

基于 Agent 的网格资源层次发现方法^①

A Resource Discovery Method of Grid Based on Agent

黄文明 崔亚楠 朱 英 (桂林电子科技大学 计算机与控制学院 广西 桂林 541004)

摘要: 在分层网格资源管理模型思想基础上,本文提出了一种基于 Agent 的网格资源发现方法,通过对网格资源进行虚拟组织的划分、建立虚拟组织信息表 and 用户最近访问多次的记忆表,进行有效的资源信息管理。在资源信息查找中引入 Agent 技术,通过 Agent 携带用户请求信息,进行快速资源发现查找,节省了用户比较选择的时间,有利于用户对资源进行选择,减轻网络负载,提高资源查找效率,最后通过模拟仿真证明该方法的有效性。

关键词: 网格 资源发现 虚拟组织 记忆表 Agent 技术

1 引言

网格应用就是指根据给定的条件去查找满足需要的资源供用户使用。由于资源的广域分布、现有 Internet 存在的带宽和延迟限制以及网络的不可靠性,资源查找将在很大程度上影响计算性能^[1]。因此,需要一种有效的资源发现方法来解决广域资源的快速定位问题。

本文首先分析网格资源发现机制中的最常见的三种资源查找算法,并对各个算法的复杂度进行了分析比较,找出资源查找中的问题所在;再对网格中资源虚拟组织划分和信息管理作了介绍,在此基础上提出了基于 Agent 的网格资源发现方法,并对其进行模拟仿真比较,证明其可行性。

2 网格资源发现

资源发现^[2]是根据资源请求者提供的资源请求描述,从网络上为请求者找到满足请求描述要求的合适资源(集)的过程。网格中的资源种类繁多,功能各异,网格应该为用户提供一种功能,能够根据用户的请求从网格资源中找到满足用户请求的资源。资源发现将网格中不被用户知道的资源和请求使用资源的用户联系起来。资源发现功能的强弱直接决定了网格的使用效率和友好程度。

2.1 现有资源查找技术研究

资源的查找问题可以看成是从网络中的某个节点开始,通过网络到达所需资源时经过的路径,资源查找算法的目标就是如何在较短的时间内找到资源所在的节点。现有 3 种主要查询方法:穷举查找法、集中查找法和路由转发法,下面针对这些算法简要介绍如下^[3]。

2.1.1 穷举查找法

在穷举查找法中,整个网格环境由若干个资源节点组成,每个节点都包含了其他所有节点的位置信息,所以从任何一点开始都可以到达目标节点。

它的时间复杂度是 $O(n)$, n 是网格中资源节点的数量,最坏情况下的空间复杂度为 $O(n^2)$ 。该算法时间和空间复杂度都比较大,而且每一个节点都必须包含整个网格环境的所有资源节点的位置信息,导致每个节点存储的位置信息随着网格规模的扩大而急剧增大。由于各节点可能随时加入或退出,每一个节点都必须在较短的周期内更新信息表,这样除了大量占用存储资源以外,还将导致网络因传递大量的更新数据的增加而造成严重负载甚至网络堵塞。

2.1.2 集中查找法

在集中查找方法中,整个环境中只有一个中心服务器信息节点 V 。时间复杂度最坏情况下应该 $O(WP)$

① 基金项目:广西教育厅项目(2004(20));广西研究生创新计划项目(2008105950812M428);广西研究生教育创新项目(2007105950812M18);桂林电子科技大学 06 年度学科软环境项目

收稿时间:2008-10-15

-S+n+Wpe) , n 为网格计算环境内的节点个数, Wps 为从起点至中心服务信息节点的时间, Wpe 为查询起点至目标节点的时间, 空间复杂度为 O(n)。

此方法具有严重的缺陷: 可靠性问题; 中心节点成为整个系统的瓶颈; 中心服务器信息节点附近的网络将会由于进行更新信息表而造成网络的严重负载。

2.1.3 RT 查找法

RT(Routing-Transfer)路由转发资源查找法, 在此算法中, 网格中的每个节点都是信息节点。每个信息节点中存放着所有资源的路由信息, 并不是该资源的直接位置信息, 即在查找某个资源时, 路由信息需要把搜索的方向指向相邻的某个节点。

