An Improved Ant Colony Algorithm for Virtual Resource Scheduling in Cloud Computing

Methods to Improve the Performance of Virtual Resource Scheduling

Chunlei Zhong*[1], Gang Yang[2]

Huai'an Bioengineering Branch Institute, Jiangsu Union Technical Institute,
Huai'an 223200, China[1]
College of Teacher Education, Wenzhou University,
Wenzhou 325035, China[2]

Abstract—In order to solve the problems of uneven spatial distribution of data nodes and unclear weight relationship of virtual scheduling features in cloud computing platform, a virtual resource scheduling method based on improved ant colony algorithm is studied and designed to improve the performance of virtual resource scheduling in cloud computing platform by this method. After analyzing the information resource sequence change of the cloud computing platform, according to the STR - Tree partition graph, a simulated annealing-based algorithm is employed to classify the resource types after optimal scheduling into IO types, middle types and CPU types, and the time span and load balance are set as the measurement indexes. The simulation results show that after applying this method, the occupied resources of the main platform are 535 MB, which are much lower than the other two comparison algorithms, and the method has improved the allocation rationality, resource balance, maximum queue length and energy consumption. This result indicates that applying this virtual resource scheduling method can effectively improve the intelligent scheduling of virtual resources in the cloud computing platform.

Keywords—*Improved ant colony algorithm; cloud computing; virtual resources; intelligent scheduling*

I. INTRODUCTION

Cloud computing is the mainstream way to provide services to users relying on resources deployed over numerous physical machines to be available regardless of their location. *Virtualization* is the key concept employed to provide a logical abstraction over physical resources running on data centers. This model considers data centers as nodes suitable for cloud computing and provides a unified and easy-to-use resource organization.

Different users request access and obtain resources according to their own needs, without paying attention to the specific location of resources. In order to provide users with the same or similar functions as the real layer, and to facilitate the stable operation of the upper layer system in the middle layer, cloud virtual resource scheduling is needed. Unlike other distributed computing platforms, virtualization provides more computing power and flexibility for applications, an advantage that fits well with the needs and main concepts of cloud computing [1].

In order to improve the usage of virtual resource in cloud computing, which considers hosting, network traffic, and other costs optimization, is not uncommon for multiple system resources, even from different organizations, to share a single physical server. Besides, a service application can also run over multiple service provider platforms to be accessed by multiple clients simultaneously. This scenario is tackled by the multi-tenant scheduling process, which aims for high resource utilization using the cloud computing network.

Laith & Ali [2] presented a hybrid ant colony optimization algorithm based on elitist differential evolution is proposed to solve the multi-objective task scheduling problem in cloud computing environment, which minimizes the maximum completion time, maximizes the resource utilization, and uses elitist differential evolution as a local search technique to improve its utilization ability and avoid falling into local optimization. In [3], by satisfying the constraint of security requirements, we find an optimal scheduling algorithm with minimum completion time and cost, and users can submit their security requirements to the cloud provider during the negotiation period. However, none of the methods mentioned before has solved the problems of indefinite weight relation of virtual scheduling features and uneven distribution of cloud platform information space, so it is difficult to realize virtual resource scheduling efficiently.

In order to solve the problems of the above methods, a cloud computing virtual resource scheduling method based on improved ant colony algorithm is proposed. The overall research technical route of this method is as follows:

1) This paper analyzes the characteristics of the sequence change of information resources on the cloud computing platform, and establishes the index relationship between spatial information objects through STR tree partition graph to solve the problem of data imbalance.

2) Simulated annealing algorithm is used to divide the optimized resource types into IO type, intermediate type and CPU type, and time span and load balance are used as indicators to complete the scheduling of virtual resources.

3) Experimental verification, taking the occupied space, load fairness, Allocation Rationality, resource balance, the longest queue length and energy consumption value as the test

indicators, the comparative tests of different methods are carried out.

By updating the above techniques, the proposed cloud computing virtual resource scheduling method can effectively solve the problems of uncertain feature weight relationship and uneven information space distribution in the virtual scheduling process, and it is expected that this method will improve the performance of cloud computing virtual resource scheduling and enhance new ideas for the field of virtual resource scheduling.

The structure of the study is divided into four main areas as follows. The first part introduces the current research status of cloud computing resource scheduling and compares it with the proposed algorithm. The second part introduces the virtual resource scheduling method based on the improved ant colony algorithm and describes its practical computing process. The third part is a performance test and application analysis of the virtual resource scheduling algorithm proposed in the study, and the results are organized and analyzed. The last part is a summary and discussion of the whole paper.

II. CLOUD COMPUTING PLATFORM VIRTUALIZATION RESOURCE EXTRACTION

A. Changes in the Sequence of Communication Network Information Resources of Cloud Computing Platforms

Each node in the cloud computing platform sends information to the center node to report according to the system load and user's demand [4]. In most cases, passive real-time monitoring is used to schedule resources. The client connects to the cloud server based on the control server.

When the client sends the computer application, Computer application is to study the theory, method, technology and system of computer application in various fields. Computer application is an activity that gives guidelines on how to participate and implement in social activities. The cloud server can store some data related to the user, Distributed storage is used to consolidate a large number of servers into a supercomputer, providing a large amount of data storage and processing services. Distributed file systems and distributed databases allow access to common storage resources and realize IO sharing of application data files. By integrating computer, the resource scheduling server sends the calculation results to the user's computer, and the computer client returns the results. Calculate the data source through the server and wait for the next user request to obtain the communication network model of the cloud computing platform, as shown in Fig. 1.

In Fig. 2, the resource storage module adopts idt72v3680 memory produced by IDT Company. Idt72v3680 is one of the high-density supersynctm II 36 bit series memories idt72v3640 ~ 3690 of IDT company. Its storage structure is 16384×36. This series of CMOS process FIFO (first in first out) chips have great depth. The storage structure of this memory is $16384 * 36$, and the storage space can reach 1.125MB [5-8]. The internal circuit diagram of the memory is shown in Fig. 3.

