Data Scheduling Method of Social Network Resources Based on Multi-Agent Technology

Xing-hua Lu(✉), Ling-feng Zeng, Hao-han Huang, and Wei-hao Yan

Huali College Guangdong University of Technology, Guangzhou 511325, China
luxinghua5454565@163.com

Abstract. Aiming at the problem that the traditional scheduling method can't deal with a large number of data quickly when dealing with social network resource data, a new scheduling method of social network resource data based on multi-Agent technology is proposed. Firstly, the social network scheduling framework is designed, using the two-level structure of Agent and three CDN management domains to hide the distribution and heterogeneity of different resources, setting the upper limit trigger conditions of data in each management domain of the framework, using reasoning tools to infer and calculate the SLA comprehensive level of each network operation node, calculating the proportion difference between various resources, selecting the appropriate bias a two-stage resource scheduling method is used to realize resource data scheduling. The experimental results show that: compared with the traditional scheduling method, the social network resource data scheduling method based on multi-Agent technology can maintain the processing time in about 10 s with the increase of data volume, and the processing time is shorter, which is more suitable for practical use.

Index Terms: Multi Agent technology · Social network resources · Data scheduling · Processing time

1 Introduction

Agent technology has become a hot topic in the 1990s, even referred to as the next far-reaching breakthrough in the field of software by some literatures. One of the important reasons is that Agent technology is the mainstream technology in the field of computer technology in the network distributed computing, which is also playing an increasingly important role. On the one hand, Agent technology provides an effective way to solve new distributed application problems. On the other hand, it provides a reasonable conceptual model for the comprehensive and accurate study of the characteristics of distributed computing system. The concept and technology of Agent appear in the development of distributed application system, which shows obvious effectiveness [1].

Agent has its own computing resources and its own behavior control mechanism. It can decide and control its own behavior according to its internal state and perceived environmental information without direct external manipulation. Agent can interact with other Agent and implement flexible and diverse interaction with Agent communication language, and can work effectively with other Agent. For example, a user on

Y.-D. Zhang et al. (Eds.): ICMTEL 2020, LNICST 326, pp. 148–158, 2020.
https://doi.org/10.1007/978-3-030-51100-5_13

the Internet needs to use the Agent communication language to present information requirements to the active service Agent. The Agent can sense the environment and respond to related events [2].

Scheduling method refers to the method used by the dispatchers of automobile transportation enterprises when they perform vehicle scheduling. The main scheduling method is empirical scheduling. Experiential scheduling is carried out by means of simple and visualized tools, such as using marks on the scheduling board or transportation network diagram to display the vehicle running dynamics and arrange the vehicle running routes. Using the method of mathematics or artificial intelligence combined with electronic computer to make the plan, at the same time using modern communication technology to transmit and feedback scheduling information, commanding and supervising the vehicle operation, the research on the social network resource data scheduling method based on multi-Agent technology can schedule the social resources in the way of data, and more reasonable allocation of social resources.

2 Data Scheduling Method of Social Network Resources Based on Multi-Agent Technology

2.1 Design the Social Network Scheduling Framework

When designing the social network scheduling framework, virtualization technology is used to hide the distribution and heterogeneity between different resources, so that the network resource data is no longer an independent entity, but all resources are integrated together, so that many virtual machines form a virtual resource pool, which is managed by the management system, the preliminary design of the management framework is shown in the Fig. 1 below:

When there are application request resources, the above management framework will allocate corresponding amount of resources from the resource pool to provide the service in the form of virtual machine. When the application request is completed, the service will release the resources, the system will recycle the virtual machine and put the released resources back to the resource pool. Users do not need to care about the structure and specific implementation methods of the system, but only about the quality of their own services. Because there are different users in the social network and different users have different needs, there will be a great demand fluctuation in the network access end, which requires that the management and scheduling of resources in the network management system can dynamically meet the needs of users and solve the resource utilization rate and application availability, to achieve on-demand allocation [3].

