

一种改进的非结构化 P2P 网络资源搜索策略

唐冲^{1,2}, 石磊^{1,2}

(1. 山东师范大学 信息科学与工程学院, 山东 济南 250014;

2. 山东省分布式计算机软件新技术重点实验室, 山东 济南 250014)

摘要: 非结构化 P2P 网络具有资源搜索效率不高, 容易产生大量冗余信息等问题, 为此, 提出了一种改进的资源搜索策略。通过为网络中的节点建立朋友节点来改进传统的非结构化对等网络资源搜索, 并在此基础上设计了一种新的资源搜索算法。仿真试验证明, 该策略在一定程度上提高了非结构化 P2P 资源搜索的效率, 同时减少了网络中的冗余信息量。

关键词: 非结构化 P2P 网络; 资源搜索; 朋友节点; 冗余信息

中图分类号: TP393

文献标识码: A

文章编号: 1674-7720(2012)21-0056-04

An improved unstructured P2P network resources searching strategy

Tang Chong^{1,2}, Shi Lei^{1,2}

(1. School of Information Science & Engineering, Shandong Normal University, Ji'nan 250014, China;

2. Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Ji'nan 250014, China)

Abstract: The low efficiency of resources searching and information redundancy is the main problems of unstructured P2P network. To resolve this problem, an advanced resource searching strategy is given in this paper. Through establishing friend-nodes in the net to improve the traditional unstructured P2P network resources searching, and on this basis, a new resources searching algorithm is designed. The simulation shows that the strategy can improve the resources searching efficiency of unstructured P2P network, while reducing the information redundancy in the net.

Key words: unstructured P2P network; resources searching; friend-nodes; information redundancy

对等网络 P2P 网络 (Peer-to-Peer Network) 技术是 Internet 上实施分布式计算的新模式, 其致力于合理、高效地组织和利用 Internet 上大量分布的计算、存储以及信息等资源, 充分释放互联网蕴含的巨大的边缘资源, 以实现信息共享、即时通信、超级计算等目标。P2P 技术在当今互联网中有着广泛的应用, 美国财富杂志更是将 P2P 技术列为未来影响 IT 技术的四大关键技术之一^[1]。然而, 计算机网络是一个用户分布广泛、数量巨大, 节点行为不可控、计算能力和网络连接不均匀的复杂网络, 如何实现资源高效地搜索服务是 P2P 技术面临的一个难题。

本文针对现有非结构化 P2P 网络资源搜索效率不高, 容易产生冗余信息等问题, 提出了一种改进策略。通过增加节点的智能性来提高非结构化 P2P 网络资源的搜索效率, 并采取一定的措施来减少网络中冗余信息的

产生。具体实现方法如下: 在资源的搜索过程中, 逐渐建立节点的朋友节点列表, 相对于其他节点, 朋友节点具有资源丰富及搜索成功率比较高等优点; 资源搜索请求首先在朋友节点上进行, 当搜索失败时, 再采用最基本的搜索算法。

1 相关工作

目前, P2P 网络按照拓扑结构可以分为结构化 P2P 网络和非结构化 P2P 网络。前者利用分布式 Hash 表 (DHT), 将节点组织成严格的拓扑结构, 并将共享资源映射到特定的节点上, 在资源搜索时采用相应的定位方法, 使得资源搜索能够在确定的跳数内完成, 效率较高^[2]。然而, P2P 网络的高动态性使得结构化 P2P 网络的维护开销巨大; 只支持关键字的精确查询, 不支持内容、语义查询; 网络节点的物理结构被破坏, 实际延迟大。

非结构化 P2P 网络是以 Gnutella 为代表的一类网

络。这种网络的结构松散,共享资源的分布具有随机性和不确定性,其主要采用中心搜索和泛洪(Flooding)搜索算法。中心搜索是指在网络加入一个主要用于资源搜索的服务器,资源的搜索都在该服务器上完成,其容易产生服务瓶颈问题,一旦服务器失效,整个网络将崩溃;Flooding算法向所有邻居节点广播查询信息,搜索根据预先设置的生命值TTL(Time to Live)来终止广播查询请求,其容易产生大量冗余信息,且搜索效率低。在以后的算法改进中出现了Modified-BFS和Random Walk等算法。Modified-BFS算法虽然按照一定的概率随机地选择部分的节点发送消息,减少了网络中的路由消息,降低了网络的负载,但是资源搜索的成功率降低了;而Random Walk算法的最大劣势在其表现的不稳定性上,而且该算法的两个参数(漫游消息的数量 n 以及漫游消息的步长 s)的恰当取值也是个难点^[3-4]。

