Research on Big Data Equilibrium Scheduling Method in Cloud Computing

Xiangjun Zhou[1], Hao Li [2]

[1]School of Information Technology, Guangdong Teachers College of Foreign Language and Arts, Guangzhou, China

[2]GuangZhou Think-Height Information Technology Co. Ltd, Guangzhou, China

Abstract

Under the cloud computing background, big data constructs the large-scale cloud database through the cloud storage form, it needs to carry on the balanced scheduling to the cloud computing background big data, enhances the cloud storage resources equilibrium disposition equilibrium and the management efficiency. The traditional big data equilibrium scheduling method adopts subspace scheduling method, which has poor adaptive scheduling performance for big data in cloud computing background. An adaptive link forwarding control based big data equalization scheduling algorithm in cloud computing is proposed. In the background of cloud computing, big data stack storage is designed to extract the wavelet information entropy characteristics of big data information flow in cloud computing background, to construct the link transceiver control mechanism in cloud computing, and to calculate the channel equilibrium configuration parameters of big data scheduling. Adaptive link forwarding control to achieve big data balanced scheduling. Simulation results show that the proposed method can improve big data's load balance in cloud computing context, and improve the efficiency of big data's balanced scheduling and the ability to balance the allocation of cloud storage resources in cloud computing background.

Keywords

Cloud computing; big data; load balancing; scheduling; link forwarding control.

1. Introduction

Computer technology and Internet technology hasten the rapid development of Internet application, users share computer resources through the network to the emergence of cloud computing business model epoch-making, its computing power is almost unlimited, information services are rich and diverse. Users of computing resources and services are readily available, easy to expand, on-demand billing. In the cloud computing business model, the cloud data center has become a high performance computer centralized, physical server and network equipment scale, but due to the diversity of user needs, Dynamic change and the resource heterogeneity of the server will lead to the unbalanced load in the data center, which causes some physical computers to be overloaded and inefficient, while others are light and idle. Therefore, how to balance the load between physical computers through the appropriate resource scheduling algorithm to improve the resource utilization and overall performance of cloud data center is a key problem in cloud computing field at present[1].

Big data load balancing in cloud computing is an important means to realize the effective utilization and sharing of resources, which is divided into task-based load balancing and resource-based load balancing. Task-based load balancing refers to the allocation of tasks between multiple computations, disks, processes or other resources in order to achieve optimal resource utilization, which is used to reduce computing time. Resource-based load balancing is to achieve the approximate balance of the load of each node through the allocation and redistribution of resources, so as to improve the performance of the whole system[2]. The commonly used load balancing algorithms include two categories: (1) static load balancing algorithms, such as (RR), Weighted Round Robin(WRR), random placement algorithm, etc. (2) dynamic load balancing algorithms, such as minimum link algorithm (LC) and Weighted Least-Connection Scheduling (WLC), which distribute requests dynamically according to the real-time load of the system. In reference [4], it is proposed that Eucalyptus platform uses RR scheduling algorithm to distribute virtual machines to different physical machines in order to realize load balancing[3]. WRR algorithm uses the corresponding weights to represent the processing power of the server, and the larger weights will be allocated to more requests. This algorithm is used in VMware resource load balancing. Random algorithm is to randomly assign virtual machine requests to the appropriate physical machine[4]. In reference [5], a balanced scheduling model of big data in cloud computing based on random scheduling of virtual information resources is proposed, which realizes task tradeoff and scheduling through cloud computing, and improves the efficiency of big data in cloud computing. However, the algorithm has the disadvantages of long time delay and complex computation.

Aiming at the above problems, this paper proposes a big data equalization scheduling algorithm based on adaptive link forwarding control in cloud computing. In the background of cloud computing, big data stack storage is designed to extract the wavelet information entropy characteristics of big data information flow in cloud computing background, to construct the link transceiver control mechanism in cloud computing, and to calculate the channel equilibrium configuration parameters of big data scheduling. Adaptive link forwarding control to achieve big data balanced scheduling. Finally, the experimental test and analysis are carried out to demonstrate the superior performance of this method in improving big data's balanced scheduling capability in cloud computing.