According to the nature of the Agent itself, integrate the framework of Fig. 1, use the interaction between each Agent, combine the learning and adaptability of the Agent, and make automatic adjustment for the current environment. When scheduling, multiple Agent are used. Through coordination and cooperation, the system can make reasonable arrangements for different goals, and finally form an overall scheme. Therefore, when integrating the design framework of Fig. 1, the first level Agent can effectively manage and schedule network resources by controlling multiple second

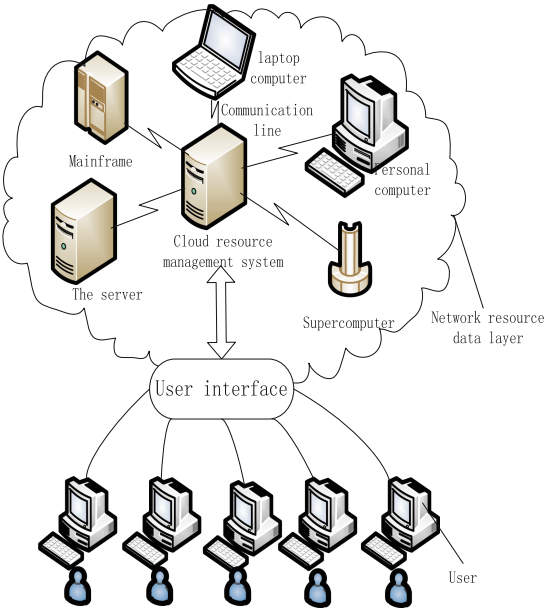

Fig. 1. Management framework designed

level Agent. Then use the hierarchical architecture, two-level structure and three CDN management domains, and appropriately increase the number of Agents and the number of Agent management domains in each layer, the overall framework structure is shown in Fig. 2 below:

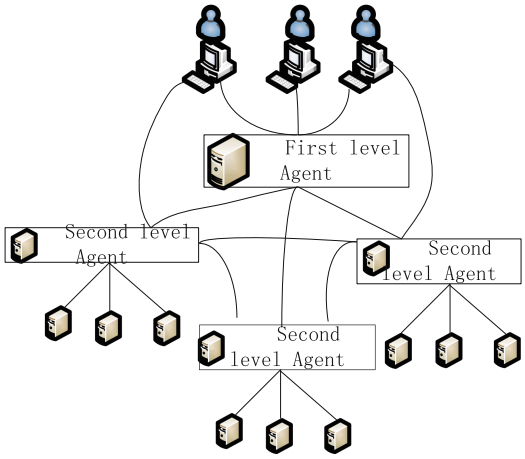

Fig. 2. Architecture diagram for integrating Agent technology

Using the architecture diagram designed in the above figure, the number of monitoring items includes memory utilization and network utilization. In actual monitoring, we only need to ensure that the user determines the upper limit of the amount of usage in the process of establishing the virtual machine instance, that is, the upper limit of the memory size and the bandwidth of the virtual machine can be satisfied without affecting the operation of the scheduling task in the virtual machine. In this way, only the sum of the upper limit of memory and the upper limit of bandwidth of all virtual machine instances on the host needs to be guaranteed within the range of memory and bandwidth resources owned by the physical server. However, in terms of cost, users usually want to use the actual amount of resources used by the virtual machine to count the use of the virtual machine, so a relatively complex scheduling method is needed to schedule effectively, and then set an upper limit trigger condition [4].

2.2 Set the Upper Limit Trigger Condition

The factors that need to be considered for the upper limit trigger conditions mainly include utilization, memory utilization and network utilization. These three monitoring factors represent two different monitoring standards and cannot be monitored by a unified measurement method. Therefore, the three monitoring factors are divided into two groups. The first group is quality monitoring item, which only includes utilization. The main reason why it is called quality monitoring item is that its performance, that is, the computing power provided by the computing center, cannot be measured by the utilization of virtual machine instances alone, because even though the utilization of a specific virtual machine is very low, other virtual machine loads on the same host are the same all of them are very high, resulting in a very high utilization rate of the host, resulting in less time slice allocated by the host to each virtual machine instance, so its computing power is still very low, so it is unscientific to use the utilization rate of the virtual machine as the measurement standard of the virtual machine performance [5].