最坏情况下的时间复杂度是 O(Wmax)。Wmax 是从起点开始查找的最大可能时间, 即跨越整个网格的“直径”时间。由于每个信息结点要保存所有资源和相邻节点的对应关系, 所以如果资源结点的数量等于网格结点数量时, 其空间复杂度与穷举法相同, 即此算法最坏情况下的空间复杂度是 O(n·|R|)。

以上三种算法的时间和空间复杂度比较如下:

表 1 算法比较

算法	时间复杂度	空间复杂度
穷举查找算法	O(n)	O(n ²)
集中查找算法	O(Wps+n+Wpe)	O(n)
RT 查找算法	O(Wmax)	O(n· R)

从时间复杂度看, 路由转发查找法最好, 从空间复杂度看集中查找法最好。但整体上看还没有一个让人十分满意的算法, 因此, 还有待进一步完善。这三种查找算法都是以“大一统”的思想来看待网格, 认为整个网格的每一部分都是密不可分的。然而实际的网格可能非常大, 这种“平均用力”的思想不便于管理更大的网格。

3 网格资源VO^[4]划分及信息组织

3.1 VO 划分

由于网格中资源类型繁多, 数量巨大, 变化频繁, 所以若采用一种有效的资源管理, 会降低资源发现效率。为了便于资源或资源信息的管理, 对资源或资源信息进行分割形成的若干个整体, 称为虚拟组织(Virtual Organization)。

虚拟组织划分原则是: 将网格系统中地理位置相邻或属于同一局域网的资源信息节点组织在一起, 按

照一定的协议或规则, 构成一个虚拟组织。当存在一个虚拟组织无法完成的任务时, 可以由各个虚拟组织之间相互合作, 相互提供资源, 共同完成一些在单个虚拟组织上无法完成的任务。为避免出现性能瓶颈、单点失效、可扩展性差等缺点, 应当把虚拟组织中的资源节点数控制在一定的范围之内, 即对虚拟组织中的资源节点数量设置上、下界, 使虚拟组织保持一定规模。

3.2 资源信息库信息组织

为了提高资源发现的效率, 需要对虚拟组织中的资源信息节点进行合理的组织。由于资源发现时是给定的资源属性, 而不是具体的资源, 因此根据资源属性对资源信息节点进行组织。为了在资源请求到来时, 可以大体确定所需资源所在位置, 从而提高资源发现效率, 需要为虚拟组织建立信息库, 用来存放虚拟组织拥有资源的类型、数量等一些信息。

(1) 为了便于虚拟组织中资源信息的查找, 用来存放虚拟组织信息库和邻接表以概述虚拟组织资源信息的节点, 称为该虚拟组织的超节点^[5](Super Node)。信息库中的超节点主要用来存放虚拟组织的信息库和邻接表, 信息库所具有字段如下图所示:

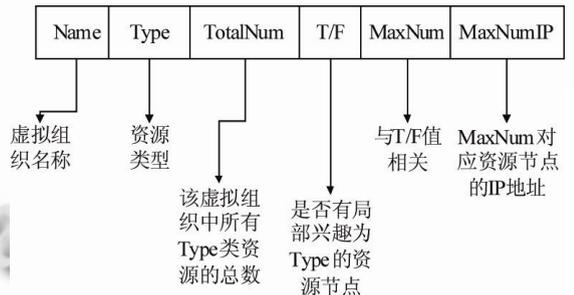


图 1 信息库字段

其中, MaxNum 的值是随 T/F 字段的值而变化的:

- ① 若 T/F 字段的值为 T, 则 MaxNum 的值是所有局部兴趣为 Type 类的资源节点中含 Type 类资源的最大数量。
- ② 若 T/F 字段的值为 F, 则 MaxNum 的值是所有资源节点中含 Type 类资源的最大数量。