2. Big data stack storage model and load information flow construction under cloud computing background

2.1 Big data stack storage design in Cloud Computing

Large cloud database cloud storage technology distributes a large amount of data to multiple service nodes for cache analysis, realizes big data's load balance and cloud storage in the cloud computing background, and studies the big data equilibrium scheduling in the cloud computing background[6]. First of all, build big data stack storage model in cloud computing background to realize the storage of cloud resources. Services and cloud computing big data load information scheduling. The entire big data equilibrium scheduling control model is represented by a connected stack graph $G=(V,E)$, where V is the set of all nodes in the stack, v_0 represents the Sink node, and in the big data equilibrium scheduling model, $A\subset V$, $B\subset V$ and $A\cap B=\varphi$ are set and each cloud is backed by cloud computing. The big data transmission and scheduling set $S_i(i=1,2,\cdots,L)$ satisfies the $N_i^2=N_i^1\cup\left(\bigcup_{j\in N_i^1}N_j^1\right)$, in the scene. In the k th time slice, $k=1,2,\cdots,L$, aggregates the data into the big data equilibrium scheduling data set $V=[v_1, v_2,\ldots,v_n]$, forms the data aggregation on the stack control center, in the cloud computing background, supposes that, in the cloud computing background, $\hat{q}_{i+1,i}:=c_{i+1,i}$ represents the task scheduling vector. For cloud storage system, large big data is divided into three layers structure model. The maximum feasible resource pooling flow of big data scheduling set should be satisfied with $c_{i,i-1}\le\min\{c_{i+1,i},d_{i',i}\}$., respectively. Obviously, this assumption is $\hat{q}_{i',i}:=d_{i',i}$. The cloud storage resource scheduling scheme of large database is designed and transferred to each node, and the cloud storage resource information of large cloud database is obtained as follows:

$$s_m(t)=\cos\left\{2\pi f_0\left[t+\tau_m(\theta)\right]\right\} \tag{1}$$

In the big data stack stored procedure in the cloud computing context, s_h^w can be expressed as:

$$s_h^w = \mathrm{E}\left[\min_{k\in R_w}\left\{\mathrm{H}_{h,k}^w\right\}\middle|\boldsymbol{\eta}^w\right] = -\frac{1}{\theta}\ln\sum_{k\in R_w}\exp\left(-\theta\eta_{h,k}^w(\omega)\right),\ w\in W, h\in H \tag{2}$$

Therefore, big data stack storage design in cloud computing background is realized, which provides parameter input for load balancing scheduling[7].

2.2 Construction of data flow and feature of wavelet information entropy

On the basis of the background design and stack storage of large-scale big data cloud computing, the grid topology of big data in cloud computing background is divided into three layers, and the feature extraction is carried out, and the grid topology of big data in cloud computing background is obtained[8]. The structure is shown in figure 1.

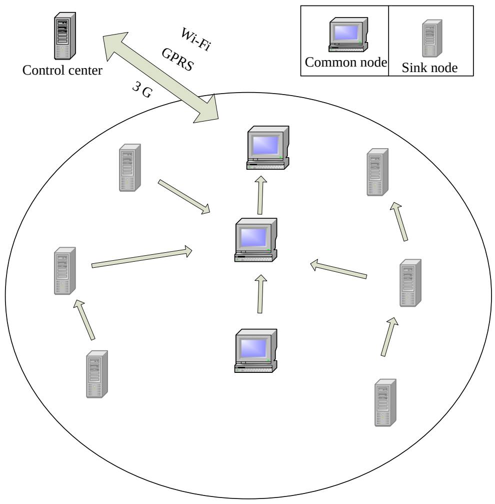


Figure 1. Big data's grid topology in Cloud Computing

According to figure 1, the base station item in big data in the cloud computing background involves a large number of multi-source heterogeneous resources. In the cloud computing background, big data scheduling, the storage component of the first layer of the network connected to the Internet, and the second layer are connected to the Internet. It is composed of LAN (Local area Network), the third layer is session layer, which carries on the information exchange of big data's equilibrium scheduling[9]. The gathering scheduling control model of big data's equilibrium scheduling model can be formally described as: given a stack. The control set $G=(V,E)$ and $Sink \in V$, look for a data aggregation scheduling set $\{S_1, S_2, \cdots, S_L\}$, so that big data $TS(u)=\max(TS(v)+1, TS(u))$ is the minimum. At this time, the delay response system expression corresponding to big data equilibrium scheduling node is defined as follows:

$$\dot{Y} = AY + B[f(Y)+u] \tag{3}$$

In the cloud storage resource scheduling of large database in network cloud computing, the length of each idle time slice is recorded as $x_1, x_2, \cdots, x_{m+1}$, the information of big data control node in k cloud computing background is defined as δ_k:

