Design of parallel scheduling algorithm for distributed tasks in flash database

Ying Li, Mingchen Shao, Chaoying Gao
Admission and Employment Office, Zaozhuang Vocational College of Science & Technology,Tengzhou Sha ndong, China
ly0013@126.com, smc188@188.com, tzgchy@163.com

Keywords: flash database; distributed tasks; parallel scheduling; throughput

Abstract: The distributed task parallel scheduling problem of flash memory database is studied to improve the ability of optimal access to flash memory database and parallel scheduling of distributed tasks. The traditional method uses particle swarm optimization control algorithm for parallel scheduling of distributed tasks in flash database. The coupling characteristics of association rules of mass distributed task data stream in flash database cannot be effectively utilized, which leads to poor scheduling performance. In this paper, a distributed task parallel scheduling algorithm for flash database based on chaotic evolution feature clustering is proposed. The dual state stationary chaotic evolution feature clustering flash database scheduling model is designed, and the complex activation function is set. The layer excavates the frequent pattern set of distributed task parallel scheduling in flash database, extracts the characteristic of data information flow, and designs the transfer balance operator of distributed task parallel scheduling in flash database. Distributed task parallel scheduling of distributed task flash memory database is carried out according to real part and virtual part path, so as to improve database access ability. The simulation results show that the algorithm can effectively improve the throughput and scheduling success rate of the distributed task parallel scheduling of flash database, and the distributed task parallel scheduling of flash database is universal.

1. Introduction

Database scheduling is a hot issue in the field of computer research, its significance lies in mining effective, novel, potentially valuable and ultimately understandable knowledge from massive data. Based on this feature, distributed task parallel scheduling is carried out. In recent years, with the rapid development of streaming media network technology, mass data carried in streaming media is called explosive growth, and the research on mass data carried in streaming media is becoming more and more popular[1]. The so-called streaming media refers to the media format in which streaming transmission is used in Internet. In the flash memory database, after the user decompresses the data through the decompression device, the streaming media audio and video data will be displayed as before the transmission, and the distributed task flash database with the network technology as the carrier is in the video conference. Video-on-demand (VOD), distance teaching, online game entertainment and other fields are widely used and require a great deal of research[2].

In distributed task flash memory database, there exists cross-type data, which is characterized by uncertainty. Effective mining is the basis of realizing accurate access and scheduling of large-scale flash memory database. It is necessary to optimize the scheduling and mining the data information in the distributed task flash memory database to improve the friendliness of large-scale streaming media applications[3]. The research on the building method of cross data mining structure model in flash memory database has been paid attention to by people and experts. The traditional method uses particle swarm optimization control algorithm for parallel scheduling of distributed tasks in flash database. The coupling characteristics of association rules of mass distributed task data stream in flash database cannot be effectively utilized, which leads to poor scheduling performance. In

  DOI: 10.25236/isaicn.2019.081

order to solve this problem, a distributed task parallel scheduling algorithm based on chaotic evolution feature clustering is proposed in this paper. Firstly, the chaotic evolution feature clustering model of streaming media flash database is constructed. Based on this, the distributed task parallel scheduling of flash memory database is carried out to improve the database access performance. The simulation results show the feasibility and effectiveness of the proposed algorithm[4].

2. Chaos evolution characteristic clustering and flash database scheduling model

2.1. Chaotic evolution feature clustering

Chaos evolution feature clustering was discovered in 1963 by American meteorologist E. Lorenz in studying the bifurcation process of mechanical motion in regional microclimate and classical mechanics[5]. The chaotic evolution characteristic clustering mathematical model is expressed as follows: f is a continuous self-mapping on a closed interval I. if $f: I \to I$, it has three periodic points, there is an uncountable set $S \subset I - P(f)$, satisfying:

$$\lim_{n\to\infty}\sup\left|f^n(x)-f^n(y)\right|>0, \quad \forall x,y\in S, x\neq y \quad (1)$$

$$\lim_{n\to\infty}\inf\left|f^n(x)-f^n(y)\right|=0, \quad \forall x,y\in S \quad (2)$$

$$\lim_{n\to\infty}\sup\left|f^n(x)-f^n(y)\right|>0, \quad \forall x\in S, \forall y\in P(f) \quad (3)$$

