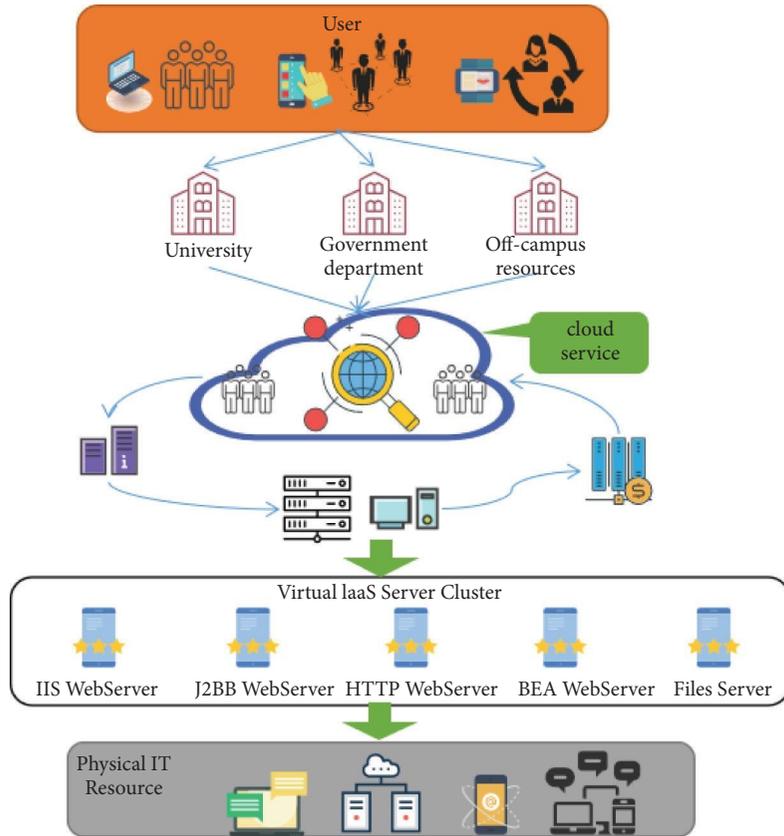


FIGURE 3: Architecture diagram of a multimedia-based cloud service platform.

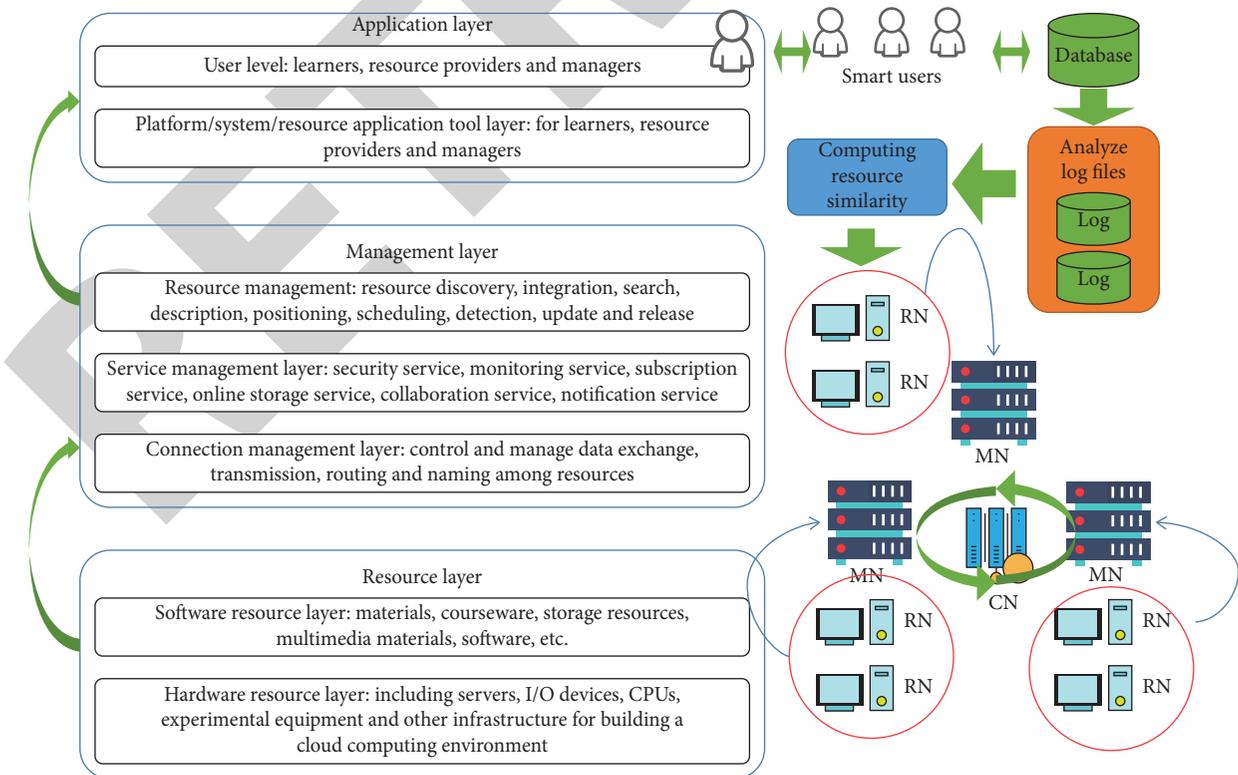


FIGURE 4: Online learning resource-sharing model based on smart education cloud service platform.