P2P网络的可用性依赖于网络中各个节点贡献资源,然而有参考文献[5]显示,在Gnutella中有70%的用户节点从网络中获取资源,但它们对网络的贡献率却几乎为零,这种称为“搭便车”的现象普遍存在于P2P网络中,阻碍了网络的高效运行。因此,在非结构化P2P网络的资源搜索过程中,访问一些不存在共享资源或者共享资源很少的节点本身就是对网络带宽和节点计算能力的一种浪费,完全可以通过一些策略将资源的搜索发生在那些搜索成功率比较高的节点上,这样就大大提高了非结构化P2P网络资源搜索的效率。

关于提高非结构化P2P网络资源搜索,参考文献[6]提出了一种基于学习的搜索算法,其采用分布式的被动学习方式,从历史搜索结果中学习节点之间的兴趣相似度,并指出保存历史记录可以对后面的节点发起的资源搜索产生借鉴意义。参考文献[7]提出将网络中的节点划分为超级节点和普通节点,资源搜索首先在超级节点上进行,搜索失败后再通过广播搜索信息在普通节点上进行。

2 改进的资源搜索策略

2.1 朋友节点

在本文中,朋友节点是指那些对以后的资源搜索具有重要意义。和其他节点相比,朋友节点可能具有以下特点:(1)丰富的资源,很大程度上能满足资源的搜索请求;(2)和搜索节点具有相同的兴趣^[6]。网络中的任意节点可以选择多个节点作为自己的朋友节点,并建立起自己的朋友节点列表。当一个节点刚加入网络时,其朋友节点列表长度为零,在以后的资源搜索过程中,朋友节点列表逐渐建立起来。朋友节点列表长度不能无限大,否则就失去了意义。朋友节点列表需要不断地更新和维护,以便增加新的更有意义的节点和淘汰那些旧的逐渐失去意义的节点。

2.2 朋友节点列表

本文将朋友节点列表设计成一个线性链表,该线性

链表除了有一个指示其入口地址的指针* L 外,还需要增加size和Maxsize两个变量。其中,size指示现有的朋友节点列表的长度;Maxsize指示该线性链表不能超过的最大长度。因此,网络中某一节点的朋友节点列表定义为:FriendNode_List(* L ,size,Maxsize)。

当网络中某一节点在某一时刻发起资源搜索请求并成功后,就将包含该目标资源的节点视为自己的朋友节点,并将其加入到自己的朋友节点列表中。很显然,在朋友节点列表中只需要记录该目标节点的地址即可,即IP_Address。另外,由于朋友节点列表需要不断地更新和维护,因此需要引入一个变量来衡量朋友节点的意义程度,很显然,在某一节点上获取资源的次数越多,该节点就越有意义。因此,引入一个变量num指示成功获取资源的次数;再增加一个变量Extra表示一些额外的附加信息,例如Extra可以记录最后一次成功获取资源的时间。因此,朋友节点定义为FriendNode(*next,IP_Address,num,Extra)。朋友节点列表的组织结构如图1所示。



图1 朋友节点列表结构图

图1中,为了取参数方便,将变量size的值存放在头节点的num值域中。下面给出了定义朋友节点和初始化线性链表(由于文章篇幅限制,只给出部分实现代码)。

//定义朋友节点的数据结构

```
#define Maxsize 50 //定义朋友节点列表的最大长度为50
typedef struct FriendNode{
    struct FriendNode *next;
    DataType IP_Address;
    int num;
    DataType Extra;
}FriendNode, *Lnode;
```

//初始化朋友节点列表,

//返回指向第一个节点(头结点)的指针

```
Lnode*FriendNode_List(){
```

```
L=(Lnode*)malloc(sizeof(Lnode));
```

//开辟一个存储朋友节点信息的空间

```
L->next=null;
```

//将next域置为空

```
L->num=0;
```

//朋友节点列表的初始长度为0

```
return L;}
```

2.3 朋友节点列表的维护

此时,网络中任意一个节点A维护自己的朋友节点列表FriendNode_List(* L ,size,Maxsize)的算法如下(文字描述):

