

# 基于 P2P 框架的网格资源发现算法研究

郑秀颖<sup>1,2</sup>, 常桂然<sup>2</sup>, 田翠华<sup>1,3</sup>

(1. 东北大学 信息学院, 辽宁 沈阳 110004;

2. 沈阳工程学院 信息工程系, 辽宁 沈阳 110136;

3. 沈阳工业大学 信息科学与工程学院, 辽宁 沈阳 110023)

**摘 要:** 由于资源数量的巨大, 典型网格系统中的资源发现问题就成为关键, 而现有网格中有许多的资源发现算法主要是集中或分层方法。P2P 已经被认为是一种合适的分布式模型, 它具有动态性、可伸缩等优点。在现有网格之上提出了一个 P2P 虚拟层, 并据此提出了一个用于网格资源发现的分布式算法, 从而解决现有网格系统中资源发现效率低、性能差、不具有伸缩性和可扩展性等一系列问题。

**关键词:** P2P; 位图索引; 资源发现

**中图分类号:** TP393.01

**文献标识码:** A

**文章编号:** 1673-629X(2008)05-0123-03

## Resource Discovery Algorithm in Grid Based on P2P Frame

ZHENG Xiu-ying<sup>1,2</sup>, CHANG Gui-ran<sup>1</sup>, TIAN Cui-hua<sup>1,3</sup>

(1. College of Information Science and Engineering, Northeastern University, Shenyang 110004, China;;

2. Dept. of Information Science and Engineering, Shenyang Institute of Engineering, Shenyang 110136, China;

3. School of Information Science and Engineering, Shenyang University of Technology, Shenyang 110023, China)

**Abstract:** Because of large amounts of resources, it is very important for resource discovery in grid, but in the existing grid, many resource discovery algorithms are centralized or hierarchical. P2P system has been considered a suitable and distributed paradigm, and it is dynamic, scalable. In this paper, propose a P2P virtual level on the existing grid and based on which provided a distributed grid resource discovery algorithm, so it can solve a series of problems that existing grid resource discovery algorithm has low efficiency, poor performance, no scalability and no extensibility.

**Key words:** P2P; bitmap index; resource discovery

### 1 问题的提出

网格系统中为一个应用程序匹配合适的资源是基本的也是关键的要求。实际的几个网格实现中, 资源发现方法往往采用集中式的方法或是分层的方法<sup>[1]</sup>, 无论是哪一种方法, 每个 VO<sup>[2]</sup> 都有一个负责信息汇总的类似于网关的节点, 通过这些节点将不同的 VO 中的资源信息汇总到一起, 而信息访问的方向正好相反。通常来说, 分层方法具有较好的查询效率和伸缩性以及扩展性, 而集中式方法查询性能差, 不具有伸缩性和可扩展性。采用集中式还是分层式在网格创建之初随着网格体系结构的建立已经确定下来了。对这些

已经建立好的, 但资源发现算法效率差的网格, 可以考虑在网格之上再加一个虚拟层, 这一层虚拟层采用 P2P 网络进行组织。具体来说, 现有网格中 VO 的信息汇聚与查询大都通过专门的网格构件如 WMS(负载管理系统)来完成, 这里称其为 supernode, 在文中所提出的虚拟网络层中, 节点就是这些具有 WMS<sup>[3]</sup> 功能的 supernode。

### 2 解决方案

P2P 框架具有天然的伸缩性和扩展性, 而且在资源发现方面的研究较深入, 可将其研究成果用于网格的资源发现<sup>[4,5]</sup>。具体来说, 这样的 P2P 系统是一个由多个 supernode 构成的树型覆盖网络<sup>[6]</sup>, 网络中的每个 supernode 管理其孩子节点的资源信息, 并具有每个邻居 supernode 的压缩描述, 主要是资源属性值的位图索引<sup>[7]</sup>。在这种 P2P 系统中查询路由的主要根据就

收稿日期: 2007-08-16

基金项目: 辽宁省博士学科点专项科研基金(20030145017)

作者简介: 郑秀颖(1971-), 女, 辽宁盘锦人, 副教授, 博士研究生, 研究方向为分布式计算、网管; 常桂然, 博士生导师, 研究方向为分布式计算、计算机网络。