Through actual analysis and application, it is found that, in a local area, the user requests of the educational resource grid sometimes have the characteristics of dispersion. In this case, we introduced a new strategy to solve the problem.

The basic idea of this strategy is similar to that of ABU, which aggregates the same historical records layer by layer from bottom to top. When aggregating, the f of each record with the same parent node in the history record and requesting the same logical file name is aggregated. In addition, the education resource grid generally does not exceed 6 layers from the root node to the bottom node, which are the central node, provincial node, municipal node, county node, school management node, and bottom school resource node. According to the actual situation, within the same period of time, the similarity of resources requested by users in the jurisdiction of a node decreases as the level of the node increases. Therefore, each time it aggregates, it does not aggregate to the central node from the bottom up but only aggregates to the city-level nodes.

3.3. Dynamic Resource Scheduling Algorithm. In this paper, a dynamic resource scheduling algorithm is designed. Our approach is based on a multivirtual machine cloud computing resource sharing system. On this basis, a dynamic multiresource equitable distribution algorithm is proposed for users to submit multiple sets of limited time-changing tasks. Considering a cloud computing resource sharing system with multiple virtual machines, resource requirements for each task submitted by different users may be different. Therefore, it is necessary to select the most suitable virtual machine to run according to the resource requirements of the task submitted by the user to meet higher resource utilization.

We define the remaining resources of the m -th virtual machine in the system at time a as

$$R = (r_1, \dots, r_m) = (n - m_1(a), n - m_2(a), \dots, n - m_m(a)). \quad (6)$$

Among them, $ri = n - mi(a)$ is the remaining amount of the i -th resource in the m virtual machines at time a .

The amount of resources required by each task of user u at time a is

$$Ru_i(a) = (Ru_{i1}(a), Ru_{i2}(a), \dots, Ru_{im}(a)). \quad (7)$$

Among them, Ru_i is the demand of user u for the i -th resource in each task at time a .

We define the following heuristic formula to measure the matching degree between user task resource requirements and virtual machines to complete the matching between user tasks and virtual machines:

$$Q(u, m) = \frac{Ru_i(a)}{\sum_i Ru_i(a)} - \frac{n - m_i(a)}{\sum_i n - m_i(a)}. \quad (8)$$

When the time a in $Q(u, m)$ is smaller, it means that the task demand of user u at time a is more matched with the remaining resource capacity of the m -th virtual machine. We select the virtual machine with the smallest matching degree to allocate task resources to user u .

The main idea of the algorithm has three aspects:

- (1) When there are n users and m virtual machines in the system, according to the resource requirements $Ru_i(a)$ of each task of any user $u \in U$ in the system at different times, select the best match in each round according to formula (8). The server m allocates the resources needed to perform a task to the user u . And update user u 's cumulative global dominant resource share and global sharing coefficient p , sort from small to large, and select the user with the smallest sharing coefficient among users:

$$u = \min_i (1, n * p). \quad (9)$$

If $p < 1$, find the most matching virtual machine to allocate a task resource to the user u and update the system resource allocation status.

- (2) Perform the above steps in sequence. If the minimum global sharing coefficient $p \geq 1$ for all users in the system, it means that all users in the system will no longer share defects. Select the user with the smallest cumulative share of dominant resources among all users:

$$\min_u = \min(1, n * u), \quad (10)$$

and allocate the resources needed to perform a task for the user \min_u and update the system resource allocation status.

- (3) Perform Step 1 and Step 2. When a certain resource i , $1 \leq i \leq m$ in the system is allocated, or when all user tasks are executed, the resource allocation ends.

4. Results and Discussion

4.1. Configuration of Teaching Resource Platform and Experimental Settings. This paper uses the CPU and Memory resources required for running 12,951 tasks from the Alibaba cluster dataset [36] as the user resource requirements in this experiment. Each user in the dataset submits multiple sets of task resource requirements in the form of job. Each job contains multiple sets of task requirements. Each set of tasks consists of multiple instances with the same resource requirements. The experimental simulation server configuration in this paper is shown in Table 1, where the number of servers is the server resources of each user. This article assumes that the number of virtual machines is the same as the number of physical servers after users share resources (see Table 1).

4.2. The Impact of Different Copy Strategies on the Sharing Model. The average job execution time refers to the total execution time of jobs submitted by users divided by the number of jobs, which directly reflects the impact of replica creation strategy on grid system performance. Under the same workload, the shorter the average job execution time of the system, the better the performance. Based on this indicator, we, respectively, simulated, compared, and analysed

TABLE 1: Cluster server configuration.