Another measure is to represent the performance of all virtual machines on the host through the utilization of the host of each virtual machine. This measurement standard is more scientific than the former one, because the higher the utilization rate of the host computer is, the busier the host computer is, the less time can be allocated to each virtual machine, and its computing power will be lower for each virtual machine. This measurement method will perform well in the data center of the same structure, that is, the configuration of the running nodes in the data center is the same. However, due to the different computing power requirements of virtual machine instances running on the work nodes, and the servers in the data center are often heterogeneous, that is, the configuration of servers is often different. Because each running node is different, there are some differences in the computing power between CPUs. Therefore, there must be a big deviation in using the CPU utilization rate to measure the CPU performance of each running node. Therefore, when inferring the upper limit, use the inference tool to synthesize the data value processed by CPU to get the upper limit trigger condition. Use the inference tool as shown in the following Fig. 3 to infer the upper limit trigger condition:

After using reasoning tools, SLA is used to enhance the data quality of conditions. Before evaluation, a group of representative computing tasks will be generated, which

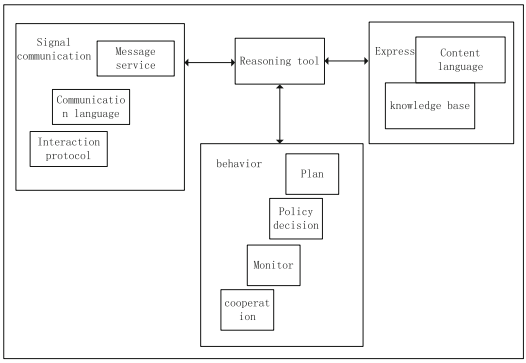

Fig. 3. Reasoning tools used

represent different aspects of CPU computing power, and specify the ideal completion time of each task. After that, the completion time of each task is calculated by executing this group of tasks in all running nodes. Through the combination of these two data and SLA level strategy, the SLA level of each node for each task is calculated. And sum up all SLA levels of each node, and then calculate the SLA comprehensive level of each running node. This is a measure of the CPU's computing power based on the quality of work. It allows the CPU to perform real tasks, and then evaluate its computing power according to its performance. In this way, there is no problem of heterogeneous servers to affect the evaluation of CPU performance of running nodes. And through the selection of SLA level strategy, it can effectively highlight the sensitivity of each virtual machine to the change of CPU computing power and effectively guarantee the work quality of virtual machines with high demand for CPU [6].

One of the problems that should be considered is the virtual machine migration triggered by the instantaneous peak value of all monitoring items on the work node. In the process of virtual machine running, the instantaneous utilization of a monitoring item may increase for some reason, but it will return to the normal level only for a short time. When the monitoring items exceed the preset threshold, the virtual machine will be migrated. When the instantaneous peak value is encountered, many unnecessary migrations will be carried out to reduce the overall performance of the data center, and this instantaneous peak value often occurs. So when calculating the upper limit, follow the flow chart below (Fig. 4):

Run the device at the running node, execute the pre-defined test tasks on all the running nodes every t, record the execution time of each task, calculate the integration $L = (l_1, l_2, l_3, \ldots, l_n)$ of SLA level L of each node for each task combined with the expected execution time of each task, finally calculate the SLA comprehensive level Lr of each node r, and complete the upper limit setting [7].

2.3 Determine Scheduling Parameters

When determining the scheduling parameters, first deploy the tasks that users apply for various types of resources to the cloud to complete their application deployment or

