Volunteer Computing Resource Discovery Algorithm in Grid Environments

Jiang Hong
Information Center
Hohai University
No.1 Xikang Rd, Nanjing City, JIangsu Province
R.P.China, 210098

Cai Hong
College of Computer & Information Engineering
Hohai University
No.1 Xikang Rd, Nanjing City, JIangsu Province
R.P.China, 210098

Abstract: In grid environment, the peer-to-peer computer network is beneficial to build a scalable volunteer computing environment. Large-scale computations often need a great deal of scalable volunteer computing resources. In this paper a resource discovery and organization algorithm was proposed in the peer-to-peer computer network environment. This algorithm improved on the resource discovery methods in existence. At last, it can be shown that this algorithm have many advantages, compared with others.

Key-Words: volunteer computing; grid computing; resource discovery

1 Introduction

With the development of sciences and techniques, the scale of high performance computing increasingly enlarges, and computing grid becomes one of the best method for this demand. Compared with the traditional clusters and PCs, the computing resources are decentralized, so the volunteer computing is often applied. For this, we should improve the speed of searching so that the resource in the large field can be full applied, meanwhile our computing speed can be improved too. This is why we research the computing grid. In this paper, we focus on large-scale resources discovery in the grid environments.

Recently, there are three algorithms based on content discovery [6]:

1. Central index algorithm. This algorithm was made porpular by Napster. It was not a pure P2P model and depended on a server to discover resources. Hence, there was the disadvantage of the C/S model in this algorithm.
2. Document routing algorithm. This algorithm was used by FreeNet and based on a hash of the document's contents and its name. It has the disadvantage that the network is apt to become many information islands.
3. Flood request algorithm. This algorithm discovered resources by means of broadcast. As a result, the network bandwidth consume will increase.

Besides these, there are many other algorithms focusing on research precise content discovery , which is differ from our research [6].

We can use these algorithms into large-scale resources discovery. Based on this, Central index algorithm and Flood request algorithm were applicable for not only particular resources, but also discovery of large-scale computing resources. However, these algorithms had the disadvantages as before, while they were applicable for large-scale computing resources discovery. In this paper we proposed a extendable algorithm based on volunteer computing in P2P network.

The rest of this paper is as follows. In section 2, we introduce our related work.

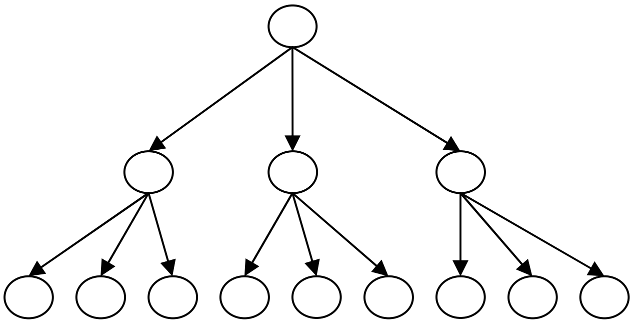

Fig. 1 Tree

Section 3 describe our design algorithm. Section 4 evaluate our algorithm with others. We conclude in section 5.

2 Problem Formulation

In this section we consider the problem of how to research a set of key in a research tree to maximize the number of resources as quick as possible.

Fig.1 a tree that turn the problem of resource discovery into a problem of searching a tree.

Definition 1 field tree
In a general tree, all of nodes are in the equal position on the network.

Every node in the tree present a computing node, and it might be a PC or a cluster.

3 Algorithm Design

The algorithm we designed called field algorithm, which is based on volunteer computing and has a good expandability.

Algorithm

Design: this algorithm is same with discovery of large-scale computing resources, and it orients FIELD (FIELD, a computers' address computed by field algorithm), then it accomplishes the project which is most beneficial to the host computer for submitting the computing tasks in order to perform the goal--the most computing resource can be found in the shortest time.

In:

Hypothesis: assume that the total number of computing nodes is m; among these nodes, h1 is the node that submits the computing task.

Out:

Find the acquirement computing resources; finish the task.

3.1 The computing node that submits the task sends out the requirement to other computing nodes by searching the FIELD;

3.1.1 FIELD

The nodes in the network are given 2 property: (1) close IP; (2) same network type; we can mark out some subnets based the two attributes and n presenting the number given for rational nodes. FIELD is an address table preserving the deputy nodes of as many FIELDs. LOCAL is also an address table preserving all of nodes in its FIELD. This method can improve the speed of searching computing resources.

Description:

1. The node h1 sends out requirements to its neighbors. And its neighbor node compares with the criterions as follow:

   a. it is a close node to h1;
   b. the network type is same with h1's;
   c. flag=0;

   if the neighbor node accords with c and one of conditions a and b, the counter will add 1 and the flag of neighbor as a sign of being visited will become 1, or only the flag will become 1.
2. The node that joined in a FIELD can search other nodes. If its neighbor node is according with condition a or b, it will send its neighbor IP address to h1.
3. H1 received an IP address, and it will examine the number of counter. It puts the new address into its LOCAL. if the number of counter is less than the n, the counter mount will add 1 up
4. Repeat 2, 3 until the number of the counter is equal to the n. H1 will put the last node that we called h2 into its FIELD, and refuse to receive other nodes. H1 informs h2 about establishing a new FIELD, and informs the nodes in its LOCAL about stopping search other nodes.
5. H2 establishes a new FIELD and puts h1 in it. Like h1, it repeats 2, 3 until 4. And h2 can search the last node during