User number	5	20	100
Per user own machines	1	1	1
CPU capacity	250	250	250
Memory capacity	40	40	40

the impact of factors such as the number of jobs, node storage capacity, and user access patterns on the performance of educational resource grid systems using different copy creation strategies.

4.2.1. Number of Jobs. The storage capacity of the underlying node is set to 100 G and 50 G, respectively, to conduct two sets of experiments to analyse the impact of the number of jobs on the performance of the educational resource grid system using different strategies. It can be seen from Figure 5 that, compared with no copy, various copy placement strategies can significantly reduce the average execution time of the job, which fully verifies that creating a copy can improve the performance of the grid system.

Figure 5(a) is the experimental result when the node storage capacity is 50 G. We observe that when the node storage capacity becomes 50 G, the performance of our strategy is significantly better than the other three strategies, and only when our strategy is adopted, time of the average job execution can be reduced as the number of jobs increases. This is because when the storage capacity of a node is insufficient, our strategy can select neighbour nodes to create a copy instead of frequent copy replacement, which saves a lot of remote file access time.

Figure 5(b) is the experimental result when the node storage capacity is 100 G. First of all, we notice that as the number of jobs increases, the average job execution time of the four copy creation strategies decreases, and the performance is not much different. This is because when the storage capacity of the underlying node is large, these strategies can create copies on the underlying node, reducing the number of remote file accesses. In addition, we can also observe that when the number of jobs is the same, the performance of our strategy is the best, and the performance of the copy creation strategy based on the economic model is the worst (see Figure 5).

4.2.2. Node Storage Capacity. In the two cases of 100 and 500 jobs, the impact of node storage capacity on the performance of the educational resource grid system using different strategies is analysed. It can be seen from Figure 6(a) and Figure 6(b) that the average job execution time of the three strategies of LRU, LFU, and the economic model is much longer than that of ours. When the storage capacity of the node is small, the difference is more obvious. This fully reflects the superiority of our strategy to select neighbour nodes to create copies when the storage capacity of the node is small.

We can also observe that as the storage capacity of the node decreases, the average job execution time of the three strategies of LRU, LFU, and the economic model gradually

approaches the state of no copy. This fully shows that these three strategies are not suitable for educational resource grids with small node storage capacity. In addition, by comparing Figure 6(a) and Figure 6(b), we can also find that even when the storage capacity of the node is small, the average job execution time of our strategy can be reduced with the increase of the number of jobs.

4.2.3. Access Mode. The storage capacity of the underlying node is set to 50 G and the number of jobs to 500. Under the two modes of decentralized access and centralized access, the performance of the system when different strategies are adopted is analysed. It can be seen from Figure 7 that when the same job is submitted, the execution time of the distributed access job is less than the execution time of the centralized access job. This is because, in centralized access, all jobs are concentrated on a few specific nodes, making the storage capacity of the underlying nodes insufficient, resulting in remote access to files and frequent replacement of copies (see Figure 6).

4.2.4. Number of Copies Created. The number of copies created refers to the total number of copies created during the execution of the job. Since the creation of a copy requires the IO and network bandwidth of the storage node, when the same job is executed, the fewer the number of times the copy is created, the better while the job execution time is guaranteed. In our statistics, the number of jobs is 100 and 500. The number of replica creations of different strategies varies with the storage capacity of the node (see Figure 7).

It can be seen from Figure 8 that when the same job is executed, as the storage capacity of the node increases, the number of copies created by the three strategies of LRU, LFU, and the economic model continues to increase, while our strategy can always keep the number of copies created. This fully shows that in this paper, we analyse the three strategies of LRU, LFU, and the economic model. When the storage capacity of the node is small, it is easy to cause frequent creation and deletion of replicas. This conclusion is correct. In addition, from the comparison between Figure 8(a) and Figure 8(b), it can be seen that the number of copies of our strategy does not increase with the increase in the number of jobs. This shows the superiority of choosing neighbour nodes to establish replicas when the storage capacity of the underlying nodes is insufficient (see Figure 8).

4.2.5. The Impact of Our Strategy on System Performance. According to the experimental results, three factors including the number of jobs, the storage capacity of nodes, and the user access mode are integrated to analyse the impact of different strategies on the performance of the educational resource grid system.

Taking into account the actual situation of the educational resource grid, a distributed user access mode is adopted. When the storage capacity of the underlying node is 100 G, 50 G, and 10 G, 500 jobs are submitted, and the