According to Fig. 3, Memory is a memory device in a computer system that stores programs and data. All information in the computer, including input raw data, computer programs, intermediate running results and final running results are stored in the memory. It stores and retrieves information according to the location specified by the controller. The main function of the memory is to store programs and various data, and to complete the program or data access at high speed and automatically during the running of the computer. The memory has an independent chip structure. The storage rate of the memory can reach 100MHz, the working voltage is 3.15-3.45v, and the maximum working current is 400mA. In addition, the memory uses CMOS technology, belongs to high-density memory, and has great depth. During operation, idt72v3680 memory, as the transmission link between acquisition module and control module, can still realize efficient data storage without additional coordination under the condition of different rates of the two sides.
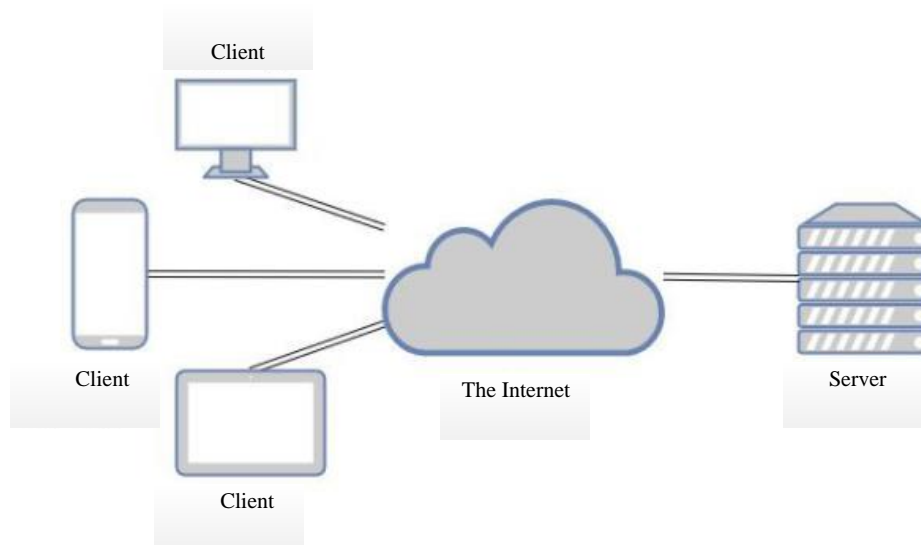

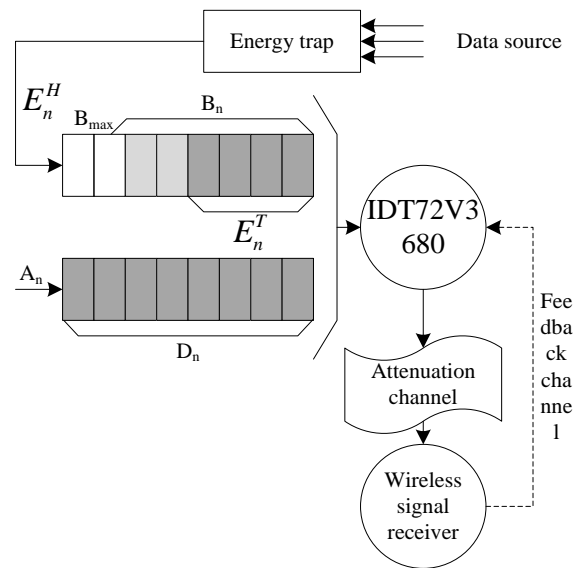Fig. 1. The cloud server

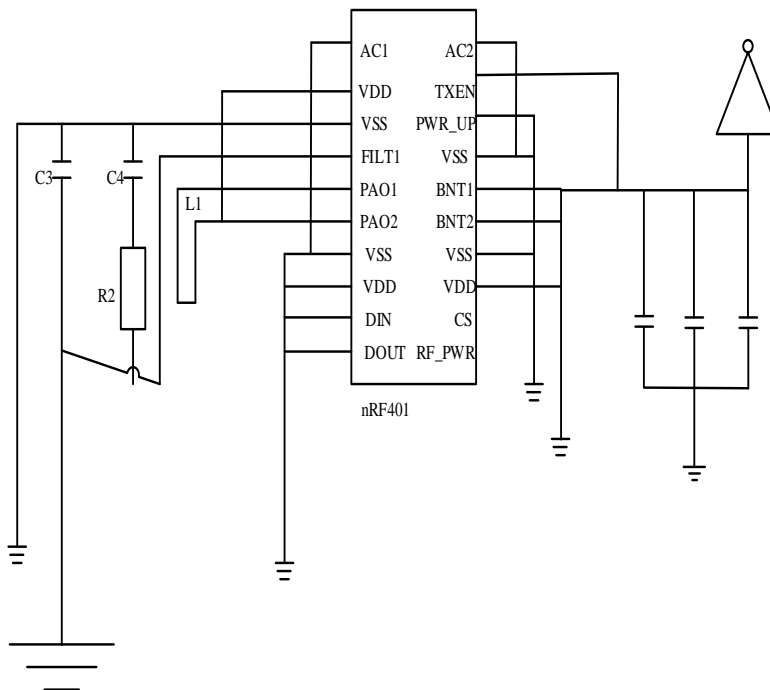Fig. 2.   Communication network model of cloud computing platform

Fig. 3.   Internal circuit diagram of memory

If it is simply stored, it is difficult to complete the subsequent indexing steps. Therefore, in the process of establishing the storage mechanism, it is necessary to consider the indexing efficiency and divide and encode the data attributes [9]. The types and sizes of cloud data are different, and it is difficult to divide them. It is necessary to develop a reasonable strategy to divide them effectively [10]. In the process of partition coding, only the size, type and spatial distribution of data need to be considered to ensure the relative integrity of the cloud platform spatial information entity as far as possible, so as not to divide it into multiple scattered data blocks. If two data blocks are adjacent data in the information space, they can be consciously divided into the same interval and given adjacent codes. In addition, if the data volume difference between the divided intervals is too large, it is easy to lead to the low balance of the distributed cluster, resulting in logical vulnerabilities. Therefore, when coding data nodes in partitions, this kind of data imbalance can be solved through STR tree. The main operation mechanism of STR tree is to establish the index relationship between spatial information objects through the external matrix between two data nodes, and code and sort their abscissa and ordinate between the two dimensions.

In the data storage space of a cloud platform, it is first necessary to determine the number of data partitions n, code

other data entities into $A, B, C, \cdots, X, Y, Z$, etc., and divide the average total amount of data block space . $\sqrt{n}$ .. In this way, the horizontal and vertical coordinates of all data can be indexed to $\dfrac{N}{\sqrt{n}}$, all data coordinates can be grouped into different data blocks, and all vertical and horizontal coordinates are $\dfrac{N}{\sqrt{n}}$. Determine the partition coding basis of STR tree, as shown in Fig. 4.
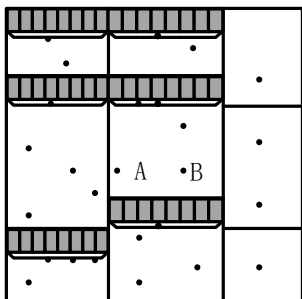

Fig. 4.   STR tree zoning diagram

As shown in Fig. 4, STR tree can effectively solve the problem of uneven spatial distribution of information of data nodes in the cloud platform, so as to reduce the repeatability of data nodes. Before designing the virtual resource scheduling model of cloud computing platform, firstly, set the number of cellular users in heterogeneous units as m and the number of communication network users of cloud computing platform as N, $M = \{1, 2, \ldots, M\}$ and $N = \{1, 2, \ldots, N\}$. Multiple uplink resources are used simultaneously in the scheduling process, and the resource blocks do not affect each other.

