

A REGION GROWING ALGORITHM FOR SOLVING CVE PARTITIONING PROBLEM

Xiaomei Hu, Zhengjun Zhai , Xiaobin Cai

School of Computer, No.127 West Youyi Road, Northwestern Polytechnical University, Xi'an, China 710072,sufeimasohxm@163.com

Abstract: Collaborative Virtual Environment (CVE) systems offer multi-user support, in which many users share a virtual world and work collaboratively, and in which multiple-server architectures are widely used and multiple servers deployed, with each one handling a part of the virtual world. One of the key issues in the design of scalable CVE systems based on multiple-server architecture is the partitioning problem. In this paper, a new quality function of partitioning is used to evaluate the partitioning method and a region growing (RG) algorithm is proposed for solving the partitioning problem in CVE systems. The RG algorithm regards the partitioning problem as a dynamic assignment of cells to the servers. Some cells are chosen as growing points, and their neighbor cells are merged into the same region. Users in the same region are connected with the same server until the server is overloaded. Through the experiment, we discover that our method can provide better partitioning solutions with shorter execution time than the methods proposed in the previous literature of large-scale CVE systems. Therefore, the proposed implementation of RG algorithm improves scalability and efficiency of CVE systems.

Key words: collaborative virtual environment; partitioning; region growing.

1. INTRODUCTION

Collaborative virtual environment (CVE) supports users working on different computers to share a virtual environment and accomplish collaborative works. A scalable CVE system usually adopts the multiple-server architecture so that more users can participate in the system. A CVE is partitioned into several parts and each server is in charge of a part of the virtual world. When a new user logs in the CVE system, it will select one of

Please use the following format when citing this chapter:

Hu, Xiaomei, Zhai, Zhengjun, Cai, Xiaobin, 2006, in International Federation for Information Processing (IFIP), Volume 207, Knowledge Enterprise: Intelligent Strategies In Product Design, Manufacturing, and Management, eds. K. Wang, Kovacs G., Wozny M., Fang M., (Boston: Springer), pp. 960-965.

the servers in the system to connect with, which makes the server side increase the computation workload. In addition, when the user modifies the state of system, it sends the update message to its server, which in turn propagates this message to other servers or clients to maintain the consistency of the system, which makes the server side increase the communication cost. There are two kinds of communication messages in the CVE systems: Inter-server ones and Intra-server ones as shown in Figure 1.

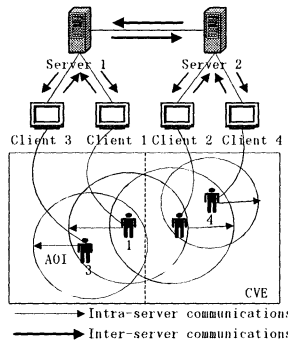


Figure 1. Two kinds of communication messages in CVE

Lui and Chan¹ defined a quality function C_p consisting of the computing workload C_p^c and overall inter-server communication cost C_p^i to evaluate the partitioning algorithm and used Linear Optimization Technique (LOT) algorithm to minimize the value of function C_p . The LOT algorithm is periodically executed for adapting the partition to the current state of the CVE system, which may take much execution time.

P. Morillo et. al.^{2,3,4,5} put forward several heuristic search methods and presented a comparison study of several heuristics methods for solving the partitioning problem in CVE systems. These methods improve the performance of the partitioning algorithm by providing better C_p values and requiring shorter execution time than LOT method. The characterization study of CVE system⁶ shows that the partitioning problem seriously affects the overall performance of the CVE system, since it determines not only the workload each server is assigned to, but also the inter-server communication requirements. According to the character of the CVE system, the effective partitioning algorithm should not only balance the workload among the servers in the system in such a way that none of them reaches its saturation point to improve the system throughput, but also assign the users doing the same collaborative task to the same server to minimize the inter-server messages in order to reduce the network traffic.

2. A NEW PARTITIONING APPROACH

In this section, a new quality function of partitioning is introduced and a region growing algorithm for solving the partitioning problem is proposed.