是这种位图索引。为适应网格资源的动态特性,还要体现出资源属性值的时时变化,同时还要解决索引更新的问题。

## 2.1 体系结构

相应的体系结构如图 1 所示。

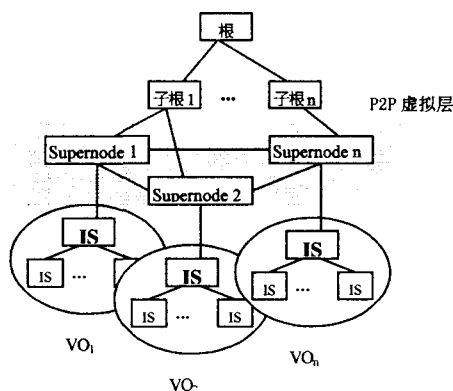


图 1 算法的体系结构

其中,IS 表示 VO 中的资源集合的信息服务,由图中可看到每个 VO 中有一个 supernode 对应于 P2P 网络中的一个节点,需要注意的是不同 VO 的 supernode 构成的是一个树型的覆盖网络,以实现传统网格的信息和资源发现的分层管理,从而提高查询效率。通常,按照节点组织的不同以及所采用的连接协议的不同,P2P 网络分为有结构和无结构网络,有结构网络中节点以一种严格的结构来组织,而无结构网络中每个节点的连接是随机的。研究表明有结构网络中,资源发现算法具有很好的伸缩性,但由于节点状态更新所引起的流量很大;无结构网络中资源发现算法主要以洪泛(flooding)为基础,因此伸缩性较差,在网络规模较小的时候具有良好的效率,且实现简单。这里的 P2P 网络采用的是无结构网络。

## 2.2 算法分析

设网格系统中的资源集合为  $\{R_1, R_2, \dots, R_M\}$ , 其中的不相交子集被分成不同的 VO, 由不同的 IS(supernode) 负责。资源  $R$  的属性  $A$  可表示为  $A[R]$ 。在实际的部署网格中,资源类型可分为:计算资源和存储资源。计算资源  $C$  的属性可能包括  $CpuSpeed[C]$ ,  $NumCpus[C]$ ,  $RamSize[C]$  等等。存储资源  $S$  的属性可能包括  $Capacity[S]$ ,  $FreeSpace[S]$  等等。这样,资源发现的依据就是满足某种属性一定的值或值范围,从而可将资源查找表示成资源属性值限定在某一个区域的表达式。例如:

$Q = \{R \in \{R_1, \dots, R_N\} \mid CpuSpeed[R] \geq 2.0GHz \text{ and } RamSize[R] \geq 512MB \text{ and } Utilization10[R] \leq 0.3\}$

此查询可查找满足如下条件的计算资源:CPU 速度至少在 2.0GHz 以上,RAM 至少为 512MB 且最后 10

分钟的利用率最多为 0.3。如此,可将查询看成是一些属性区间上的条件布尔组合,则这种查询策略能适用于任意类型上的属性区间查询。

为将这种属性区间查询转化成为现实中的资源发现算法,在由 supernode 构成的 P2P 网络采用位图索引<sup>[6]</sup>进行资源属性查询的路由依据。前面提到过,无结构 P2P 网络中的资源发现主要以洪泛算法为基础,位图索引是在洪泛基础上一个改进。位图索引的创建为:设  $A[R]$  值在区间  $[a, b]$  上,选择一个由  $k$  个元素构成的集  $a = a_0 < a_1 < \dots < a_{k-1} < a_k = b$ , 用一个  $k$  位二进制串对  $A[R]$  值编码,这样当且仅当  $A[R] \in [a_i, a_{i+1})$  时,第  $i$  位为 1。其基本思想是:针对每一个可能的值  $x$ ,其位图索引中每个 bit 为 1 代表这个位置的值为  $x$ ,为 0 则不为  $x$ 。值  $v$  的位图索引表示为  $BitIdx(v)$ 。

在 supernode 构成的树形覆盖网络之上,节点  $S$  的邻居集合设为  $Nb(S)$ ,即那些与  $S$  直接相连的节点集合。对每个  $S' \in Nb(S)$  来说,设  $R_S(S')$  为以  $S'$  为根的子树(不包含  $S$ )资源集合。用  $S' \rightarrow S$  表示从  $S'$  到  $S$  的覆盖链路,对于每个  $S' \in Nb(S)$  以及  $R_S(S')$  的属性  $A$  来说,等式(1)表示以  $S'$  为根的子树上所有资源的位图索引  $BitIdx(A)$  的总和(或)。它反映了节点  $S$  与链路  $S \rightarrow S'$  在属性  $A$  方面的相关程度:

$$LinkBitIdx(S \rightarrow S', A) \equiv \bigvee_{R \in R_S(S')} BitIdx(A[R]) \quad (1)$$