The above formula shows a narrow window in the chaotic region with chaos on both sides. At this time, the continuous self-mapping I on the closed interval $f(x)$ appears chaotic evolution characteristic clustering phenomenon if there is a periodic point with a period of 3. In order to illustrate the phenomenon of period-doubling bifurcation intuitively, the famous Rossler system is firstly considered:

$$x=\frac{1}{2\mu}(1+\mu-\sqrt{(\mu+1)(\mu-3)}) \quad (4)$$

When $0\le\mu\le1$, the iterative system has only one stable periodic point $x=0$, and the system has two unstable 1-cycle points $x=0$ and $x=1-\frac{1}{\mu}$,. When $3.449\le\mu\le3.544$ occurs, the 2-cycle point becomes unstable, and then the two-state is stationary. The clustering model of chaotic evolution features is shown in figure 1.

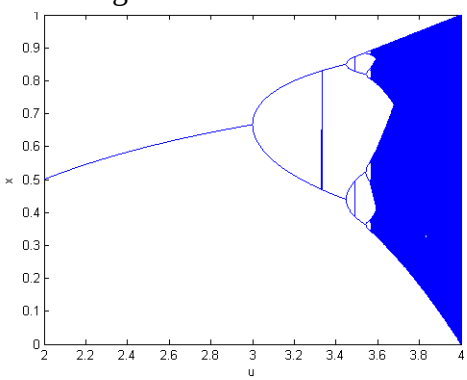

Fig. 1 Feature clustering model for distributed task scheduling

2.2. The principle of database distributed task parallel scheduling based on chaotic evolution clustering

In the process of clustering chaotic evolution features, there is a universal quantitative rule. According to this rule, the distributed task parallel scheduling of flash memory database can be carried out[6]. The convergence of the parameter μ at the bifurcation point obeys the universal law as follows.

$$\mu_n = \mu_\infty - \frac{c}{\delta^n}, \quad (n \to \infty) \tag{5}$$

The weights are reallocated to reconstruct the chaotic period-doubling nonlinear region, and the allocation time is expressed as follows:

$$T_c = \max\{0, r_d^i(k)(w_k^i = 1/N/N_c) \tag{6}$$

In the above formula, $r_d^i(k)$ represents the neighborhood change of the streaming media data in the virtual cluster transmission bandwidth, and w_k^i represents the scheduling cost weighted value of the streaming media server deployment, and the $\forall j \in [0, N_f - 1]$. The distance between the parameter sequences μ_m and μ_{m+1} form a gradual equal-to-equal ratio, and the streaming media data generates a bifurcation behavior during the scheduling process[7], and the behavior is controlled by a public-to-public ratio of Fegenbaum universal constant δ:

$$\delta = \lim_{n\to\infty} \frac{\mu_n - \mu_{n-1}}{\mu_{n+1} - \mu_n} = \lim_{n\to\infty} \frac{\Delta_{n-1}}{\Delta_n} = 4.669201\ 609\cdots \tag{7}$$

In that above formula, when $1 \le \mu \le 3$, the iteration system has an unstable 1-period point $x = 0$, the weight of the particle in the evolution of the hybrid evolutionary feature is calculated as follows:

$$\tilde{w}_k^i = \tilde{w}_{k-1}^i \frac{p(z_k / \tilde{x}_k^i) p(\tilde{x}_k^i / x_{k-1}^i)}{q(\tilde{x}_k^i / x_{k-1}^i)} \tag{8}$$

The effective sampling $N_{eff} \approx 1/\sum_{i=1}^{N} (\tilde{\tilde{w}}_k^i)^2$, is defined to set the threshold value N_{th}, and when $N_{eff} < N_{th}$ is used, the adoption time series of the data obtained from the parallel scheduling genetic evolution model of streaming media distributed tasks is $\{\tilde{x}_k^i\}_{i=1}^N$. Through the above bifurcation behavior, when the parameter μ continues to increase and $\mu > 3.544$, the scheduling of large-scale data streams becomes stable[8].