In time period T, the transmission power of the nth cloud computing platform communication network at the m-th cellular user and its uplink resources is $P^m(t)$ and $P_n^m(t)$ respectively; The signal to dryness ratios of N on M to cellular users in time period T are $SINR_n^m(t)$ and $SINR^m(t)$ respectively. Set $i, j \in N$. Within the time period T [11], the communication between the cloud computing platform communication network and the cellular user is dry, as shown in formula (1):

$$SINR_i^m(t) = \frac{P_i^m(t) h_{ii}(t)}{N_0 + \sum_{j \in S_m, i \neq j} P_j^m(t) h_{ji}(t) + P^m(t) h_{mi}(t)} \quad (1)$$

Among them, $h_{ii}(t), h_{ji}(t), h_{mi}(t)$, and are the channel power gain of the communication network of the cloud computing platform, which are respectively used to describe the channel power gain of uplink resources from the information transmission end to the information receiving end at different times, and calculate the ratio of output power and input power of uplink resources at different times. If all multiplexing resource sets are S_m and the noise power spectral density is N_0, the signal to dryness ratio of the primary user is:

$$SINR^m(t) = \frac{P^m(t) h_{mB}(t)}{N_0 + \sum_{j \in S_m} P_j^m(t) h_{jB}(t)} \quad (2)$$

Among them, the channel power gains from the user and the network to the base station are $h_{mB}(t)$, $h_{jB}(t)$ [12]. Calculate the maximum communication rate of cloud computing platform communication network to cellular users through Shannon channel capacity calculation formula:

$$R^m\left(P^m\right)_t = \log_2\left(1 + SINR^m(t)\right) \quad (3)$$

Where P^m represents the transmission power passing on the uplink resource.

Only when the uplink interference of the communication network users of the cloud computing platform to the cellular users is minimized can the communication efficiency of the cellular users be maximized. The resource allocation of the resource block can be expressed by formula (4), the time period is t, and the signal to dryness ratio threshold of the cellular user is $\xi_{Threshold}^m$. Average maximum throughput of communication network resource scheduling in t [13]:

$$\arg\max = \frac{1}{T}\left[\sum_{t=1}^{T}\left(\sum_{m=1}^{N} \log_2\left(1 + SINR^m(t)\right)\right)\right] \quad (4)$$

Among them, $SINR^m(t) \geq \xi_{Threshold}^m$. There are network information resources arriving at the wireless signal transmitter in each time slot, and X_n is used to represent the number of information resources contained in the communication network, so the change process of the information resource sequence of the communication network can be described as

$$D_{n+1} = \max\{D_n - X_n, 0\} + A_n \quad (5)$$

In formula (1), D_{n+1} and D_n respectively represent the information resource sequence $n+1$ of the communication network of the n th and th time slots; A_n represents the number of data packets arriving at the wireless signal transmitter in the time slot.

B. Cloud Platform Virtualization Resource Association Feature Extraction

Advantages of ant colony algorithm over other swarm intelligence algorithms: (1) the algorithm has general applicability: as long as the problem can be transformed into a connected graph, it can generally be solved by ant colony algorithm in scenes with low real-time requirements. (2) Randomness and certainty are interdependent: the heuristic function represents a priori knowledge, its weight represents the degree of certainty in selection, and the pheromone update criterion represents the mutual restriction between the two. Pheromone update, as the performance mechanism of positive feedback, reflects the degree to which the path is focused (certainty), while the volatilization coefficient participates as the negative feedback mechanism, the excellent path with too

much pheromone behind has the opportunity to catch up. The roulette algorithm increases the possibility of exploring a larger solution space (randomness). Under normal circumstances, the cloud platform communication process acts on the attenuation channel, which means that the network channel gain remains the same in any time slot, but there are some differences in the network channel states in different time slots. If the wireless network is in the form of adaptive coding modulation, the calculation formula of information resource transmission rate of communication network is

$$R_n = W \log_2 \left(1 + \frac{G_n P_n}{\Gamma W N_0} \right) \quad (6)$$

Wherein, W represents the channel bandwidth of the communication network; G_n represents the channel gain in the time slot; P_n represents the transmission power of the wireless communication network; Γ represents the auxiliary constant; N_0 represents the noise power density.

Then the energy required for information resource scheduling of communication network in time slot is

$$E_n^T = P_n \times \Delta T \quad (7)$$

Where ΔT represents the length of each time slot.

Since there are many task types of cloud computing resources and applied resources, only tasks with matching types will be assigned to corresponding computing resources for task processing. In addition, multiple computing resources can work at the same time. The steps are as follows:

Step 1: after obtaining the energy data required for scheduling, place k ants on the energy samples required for scheduling, record the number of samples as M, and open the sample scheduling feature association pair at the same time;

Step 2: ants select the energy characteristics required for scheduling with transfer probability. At the same time, taboo table $tabu_k$ and temporary pool $temppool(k)$ are set, $tabu_k$ is used to record the sample scheduling feature association pair selected by ant k, and application $temppool(k)$ is used to save the probability value selected by ant [14]. Among them, under the limitation of energy required for information resource scheduling of communication network in time slot, the calculation formula of transfer probability is

$$p_{ij}^k(t) = \begin{cases} \dfrac{\alpha \cdot \tau_{ij} + \beta \cdot \eta_{ij}}{\sum\limits_{r \neq tabu_k} \alpha \cdot \tau_{ij} + \beta \cdot \eta_{ij}} & j \neq tabu_k \\ 0 & others \end{cases} \quad (8)$$

Wherein, α represents the weight parameter of pheromone quantity τ_{ij}; β represents the weight parameter of visibility parameter η_{ij}, which is calculated by $1/d_{ij}$, where d_{ij} represents the distance of a sample scheduling

feature association pair (i, j) among a plurality of pedestrians.