虽然信息库不能完全反映虚拟组织中全部资源的情况, 但可以通过各个信息库确定各个虚拟组织存在所需资源的概率, 并选择概率最大的虚拟组织查找资源。

在为某类资源建立邻接表时, 首先按照提供该类资源的节点的局部兴趣进行分类。邻接表中的每个结

点需要存储的内容具体格式如图 2 所示:

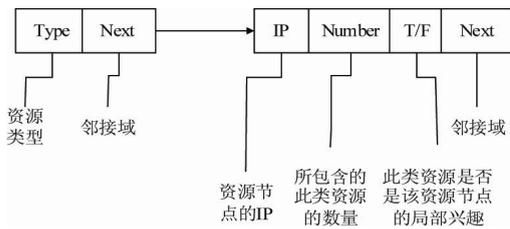


图 2 邻接表的格式

邻接表的组织是根据资源的属性建立,对每一类资源都建立一个链表,首先是按照资源的局部兴趣进行排列,再按所含该类资源的数量进行排列,形成有序的资源链表。这种方式有利于资源发现时很快能确定所需资源大体所在方向。

(2) 资源记忆信息表

记忆信息表主要存放的是最近该计算机节点访问过的资源信息。根据本地计算机访问资源的局部性,该节点可能在一段时间内重复多次有相同或相近的资源需求,以及同一个资源的在网格中的定位地址在短时间内一般不会改变,资源失效的机率较小^[6]。记忆表的主要属性如下:

表 2 记忆表属性

资源 ID;	资源类型	最近访问时间	访问次数
--------	------	--------	------

因为表结构较简单,表项数目较少,而且考虑到网格中资源请求节点多为个人 PC,性能有限,所以无须为本地计算机的记忆表建立数据库,可以将记忆表以数据结构中循环链表的形式存在于本地计算机的内存中,按最近访问时间排序,当记忆表存满时使用近期最少使用算法 LRU (Least Recently Used)将一段时间内使用最少的资源信息表项替换出去。

在本地计算机设置“记忆表”的好处是可以最大程度的减少访问次数,最大程度地降低网络开销。超级节点的“记忆表”存放最近通过该 SN 查询并成功获得资源访问权限的资源信息,SN“记忆表”主要属性如下:

表 3 SN 记忆表属性

资源 ID	资源类型	最近访问时间	访问次数	所在网络的路由器地址
-------	------	--------	------	------------

“记忆表”在数据库实现,可以是以数据结构的形式存放在内存中,也可以是以表的形式一般情况下按最近访问时间排序,当有新的资源信息需要加入

而“记忆表”本身已经存满时,也可采用近期最少使用算法进行替换。

4 基于Agent的网格资源层次发现算法

4.1 Agent

移动 Agent 是一种软件对象,它能携带执行代码、数据和运行状态,在复杂的网络系统中自治地、有目的地迁移,并能响应外部事件,在迁移过程中能保持其状态的一致性。简单地说,移动 Agent 就是一个能在异构网络中自主地从一台主机迁移到另一台主机,并可与其它 Agent 或资源交互的程序。

4.2 基于 Agent 的网格资源层次发现算法

(1) 资源发现算法描述

基于 Agent 层次资源发现算法描述:

输入:所需资源的信息; 输出:资源发现结果
算法步骤如下:

Step 1: 用户发送请求时,通过 JA(作业 Agent)提交所需资源的信息,同过资源调度器在本区域记忆表查找是否存在所需资源,如果有满足条件的资源,结合资源评估策略^[6]进行资源选择,确认访问后,更新本区域所在记忆表,并通过资源预留分配器和局部分配器对资源进行预约或者分配,直接访问最佳资源对应信息;如果不可访问则更新记忆表,删除该信息选项;如果没有满足要求的资源信息,则派发 JA 将所需资源信息发送到 SN 记忆表,并执行 Step2;

Step2: 通过 JA 将信息发给 SN 记忆表进行查询如果有满足要求的资源则结合资源评估策略进行资源选择,直接访问。如果没有执行 Step3;