3. Improved design of database distributed task parallel scheduling algorithm

3.1. Convergence clustering of database scheduling.

From the above, because the traditional method adopts the particle swarm optimization control algorithm to carry out the distributed task parallel scheduling of the flash memory database, the association rule coupling characteristics of the mass distributed task data flow in the flash memory database cannot be effectively utilized, and the scheduling performance is not good. In this paper, based on the hybrid evolution feature clustering theory to improve the algorithm[9], a distributed task parallel scheduling algorithm based on the hybrid evolution feature clustering is proposed, and it is assumed that each data node $\tilde{\tilde{w}}_k^i$ in the hybrid evolution feature clustering process is in the flash memory database. the cross-type data set is generated, and the information flow model of the cross data set is expressed as follows:

$$D_w(S) = \frac{\sum_{u,v\in S} w_{u,v}}{|S| \times (|S| - 1)/2} \tag{9}$$

Assuming that the threshold of the weighted density of a subgraph is S = 0.2, node 0 is selected as the seed and node 0 is the first element of the subgraph (complex) C. In the neighbor of node 0, the edge weight of node 1 is the largest, assuming that the data volume of data set D is n, the dimension of data point is $\{\lambda_i : 1 \le i \le S\}$, k is the number of nearest neighbors set by the algorithm, and the information chain single flow analytical model is designed for the cross item of data set. It is assumed that the performance of packet information prediction is the GS, criterion $\{R_j : 1 \le j \le L\}$. Where S represents the amount of input data, L indicates the total bandwidth ratio used by the channel. At this point, the channel allocation packet translation wait time is:

$$W_q = W - \overline{X} = \frac{1}{\gamma}\sum_{k=1}^{K}\sum_{n=1}^{N} kp_{k,n} - \frac{(N-1)\mu + r}{\mu r} \quad (10)$$

The complex activation function is set, and the distributed task parallel scheduling is performed on the distributed task flash memory database according to the real part and the virtual part[10]. The balance processing operation expression of the activation function is expressed as follows:

$$\varphi(y) = f(y_R) + jf(y_I) \quad (11)$$

The reliability estimation error of distributed task parallel scheduling for distributed task flash database is as follows:

$$e(k) = z(k)[|z(k)|^2 - R] \quad (12)$$

Where, $e_{MDMMA\_R}(k)$, $e_{MDMMA\_I}(k)$ are the real and virtual parts of $e_{MDMMA}(k)$, $e_R(k)$, $e_I(k)$ are the virtual parts of $e(k)$. The distributed task parallel scheduling flow of flash database based on chaos evolution feature clustering is shown in figure 2.

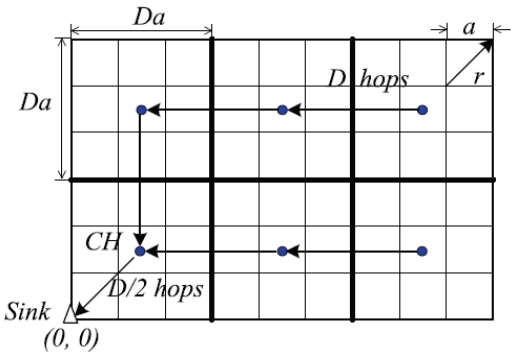

Fig.2 Distributed task parallel scheduling process of flash memory database based on hybrid evolution feature clustering

3.2. Database scheduling algorithm optimization

The K-means clustering algorithm is used to cluster and schedule the information attributes of the database. The subspace pairing between data individuals can easily lead to a local optimal solution in the process of data access[11-13]. Suppose the flash database distributed task parallel scheduling response function is:

$$\begin{aligned} H(t) &= \hat{h}(t) * p(t) * p(-t) \\ &= (\sum_{i=1}^{M} h_i^{'}(t) * h_i(-t)) * p(t) * p(-t) \end{aligned} \quad (13)$$