Step 3: in the process of path optimization, the population ant will leave the pheromone quantity τ_{ij} in real time. The pheromone is the pheromone related to the scheduling order of the scheduled task. The scheduling order is the order in which the tasks in the taboo table are arranged to be processed by the corresponding computing resources. The pheromone here indicates whether a computing task may be assigned a specific order, In order to prevent the pheromone quantity of a path (Association pair (i, j)) from increasing rapidly, resulting in the algorithm falling into local optimization, the initial pheromone quantity is set to τ_{max} and the value range is $\tau_{ij} \in [\tau_{min}, \tau_{max}]$;

Step 4: in the process of path selection [15-16], update the pheromone quantity based on local update rules. After the path selection, the pheromone update rule is changed to the global update rule. The global update rule expression is

$$\tau_{ij}(t+1) = (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^* \quad (9)$$

Wherein, ρ represents the pheromone quantity update coefficient; $(1 - \rho)$ represents the pheromone residue coefficient; $\Delta\tau_{ij}$ means that (i, j) path pheromones are updated according to the path length ranking information of $\sigma - 1$ excellent ants, and the calculation formula is

$$\Delta\tau_{ij} = \sum_{\mu=1}^{\sigma-1} \Delta\tau_{ij}^\mu \quad (10)$$

Wherein, μ represents the ranking of ants participating in path search in ant colony; $\Delta\tau_{ij}^\mu$ indicates the amount of pheromone retained by the first ant on the associated pair.

Step 5: after determining the incidence matrix, record all the sample scheduling feature information to obtain the correct multi scheduling energy target feature data association results.

C. Cloud Computing Virtualization Resource Scheduling based on Improved Ant Colony Algorithm

The original ant colony algorithm also has many shortcomings that are difficult to overcome by itself:

1) The ant colony algorithm framework operates on the basis of roulette algorithm. In the initial stage of solution space exploration, the utilization of prior knowledge is insufficient, resulting in a long time to search for a satisfactory solution. The certainty of the algorithm should be appropriately increased in the whole algorithm cycle [17-18].

2) After several rounds of search iteration, the evolution rate of the solution often decreases sharply, and even the basic gap between the found solutions is small, that is, the search stalls. The weight relationship between randomness and certainty should be dynamically adjusted according to different operation stages of the algorithm [19-20].

Therefore, simulated annealing algorithm is used to

optimize ant colony algorithm to overcome the above problems.

Combinatorial optimization problem is to solve the optimal solution. In the time period, set a non-negative objective function representing the solution goal on the ant colony path. The solid annealing process and combinatorial optimization problem have certain similarities, and the main advantages are as follows:

1) Perform complex area search: The algorithm is more suitable for searching in complex regions and obtaining regions with higher region values.

2) Good parallelism: The good parallelism of ant colony algorithm and simulated annealing algorithm can effectively solve various nonlinear problems. The good parallelism of the ant colony algorithm and the simulated annealing algorithm can solve various nonlinear problems effectively. The ant colony algorithm has very good time efficiency, but the ability to find the optimal solution needs to be improved, while the simulated annealing algorithm has very good ability to find the optimal solution, but the time complexity is high. By parallelizing the two, the advantages of both can be combined and the disadvantages of both can be compensated, and the final parallel algorithm not only has good ability to find the optimal solution, but also its time efficiency is high.

3) Search with object functions: Directly convert the objective function into fitness value to determine the search scope and direction in the next step.

Due to the slow development of physical state to low-energy state, in order to obtain better results, simulated annealing algorithm mainly selects the state with important contribution rate in the process of sampling.

Set the position representation state of the particle solid as the initial state, the energy is fixed, randomly form a small displacement change and produce a new state, set this state as an important state, the next state of the solid is sum, and the factor ratio corresponding to the probability ratio of the two states can be expressed as:

$$p(t) = \begin{cases} \left(E_i - E_j \right) \\ \dfrac{1}{Z(t)} \exp\left(-\dfrac{E_j - E_i}{k_B(t)} \right) \end{cases} \quad (11)$$

Where Z (T) represents the modified pheromone value on the ant path (i, j) at time t. It is set that the random number is a value formed by the random number generator, and the value range is within the range of $[0,1]$. Since p represents a value less than 1, assume $p \geq A$, indicating that it is sufficient to maintain the current state.

The resource types after optimized scheduling are divided into IO type, intermediate type and CPU type. When optimizing the scheduling, the resources are divided in combination with the running time sequence. The resulting resource division scheme is shown in Fig. 5.
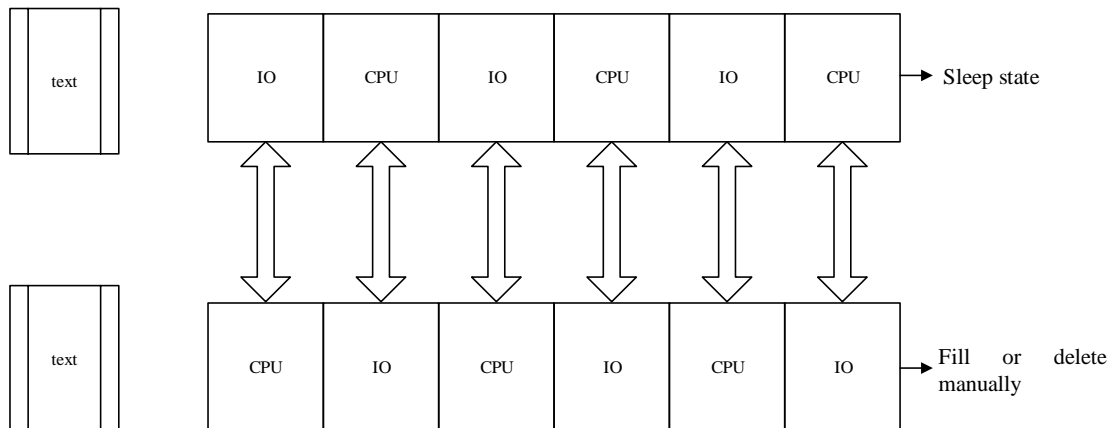

Fig. 5. Resource division scenario

As shown in Fig. 5, resource A and resource B consume the same disk (IO) and CPU, so both can be divided into intermediate types of resources. However, it can be seen from Fig. 4 that the time occupied by disk (IO) and CPU is completely crossed. Therefore, in the process of resource scheduling, the existence of AA or BB should be avoided as far as possible. In addition, missing and redundant data will be automatically filled or manually deleted in the process of resource optimal scheduling.

It is worth noting that the goal of the algorithm is to limit the number of physical hosts providing services in the cloud platform, and the trigger of any migration scheme is to migrate the virtual machine to an active physical machine.

However, when the current active physical machine cannot meet the requirements of the virtual machine to be migrated. The dormant physical machine will be activated as the destination physical host in the migration plan. A physical host is switched to sleep if and only if the virtual machines on it have been migrated. That is, when a physical host does not provide resources for any virtual machine, it will be converted to sleep state.

The important state is mainly set as the state with small probability acceptance and large current energy difference; In the high temperature state, the new state with large energy difference from the current state is accepted with a high probability and set as the important state.