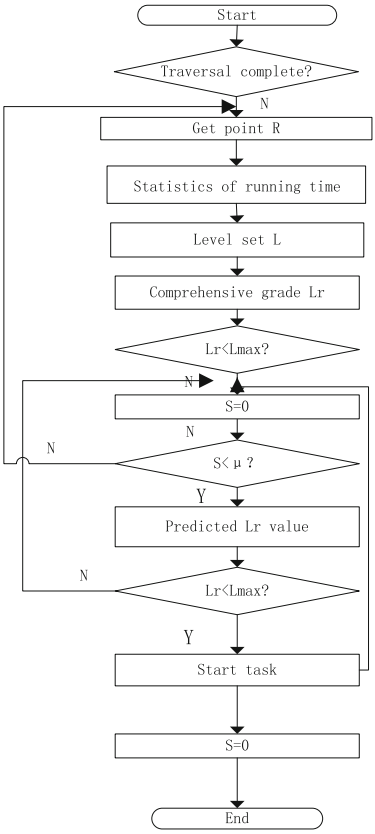

Fig. 4. Execution process

calculation work. Different types of users work with different resource requirements, in addition, the physical devices in the cloud environment data center have different specifications, to design an effective method to get the optimal creation scheme of the virtual machine, make efficient use of resources, reduce resource waste, and at the same time, make the available resources in the cloud be allocated to many users fairly, this is the question that needs to be considered in the resource scheduling algorithm of the virtual machine layer question. Using a game theory based fairness effectiveness trade-off resource scheduling algorithm to solve the resource scheduling problem considering the fairness of resource utilization and resource allocation [8].

In the process of scheduling, we should choose the resource that has the lowest utilization on a single physical machine to maximize the utilization value. First, quantify the utilization value of network resources, and the quantification formula is as follows:

$$utl_k^{(m_i)} = 1 - \frac{R_k(m_i) - \sum x_{\tau k}^{(m_i)}}{PM_k(m_i)} \tag{1}$$

Among them, $R_k(mi)$ is the total number of available resources K on the network resource mi, τ is the task processed on the physical machine m_i, $\sum x_{\tau k}^{(mi)}$ is the number of resources of type k of the assignment task τ on the network resource data, $PM_k(m_i)$ is the total number of resources k in the initial state of the network data.

In order to avoid the large difference in the proportion of various resources, when the proportion of a certain type of resources reaches the threshold value, the physical machine has no ability to create a new virtual machine, at this time, resources with a small proportion are very easy to waste resources, also known as resource fragments [9]. In order to reduce the generation of resource fragments, it is necessary to reduce the difference of the proportion of various types of resource occupation on a single physical machine as much as possible, so that they tend to balance in all resource dimensions, so as to reduce the probability of fragment generation. Skewness is used to measure the imbalance of the proportion of various types of resources on the physical machine [10]. The smaller the skewness is, the less the residual resource fragments will be and the higher the resource utilization will be. The calculation formula of deflection is as follows:

$$ske(m_i) = \sqrt{\sum_{k=1}^{K} \left(\frac{utl_k^{(m_i)}}{\underline{utl^{(m_i)}}} - 1\right)} \tag{2}$$

Among them, $utl_k^{(m_i)}$ represents the utilization rate of resource k on network resource m_i and $\underline{utl_k^{(m_i)}} = \frac{1}{K}\sum_k utl_k^{(m_i)}$ represents the average utilization rate of resource m_i. Select the scheduling data with appropriate skewness, and use a two-stage resource scheduling method to achieve resource data scheduling [11].

2.4 Realization of Resource Data Scheduling

In order to achieve resource data scheduling, a two-stage resource scheduling method is used. In the first stage, it is appropriate to find out how much data each network resource data needs to be allocated. In the second stage, a stable placement algorithm is proposed to determine which scheduling algorithm the allocated network resource data needs to use. Most data centers often run a large number of servers. Due to the low average utilization rate of resources on the servers, most resources are idle [12]. In view of the heterogeneity of the physical devices in the data center and the difference of resource requests from a large number of user groups, there must be some physical resources that cannot be fully used in the scheduling process, resulting in a waste of resources. In the multi-type resource allocation problem, the optimization of resource utilization is more complex, and each type of resource consumption needs to be considered [13].