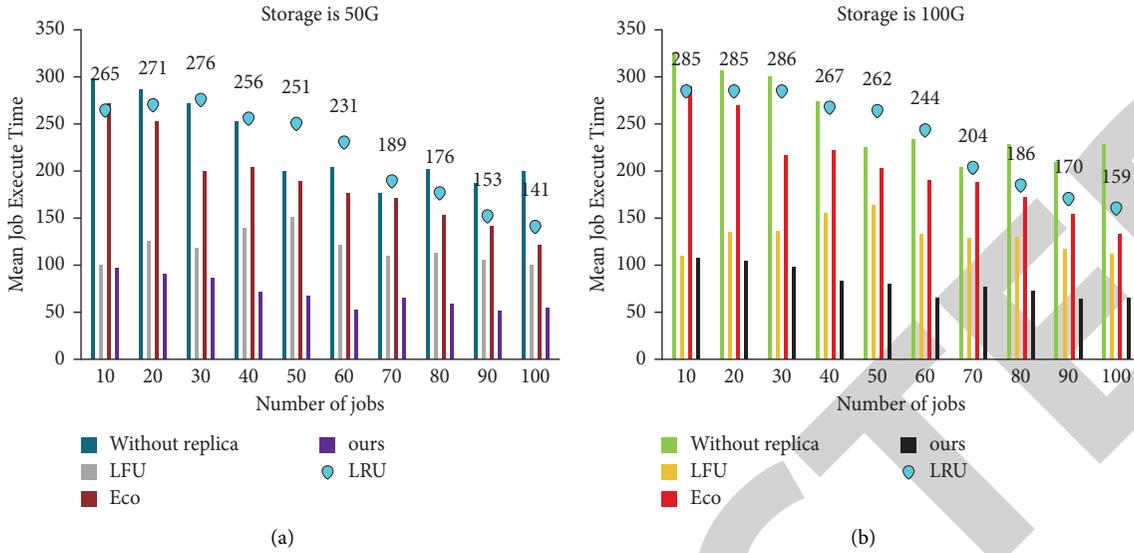


FIGURE 5: The relationship between the number of jobs and the average job execution time. (a) Storage is 50 G. (b) Storage is 100 G.

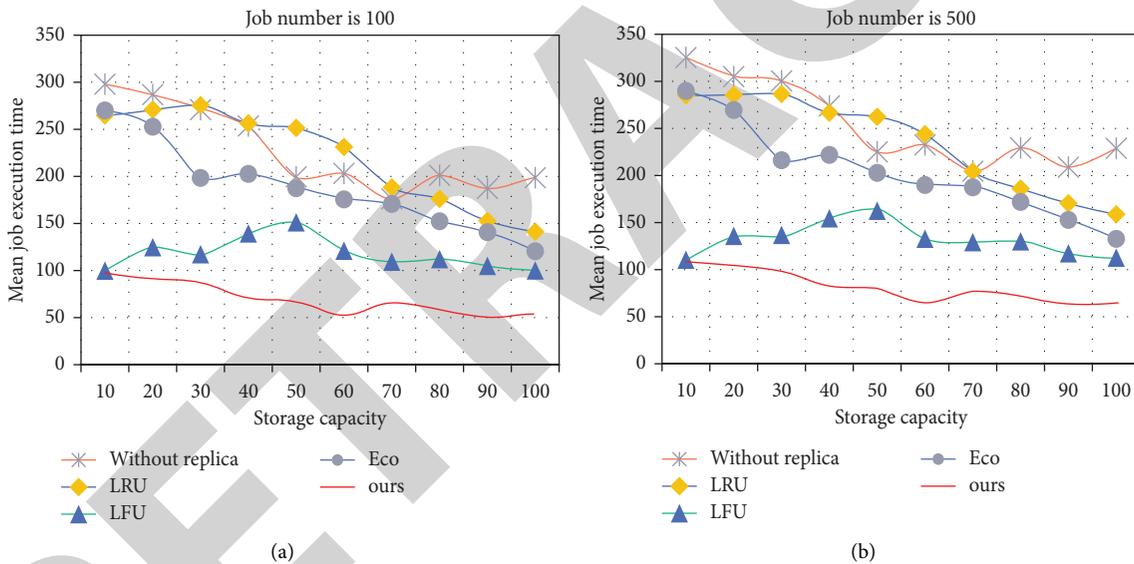


FIGURE 6: The relationship between storage capacity and average job execution time. (a) Job number is 100. (b) Job number is 500.

average job execution time and copy creation times are counted. The results are shown in Table 2.

When the node storage capacity is 100 G, that is, when the node capacity is large, there is little difference between different strategies. At this time, compared with LRU, the performance of our strategy is only improved by 0.5%. When the storage capacity of the node becomes 50 G, the average job execution time of LRU, LFU, and the economic model is not much different, but the number of copies of the economic model is significantly better than that of LRU and LFU. For our strategy, the average job execution time and the number of replica creation times are significantly better than the other three strategies. At this time, compared with the economic model, the two performance parameters of our strategy are increased by 78% and 89%, respectively. When

the storage capacity of the node is further reduced to 10 G, the performance of the economic model begins to deteriorate, and our strategy improves the system performance more obviously. At this time, compared with the LFU strategy, the two performance parameters of our strategy are increased by 84% and 95%, respectively (see Table 2).

From the above analysis, it can be seen that, in various situations, the performance of the educational resource grid system using our strategy is better than that using the other three strategies. Especially, when the node storage capacity is small, our strategy can create fewer copies. Under the premise, a smaller average job execution time is obtained. It can be seen that it is appropriate to choose our strategy in the educational resource grid where the nodes are based on the general network environment.