相关程度越大表明满足范围要求的属性  $A$  的查找方向,因此也就成为查询的路由依据。当向一个 supernode 提交查询请求时,详细的查询的路由策略如下所述:

每个 supernode  $S$  从其邻居节点收到一个查询  $Q$ ,首先  $S$  检查自身是否有合适的资源;若有,则查询满足;否则查询  $Q$  将被洪泛(广度优先搜索)。为避免洪泛过度,一个邻居子集在转发查询之前,先排除一些可确定不会含有任何有用资源的路径;每个  $S$  通过检查与每个邻居相关的位图索引来执行 DBFS。若位图索引  $LinkBitIdx(S \rightarrow S', A)$  满足  $Q$  的位图表示时,每个节点  $S$  仅向其邻居节点  $S' \in Nb(S)$  转发查询  $Q$ 。从而确保查询到所有满足该查询的资源。

思路如下:在域  $[a, b]$  中考虑属性  $A[R]$ ,将其划分成  $k$  个独立的分区  $[a_i, a_{i+1})$ ,  $i = 0, \dots, k-1$ 。若  $A[R] = v$ ,则属性位图索引  $B = (b_0, b_1, \dots, b_{k-1})$  定义为  $b_i = 1 \Leftrightarrow v \in [a_i, a_{i+1})$ 。考虑查询  $Q \equiv "v_1 \leq A[R] \leq v_2"$ ,建立的查询位图表示为  $B_Q = (b_{Q,0}, b_{Q,1}, \dots, b_{Q,k-1})$  对于每个  $i = 0, 1, \dots, k-1$  来说存在

如下关系:

$$b_{Q,i} = \begin{cases} 1 & \text{if } [a_i, a_{i+1}) \cap [v_1, v_2] \neq \emptyset \\ 0 & \text{其他} \end{cases} \quad (2)$$

若  $B \wedge B_Q = (0, 0, \dots, 0)$ , 则  $R$  与查询  $Q$  不匹配; 若  $B \wedge B_Q \neq (0, 0, \dots, 0)$ ,  $R$  可能满足  $Q$  查询, 但是要进行进一步的比较。即将  $R$  的属性值  $A[R] = v$  与区域  $[v_1, v_2]$  比较以确定  $R$  是否满足  $Q$  的需求。确切查询  $Q \equiv "A[R] = v"$  可以表示为  $Q \equiv "v \leq A[R] \leq v"$ , 其相应的位图表示  $B_Q$  计算如等式(2)。

需要注意的是资源属性值的即时性, 邻居的选择不能根据之前的查询结果。因此除了查询处理算法之外, 还需要体现属性值更新的位图索引更新算法。

### 2.3 算法描述

算法 1 中描述每个节点  $S$  路由和处理查询请求的过程。响应的路由方向与查询相反。下面采用类 C 语言进行算法描述。主要的算法包括查询处理算法和位图索引更新算法。

#### 算法 1 处理查询

前提:  $S$  是运行此程序的 supernode

process( $Q$ )

query  $Q$ ; // 查请求  $Q$  来自于 supernode  $S_m$

{  $S_{out} = Nb(S) - S_m$ ;

$i = 0$ ;

rpl = false;

while( $S_{out} \neq \emptyset$ )

{  $S' = \text{getfrom}(S_{out})$ ;

if(LinkBitIdx( $S \rightarrow S'$ ,  $A$ )满足  $B_Q$ )

(

send( $S', Q$ );

reply[i] = receive( $S'$ );

$S_{out} = S_{out} - S'$ ;

}

}

for( $j = 0; j < i; j++$ )

{ if(reply[j] !=  $\emptyset$ )

send( $S_m$ , reply[j]);

rpl = true;

}

if(! rpl) send( $S_m$ , "failed");

}

#### 算法 2 位图索引更新

设资源  $R$  的属性  $A$  的值从  $v$  变为  $v'$ 。资源  $R$  的所有者 supernode  $S$  运行算法 2, 用以产生更新消息。