2.1 Quality function of partitioning

Average system response time is the optimal parameter to evaluate the partitioning algorithm. But it is hard to get the actual data before a real CVE system is constructed. Through the analysis of the CVE system response character⁶, we propose the quality function of partitioning R_p . Assuming that CVE where the load is W is partitioned into p regions and the network traffic of inter-server communication of region i and region j is denoted as $D_{i,j}$, the inter-server communication is defined as

$$C_p^L = \sum_{j=1}^p \sum_{i=1}^p D_{i,j} \quad (1)$$

Let the threshold δ be the maximum number of clients that a server can handle smoothly. Assuming the workload of the server i is W_i and the threshold of the server i is δ_i , the CPU utilization of the server i can be approximated by:

$$L_i = \frac{W_i}{\delta_i} \quad (2)$$

The quality function R_p is defined as

$$R_p = C_p^L \times \max\{1, L_1, L_2, \dots, L_p\} \quad (3)$$

The best partitioning strategy P^* should make the quality function R_p^* minimum. In order to get R_p^* , the inter-communication should be minimized and the workload of servers should be lower δ . If the workload of servers does not balance, some servers can easily reach to the saturation point to make them overloaded. If servers are overloaded seriously, namely the value of L_i is very high, the value of R_p will increase rapidly. Therefore, R_p become a near optimal quality function to evaluate the partitioning algorithm.

2.2 A RG algorithm for partitioning the CVE

CVE is divided into $N \times M$ cells and the size of cell is the average value of the users' vision domain radius. The neighborhood of a cell is the adjacent

cells sharing the same border of the cell. The users only communicate with the other users in the neighborhood of its residing cells. So communication cost between two partitions can be simplified to the case that we only need to consider the users of the adjacent cells that are divided into different partitions. Suppose each cell c_i in CVE is labeled with the number of users that it contains $L(c_i)$ and the assigned server $S(c_i)$, the communication cost in the system is considered the sum of users in the cells c_i whose adjacent cells have different value of $S(c_i)$. In order to find the optimal partitioning, the communication cost should be reduced as the same time each server should not exceed δ . Therefore, the cells which contain fewer users can be considered as the border of partitions.

RG algorithm first initializes growing points which is near the general idea of the graph theoretic method. Assuming that there are k servers in the system, CVE is initially separated into k regions. Then all the cells in each region are sorted by their $L(c_i)$ value, and the first k cells called mass-center (mc_k) that are located in different regions are chosen as growing point of each region and assigned to different servers. At this time, the workload of each server is the value of $L(mc_k)$. Next is to assign the other cells to servers. The neighborhood of the cell mc_k is defined as $N(mc_k)$ and the cell which has the maximum value $L(c_j)$ ($c_j \in N(mc_k)$) and is not assigned to servers is merged into the cell mc_k and assigned to the server k if the workload of the server k is not beyond the threshold. This process is cycled until all the cells are assigned to servers and the pseudo-code is shown in Algorithm 1:

- Algorithm 1. Region Growing algorithm

Begin

Divide CVE into k regions

let $S(c_i)=0$;

For $i=1$ to k do

Choose the biggest element mc_i of the region i ;

$S(mc_i)=i$;

$n--$;

Workload(Server _{i})= $L(mc_i)$;

Endfor

While($n <$ the number of the cells)

For $i=1$ to k do

List=obtain($N(mc_i)$);

Sorted-list=Quicksort(List);

c_j =The first element in Sorted-list;

If ($S(c_j)=0 \& \& L(c_j)+Workload(Server_i) < \delta$)

Merge(c_j, mc_i);

$S(c_j)=i$;

$n--$;

Workload(Server _{i}) += $L(c_j)$;

Endif

Endfor
Endwhile
End.

According to the RG algorithm, partitioning problem is the dynamic assignment of cells to the servers. In this process, the workload of server i increases gradually until server i reaches to a saturation point δ . The CVE systems are extremely changing environments since users can join or leave the system at any time and they can also freely move throughout the virtual world. The RG algorithm should be executed periodically. With this algorithm, the servers cannot be overloaded because new cells are not assigned to the server when the number of users in the region managed by the server is close to the threshold δ . Since the cells with high $L(c_i)$ value in the neighborhood of mc are first assigned to the same region with mc , the cells with low $L(c_i)$ value become the border of regions, which reduces the inter-server communication.