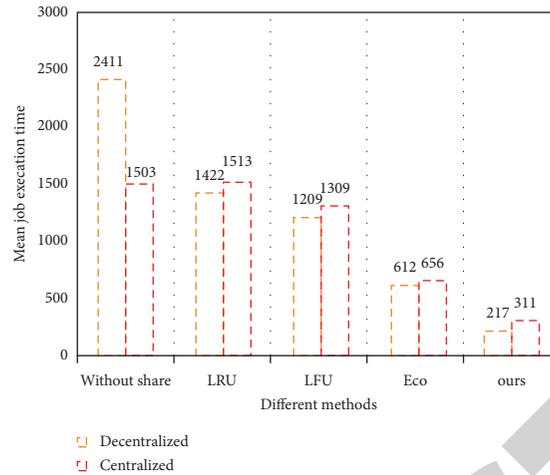


FIGURE 7: The relationship between access patterns and average job execution time.

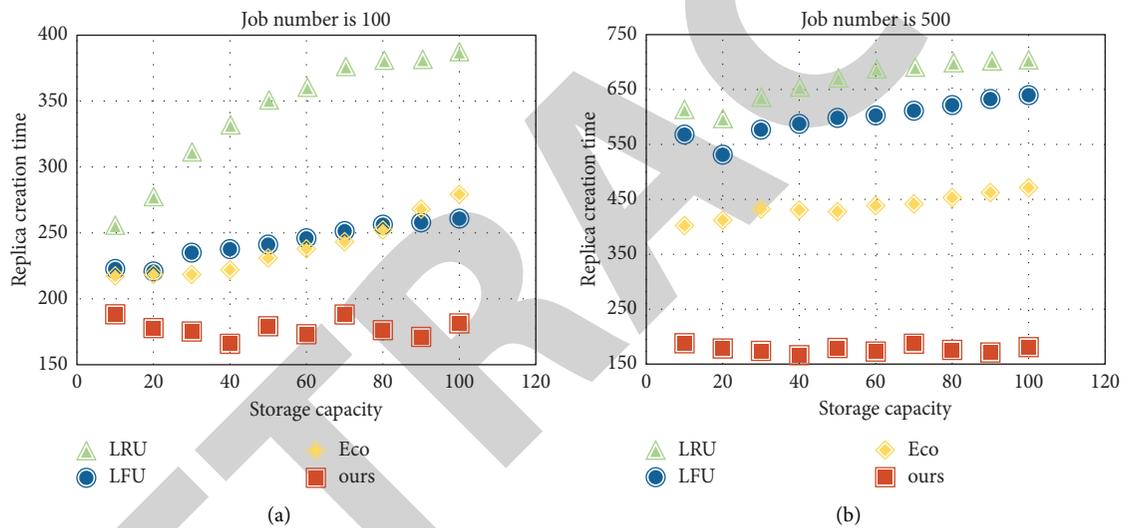


FIGURE 8: The relationship between storage capacity and the number of copies created. (a) Job number is 100. (b) Job number is 500.

TABLE 2: The impact of different strategies on grid system performance.

(G)		Without replica	LRU	LFU	Eco	Ours
10	Job execution time	2318.761	2276.512	2187.231	2213.876	521.876
	Replica creation time	0	3601	4672	2562	245
50	Job execution time	2211.876	1307.982	1588.657	1267.091	289.009
	Replica creation time	0	3287	3211	1678	365
100	Job execution time	1932.091	138.091	142.651	199.092	156.781
	Replica creation time	0	183	188	181	179

4.3. Performance of Dynamic Resource Scheduling Algorithm. This paper compares the algorithm in this paper with the user in the case of nonsharing of resources, and the user's smallest user accumulatively dominates the resource share and the resource utilization in the case of 20,100 users through experiments and executes 100 times to average to evaluate the performance of the algorithm in this paper.

4.3.1. Making the User's Accumulated Dominant Resource Share Meet the Lexicographical Optimal. It can be seen from Figure 9 that the goal of the algorithm in this paper is to make the user's accumulated dominant resource share meet the maximum and minimum. Therefore, compared with the situation where users do not share resources, the algorithm in this paper has the smallest cumulative resource share. As

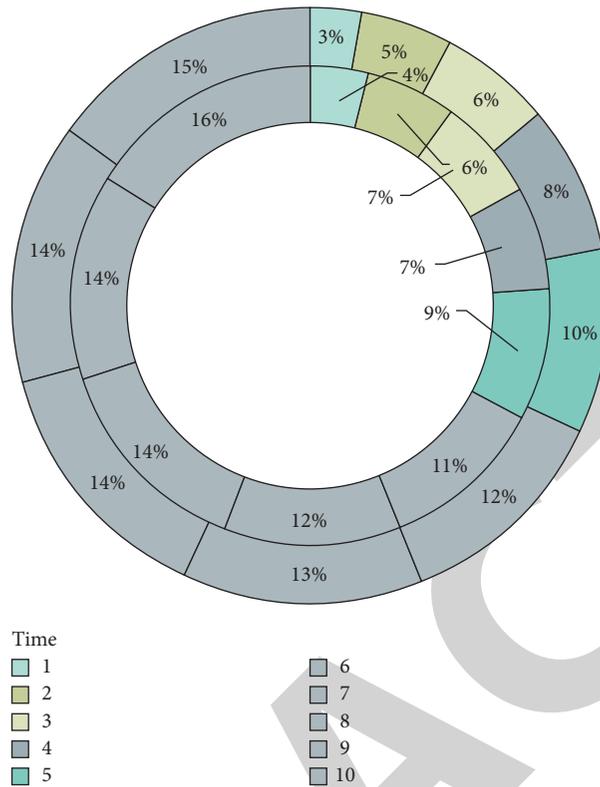


FIGURE 9: Cumulative share of resources.