Step3: 根据 JA 请求判断资源类型,确定需求资源的类型 T,根据 T 类型将各超节点中信息库对应 T 类型资源进行比较,根据需要进行排序比较,并存储,从而保证当资源在一个虚拟组织内不存在时,通过信息表和邻接表继续在其他虚拟组织中查找,或者对多个虚拟组织查找。

Step4: 根据排序的结果,用串行或者并行的方法,对一个或几个超节点中的邻接表进行遍历,对已遍历的资源节点作已遍历的标志;

Step5: 在本地资源信息库中查找到可能存在需求资源的资源节点的信息传输到资源 VO 信息管理器,对该资源节点的信息进行核查,若满足需求资源的要求,结合资源评估策略直接返回该资源的地址给

请求的用户。否则，返回未发现资源的信息至资源管理器，继续进行资源的发现，即重复 Step4。

(2) 资源发现过程

本算法资源发现过程如图所示：

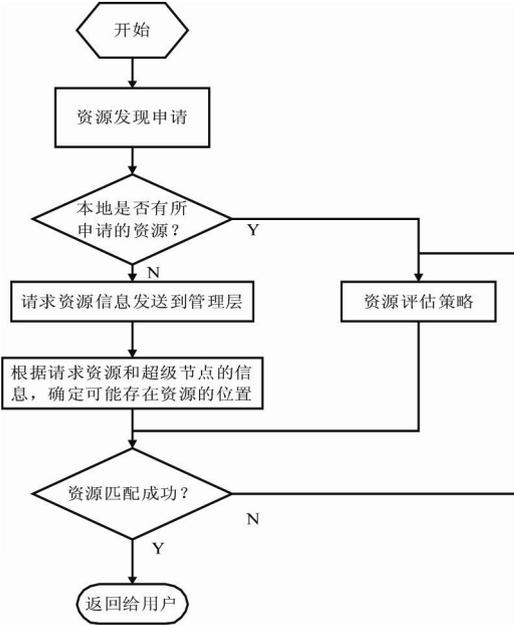


图 3 基于 Agent 的层次资源发现方法流程

移动 Agent 具有动态性、自主性，可以断开网络连接，在本地运行，减轻网络负载。引入移动 Agent，在虚拟组织之间进行动态网格资源发现。在 VO 的资源信息表没有合适的资源时，则转发移动 Agent 向总的资源信息管理库进行资源发现。其基本工作方式为：当资源请求 Agent 在作为用户资源请求任务的载体在本地 VO 间进行查找的过程中，接收到请求的 VO 总是将请求 Agent 转发给拥有资源能力最强、且优于自己的邻节点，若无法继续，则接受请求。否则继续进行转发。

4.3 层次发现算法复杂度分析

假设所需资源类型为 a，网格中有 n 个虚拟组织 VO_i(i=1, 2, 3,...n)，即有 n 个超节点，其中 m 个虚拟组织中有 a 类资源。假设第 i 个虚拟组织具有 KN_i 个资源节点，该虚拟组织中各个资源节点存在所需资源的概率相同，即为 1/KN_i。显然，时间复杂性由两部分组成：

表 4 复杂性比较

	最优的排序方法	串行查找方法
平均时间复杂性	$O(m \ln m)$	$O(\ln KN_i)$
最坏时间复杂性	$O(m \ln m)$	$O(KN_i)$

因此，整个资源发现算法的平均时间复杂性为 $O(m \ln m) + O(\ln KN_i)$ 。

最佳情况下，资源需求信息提交后，只遍历一个虚拟组织的邻接表；这种情况下所需的时间为选择合适的虚拟组织所需要的时间，因此最佳时间复杂度为 $O(O(m \ln m) + O(1)) = O(m \ln m)$

最坏情况下，需要将网格中的所有资源信息节点都查找一遍，才能确定所需资源是否存在。这种情况下所需时间为对信息库中的信息排序所需的时间、遍历邻接表的时间及在虚拟组织中确定资源所需时间之和，因此最坏时间复杂度为 $O(O(m \ln m) + m * O(KN_i))$