In the actual application scenario of scheduling algorithm, due to the complexity and heterogeneity of cloud platform resources, the performance of physical hosts varies greatly, and the state of virtual machines relying on them will change dynamically with the deepening of task execution. In this case, it is easy to cause a large number of tasks to flow into the waiting queue of high-performance virtual machines and cannot be executed, resulting in service degradation and user experience degradation. On the other hand, there are a large number of idle virtual machines in the cloud platform, which are not favored by tasks due to limited performance, resulting in a great waste of cloud platform resources. As a result, the service life of overloaded machines is shortened, and the completion time of single tasks is not satisfactory. In order to solve this problem, it is necessary to analyze the needs of different users, calculate the matching degree of each demand corresponding to the task to the resource node, and add the matching factor Tm_i. Then the scheduling requirements in important states are expressed as follows:

$$Match_{ij} = \frac{1}{\sqrt{\sum_{i=1}^{5}\left(Tm_i - Vm_j\right)^2}}$$

(12)

Set the mapping sequence Tm_i representing tasks and resources, Vm_j represents a sequence node. And the load balance degree can be expressed as:

$$Load(X) = \sqrt{\sum_{i=1}^{n}\left(1 - C_1\frac{Y_i}{Lv_i}\right)^2}$$

(13)

Calculate the computing power of each resource node by weighting, that is:

$$Lv_i = w_1 \times n(v_i) + w_2 \times p(v_i) + w_3 \times r(v_i)$$ (14)

When pheromones accumulate to a certain extent, positive feedback suppression is needed, because with the deepening of iterative search, the total amount of pheromones rises to a certain threshold, which is prone to excessive positive feedback, that is, the influence of pheromones on ant colony covers the prior effect of heuristic function. In addition to ensuring the update speed of pheromones and highlighting the differences between high-quality paths and general paths, it is very important to comprehensively improve the load balance of resources. Therefore, the time span and load balance are set as the measurement indicators:

$$\begin{cases} F(X) = Load(X) \cdot Makspan(X) \\ Makspan(X) = \sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij} \times E_{ij} \end{cases}$$

(15)

Above, mainly through simulated annealing algorithm for data center resource scheduling, the specific operation process is as follows:

1) Initialize all pheromones;

2) Setting up the main objective function and initializing each parameter;

3) When the ant starts the cycle, it needs to place all the tasks on a single resource.

4) Calculate the matching factor, set it as heuristic information, and select nodes in combination with the transfer probability.

5) Calculating the load balance degree and the value of the objective function, and the optimal solution of the ant colony algorithm.

6) The simulated annealing algorithm is used to optimize the path and update the pheromone.

7) Loop through the above steps until the maximum number of iterations is reached, then stop the operation and vice versa return step (3).

III. SIMULATION EXPERIMENT

A. Experimental Data and Process

In order to verify the scheduling performance of the proposed method, comparative experiments are carried out. Provide 600 to 1000 cloud tasks in the resource submission window, and use the random number generator to randomly generate 50 waiting tasks with different requirements for cloud computing resources in the scheduling time window. It is estimated that the starting processing time is 0. Assuming that the time window is 40 minutes in length, the iteration is terminated under the condition that 6,000 ants all complete the iteration and the number of intelligent population is 100. When there is no change in the results of 10 consecutive iterations, the operation of the algorithm is stopped and an error is reported. Each set of data was analyzed 10 times and the average value was taken, and a parameter change was made to observe its impact on the delay time and solving efficiency of the task.

Hybrid cloud is a more advantageous infrastructure, which flexibly combines the internal capabilities of the system with external service resources, and ensures low cost. Hybrid cloud is a model that organically integrates public cloud and private cloud, allowing users to choose between the privacy of private cloud and the flexibility and cheapness of public cloud. Hybrid cloud can provide enterprises with more flexible cloud computing solutions. In the next few years, hybrid cloud will become the leading enterprise IT architecture, and more and more enterprises will benefit from hybrid cloud services. Considering the transport efficiency elements of cloud platform virtualization resources, a more commonly used cloud computing architecture is chosen as the experimental basis, as shown in Fig. 6.
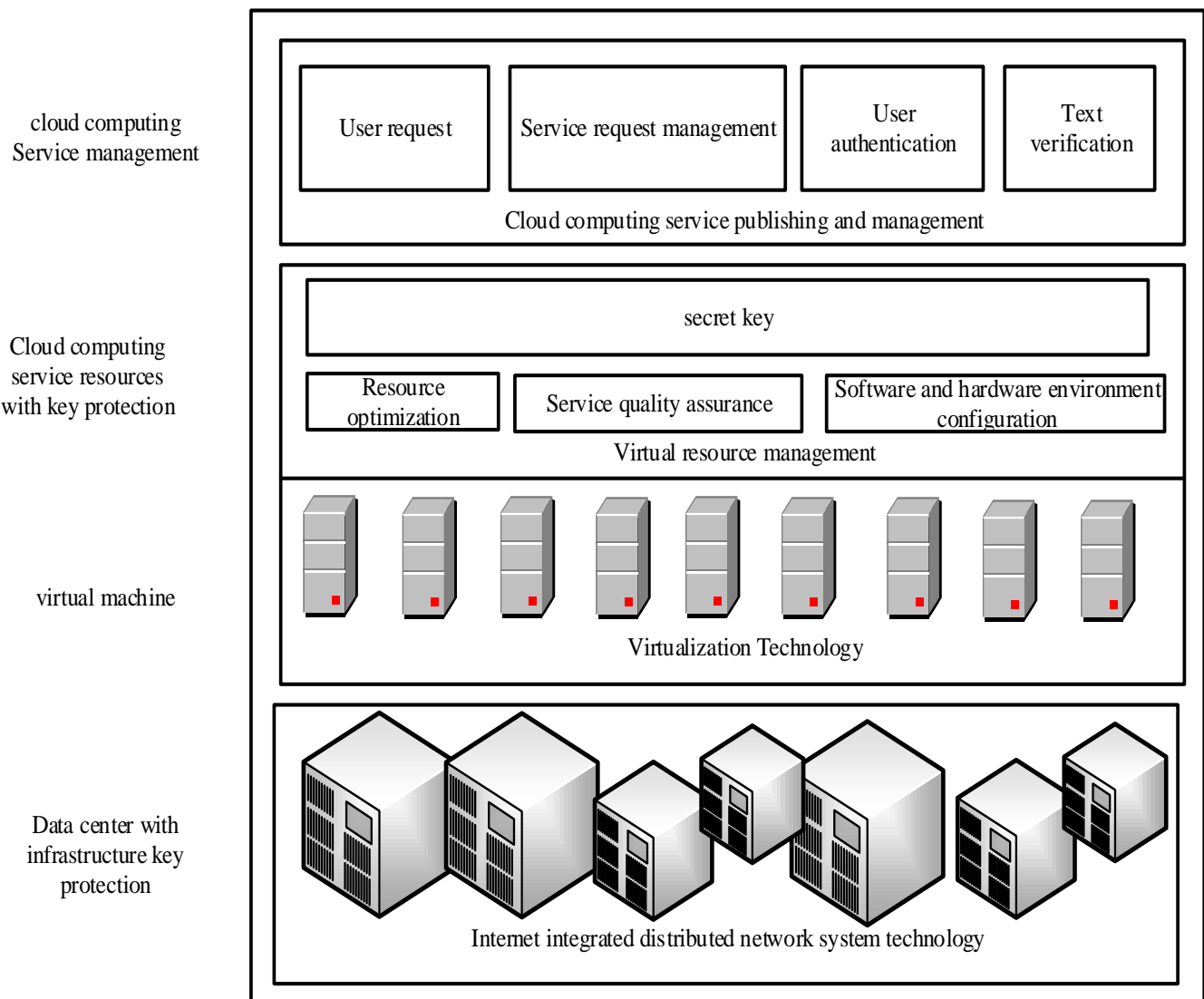
Fig. 6. Cloud computing architecture