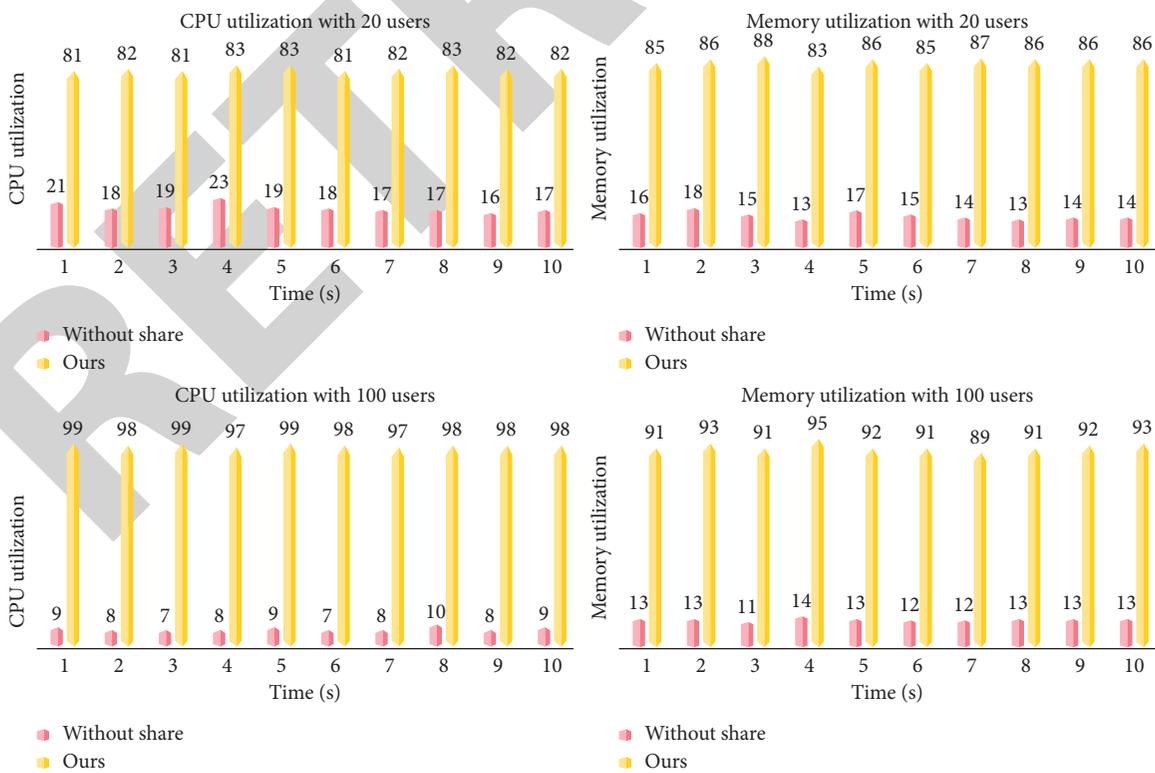


FIGURE 10: Utilization of CPU and Memory.

time goes by, the gap is getting bigger and bigger. This shows that the algorithm in this paper ensures that the user with the smallest accumulated dominant resource in the allocation process can allocate the most resources, that is, the user completes the largest number of tasks. Compared with the nonsharing algorithm, it ensures that the number of accumulated tasks completed by the user is as large as possible (see Figure 9).

4.3.2. High Resource Utilization. This paper compares the resource utilization rate under the algorithm of this paper with the resource rate of the system under the condition that users do not share resources. Figure 10 shows the resource utilization of the system when the number of users is 20 and the number of users is 100, respectively. It can be seen that the system resource utilization rate of the algorithm in this paper at any time is higher than the system resource utilization rate under the condition that users do not share resources.

In short, the algorithm in this paper satisfies fairness and enables all assigned users to complete more tasks while having high resource utilization (see Figure 10).

