Research on Optimization Strategy of Virtual Resource Scheduling Based on Improved Frog Leaping Algorithm

Xinxin Guo

Address: Jilin Agricultural Science and Technology college, No.77, Hanlin Road, Changyi District, Jilin city, Jilin province, China.

15043237210@163.com

Abstract. In order to mobilize valid data in virtualized resource pool quickly, reduce resource scheduling execution time and improve resource utilization. The paper proposed a virtual resource scheduling strategy based on improved frog leaping algorithm. A mathematic model of leapfrog algorithm in cloud computing environment is established to carry out the task of the shortest time and the maximum utilization of resources for the purpose of incorporating greedy algorithm to find the optimal scheduling strategy. The results of Cloudsim simulation show that this research can improve the utilization of virtual data center resources, get efficient dynamic resource strategy, save resource overhead. The algorithm provides important theories for the future of cloud computing virtualized complex dynamic resource scheduling and hardware resource loading.

Introduction

Virtualization of the data center resources will be integrated into a huge pool of resources. Through the Internet and virtual server "computing pool", "network pool" and "storage pool" abstraction[1], virtualization will show the fast and effective resource sharing and transparency. Therefore, it is very important to design the optimal resource scheduling strategy in virtual resource scheduling[2-4].

In the traditional sense of data center is resource allocation, management strategies and scheduling methods[5]. It can no longer meet the needs of the use of big data. In this paper, we propose a virtual resource scheduling optimization strategy based on improved frog leaping algorithm to find the optimal scheduling strategy and improve the overall energy efficiency of the system.

Virtualization Resource Pool Scheduling

The basic framework of virtualized resource allocation includes resource pool, service layer and task layer scheduling, and scheduling overall performance[6].The multiple problems lie in the server CPU performance; capacity in the resource pool and the bandwidth of the "network pool". The virtualized computing resources V_m are:

$$V_m = \{mips, nbw, mem, ...\} \tag{1}$$

Eq. 1 show that, $mips$ was the average server speed, nbw was network bandwidth, mem was the CPU capacity.According to the above task nodes scheduling resources, we design efficient and rapid processing of resource scheduling algorithm.

Resource Pool Model of Frog Leaping Algorithm

It generates request tasks and tasks group randomly in the resource pool task layer, each task is defined , mark the global optimal solution as p_b at last. Then the whole task group is divided into m subgroups, each subgroup has n tasks.

$$D_s = rand(\ )*(p_b - p_w) \tag{2}$$

In each subgroup, the optimal individual task is marked as p_o, and the worst value task entity is recorded as p_w. After the above mathematical model of the task packetization operation, the local worst-case solution for the distance update operation, the update formula is as follows:

$$STEP = \min\left(rand(\ )\times(p_b - p_w), S_{max}\right) \tag{3}$$

In the Eq. 2; Eq. 3 and Eq. 4, parameter as: p: request mission (number of frogs); m: task subgroups; n: the number of tasks for each subgroup; p_b: global optimal solution; p_o: local optimal solution; p_w: local worst-case solution; T: number of local iterations; D: global iteration number, $rand(\ )$ generate a random number in the range $[0,1]$.

$$ND_w = STEP + D_s\left(-D_{max} \le D_s \le D_{max}\right) \tag{4}$$

If it defines $S_{max} = 3$, $rand() = 0.5$, $P_b = \{2,1,5,3,4\}$, $P_w = \{1,3,5,4,2\}$, then it will get the latest optimization node $newD_w = \{1,2,5,4,3\}$.

Application of Improved Frog Leaping Algorithm in Virtual Resource Scheduling

Local Optimization Strategy. through iterative distance formula, local search strategy obtains a high degree of adaptability of the individual tasks.