There are two ways to improve resource utilization, first, the minimum value maximization method is used to consider the bottleneck of resource consumption in multiple types of resources [14]. For a large cluster of physical machines in the data center, there may be multiple physical machines carrying sporadic virtual machines, which idle a large number of resources, but because there are virtual machines running, they must maintain normal working conditions, consume energy, and increase maintenance costs. The best running state of a single physical machine should make the utilization rate of all types of resources reach the closed value of the best workload. Therefore, when selecting physical machines, the resource utilization on each physical machine should be maximized within the threshold range [15].

Using each data subtask type τ, each algorithm is regarded as a service platform. Assuming that the data rate of network resources is distributed in a compound *Poisson distribution*, set the server rate as μ, and calculate the probability of free time occurrence of resource data scheduling as follows:

$$P_0(\tau) = \left[\sum_{n=1}^{K} \left(\frac{P^n}{n!}\mu - 1\right)\right] \tag{3}$$

where, $P_0(\tau)$ represents the number of blank data in the whole scheduling method. According to the number of blanks, we remove the blank scheduling data from the network resource data, and complete the research on the scheduling method of social network resource data based on multi-Agent technology.

3 Simulation Test Experiment

3.1 Experimental Data Preparation

Using the configuration of the ClouldSim experimental environment, install several corresponding software packages in the configuration. The installed software packages are as follows (Table 1):

Table 1. Packages used

Serial number	Name	Model
1	Operating system	Windows
2	Memory	4G
3	Development tool	Eclipse 3.7.1
4	Java environment	JDK 1.7
5	CloudSim	CloudSim3.1

Cloud computing is used to provide various types of services. In order to simplify the operation, four types of services are set. According to the setting of the Cloudlet

class in ClouldSim, different requirements are quantified, and the task parameters in the following Table 2 are obtained:

Table 2. List of task parameters

Task type	Length	File size	Output size	Network
1	40	100	80	120
2	200	60	100	60
3	600	80	50	100
4	800	20	10	50

In contrast to user services, there are many kinds of virtual machines with different performance in the cloud platform to meet the needs of users. This paper sets up three virtual machines with different performance to handle tasks with different requirements, as shown in the following Table 3:

Table 3. List of parameters for virtual machines

VM type	A	B	C
CPU	Single-core	Dual-core	Quad-core
Memory	256	1024	2048
Network	4000	1000	2000
Storage	40000	20000	10000
Power	100	400	1000

After initializing CloudSim, use the DataCenter in the software to create the data center, submit the user request according to the rules of Agent technology, create the task list and virtual machine list after creating the data center Agent, call the functions defined when creating the database center, start the experiment, record the two traditional methods and the social network resource data scheduling method based on multi Agent technology the time to complete the task of scheduling social network resource data.

3.2 Experimental Result

In the experiment, according to the order of data task quantity from more to less, the time of completing different data quantity by three methods is compared, the results are as follows (Fig. 5):

It can be seen from the above figure that when the task volume is small, with the increase of task volume, the execution time of the three scheduling methods is almost linear growth, and the social network resource data scheduling method of multi-Agent technology does not show obvious advantages. When the network resource data task volume is greater than 60, the processing time of traditional scheduling method 1

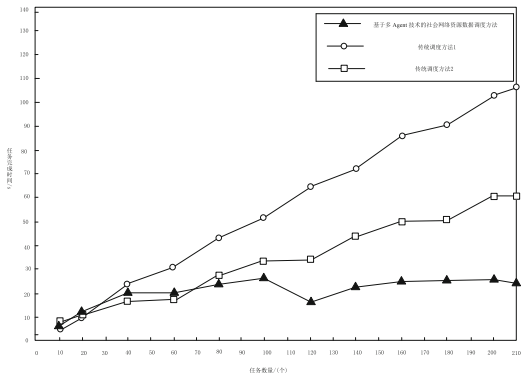

Fig. 5. Test results of three scheduling methods

increases linearly, while that of traditional scheduling method 2 and multi Agent based scheduling method 2 increases linearly the processing time of social network resource data scheduling method based on t technology gradually slows down. When the number of network resource data scheduling tasks reaches 120, the processing time of traditional scheduling method 1 and traditional scheduling method 2 still increases rapidly, while the social network resource data scheduling method based on multi-Agent technology maintains a stable state. To sum up, with the increase of scheduling data tasks, the processing time of social network resource data scheduling method based on multi-Agent technology can be maintained at about 10 s, while the two traditional scheduling methods, with the increase of scheduling tasks, the processing time will also increase, with weak timeliness, which is not conducive to the scheduling of social network resource data.