前提:  $S$  是运行此程序的 supernode, 其中  $A[R]$  值从  $v$  变成  $v'$ 。

update( $S$ )

{

if(BitIdx( $v$ )  $\neq$  BitIdx( $v'$ ))

{  $S_{out} = \text{getfrom}(Nb(S))$ ;

while( $S_{out} \neq \emptyset$ )

{  $B = \text{BitIdx}(v')$ ;

$B = B \vee (\bigvee_{S' \in Nb(S) - S_m} \text{LinkBitIdx}(S \rightarrow S', A))$ ;

$Nb(S) = Nb(S) - S_{out}$ ;

send( $S_{out}$ ,  $B$ );

}

若新位图索引有变化,  $S$  发送更新消息到所有的邻居节点。这些更新作为专有性质来计算: 对于每个邻居  $S_{out}$ , 新发送给  $S_{out}$  的索引计算如下:

$\text{LinkBitIdx}(S_{out} \rightarrow S', A) =$

$\text{BitIdx}(A[R]) \vee (\bigvee_{S' \in Nb(S) - S_m} \text{LinkBitIdx}(S \rightarrow S', A))$

(3)

### 3 实验结果

实验主要判定查询的路由性能, 即考虑位图索引的位数对查询节点规模的影响。拓扑为一棵带有 500 资源的树; 设资源具有相同属性, 每个资源被映射到不同的 supernode。为简单性, 只考虑资源的单个属性, 并设其值在  $[0, 1]$  范围内。初始时, 所有资源被统一分配了随机值。在实验中, 将  $k$  位图索引对应到将  $[0, 1]$  分成  $k$  份独立的大小为  $1/k$  的分区。考虑选择 8 位到 64 位的位图索引, 而属性值分布在  $[0, 1]$  区间。为了测量查询性能, 对于统一选择的  $v_1, v_2 \in [0, 1]$ , 形如  $v_1 \leq A[R] \leq v_2$  的区域查询连续执行 100 次。从一个随机选择的节点开始, 结果显示查询传播规模随着位图索引位数的增加而减少。因为索引越大, 就越能够过滤掉许多不必要的查询。如图 2 所示。

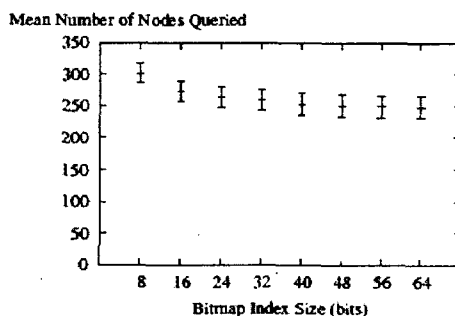


图 2 查询传播规模与位图索引位数的关系

### 4 结 语

文中提出了一个基于 P2P 虚拟网络资源发现分布式算法。主要解决传统网格集中式资源发现的性能差、不能伸缩和不能扩展等特性, 从实验结果来看, 所

(下转第 130 页)