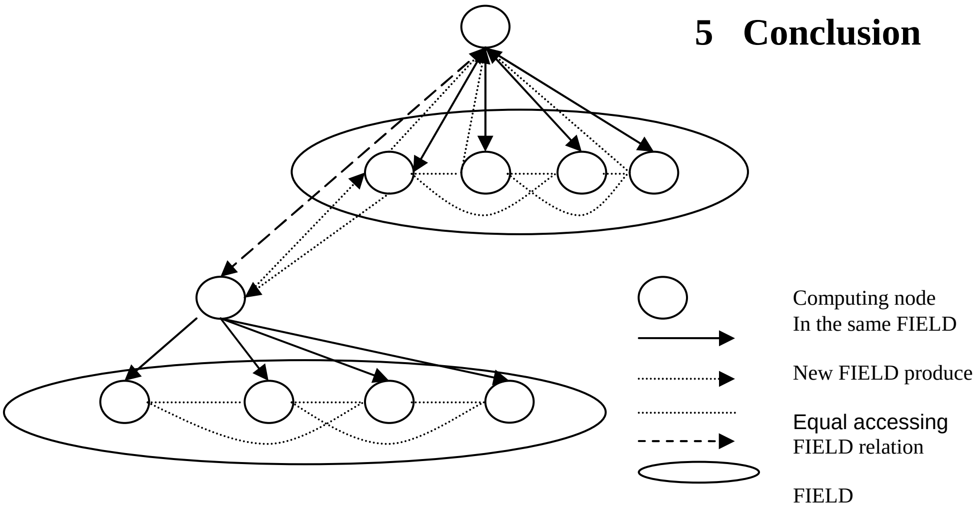

Fig.2 FILED

the time of searching new nodes. We named the last one h3, and h3 like h2 establishes a new FIELD. On the other hand, h2 puts h3 into its FIELD and sends h3's IP address to h1. At last, h1 puts h3's IP address into its FILED, too.

6. Node h3 repeats step 4 and 5. We can find m/n FIELD with repeating the above steps.

4 Analysis

This algorithm combines the advantage of C/S(Fig. 3) model and pure P2P model. Under the rational partition of subnet, accessing to the node submitting the task is finite. Hence, it can avoid the server to be a bottleneck in C/S model. For example, the Napster system always depends on a central server. As a result, our algorithm can resolve discovery of large-scale computing resources better than others.

At the same time the respond notes only connect with the requirement node, so we can avoid the disadvantage of P2P model (Fig. 4). What's more, the network bandwidth consume will not increase too much. We can see that our algorithm adopt the measure of controllable broadcast. It is available to avoid the information islands produced by document routing algorithm.

On the other hand, compared with other algorithm on GC, our algorithm also has many merits. It will consume less resource than other algorithms, so the speed of response is more quickly.

In conclusion, based on the factors concerned our algorithm has advantages to a certainty in discovery large-scale resources.

5 Conclusion

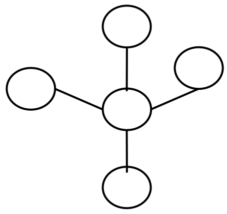

Fig. 3 C/S model

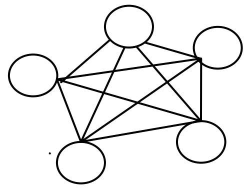

Fig. 4 P2P model

We can conclude that in this paper, FILED algorithm can effectively discover computing resources in grid environments. Compared with other algorithms, our algorithms combined the merit of the C/S model and the P2P model. The most important outcomes of this work are still to come. The work we want to do in the future is that we will do more tests and analysis in grid environments. Furthermore, we are planning to enrich our algorithm through developing and turning it into product.

References:

[1] Sarmenta L. . *Volunteer computing[Ph d dissertation]*. Department of Electrical Engineering and Science, MIT,Cambridge, Massachusetts, 2001

[2] Foster I , Kesselman C eds. *The Grid 2: Blueprint for a New Computing Infrastructure*. MorganKaufmann ,San Francisco , CA , 1998

[3] Foster I , Kesselman C , Nick J , Tuecke S. *Grid Services for Distributed Systems Integration* ,*Computer* ,2002 ,35 (6) :37～46

[4] Foster I , Kesselman C , Tuecke S. The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of Supercomputer Applications* , 2001 ,15 (3) :200～222

[5] LI Wei , XU Zhi-Wei , BU Guan-Ying , CHA Li , An Effective Resource Locating Algorithm in Grid Environments. *Chinese*

Journal of Computers , 2003 , 26 (11) : 1546~1549

[6] Dejan Milojicic S, et.al. Peer-to-Peer computing, *HP technical report[c]*.HPL-2002-57.

[7] Li Zhenwu, Yang Jian, Bai Yingcai , CHALLENGES IN STUDY OF PEER-TO-PEER NETWOKS , *Computers Applications and Software* , 2004 , 21 (2) : 54~55 , 110

[8] QU Xiaoming , ZHOU Xinran, Analysis of IP Address Lookup Algorithms, *Computer Engineering* , 2003 , 29 (14) : 143~145

9 .WANG Dan, YU Ge, Research of Peer-to-Peer System Model, Computing Engineering, 2005, 31(4): 128~130