$$\delta_{ik}(t) = G(V=k|U_i, \Theta(t)) \tag{4}$$

Where, U_i means cloud computing big data classification attribute, k is load intensity, $\Theta(t)$ is the trust value obtained by the cloud computing big data query at the last moment, on the basis of this, the information flow of big data is constructed, as follows:

$$x_1 + x_2 + \cdots + x_{m+1} = T + t - m\times t \tag{5}$$

Wherein, T indicates the length of big data load balancing scheduling time under the general cloud computing background, constructs according to the above information flow, carries on the feature extraction, extracts the wavelet information entropy feature of the big data information flow, and the cloud storage of the big database. The frequency modulation law of resource information flow is a hyperbolic function, that is:

$$f_i(t) = \frac{K}{t_0 - t} \qquad |t| \le \frac{T}{2} \tag{6}$$

On the basis of the construction of big data information flow, the wavelet information entropy feature of big data information flow is extracted, and the wavelet information entropy feature is estimated as follows:

$$\begin{cases} d_i = \min \sqrt{\sum_{m=1}^{k} \left(\frac{f_m(a_i) - f_m(P_j)}{f_m^{\max} - f_m^{\min}} \right)^2} \\ j \ne 1,\ j \in P^* \end{cases} \tag{7}$$

Where, eigenvalues are $\lambda_1, \lambda_2, \cdots, \lambda_l$ and eigenvector matrix $Y = [y_1, y_2, \cdots, y_l]$, they are used to control the loop stack through feature extraction to achieve balanced scheduling.

3. Improved algorithm implementation

Under the background of cloud computing, big data constructs a large cloud database through cloud storage, which requires the balanced scheduling of big data to improve the balanced allocation and management efficiency of cloud storage resources. The traditional big data equilibrium scheduling method adopts subspace scheduling method, which cannot self-adaptively track the time-varying coupling characteristics of big data in cloud computing background, and the scheduling performance is not good[10]. Aiming at the disadvantages of traditional methods, this paper proposes a big data equalization scheduling algorithm based on adaptive link forwarding control in cloud computing. Based on the above feature extraction, adaptive link forwarding control is carried out, assuming that big data $\{x_n\}_{n=1}^{N}$ forms a new mapping in stack space in the form of vector control as follows:

$$u_i = \frac{1}{N}\sum_{i=1}^{N} u_i = \frac{1}{MN}\sum_{m=1}^{M}\sum_{i=1}^{N} x_{mi} \tag{8}$$

Where, x is the time series of load sampling points, and τ is cloud computing time delay. A time span of big data scheduling is recorded as follows:

$$h(\tau_{i,} t) = \sum_{i=1}^{N_m} a_i(t) e^{j\theta_i(t)} \delta(t - \tau_i(t)) \tag{9}$$

In the background of cloud computing, we construct the big data link transceiver control mechanism of cloud computing, calculate the channel equilibrium configuration parameters of big data scheduling, evaluate the weight of classifying attributes[11], and give the efficient function $E(i, j)$ of the large-scale big data task:

$$E(i,j) = \begin{cases} \dfrac{e_{ij} - e(i,j)}{e_{\max} - e(i,j)} & e(i,j) < e_{ij} \\ \dfrac{e_{ij} - e(i,j)}{e(i,j) - e_{\min}} & e(i,j) \ge e_{ij} \end{cases} \tag{10}$$

Big data resources comprehensive load is:

$$R = w_1 C_i + w_2 D_i + w_3 M_i + w_4 N_i \tag{11}$$

Under the influence of interference and noise[12-14], the information appears load imbalance. At this time, big data scheduling task trajectory A, B space direction distance is:

$$h(A,B) = \frac{1}{N_A}\sum_{i \in A} \left\| x_i^a - x_{\phi(i)}^b, y_i^a - y_{\phi(i)}^b \right\| \tag{12}$$

Where N_A is the covariance vector of the data information flow of A, and big data equalization scheduling algorithm produces information acquisition delay as follows:

$$
\begin{aligned}
T &= t_0 \le t_1 + 12 \le 12 + 5 + \max(t_2, \Delta) \le \cdots \\
&\le \underbrace{(12+5) + (9+5) + \cdots (9+5)}_{k-1} + \max(t_{2k-1}, \Delta) \\
&\le \underbrace{(12+5) + (9+5) + \cdots (9+5)}_{k-1} + 9 + \max(t_{2k}, \Delta) \\
&\le 14R + \Delta
\end{aligned} \tag{13}
$$

Using adaptive link forwarding control, assuming that big data balanced scheduling stack node, all nodes conform to multidimensional normal distribution probability[15], and obtain:

$$
H(A,B) = \begin{cases} h(A,B), if, h(A,B) \le h(B,A) \\ h(B,A), if, h(A,B) \ge h(B,A) \end{cases} \tag{14}
$$

In the background of cloud computing, the distance between grids and time characteristics of big data scheduling are obtained, and then the optimal scheduling of big data is realized.