Mining the frequent pattern set of distributed task parallel scheduling in flash database layer by layer, extracting the characteristic of data information flow, designing the parallel task scheduling transmission operator $h_i(t)$, for convolution of flash database distributed task parallel scheduling. $n_{pi}(t)$ is the interference item of distributed task parallel scheduling of flash memory database[14], and the optimization subset of distributed task parallel scheduling of flash database is represented

as:

$$p_{ri}(t) = p(t) * h_i(t) + n_{pi}(t) \tag{14}$$

Wherein, $h_i(t)$ denotes the uniform traversal characteristic of $p(t)$ distributed task parallel scheduling query in flash memory database, and calculates the membership degree of isolated points:

$$S_{ri}(t) = S(t) * h'_i(t) + n_{si}(t) \tag{15}$$

Where, $h'_i(t)$ is a differential evolution state function in the process of parallel scheduling optimization of $S(t)$ flash database, from which the following results can be obtained:

$$\begin{aligned} r'_i(t) &= S_{ri}(t) * p_{ri}(-t) \\ &= S(t) * p(-t) * h'_i(t) * h_i(-t) + n_{1i}(t) \end{aligned} \tag{16}$$

Wherein

$$\begin{aligned} n_{1i}(t) &= S(t) * h'_i(t) * n_{pi}(-t) \\ &+ n_{si}(t) * p(-t) * h_i(-t) \\ &+ n_{si}(t) * n_{pi}(-t) \end{aligned} \tag{17}$$

Through the above-mentioned processing[15], it is set as the mutation genetic dispersion control quantity, and then the parallel scheduling algorithm for flash memory database distributed tasks is realized as shown in figure 3.

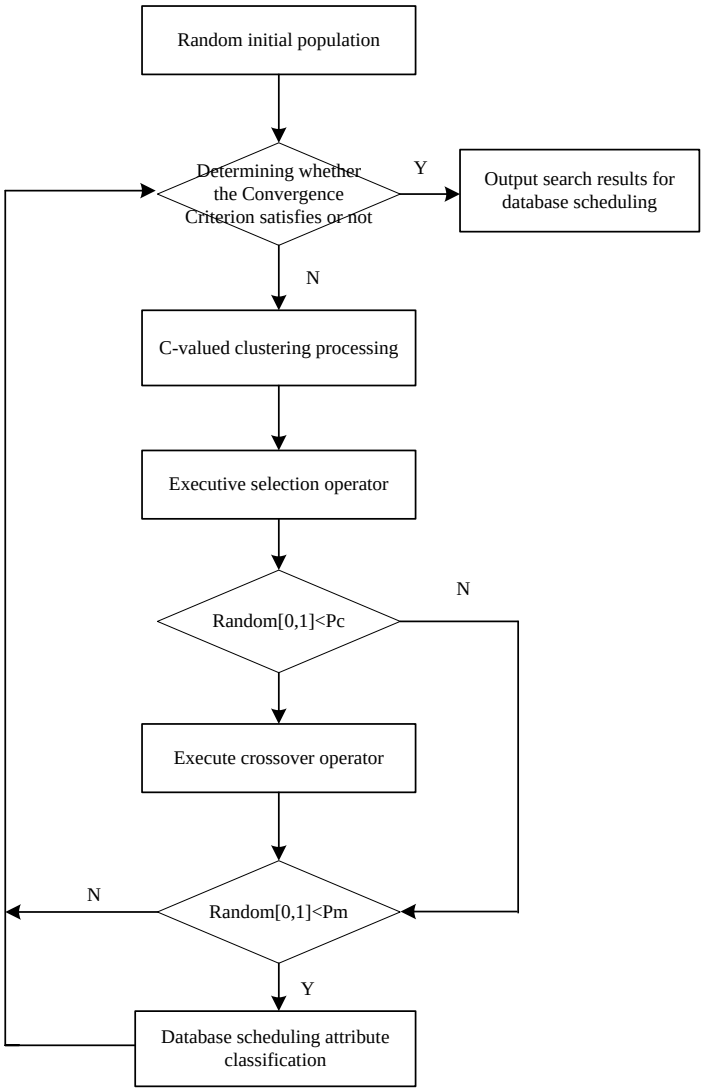

Fig. 3 Implementation flow of distributed task parallel scheduling for flash memory database