In Fig. 6, according to the characteristics of multiplexed information, the structure of integrated distributed network and virtual machine is redeployed to ensure that the structure can completely load cloud computing virtualization resources.

The experimental parameters are shown in Table I.

TABLE I. EXPERIMENTAL PARAMETERS

parameter	analysis	numerical value
H	Total number of system storage nodes	6000
Size of File	Virtualization resource size	10GB
Backup Num	Number of backup copies	6000
Availability	System availability	>95%
P	Node availability	>95%

The simulation experiment goes through four processes, which are initialization stage, virtual resource access stage, virtual resource backup stage and virtual resource recovery stage. When the system is in the initialization phase, 6000 online nodes are built and node IDs are assigned. When the system is in the virtual resource access phase, the online phase needs to be divided into three categories: one is the service scheduling node, the other is the virtual resource backup node, and the third is the backup server. These three kinds of three wire nodes can achieve virtualized resource storage and backup through coordinated work. When the system is in the virtualization resource recovery stage, judge the node availability, and then use the backup server to recover the virtualization resources.

Several storage backup nodes are simulated in the MATLAB platform, the CPU speed and network bandwidth of the data center are configured, and the number and frequency of user information requests, as well as the start, end and peak time are set.

B. Experimental Results

By comparing several different methods, taking the occupied space of the virtual resource scheduling process as the experimental index, and taking the methods of literature [2] and literature [3] as the comparison method, the simplicity of the application process of this method is verified. The experimental results are shown in Fig. 7.

As can be seen from the comparison effect of data occupation space shown in Fig. 4, the main and sub platforms have different storage capacity under the action of different data volume, but they are increasing with the increase of data volume. However, the initial occupation space of the method in this paper is the smallest and the growth fluctuation is low. This is because this method updates the pheromone quantity based on local update rules, that is, the pheromone quantity update rules are changed into global update rules to ensure that the task of type matching is accurately allocated to the

corresponding computing resources, which further increases the possibility of exploring a larger solution space.

Under different conditions, taking the load fairness test of cloud computing nodes as the experimental index, the comparison results of different methods are shown in Fig. 7.

As shown in Fig. 8, the higher the load fairness coefficient, the better the storage effect. When the amount of backup data increases, the load fairness coefficients of other methods continue to decline, while the decline rate of this method is the lowest. This is because this method analyzes the needs of different users, calculates the matching degree of tasks corresponding to each demand to resource nodes, directly converts the target function into fitness value, and sets the time span and load balance as measurement indicators, Highlight the difference between high-quality path and general path.
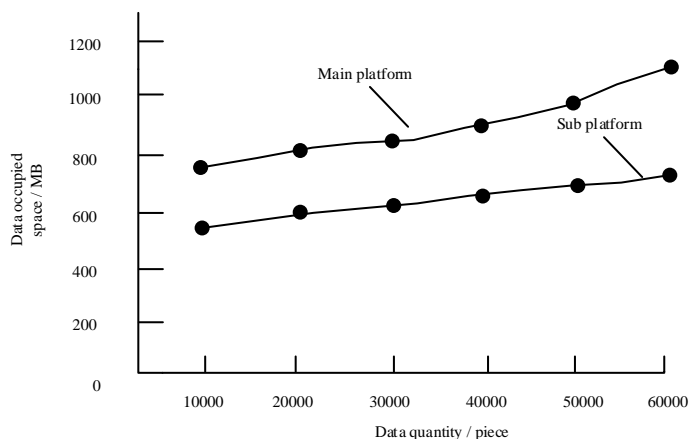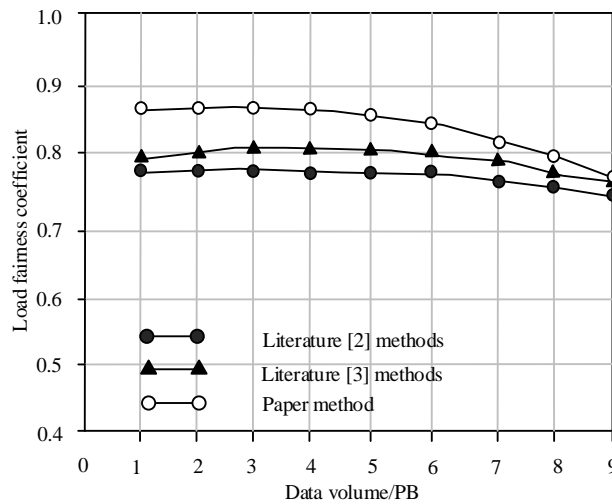

(a) Literature [2] methods
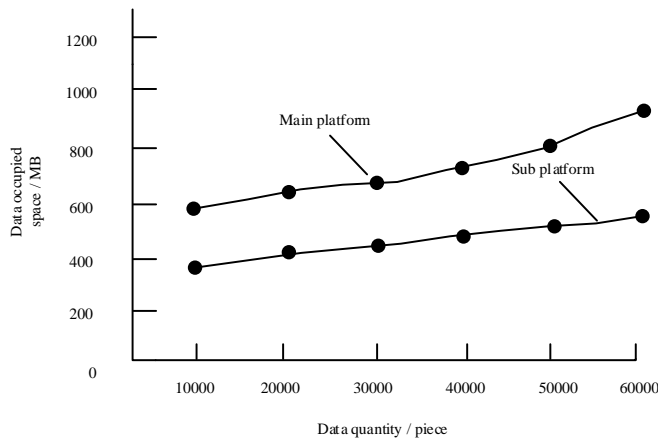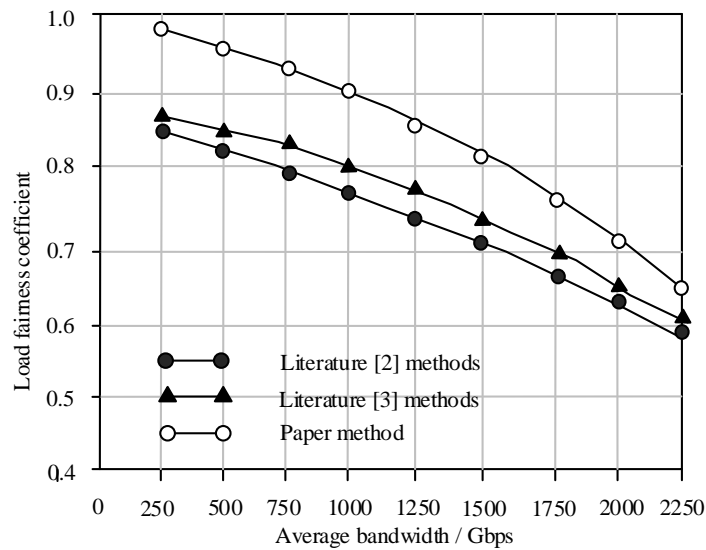