4. Simulation experiment and result analysis

In order to test the performance of the algorithm in the implementation of cloud computing in big data balanced scheduling, simulation experiments. The experiment completed the cloud computing background design in 126 computers in 4 cabinets. These machines were connected to big data in a large cloud computing background through a gigabit Ethernet switch. The half bandwidth in the cabinet was about 14 Gbps,. The machine runs on Red Hat Enterprise Linux AS 4.0. with kernel version 2.6.9. In parameter setting, big data's degree is fixed at 15 and network node number is 800 nodes in cloud computing background. A cloud storage resource scheduling set composed of 100 tasks is constructed, and the whole big data is delimited in fuzzy stack control. It is divided into 16 virtual reduction detection cells based on which data acquisition and signal model construction are carried out. The acquired time domain waveform of big data is shown in Fig. 2.

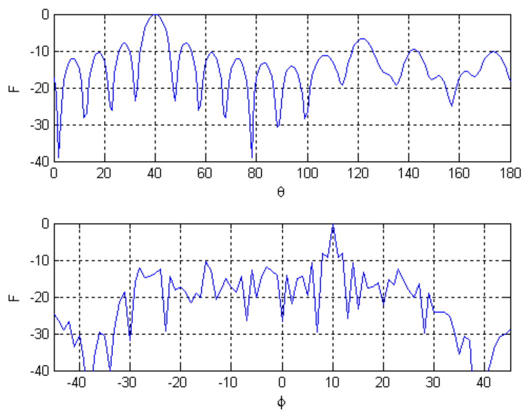
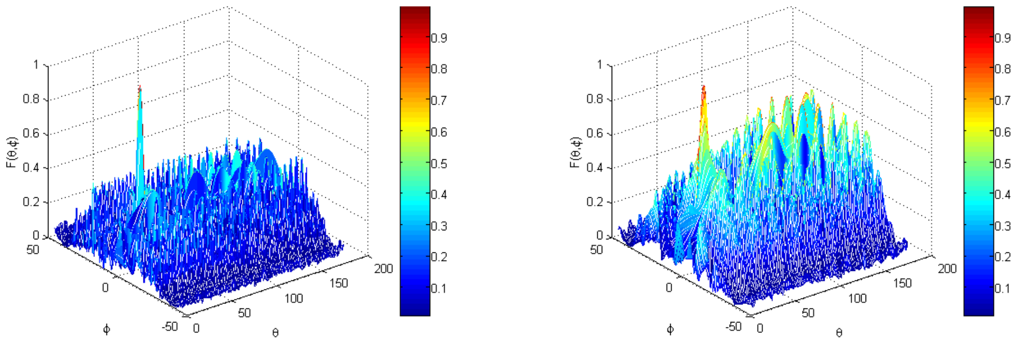


Fig. 2 Time domain waveform of big data

Taking big data given in figure 2 as the test sample, the wavelet information entropy feature is extracted and balanced scheduling is carried out. In order to compare the performance of the algorithm, we use the algorithm in this paper and the traditional algorithm, and take the response spectrum of big data equilibrium scheduling as the test index. The simulation results are shown in figure 3.

It can be seen from the figure that the spectral peak of the response spectrum of big data balanced scheduling is obvious by using the algorithm in this paper, which shows that the proposed method has a strong ability of balanced scheduling response and improves the scheduling efficiency of large data network load in cloud computing. Further quantitative analysis, this paper designed fuzzy cycle stack scheduling model for big data information flow construction and balanced scheduling, in order to schedule the fusion rate, The standard deviation and other parameters are used as the test index, and the result of big data equilibrium fusion in cloud computing is shown in figure 4.