- Learning to Control the Program Evolution Process[D]. Detroit, Michigan: Wayne State University, 1996.
- [10] Chung Chanjin, Reynolds R G. Culturing Evolution Strategies to Support the Exploration of Novel Environments by an Intelligent Robotic Agent[J]. Evolutionary Programming VII, 1998, 1447: 219 - 228.
- [11] Jacoban R, Reynolds R G, Brewster J. Cultural Swarms: Modeling the Impact of Culture on Social Interaction and Problem Solving[C]//Proceedings of IEEE Swarm Intelligence Symposium. Indianapolis, Indiana, USA: [s. n.], 2003: 205 - 211.
- [12] Reynolds R G, Michalewicz Z, Cavaretta M. Using Cultural Algorithms for Constraint Handling in Genocop[C]//McDonnell J R, Reynolds R G, Fogel D B. Proceedings of the 4th Annual Conference on Evolutionary Programming. Cambridge, Massachusetts: MIT Press, 1995: 289 - 305.
- [13] Chung Chanjin, Reynolds R G. A Testbed for Solving Optimization Problems Using Cultural Algorithms[C]//Fogel L J, Angeline P J, Back T. Proceedings of the 5th Annual Conference on Evolutionary Programming. Cambridge, Massachusetts: MIT Press, 1996: 225 - 236.
- [14] Chung Chanjin. Knowledge - Based Approaches to Self - Adaptation in Cultural Algorithms[D]. Detroit, Michigan: Wayne State University, 1997.
- [15] Jin Xidong, Reynolds R G. Using Knowledge - Based Evolutionary Computation to Solve Nonlinear Constraint Optimization Problems: a Cultural Algorithm Approach[C]//In 1999 Congress on Evolutionary Computation. Washington D. C.: [s. n.], 1999: 1672 - 1678.
- [16] Ostrowski D. Using Cultural Algorithms to Evolve Strategies in Agent - based Models[D]. Detroit, Michigan: Wayne State University, 2002.
- [17] Reynolds R G. Learning the Parameters for a Gradient - based Approach to Image Segmentation Using Cultural Algorithms[C]//Proceedings International Symposium on Intelligence in Neural and Biological Systems. Herndon, Virginia: [s. n.], 1995: 240 - 247.
- [18] Rychtyckij N. Using Cultural Algorithms to Re - engineer Semantic Networks[D]. Detroit, Michigan: Wayne State University, 2001.
- [19] Rychtyckij N, Reynolds R G. Using Cultural Algorithms to Improve Performance in Semantic Networks[C]//Proceedings of 1999 IEEE Congress on Evolutionary Computation. Washington D. C.: [s. n.], 1999: 1651 - 1656.
- [20] Jin Xidong. Solving Constrained Optimization Problems Using Cultural Algorithms and Regional Schemata [D]. Detroit, Michigan: Wayne State University, 2001.
- [21] Coello C A, Becerra R L. Evolutionary Multiobjective Optimization Using a Cultural Algorithm [C]//In: 2003 IEEE Swarm Intelligence Symposium. Indianapolis, Indiana: IEEE Service Center, 2003: 6 - 13.
- [22] Digalakis J G, Margaritis K G. A Multipopulation Cultural Algorithm for the Electrical Generator Scheduling Problem[J]. Mathematics and Computers in Simulation, 2002, 60: 293 - 301.
- [23] Ho N B, Tay J C. GENACE: An Efficient Cultural Algorithm for Solving the Flexible Job - Shop Problem[C]//Proceedings of Congress on Evolutionary Computation. [s. l.]: [s. n.], 2004: 1759 - 1766.
- [24] Abs A V, Cruz D A. Cultural Operators for a Quantum - Inspired Evolutionary Algorithm Applied to Numerical Optimization Problems[J]. Computer Science, 2005, 3562: 1 - 10.
- [25] Nguyen T T, Yao Xin. Hybridizing Cultural Algorithms and Local Search[J]. IDEAL, 2006, 4224: 586 - 594.

(上接第 125 页)

采用的资源查询路由可以避免过度的洪泛,基本符合最初的设想。

未来的工作包括一个更为详细的仿真研究,除了测量这种查询策略的精度之外,还需要考虑多种资源属性的情况。

#### 参考文献:

- [1] Zaniolas S, Sakellariou R. A taxonomy of grid monitoring systems[J]. Future Gener. Comput. Syst., 2005, 21(1): 163 - 188.
- [2] Foster I, Kesselman C, Tuecke S. The anatomy of the grid: Enabling scalable virtual organizations[J]. International J. Supercomputer Applications, 2001, 15(3): 1 - 10.
- [3] Avellino G. The datagrid workload management system: Challenges and results[J]. J. Grid Comput., 2004, 2(4): 353 - 367.
- [4] Talia D, Trunzio P. Toward a synergy between p2p and grids [J]. IEEE Internet Computing, 2003, 7(4): 94 - 96.
- [5] Yang B, Garcia - Molina H. Improving search in peer - to - peer networks[C]//In: Rodrigues L E T, Raynal M, Chen W S E, eds. Proc. of the 22nd Int'l Conf. on Distributed Computing Systems. Washington: IEEE Computer Society, 2002: 5 - 14.
- [6] Lü Q, Cao P, Cohen E, et al. Search and Replicating in Unstructured Peer - to - peer Networks[C]//Proc. of 16th Annual ACM Int. Conf. on Supercomputing (ISC'02). New York, USA: [s. n.], 2002.
- [7] O'Neil P. Model 204 architecture and performance[C]//In Proc. of the 2nd International Workshop on High Performance Transactions Systems, number 359, in Lecture Notes in Computer Science. Asilomar, CA: Springer - Verlag, 1987: 40 - 59.