(b) Literature [3] methods
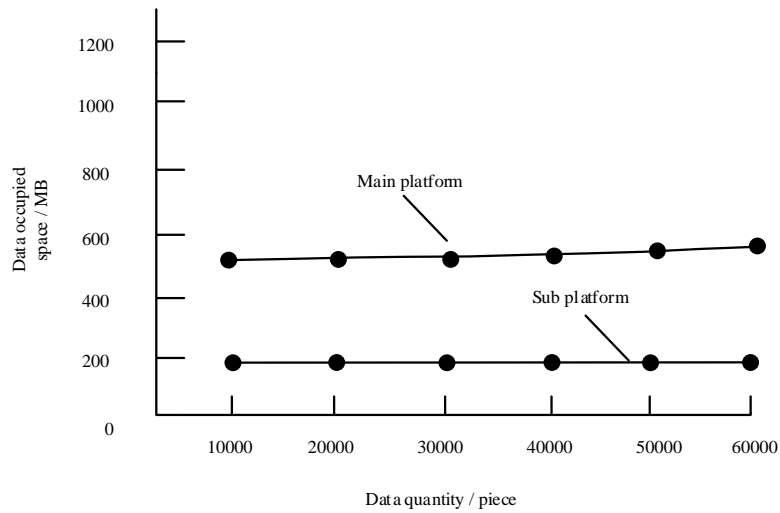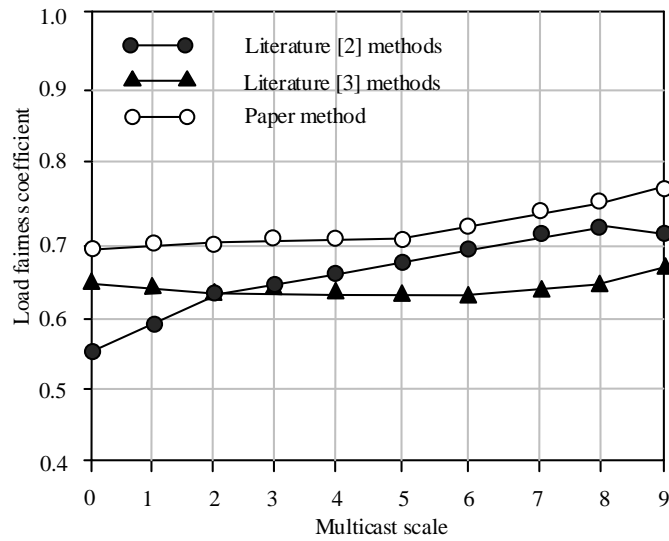
(c) Paper method

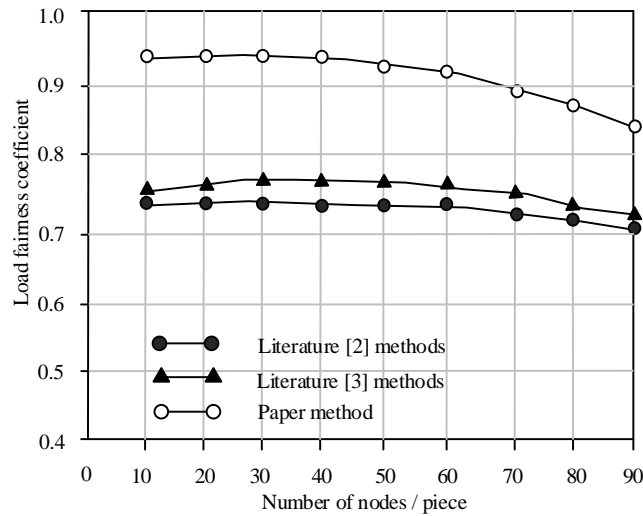Fig. 7. Space occupied by different methods under main and sub platforms

(a) More prior knowledge

(b) Complex area search

(c) Search stagnation

(d) Increase in the number of nodes

Fig. 8.   Load fairness test results under different conditions

Calculate the physical machine node allocation rationality, resource balance, longest queue length and energy consumption of the three allocation schemes. The results are shown in Table II.

As can be seen from Table II, this method performs well in terms of Allocation Rationality, resource balance, longest queue length and energy consumption value. This is because

while constructing the communication network model of cloud computing platform, this method calculates the ratio of output power and input power of uplink resources at different times, and timely adjusts the scheduling demand according to the scheduling energy to reduce the scheduling energy consumption, effectively solve the problem of uneven spatial distribution of information of each data node in the cloud platform.

TABLE II.   ALLOCATION RATIONALITY, RESOURCE BALANCE, LONGEST QUEUE LENGTH AND ENERGY CONSUMPTION

Solution method		Number of container / physical machine nodes	Rationality of distribution	Resource balance	Maximum queue length	Energy consumption value / W
Paper method		50/50	0.855	10.555	1	1000
		100/30	0.885	10.510	3	1500
		500/80	0.850	8.555	8	5000
		500/100	0.805	8.510	10	5000
Literature methods	[2]	50/50	0.751	8.155	3	1500
		100/30	0.755	8.510	7	5100
		500/80	0.555	8.551	13	5500
		500/100	0.550	7.850	18	7000
Literature methods	[3]	50/50	0.733	8.051	5	1500
		100/30	0.710	8.155	8	5550
		500/80	0.785	8.555	15	5000
		500/100	0.751	8.055	53	7500

IV.   CONCLUSIONS

In order to shorten the space occupation of cloud computing virtual resource scheduling and improve load fairness, the study combines the advantages of simulated annealing algorithm with ant colony algorithm, and applies the hybrid algorithm to cloud computing virtual resource scheduling. The results of the empirical analysis show that the virtual resource scheduling process of the proposed method in the primary and secondary platforms occupies 535 MB and 196 MB of space, respectively; the load fairness coefficient of the method is around 0.85, both of which are better than the comparison algorithms. Besides, it is also found that the resource scheduling algorithm proposed in the study has more reasonable allocation, more balanced resources, shorter longest queue length and less energy consumption compared with the comparison algorithm. The above results illustrate that the resource scheduling algorithm proposed in the study has better scheduling performance, and the method can be used to better schedule the resources in the cloud computing platform and improve the resource scheduling efficiency, thus improving the computational efficiency and reducing the running time of the platform algorithm, thus making better and faster use of virtual resources and saving time. In the future research work, we should further strengthen the research on the resource scheduling efficiency and improve the performance of the cloud computing virtual resource scheduling method comprehensively.