4. Simulation experiment and result analysis

In order to test the performance of this algorithm in distributed task flash database scheduling, simulation experiments are carried out. The hardware environment of the simulation experiment is as follows: the high-speed processing computer of CPU model is i5x2400, and the memory of 8 GB, operating system is 64-bit system of Windows 7. In the experiment, 128 deployment points were selected to construct the streaming media server cluster large network database. The parameter determination of chaos evolution feature clustering algorithm will affect the mining performance and convergence rate of cross-type data in the whole flash memory database. According to the simulated environment, the population size of chaos evolution feature cluster is 40,40. The probability of uniform hybridization is 0. The hybrid probability of PMX is 0. 32, the variation rate is 0. 09. The experiment data of flash memory database in the lab choose *Iris* dataset and H dataset of *Wine* database, in which the *Wine* dataset is divided into three categories, each class includes 50 samples, each sample has 4 attributes, and the *Wine* dataset is divided into three categories. Each category includes 59, 71, and 48 samples with 13 attributes per sample. Using this algorithm and traditional algorithm, the parallel scheduling data throughput and scheduling success rate of streaming media data distributed tasks are used as the test indexes. The simulation results are shown in figure 4 and figure 5.

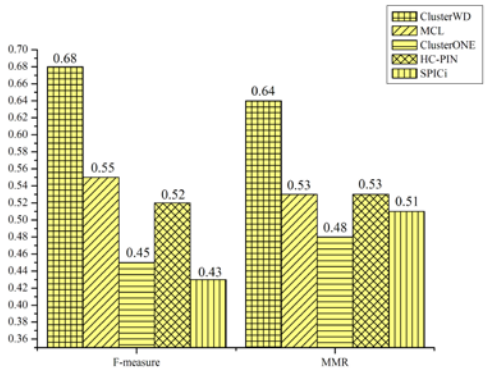

Fig. 4 Throughput of distributed task parallel scheduling in database

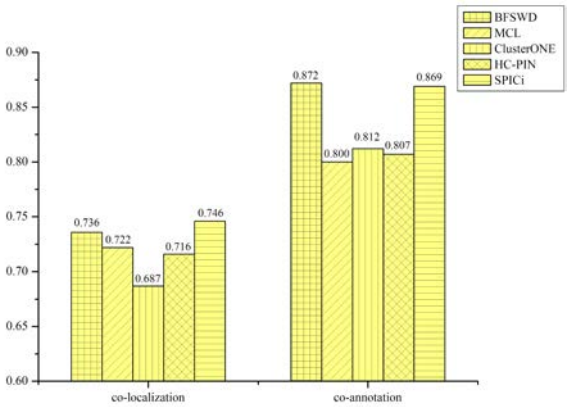

Fig. 5 Comparison of the success rate of flash database scheduler

By analyzing the results of figure 4 and figure 5, we can see that this algorithm can effectively improve the throughput and scheduling success rate of flash database scheduling. Through the improvement of the algorithm in this paper, the hierarchical fusion degree of streaming media data can reach more than 90%. The improved algorithm is only between 10% and 50%, which shows the superior performance of the proposed algorithm in the implementation of large database access scheduling. The precision of database scheduling is tested, and the comparison results are shown in

Table 1. The analysis shows that the precision of parallel scheduling of distributed tasks in flash memory database is higher than that of the method in this paper.