(1)节点A在某一节点B成功获取了资源,则节点

网络与通信 Network and Communication

A 根据节点 B 的 IP 地址判断其是否已经存在自己的朋友节点列表上,如果是,转向(2),否则转向(3);

(2)修改节点 B 的 num 值,置 $num+1$,结束;

(3)判断 size 的值是否小于 Maxsize,如果是,则转向(4),否则转向(5);

(4)增加一个朋友节点域,将其中的 IP_Address 的值设为节点 B 的 IP 地址,num 的值设为 1,朋友节点列表长度 $size+1$,结束;

(5)找出表中 num 值最小的元素,将 B 的 IP 地址取代原来节点的 IP_Address 值,同时将 num 的值置为 1,结束。

参考操作系统中的缺页中断算法,当朋友节点列表长度已经达到最大后,也可以将那些最近最久没有获得过资源的朋友节点删除。

2.4 改进的资源搜索算法

非结构化 P2P 网络资源搜索实质上就是将包含目标资源的搜索信息发送到网络中其他节点中去,如果某一节点中存在目标资源,则两节点建立网络连接完成该资源的传输。现在,网络中的节点维护着一张朋友节点列表,相对于其他节点,朋友节点具有资源丰富、资源搜索成功率高的优点,因此本文中改进的资源搜索算法叙述如下:

(1)在某一时刻,节点 A 发起资源搜索请求,设置 TTL 的值为 2,将资源搜索信息按照自己的朋友节点列表发送到每一个朋友节点上。

(2)TTL 的值减 1,假设节点 B 是 A 的朋友节点,如果节点 B 满足该资源搜索请求,则两节点建立连接完成该资源的传输同时修改自己的朋友节点列表,结束;否则转向(3)。

(3)节点 B 将该资源搜索信息根据自己的朋友节点列表转发到所有的朋友节点中去,重复过程(2),资源搜索信息随着 TTL 的值为 0 时终止传输。

(4)如果上述资源搜索失败,则采用 Flooding 搜索算法。

也许上述改进的资源搜索算法不适用于网络中刚加入的节点,因为其还没有发起资源搜索请求,所以其朋友节点列表的长度为 0。这个问题可以很容易解决,例如当一个节点刚加入网络时,直接复制其邻居节点的朋友节点列表。

3 性能评估

本文的意义是通过建立朋友节点列表尽量将资源搜索发生在一些搜索成功率比较高的节点上,这样就尽量避免了 Flooding 算法的使用,提高资源搜索效率。同时,采用基于朋友节点列表的资源搜索时,将 TTL 的值设为 2 是为了一方面扩大有意义的节点的搜索范围,另一方面避免冗余信息的产生,如图 2 所示。

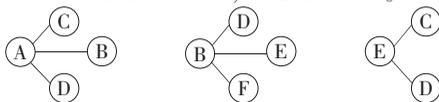


图 2 网络中节点 A、B、C 的朋友节点列表

图 2 中显示了节点 A、B、E 的朋友节点。可以看出,节点 A 和节点 B 有一个相同的朋友节点 D;节点 A 和节点 E 有一个共同的朋友节点 C;节点 B 和 E 有一个共同的朋友节点 D。当节点 A 发起资源搜索时,会把搜索信息发给节点 B、C、D。如果节点 B 上不存在节点 A 想要的资源,由于 TTL 的值设为 2,所以节点 B 将继续转发搜索信息到节点 D、E、F,可以看出节点 D 接受资源搜索信息两次,这就引起了无效的资源搜索信息转发。如果将 TTL 的值设为 3 或者更高,那么如果当节点 E 不满足节点 A 的资源请求时会继续将搜索信息发送给节点 C 和 D,很显然,由节点 E 转发的两次资源搜索信息都是无效的,这就引起了网络带宽资源的浪费,这也是 Flooding 算法的根本缺陷所在。笔者在以前参考的论文中,有的论文提出根据节点共享文档的相似性建立相同兴趣节点,将资源搜索信息首先转发到具有相同兴趣的节点上,如果该节点不满足资源搜索,则继续转发到自己的相同兴趣节点上,可以看出,这种说法是不准确的,在此给出纠正。