V.   DISCUSSION

The resource scheduling method proposed in the study combines the advantages of simulated annealing algorithm and ant colony algorithm, and compensates the disadvantages of both of them to each other, thus improving the overall algorithm performance. The parallel algorithm based on simulated annealing algorithm and ant colony algorithm has better scheduling performance in the process of resource scheduling in the cloud platform. In addition, the study also uses adaptive coding and modulation to extract the association features of virtual resources. The study transforms the pheromone update rules into global update rules through the good parallelism of ant colony algorithm and simulated annealing algorithm to schedule the cloud computing virtual resources one by one in a specific order. It expands the scale of the cloud computing center cluster, improves the scheduling capability of the whole platform, makes full use of the cloud platform resources, and gives full play to the advantages of cloud computing. In the subsequent research, the algorithms that can make up for each other's defects can be combined to obtain a better hybrid algorithm, and the optimized hybrid algorithm can be applied to the scheduling of virtual resources using the optimized hybrid algorithm in order to achieve better scheduling performance of cloud computing virtual resources.

REFERENCES

[1] W. Bo, J. L. Dang, Z. T. Li, H. F. Gong, F. Zhang, and S. Oh, "Modeling Analysis and Cost-Performance Ratio Optimization of Virtual Machine Scheduling in Cloud Computing," IEEE Transactions on Parallel and Distributed Systems: A Publication of the IEEE Computer Society, vol. 31, pp. 1518-1532, 2020.

[2] A. Ramegowda, J. Agarkhed and S. R. Patil, "Adaptive task scheduling method in multi-tenant cloud computing," International Journal of Information Technology, vol. 12, pp. 1093-1102, 2020.

[3] L. Abualigah, and A. Diabat. "A novel hybrid antlion optimization algorithm for multi-objective task scheduling problems in cloud computing environments," Cluster computing, vol. 24, pp. 205-223, 2021.

[4] J. A. J. Sujana, T. Revathi, T. S. S. Priya, and K. Muneeswaran, "Smart PSO-based secured scheduling approaches for scientific workflows in cloud computing," Soft computing: A fusion of foundations, methodologies and applications, vol. 23, pp. 1745-1765, 2019.

[5] M. Bansal, and S. K. Malik, "A multi-faceted optimization scheduling framework based on the particle swarm optimization algorithm in cloud computing," Sustainable computing: Informatics and systems, vol. 28, pp. 100429.1-100429.8, 2020.

[6] V. Priya, C. S. Kumar, and R. Kannan, "Resource scheduling algorithm with load balancing for cloud service provisioning," Applied Soft Computing, vol. 7, pp. 416-424, 2019.

[7] R. Valarmathi, and T. Sheela T, "Application of Particle Swarm Optimisation-R Factor Optimisation Algorithm for Efficient Resource Allocation and Scheduling in Cloud Environment," Journal of

computational and theoretical nanoscience, vol. 17, pp. 1990-1998, 2020.

[8] B. B. Naik, D. Singh, and A. B. Samaddar, "FHCS: Hybridised optimisation for virtual machine migration and task scheduling in cloud data center," IET communications, vol. 14, pp. 1942-1948, 2020.

[9] S. Mostafavi, and V. Hakami, "A Stochastic Approximation Approach for Foresighted Task Scheduling in Cloud Computing," Wireless personal communications: An Internaional Journal, vol. 114, pp. 901-925, 2020.

[10] A. A. H. Al-Mahruqi, G. Morison, B. G. Stewart, and V. Athinarayanan, "Hybrid Heuristic Algorithm for Better Energy Optimization and Resource Utilization in Cloud Computing," Wireless Personal Communications, vol. 118, pp. 43-73, 2021.

[11] M. Nanjappan, G. Natesan, and P. Krishnadoss, "An Adaptive Neuro-Fuzzy Inference System and Black Widow Optimization Approach for Optimal Resource Utilization and Task Scheduling in a Cloud Environment," Wireless Personal Communications, vol. 121, pp. 1891-1916, 2021.

[12] H. O. Salami, A. Bala, S. M. Sait, and I. Ismail, "An energy-efficient cuckoo search algorithm for virtual machine placement in cloud computing data centers," The Journal of Supercomputing, vol. 77, pp. 13330-13357, 2021.

[13] P. Partheeban, and V. Kavitha, "Versatile provisioning and workflow scheduling in WaaS under cost and deadline constraints for cloud computing," Transactions on Emerging Telecommunications Technologies, vol. 30, 2019.

[14] R. G. Narendrababu, and K. S. Phani, "Regressive Whale Optimization for Workflow Scheduling in Cloud Computing," International Journal of Computational Intelligence and Applications, vol. 18, pp. 7-18, 2019.

[15] L. Abualigah, and A. Diabat, "A novel hybrid antlion optimization algorithm for multi-objective task scheduling problems in cloud computing environments. Cluster Computing, vol. 24, pp. 205-223, 2020

[16] X. Li, and H. Z. Sun, "Virtual Machine Matching Scheduling Simulation under Uncertain Network Load," Computer Simulation, vol. 37, pp. 363-366, 2020.

[17] He W, Meng S, Wang J, Wang L, Pan R and Gao W. Weaving scheduling based on an improved ant colony algorithm:. Textile Research Journal,vol. 91, pp. 543-554, 2021.

[18] Zhao H, Zhang C and Zhang B. A Decomposition-Based Many-Objective Ant Colony Optimization Algorithm with Adaptive Reference Points. Information Sciences, vol. 540, pp. 435-448, 2020.

[19] Li S, You X and Liu S. Multiple ant colony optimization using both novel LSTM network and adaptive Tanimoto communication strategy. Applied Intelligence,vol. 9, pp. 1-21, 2021.

[20] Rueda R, Ruiz L, MP Cuéllar and Pegalajar M C. An Ant Colony Optimization approach for symbolic regression using Straight Line Programs. Application to energy consumption modelling. International Journal of Approximate Reasoning, vol. 121, pp. 23-28, 2020.