Assuming the average fitness of the task individuals as f_{avg}, the number of task individuals as p, the fitness of the first task $f(i)$, then it gets:

$$f_{avg} = \frac{1}{p}\sum_{i=1}^{p}f(i) \tag{5}$$

If $f(i) < f'_{avg}$, the task individuals approximate the optimal task, the fitness value is high and the ability to find the best; otherwise $f(i) < f_{avg}$ then we get：

$$S_i = \begin{cases} rand(\ )*(p_b - p_w) & f(i) \le f'_{avg} \\ rand(\ )*\left[\alpha(p_g - p_w) + (1-\alpha)(p_b - p_w)\right] & f_{avg} \le f(i) \le f'_{avg} \\ rand(\ )*(p_g - p_w) & f(i) \le f_{avg} \end{cases} \tag{6}$$

Eq. 6 show that, $rand(\ ) \in [0,1]$, α is a learning factor，f_g represents the fitness value of the global optimal task, α is decided by $f(i)$.

Global optimization strategy. The genetic operator is applied to SFLA[7], and the best individuals in each task subgroup are cross-trained between groups to achieve information exchanging and sharing. It chooses two factors betwe $p_b(1)$ and $p_b(m)$ to iterate，then make cross calculation as：

$$\begin{cases} p(i) = r_1 p_b(i) + r_2 p_b(j) + r_3 p_g \\ p(j) = r_1 p_b(j) + r_2 p_b(i) + r_3 p_g \end{cases} \tag{7}$$

Eq. 7 show that, $i, j \in [1, m]$, $r_1, r_2, r_3 \in (0,1)$, $r_1 > r_2 > r_3$, and $r_1 + r_2 + r_3 = 1$, with the best individual crossover of different subgroups, it gets better individual.

Application of Improved Frog Leaping Algorithm in Resource Scheduling. Although the virtualization resource pool is huge, leap frogging algorithm has high computational performance and global search capability, it takes time to seek the optimal solution in resource scheduling, so it integrates greedy algorithm.

Suppose A_{ij} is a subset that contains a_k, we use the recursive formula to calculate the optimal solution, $c[i, j]$ said the optimal solution size of the collection T_{ij}, then:

$$c[i, j] = \begin{cases} 0 & T_{ij} = \phi \\ \max_{a_k \in T_{ij}} \{c[i, k] + c[k, j] + 1\} & T_{ij} \neq \phi \end{cases} \quad (8)$$

Replace the frog leaping algorithm partial search links to speed up the resource scheduling process.

Simulation Experiment and Analysis

In order to verify that the experimental results are representative and effective, this paper uses Cloudsim simulation platform[8-9] to simulate SFLA, ISFLA, ACO[10] and Min-Min algorithm and compare the results. According to the above resource scheduling procedure: the number of resource nodes set in experiment is 5; the processing power of the virtualized server is 200 to 300Mips; the number of tasks is 20 to 100 and the task size is 1T to 5T.

In order to facilitate the analysis of results, the four algorithms to perform 10 times the average, the time comparison chart shown in Figure 1. The four algorithms in the overall system resource utilization than the results, the relationship shown in Figure 2.

As can be seen from Fig. 1, as the amount of tasks increases, ISFLA performs less tasks than SFLA and other two algorithms. Because SFLA repeatedly chooses the optimal path based on the amount of information the individual carries. The difference between the two algorithms is not obvious. ISFLA has an advantage in time to complete when scheduling large amounts of data in a "storage pool."
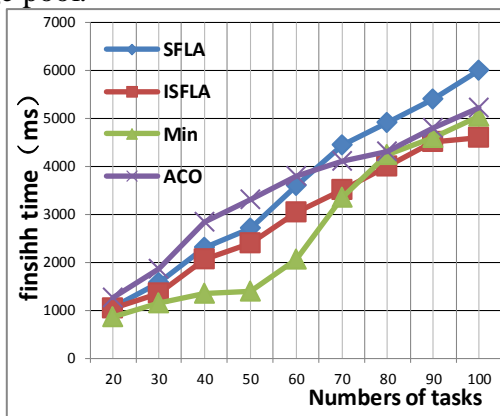