从上面的说明中可以看出,当网络中的节点逐渐建立起自己的朋友节点列表后,下次搜索资源时,通过将资源的搜索信息发送到朋友节点以及朋友的朋友节点(即两级搜索),就基本上可以做到搜索覆盖了网络中的大部分有意义的节点。也就是说,当一个节点成功建立起自己的朋友节点列表后,后来的资源搜索基本上就可以在一跳或者两跳内完成。

4 实验验证

本次模拟实验在一台 PC 上完成,网络的拓扑结构由 FLOD 算法产生,构造了一个具有 1 000 个节点和 10 000 份文档的 P2P 网络,文档在节点间采用 80:20 分布(即网络中 20%的节点拥有全网 80%的文档,剩下 80%的节点只拥有剩下的 20%的文档)。

本次试验通过设置朋友节点列表长度与全网所有节点数目的比例 P ,来观察资源搜索在一跳或两跳内的搜索成功率 S (即资源搜索通过转发朋友节点列表)。理论上,朋友节点列表长度最大取值为 \sqrt{n} (n 为网络的节点数目)。

搜索结果如图 3 所示。其中, P 的取值分别为 0.005、0.01、0.015、0.02、0.025、0.03、0.035、0.04、0.045,即朋友节点长度分别为 5、10、15、20、25、30、35、40、45 时的搜索成功率 S 。

从图 3 可以看出, S 的值随着 P 的值增大而明显增大,当朋友节点列表长度接近或略大于 40 时, S 的值接近于 1,这也就验证了预期的设想。

在网络信息流量方面,由于朋友节点列表的存在,资源搜索信息的转发是明确的,即资源搜索信息不会转发到一个先前不确定的节点上,这也就解决了 Flooding 算法产生大量冗余信息的根本问题所在。

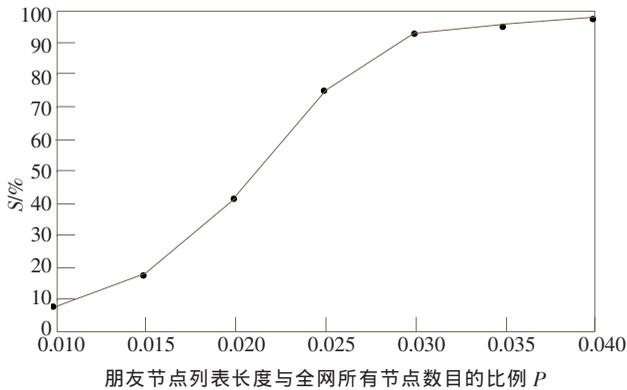


图3 资源搜索成功率

本文通过为网络中的节点建立朋友节点列表,将资源的搜索发生在资源搜索成功率比较高的节点上,这样既提高了资源的搜索效率,又避免了大量冗余信息的产生。试验证明该策略简单可行,当一个节点在发起多次资源搜索请求并成功建立起自己的朋友节点列表后,以后的资源搜索通过转发朋友节点列表就基本上能在一跳或两跳内完成,这样也验证了预期的设想。

参考文献

[1] GONG L. Peer-to-peer networks in action[J]. IEEE Internet

Computing, 2002,6(1):37-39.

- [2] 王丽莉,孙波,肖永康,等.结构化 P2P 资源搜索算法研究综述[J].计算机应用研究,2009,26(10):3621-3624.
- [3] 张文,赵子铭,杨天路,等.P2P 网络技术原理与 C++ 开发案例[M].北京:人民邮电出版社,2008.
- [4] 张伟,欧阳松.一种基于非结构化对等网络的改进搜索算法[J].计算机系统应用,2009,21(1):59-61.
- [5] ADAR E, HUBERMAN B A. Free-riding on Gnutella[J]. First Monday, 2000,5(10):1-22.
- [6] 陈海涛,龚正虎,黄遵国.一种基于学习的 P2P 搜索算法[J].计算机研究与发展,2005,42(9):1600-1604.
- [7] 曾晓云.基于 Chord 协议的混合 P2P 模型[J].计算机工程,2010,36(7):112-115.

(收稿日期:2012-05-02)

作者简介:

唐冲,男,1988年生,硕士研究生,主要研究方向:对等网络技术。

石磊,男,1970年生,硕士,研究员,硕士生导师,主要研究方向:网络信息安全、网络资源管理。