Table 1 Comparison of precision rates

Number of iteration steps	Proposed method	Reference[4]	Reference[5]
100	0.934	0.873	0.864
200	0.985	0.890	0.894
300	0.992	0.912	0.911
400	1	0.934	0.956

5. Conclusion

In this paper, a distributed task parallel scheduling algorithm for flash database based on chaotic evolution feature clustering is proposed. The dual state stationary chaotic evolution feature clustering flash database scheduling model is designed, and the complex activation function is set. The layer excavates the frequent pattern set of distributed task parallel scheduling in flash database, extracts the characteristic of data information flow, and designs the transfer balance operator of distributed task parallel scheduling in flash database. Distributed task parallel scheduling of distributed task flash memory database is carried out according to real part and virtual part path, so as to improve database access ability. The simulation results show that the algorithm can effectively improve the throughput and scheduling success rate of the distributed task parallel scheduling of flash database, and the distributed task parallel scheduling of flash database is universal. This method has good application value in distributed task parallel scheduling.

References

[1] Z ZHANG Chaohua, LI Lianhe,YUN Guohong. Study on moving dislocations in decagonal quasicrystals[J]. Chinese Journal of Solid Mechanics, 2017,38(2): 165 -169.

[2] ZHANG Wei,WANG Zhijie. Research on Join Operation of Temporal Big Data in Distributed Environment[J]. Computer Engineering, 2019, 45(3): 20-25,31.

[3] ZHANG Haoshenglun,LI Chong,KE Yong,ZHANG Shibo. A Distributed User Browse Click Model Algorithm[J]. Computer Engineering, 2019, 45(3): 1-6.

[4] GAO Jun,HUANG Xiance. Design and Implementation of Correlation Weight Algorithm Based on Hadoop Platform[J]. Computer Engineering, 2019, 45(3): 26-31.

[5] ZHOU Qi,CHAI Xiaoli,MA Kejie,YU Zeren. Design and Implementation of Tucker Decomposition Module Based on CUDA and CUBLAS[J]. Computer Engineering, 2019, 45(3): 41-46.

[6] TAO Wenjing,LU Yang,WEI Xing,JIA Xiangli. Software Implementation of Precision Clock Synchronization Based on PTPd2[J]. Computer Engineering, 2019, 45(3): 47-53,59.

[7] LI Chaofan,CHEN Qingkui. GPU Cluster Power Consumption Collection and Monitoring System Based on Sensor[J]. Computer Engineering, 2019, 45(3): 65-72.

[8] PAN Chengsheng,JIA Yaru,CAI Ruiyan,YANG Li. Routing Strategy for Spatial Information Network Based on MPLS[J]. Computer Engineering, 2019, 45(3): 85-90.

[9] CHEN Jiaojiao,ZHU Weiping,TU Mingxuan,TANG Yijie,SUN Zeyu. Mobile Group Recognition and Location Prediction Method in Large Shopping Mall[J]. Computer Engineering, 2019, 45(3): 78-84.

[10] WANG Zhihe, HUANG Mengying, DU Hui, QIN Hongwu. Integrated algorithm based on density peaks and density-based clustering. Journal of Computer Applications, 2019, 39(2):

398-402.

[11] FARNADI G, BACH S H, MOENS M F, et al. Soft quantification in statistical relational learning[J]. Machine Learning, 2017, 106(12): 1971-1991.

[12] RAMES A, RODRIGUEZ M, GETOOR L. Multi-relational influence models for online professional networks[C]// Proceedings of the 2017 International Conference on Web Intelligence. New York: ACM, 2017: 291-298.

[13] ZHANG B Z, GUI X, HE T, et al. A novel recommender algorithm on fusion heterogeneous information network and rating matrix[J]. Journal of Computer Research and Development, 2014, 51(S2):69-75.

[14] LIN X Y, DAI L L, SHI S H, et al. Matrix factorization recommendation based on topic model[J]. Journal of Computer Applications, 2015, 35(S2):122-124.

[15] Liu M,Shi Y.Model predictive control for thermostatically controlled appliances providing balancing service[J].IEEE Transactions on Control Systems Technology,2016,24(6):2082-2093.