(a) Proposed method (b) Traditional method

Fig. 3 Response spectrum of big data equilibrium scheduling

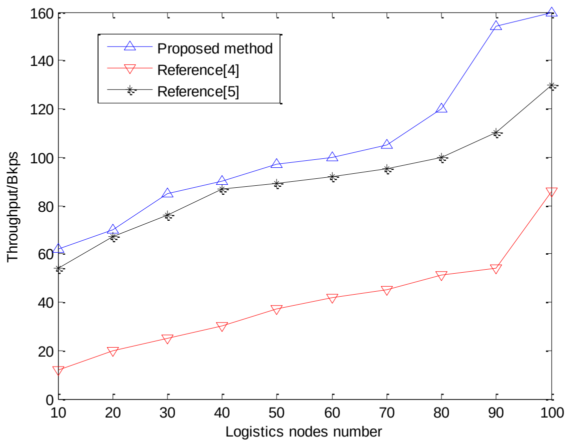


Fig. 4 Scheduling performance comparison

Figure 4 shows that by using this method, adaptive link forwarding control and balanced scheduling design can effectively eliminate incoherent data in the cluster, improve big data's load balance in cloud computing background, and improve cloud computing.

5. Conclusion

In this paper, we propose a big data equalization scheduling algorithm based on adaptive link forwarding control in cloud computing. In the background of cloud computing, big data stack storage is designed to extract the wavelet information entropy feature of big data information flow in cloud computing background, to construct the link transceiver control mechanism in cloud computing, and to calculate the channel of big data scheduling. Balanced configuration parameters, adaptive link forwarding control, to achieve big data balanced scheduling. Simulation results show that the proposed method can improve big data's load balance in cloud computing context, and improve the efficiency of big data's balanced scheduling and the ability to balance the allocation of cloud storage resources in cloud computing background. This method has good application value in big data equilibrium scheduling.

Acknowledgments

The work described in this paper was fully supported by a grant from Characteristic innovation projects of universities in Guangdong (Grant No.2017GKTSCX006)

References:

[1] GUO H P, DONG Y D, MAO H T, et al. Logistic discrimination based rare-class classification

method[J]. Journal of Chinese Computer Systems, 2016, 37(1):140-145.
[2] MA Xiantong, LUO Jingqing, and WU Shilong. Joint sorting and location method using TDOA and multi-parameter of multi-station[J]. Journal of National University of Defense Technology, 2015, 37(6): 78-83.
[3] YANG Lili and SUN Xiaowen. Precision analysis of airborne passive location of multi-stations[J]. Journal of China Academy of Electronics and Information Technology, 2014, 9(4): 348-352.
[4] YUAN Quan, GUO Jiangfan. New ensemble classification algorithm for data stream with noise[J]. Journal of Computer Applications, 2018, 38(6): 1591-1595.
[5] KOLTER J Z, MALOOF M A. Dynamic weighted majority:a new ensemble method for tracking concept drift[C]//Proceedings of the 2003 Third IEEE International Conference on Data Mining. Washington, DC:IEEE Computer Society, 2003:123-130.
[6] JU C H, ZOU J B. An incremental classification algorithm for data stream based on information entropy diversity measure[J].Telecommunications Science, 2015, 31(2):86-96.
[7] LYU Y X, WANG C Y, WANG C, et al. Online classification algorithm for uncertain data stream in big data[J].Journal of Northeastern University (Natural Science Edition), 2016, 37(9):1245-1249.
[8] HUANG S C, LIU Y. Classification algorithm for noisy and dynamic data stream[J]. Journal of Jiangsu University of Science and Technology (Natural Science Edition), 2016, 30(3):281-285.
[9] LIN J M, BAN W J, WANG J Y, et al. Query optimization for distributed database based on parallel genetic algorithm and max-min ant system[J]. Journal of Computer Applications, 2016, 36(3):675-680.
[10] ZHOU X P, ZHANG X F, ZHAO X N. Cloud storage performance evaluation research[J]. Computer Science, 2014, 41(4):190-194.
[11] QI C Y, LI Z H, ZHANG X, et al. The research of cloud storage system performance evaluation[J]. Journal of Computer Research and Development, 2014, 51(S1):223-228.
[12] CHITRA K, JEEVARANI B. Study on basically available, scalable and eventually consistent NoSQL databases[J]. International Journal of Advanced Research in Computer Science and Software Engineering, 2013, 3(7):1356-1360.
[13] MOHAN B, GOVARDHAN A. Online aggregation using MapReduce in MongoDB[J]. International Journal of Advanced Research in Computer Science and Software Engineering, 2013, 3(9):1157-1165.
[14] ZHOU Yuhao, ZHANG Hongling, LI Fangfei, et al. Local focus support vector machine algorithm[J]. Journal of Computer Applications, 2018, 38(4): 945-948.
[15] GU Q, YUAN L, XIONG Q J, et al. A comparative study of cost-sensitive learning algorithm based on imbalanced data sets[J]. Microelectronics and Computer, 2011, 28(8):146-149.