5. Conclusion

In the cloud computing environment, it is inevitable to realize the comprehensive sharing of network learning resources. It can not only make up for the shortcomings of traditional networks but also avoid the high cost and serious waste caused by the repeated construction of resources. On the other hand, it has promoted the development of education information, which has strong practical significance. Aiming at the problems existing in the sharing of learning resources in colleges and universities, this article first builds a smart education cloud service platform. Secondly, a hierarchical grid model of educational resource sharing is proposed. According to the characteristics of the educational resource grid, the key factors affecting the performance of the replica creation strategy are analysed, and a dynamic replica creation strategy is proposed. Considering the problem of multiuser resource allocation in the resource sharing system, a multiresource fair allocation algorithm based on the concept of resource sharing fairness is proposed. Experiments show that the grid sharing model proposed in this paper has better performance on the education cloud service platform. At the same time, when resource sharing users put forward multiple sets of time-varying resource requirements, the resource allocation algorithm proposed in this paper can ensure that users meet the optimal lexicographical order of accumulated dominant resource shares and have higher resource utilization.

Due to the need to judge user sharing defects and cumulative resource allocation in the process of algorithm execution, the algorithm does not have an advantage in allocating execution speed when there are a large number of users and virtual machines. Therefore, how to effectively optimize the allocation of tasks and virtual machines in order to improve the allocation speed will be the focus of the next research.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

- [1] L. Rajendran and R. Veilumuthu, "A cost effective cloud service for E-learning video on demand," *European Journal of Scientific Research*, vol. 55, no. 4, pp. 569–579, 2011.
- [2] S. Gupta and B. B. Gupta, "XSS-secure as a service for the platforms of online social network-based multimedia web applications in cloud," *Multimedia Tools and Applications*, vol. 77, no. 4, pp. 4829–4861, 2018.
- [3] J. H. Bing, "University Ideological and Political Multimedia Network Teaching Based on MOOC," in *Proceedings of the International Conference on E-Learning, E-Education, and Online Training*, pp. 167–180, Xinxiang, China, June 2021.
- [4] S. Mayouf, H. Alaswad, S. Aljeshi, A. Tarafa, and W. Elmedany, "A hybrid circuits-cloud: development of a low-cost secure cloud-based collaborative platform for A/D circuits in virtual hardware E-lab," *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 1197–1209, 2021.
- [5] J. J. Youn, S. J. Jeong, K. H. Lee, and K. Byung Man, "Development and Application of a Digital Curation System to Promote Total Creative Personality Based on Multimedia," *Multimedia Tools and Applications*, vol. 80, pp. 34369–34387, 2021.
- [6] A. McDonald, H. McGowan, M. Dollinger, R. Naylor, and H. Khosravi, "Repositioning students as co-creators of curriculum for online learning resources," *Australasian Journal of Educational Technology*, vol. 37, no. 6, pp. 102–118, 2021.
- [7] S. Namasudra, "Data access control in the cloud computing environment for bioinformatics," *International Journal of Applied Research in Bioinformatics*, vol. 11, no. 1, pp. 40–50, 2021.
- [8] S. Khan, A. Al-Dmour, V. Bali, M. R. Rabbani, and K. Thirunavukkarasu, "Cloud computing based futuristic educational model for virtual learning," *Journal of Statistics & Management Systems*, vol. 24, no. 2, pp. 357–385, 2021.
- [9] S. Yao, D. Li, A. Yohannes, and H. Song, "Exploration for network distance teaching and resource sharing system for higher education in epidemic situation of COVID-19," *Procedia Computer Science*, vol. 183, pp. 807–813, 2021.
- [10] Z. Jiang, S. Yuan, J. Ma, and Q. Wang, "The evolution of production scheduling from Industry 3.0 through Industry 4.0," *International Journal of Production Research*, pp. 1–21, 2021.
- [11] J. Zhou and W. Zou, "Construction and application of wenhua education cloud based on huawei Fusion Sphere," in *Proceedings of the 2021 IEEE international conference on artificial intelligence and computer applications (ICAICA)*, pp. 477–481, Dalian, China, June 2021.
- [12] R. G. Barreto, L. Aversari, C. N. A. P. Gomes, and N. C. Q. Lino, "H-KaaS: a Knowledge-as-a-Service architecture for E-health," *Brazilian Journal of Biological Sciences*, vol. 5, no. 9, pp. 3–12, 2018.
- [13] J. Martinez-Guijosa, A. López-Alonso, C. Gortazar, A. Pelayo, J. T. María, and V. Joaquín, "Shared use of mineral supplement in extensive farming and its potential for infection