3. EXPERIMENTS AND RESULTS

In this section, we present the performance evaluation of the RG algorithm for solving the partitioning problem in CVE systems. We use the hardware platform of 2.4 GHz Pentium IV with 512 Mbytes of RAM and develop a simulator in C++ running on windows2000 to model a CVE system based on the hardware platform. The virtual environment is divided into 609 cells. There are six servers in the CVE system and the threshold δ of each server is 100. Users are randomly distributed in the CVE in the initialization. During the simulation, they move to six hotspots and finally show clustering distributions in the CVE.

The value of the quality function R_p and execution time T/ms is used to evaluate partitioning algorithm. According to [2], the ACS search method currently provides the best heuristic method for solving the CVE partitioning problem. Table 1 shows the results of two partitioning algorithms. It proves that RG algorithm provides a lower value of R_p and requires a shorter execution time than the ACS method. Thus we learn that the ACS method is not the best partitioning solution because it does not provide the lowest R_p value though it can provide the lowest C_p value. In addition, the execution time of ACS method increases more quickly with the users increasing in the CVE because the complexity of calculation in ACS algorithm is $O(B)$ where B is the number of users in the border of regions which increases with the number of users in the system while the complexity of calculation in RG algorithm is $O(n)$ where n is the number of the cells in the CVE system which is a fixed value.

Table 1. The results of partitioning CVE

user	RG		ACS	
	R_p	$T(ms)$	R_p	$T(ms)$
300	36	613	45	1413
400	42	688	49	1920
500	53	725	57	2491
600	64	852	66	3214

4. CONCLUSIONS AND FUTURE WORK

The CVE systems are becoming an important research area in parallel and distributed processing field. In this paper, we discussed a new region growing algorithm for solving the partitioning problem in scalable CVE systems. A new quality function of partitioning is given to evaluate the partitioning algorithm for a large CVE system with clustering distributions of the existing users in the systems. The experiments prove that the region growing algorithm takes lower execution time and provides better solutions to the partitioning problem than current approaches. As future work to be done, one should consider use of grouping strategy in the partition management in CVE systems with multiple-server architecture to reduce the network traffic.

5. REFERENCES

1. John C.S.Lui and M.F.Chan,(2002), An Efficient Partitioning Algorithm for Distributed Virtual Environment Systems, *Parallel and Distributed Systems*, Vol.13, pp.193-211.
2. P. Morillo, M. Fernandez and J.M. Orduna, (2003), A comparison study of modern Heuristics for Solving the Partitioning Problem in Distributed Virtual Environment Systems, In: *proceedings of the International Conference on Computational Science and its Applications*, Montreal, Canada, Springer, pp. 458-467.
3. P. Morillo and M. Fernandez,(2003),A GRASP-based algorithm for solving CVE partitioning problem, In: *Proceedings of, Parallel and Distributed Processing Symposium*, Nice, France, IEEE Computer Society, pp.60.2
4. P. Morillo, M. Fernandez and J. M. Orduna, (2003), An ACS-Based Partitioning Method for Distributed Virtual Environment Systems, In: *Proceedings of, Parallel and Distributed Processing Symposium*, Nice, France, IEEE Computer Society, p.148.2
5. P. Morillo, P. Lopez , J. M. Orduna, and M. Fernandez, (2003), A New Genetic Approach for the Partitioning Problem in Distributed Virtual Environment Systems, In: *Proceeding of the 10th Conference of the Spanish Association for Artificial Intelligence*, Spain, Springer-Verlag GmbH, pp.76-85.
6. P. Morillo, J. M. Orduna, M. Fernandez and J. Duato, (2003), On the Characterization of Distributed Virtual Environment Systems, In: *proceedings of Euro-Par2003 Parallel Processing*, Austria, Computer Science 2790 Springer, 2003, 1190 – 1198.