5 模拟仿真

本文只对穷举法、集中法和基于移动 Agent 的层次资源发现算法进行模拟。为简单起见，假设资源发现时网格中资源节点数量不变，三种方法均是在相同资源节点数量、查找资源属性相同的情况下进行比较，并且假设每次查找时只有一个资源请求。在注册资源数量分别为 50, 100, 200, 400, 600, 800, 1000 的情况下，在 GridSim 工具中的数据统计功能得到查询到所需资源的时间。所获得的实验数据如表 5：

表 5 查找时间比较

资源数量(个)	50	100	200	400	600	800	1000
穷举查找时间(ms)	30	39	45	62	75	109	131
集中查找时间(ms)	11	22	40	47	59	81	102
Agent 查找时间(ms)	6	30	12	17	28	33	37

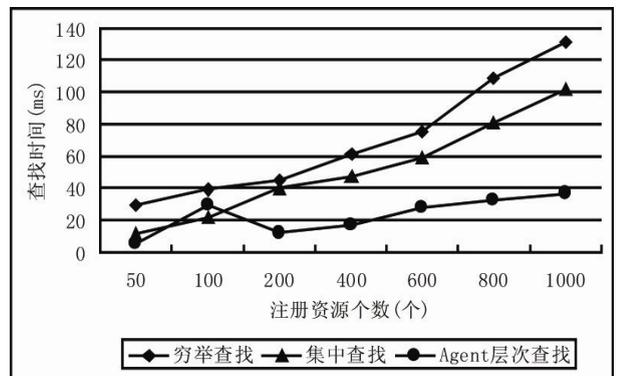


图 4 资源查找时间对比

将以上表格中的数据用坐标图表示进行形象直观的对比,如图4所示,其中横轴表示资源的注册个数,纵轴表示资源查找所使用的时间。由上述结果,分析如下:

(1) 在资源数量增加的情况下,三者的资源查找时间均会不同程度的增加,基于移动 Agent 的层次资源发现方法的增加幅度相对较小;

(2) 整体上,在资源个数相同的条件下,基于移动 Agent 的层次资源发现方法所使用时间较少,在所花时间上优于穷举查找法和集中查找法。但有时候由于查找资源属性的关系,基于 Agent 的层次资源发现方法时间会比穷举法和集中法时间长。

(3) 基于移动 Agent 的层次资源发现方法相对比较稳定所以,基于移动 Agent 的层次资源发现方法不管从时间效率来看还是从稳定性来看都是优于穷举法和集中法。

6 结束语

格资源的复杂异构性决定当前网格资源查找技术很难满足用户需求。本文对网格资源进行虚拟组织划分和信息管理,分析比较当前主要的三种网格资源查找方法的基础上,提出了基于 Agent 的网格资源发现方法,引入 Agent 技术对资源进行查找发现减少网络流量,能更好地适应网格资源动态变化的特点,使其能够对网格资源进行有效查找并提高其稳定性。

参考文献

- 1 李伟,徐志伟,卜冠英.网格环境下一种有效的资源查找方.计算机学报,2003,11(26):1546-1549.
- 2 徐志伟,冯百明,李伟.网格计算技术.北京:电子工业出版社,2004:118-119.
- 3 王国英,苏德富.基于 PZP 的资源搜索方法研究.微机发展,2004,14(11):84-86.
- 4 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International J. Supercomputer Applications, 2001,15(3).
- 5 Mastroianni C, Talia D, Verta O. A Super-peer model for resource discovery services in large-scale Grids. Future Generation Computer Systems, 2005(21):1235-1248.
- 6 赵维.Petri 网格及其资源搜索技术研究[硕士学位论文].南京:东南大学,2004:14-21.
- 7 宋凤龙,刘方爱.基于遗传算法的资源结点选择策略阴.微机发展,2005,15(10):62-67.
- 8 刘国清.网格环境下层次式资源发现方法研究[硕士学位论文].长沙:湖南大学,2007.
- 9 刘素芹,冯雪丽,邵红李.一种有利于资源选择的网格资源发现机制.中国通信学会第五届学术年会论文集,2008(1):64-68.