- transmission at the wildlife-livestock interface,” *European Journal of Wildlife Research*, vol. 67, no. 3, pp. 1–9, 2021.
- [14] H. Aldowah, H. Al-Samarraie, A. I. Alzahrani, and N. Alalwan, “Factors affecting student dropout in MOOCs: a cause and effect decision-making model,” *Journal of Computing in Higher Education*, vol. 32, no. 2, pp. 429–454, 2020.
- [15] Y. Liu, “Design and implementation of multimedia teaching platform based on SOA architecture,” *Multimedia Tools and Applications*, vol. 79, no. 15–16, pp. 10899–10914, 2020.
- [16] H. Al-Samarraie and N. Saeed, “A systematic review of cloud computing tools for collaborative learning: opportunities and challenges to the blended-learning environment,” *Computers & Education*, vol. 124, pp. 77–91, 2018.
- [17] M. R. Sandgani and S. Sirouspour, “Coordinated optimal dispatch of energy storage in a network of grid-connected microgrids,” *IEEE Transactions on Sustainable Energy*, vol. 8, no. 3, pp. 1166–1176, 2017.
- [18] H. Shirado, G. Iosifidis, L. Tassioulas, and A. C. Nicholas, “Resource sharing in technologically defined social networks [J],” *Nature Communications*, vol. 10, no. 1, pp. 1–10, 2019.
- [19] A. Weigard and C. Huang-Pollock, “The role of speed in ADHD-related working memory deficits,” *Clinical Psychological Science*, vol. 5, no. 2, pp. 195–211, 2017.
- [20] N. Sun, Y. Li, L. Ma, W. Chen, and D. Cynthia, “Research on cloud computing in the resource sharing system of university library services,” *Evolutionary Intelligence*, vol. 12, no. 3, pp. 377–384, 2019.
- [21] A. Hussain, M. Aleem, M. A. Iqbal, and M. A. Islam, “Investigation of cloud scheduling algorithms for resource utilization using CloudSim,” *Computing and Informatics*, vol. 38, no. 3, pp. 525–554, 2019.
- [22] M. Kalra and S. Singh, “An intelligent water drops-based approach for workflow scheduling with balanced resource utilisation in cloud computing,” *International Journal of Grid and Utility Computing*, vol. 10, no. 5, pp. 528–544, 2019.
- [23] X. Xu, X. Zhang, M. Khan, W. Dou, S. Xue, and S. Yu, “A balanced virtual machine scheduling method for energy-performance trade-offs in cyber-physical cloud systems,” *Future Generation Computer Systems*, vol. 105, pp. 789–799, 2020.
- [24] Z. Yimin, S. Guojun, and Y. Xiaoguang, “Cloud Service Selection Optimization Method Based on Parallel Discrete Particle Swarm Optimization,” in *Proceedings of the 2018 Chinese Control and Decision Conference (CCDC)*, pp. 2103–2107, Shenyang, China, June 2018.
- [25] Y. Sun, F. Lin, and H. Xu, “Multi-objective optimization of resource scheduling in fog computing using an improved NSGA-II,” *Wireless Personal Communications*, vol. 102, no. 2, pp. 1369–1385, 2018.
- [26] M. Kasparick, M. Schmitz, B. Andersen et al., “OR.NET: a service-oriented architecture for safe and dynamic medical device interoperability,” *Biomedical Engineering/Biomedizinische Technik*, vol. 63, no. 1, pp. 11–30, 2018.
- [27] A. Garcia-de-Prado, G. Ortiz, and J. Boubeta-Puig, “COLLECT: COLLaborativE ConText-aware service oriented architecture for intelligent decision-making in the Internet of Things,” *Expert Systems with Applications*, vol. 85, pp. 231–248, 2017.
- [28] P. Jiang, M. Elag, P. Kumar, S. D. Peckham, L. Marini, and L. Rui, “A service-oriented architecture for coupling web service models using the Basic Model Interface (BMI),” *Environmental Modelling & Software*, vol. 92, pp. 107–118, 2017.
- [29] B. Andersen, M. Kasparick, H. Ulrich et al., “Connecting the clinical IT infrastructure to a service-oriented architecture of medical devices,” *Biomedical Engineering/Biomedizinische Technik*, vol. 63, no. 1, pp. 57–68, 2018.
- [30] Y. A. M. Qasem, R. Abdullah, Y. Y. Jusoh, R. Atan, and S. Asadi, “Cloud computing adoption in higher education institutions: a systematic review,” *IEEE Access*, vol. 7, pp. 63722–63744, 2019.
- [31] N. Ramu, V. Pandi, J. D. Lazarus, and S. Radhakrishnan, “A novel trust model for secure group communication in distributed computing,” *Journal of Organizational and End User Computing*, vol. 32, no. 3, pp. 1–14, 2020.
- [32] S. S. Gill and A. Shaghaghi, “Security-Aware autonomic allocation of cloud resources,” *Journal of Organizational and End User Computing*, vol. 32, no. 3, pp. 15–22, 2020.
- [33] R. Kanniga Devi, M. Gurusamy, and P. Vijayakumar, “An efficient cloud data center allocation to the source of requests,” *Journal of Organizational and End User Computing*, vol. 32, no. 3, pp. 23–36, 2020.
- [34] M. B. Ali, T. Wood-Harper, and M. Mohamad, “Benefits and challenges of cloud computing adoption and usage in higher education,” *International Journal of Enterprise Information Systems*, vol. 14, no. 4, pp. 64–77, 2018.
- [35] I. Arpaci, “Antecedents and consequences of cloud computing adoption in education to achieve knowledge management,” *Computers in Human Behavior*, vol. 70, pp. 382–390, 2017.
- [36] C. Lu, K. Ye, G. Xu, and B. Tongxin, “Imbalance in the Cloud: An Analysis on Alibaba Cluster Trace,” in *Proceedings of the 2017 IEEE International Conference on Big Data (Big Data)*, pp. 2884–2892, Boston, MA, USA, December 2017.