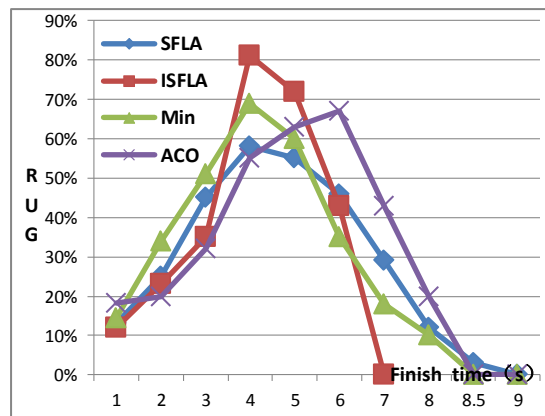Figure 1. Compare with finish time

Figure 2. Compare with resource utilization

As can be seen from Fig. 2, ISFLA is superior to SFLA and other two algorithms in handling the overall utilization of virtualized resource pools, because ISFLA mainly prioritizes urgent resource or emergency tasks, complements the idle state during global optimization, Accelerate the overall task to complete the progress.

Table 1   The result of tasks allocation on each resource

Resource ID	20（Tasks）				50				100			
	S	I	M	A	S	I	M	A	S	I	M	A
1	2	1	0	2	6	8	3	6	13	15	13	14
2	3	2	2	2	10	12	10	15	21	24	29	23
3	5	8	4	6	12	20	13	12	25	42	19	38
4	4	7	5	5	12	6	14	11	27	13	19	16
5	1	2	9	5	10	4	10	6	14	6	20	9

Table 1 shows the number of tasks assigned to each resource node using four algorithms, where S represents a frog leaping algorithm, I represents an improved frog leaping algorithm, M represents a Min-Min algorithm, and A represents an ant colony algorithm. It can be seen from the table improved frog leaping algorithm in the implementation of tasks load balancing higher.

Summary

This paper presents resource scheduling strategy based on ISFLA algorithm, which is applied in virtualized resource pool and simulated with Cloudsim platform. The results show that ISFLA performs well in task completion time and overall resource utilization, and can effectively solve the problem of cloud Computational virtualization of complex dynamic resource scheduling and hardware resource load usage.

Acknowledgements

References

[1] Subsense:a universal change detection method with local adaptive sensitivity. St-Charles P L,Bilodeau G A,Bergevin R. IEEE Transactions on Image Process. 2015

[2] Analysis of the Population-Based Ant Colony Optimization Algorithm for the TSP and the QAP. Oliveira S,Hussin M S,Roli A,et al. 2017IEEE Congress on Evolutionary Computation(CEC) . 2017

[3] A Parallel Cooperative Hybrid Method Based on Ant Colony Optimization and 3-Opt Algorithm for Solving Traveling Salesman Problem. Gülcüg,Mahi M,BaykanO K,et al. Soft Computing . 2016

[4] Research of resource sheduling based on ACA-GA in the cloud computing. Chen X. International Jouranl of Grid and Distributed Computing . 2016

[5] Arash Ghorbannia Delavar，Yalda Aryan．HSGA: a hybrid heuristicalgorithm for workflow scheduling in cloud systems．Cluster Compute，2014;17:129—137

[6] Joris Borgdorff，Jean-Luc Falcone，Eric Lorenz et al.Foundations of distributed multiscale computing: Formalization，specification，and analysis [J] Journal of Parallel and Distributed Computing，2013，73(4):465-483.

[7] Improved Shuffled Frog Leaping Algorithm And Its Application In Node Localization Of Wireless Sensor Network[J]. Fan Tanghuai, Lü Li, Zhao Jia. Intelligent Automation & Soft Computing . 2012 (7)

[8] Goyal T,Singh A,Agrawal A.CloudSim simulator for cloud computing infrastructure andmodeling[J]. Procedia Engineering,2012,38：3566-3572.

[9] Belogazov A,Abawajy J,Buyya R.Energy-Aware Resource Allocation Heuristics for Efficient Management of DataCentersforCloud Computing[J].Future Generation Computer Systems, 2012, 28(5): 755-768.

[10]An ant colony optimization-based fuzzy predictive control approach for nonlinear processes[J].S.Bououden,M.Chadli,H.R. Karimi.Information Sciences.2014

Author: Xinxin Guo (1987-10-19), female, Jilin province, Jilin Agricultural Science and Technology University, Internship researcher, Research direction: Computer Network Application; Virtual Reality