4 Conclusions

With the rapid development of social network and the continuous popularization of Agent technology, the current resource management and scheduling technology has been unable to meet the needs of all application environments. How to schedule and manage the social network resource data reasonably and realize the unity of user service availability and system resource reliability is the core issue of research. At present, most of the research on resource scheduling in the academic community considers virtual machine as the abstract unit of resource scheduling, but few of the research on resource scheduling in social network using multi-Agent technology. Therefore, this paper proposes a data scheduling method of social network resources based on multi-Agent technology, aiming at the problem that the traditional scheduling method takes too long to process multi data tasks, which is of developmental significance.

5 Fund Project

2019 “climbing Plan” Guangdong University students Science and Technology Innovation cultivation Special Fund Project;

Project name: research on crowdsourcing of complex tasks in Social Network based on Multi-Agent Technology, Project number: pdjh2019b0618.

References

1. Cai, S., Ran, X.: Research on scheduling technology of cooperative tracking based on multi-Agent. Appl. Res. Comput. **35**(2), 590–593 (2018). +599
2. Wang, L., Xu, J., Ding, M.: Research on multi-agent particle swarm optimization based on chaos in microgrid optimal dispatch. Adv. Power Syst. Hydroelectr. Eng. **33**(8), 1–7 (2017)
3. Dai, B., Li, C., Kuang, Z., et al.: Elastic scheduling method for private cloud resources based on multi-strategy. J. Comput. Appl. **37**(S1), 34–38 (2017)
4. Li, B., Li, L., Bin, D.: The dynamic diffusion and evolution of multi-agent collaboration innovation based on market orientation—a perspective of complex networks. Soft. Sci. **31** (6), 19–23 (2017). +29
5. Su, Y., Han, M., Lei, J.: An analysis of regional innovation correlation network in china based on social network analysis. Sci. Res. Manag. **39**(12), 78–85 (2018)
6. Ma, S., Nie, G., Ye, T., et al.: Dynamic distribution selection algorithm for post mobile relay nodes after optical fiber network instruction **42**(5), 44–47 (2018)
7. Shao, H.: Research on cloud computing resource scheduling method based on container technology **40**(22), 33–35 (2017)
8. Qin, R., Zeng, S., Li, J.J., et al.: Parallel enterprises resource planning based on deep reinforcement learning. Acta Automatica Sinica **43**(9), 1588–1596 (2017)
9. Ha, D., Tan, Z., Wang, H.: Research on agent-based public opinion evolution in complex network. Fire Control Command Control **42**(8), 92–96 (2017)
10. Wu, D., Yu, S., Zeng, G., et al.: Research of task allocation based on dynamic integrated relationship network for net-worked multiagent systems. J. Chin. Comput. Syst. **39**(5), 957–966 (2018)
11. Yang, T., Gbaguidi, A., Yan, P., et al.: Model elucidating the sources and formation mechanisms of severe haze pollution over northeast mega-city cluster in China. Environ. Pollut. **230**(2), 692–700 (2017)
12. Chen, X., Wang, H.H., Tian, B.: Visualization model of big data based on self-organizing feature map neural network and graphic theory for smart cities. Clust. Comput. **22**(2), 13293–13305 (2019)
13. Zhiming, C., Lianbing, D., Daming, L., et al.: A FCM cluster: cloud networking model for intelligent transportation in the city of Macau. Clust. Comput. **22**(3), 1–10 (2019)
14. Xiaoming, H., Wang, K., Huang, H., et al.: QoE-driven big data architecture for smart city. IEEE Commun. Mag. **56**(2), 88–93 (2018)
15. Wang, J., Bin, L.: Big data analysis and research on consumption demand of sports fitness leisure activities. Clust. Comput. **12**(6), 1–10 (2018)