

比特洪流协议网络通信的主机分块优化算法

高峰¹, 陈金刚¹, 陈灯², 朱红³

(1. 武汉工程大学校办, 湖北 武汉 430205; 2. 华中科技大学计算机科学与技术学院, 湖北 武汉 430074; 3. 武汉工程大学法商学院, 湖北 武汉 430205)

摘要:从理论上和模拟环境上对比特洪流(BitTorrent)中的算法进行研究. 首先分析了 BitTorrent 网络中的 P2P 文件传输协议, 并对 BitTorrent 客户端和服务端等功能进行了介绍; 然后通过以相邻为 1, 不相邻为 0, 利用图的矩阵表示方法建立连接矩阵来模拟抽象数据模型对 BitTorrent 下载过程中的关键算法既最佳虚拟主机的查找算法进行了研究和分析; 进而利用连接矩阵法构建抽象数据模型提出了在大型网络中对网络进行分块的方法来提高 BitTorrent 的效率, 优化了网络通信; 最后, 搭建测试环境, 测试了矩阵分块算法, 测试中证明了在一定程度上分块的均匀性可以有效的提高节点文件的下载速度, 缩短下载时间, 提高 BitTorrent 的效率.

关键词:比特洪流协议; 虚拟主机; 网络分块

中图分类号:TP393

文献标识码:A

doi:10.3969/j.issn.1674-2869.2012.12.015

0 引言

端到端的应用(Peer to Peer, P2P)正在逐步占据互联网业务中重要的位置. BitTorrent 简称 BT, 是由美国软件工程师 BramCohen 用 Python 语言编写的源代码公开的 P2P 软件, 用于文件的分发^[1-4]. BT 由如下几个部分组成: .torrent 文件、种子提供站点、目录服务器和内容发布者/下载者.

BT 的基本工作原理是: 首先上传者把一个文件分成若干个部分(以下用 A、B、C 等英文字母表示), 甲在服务器随机下载了第 A 部分, 乙在服务器随机下载了第 B 部分, 这样甲的 BT 就会根据情况到乙的电脑上去取乙已经下载好的 B 分, 乙的 BT 就会根据情况到甲的电脑上去取甲已经下载好的 A 部分, 交换文件的每个计算机都可以分摊文件传输的负荷, 这样不但减轻了服务器的负荷, 而且加快了用户的下载速度, 提高了效率^[5-6].

1 虚拟主机的定义

现实中的网络错综复杂, 无论是几台计算机之间形成的小型的网络, 还是各种类型的网络之间组合而形成的大型网络, BT 系统总是尽可能的使种子文件的所有片段均匀的发布到网络中去^[7], 所以在进行 BT 下载时就要考虑到最佳虚拟

主机的查找^[5].

任何一个广域网都是由许多局域网构成的, 根据局域网原理, 处于同一网段下的电脑之间传输数据不仅速度快, 而且不会占用主干的带宽, 可以选择局域网中的一台计算机为节点作为逻辑上的唯一与外部网络连接的计算机, 就可以把这台计算机称为虚拟主机. 也可以就把整个局域网看作是由虚拟主机分支出来的网络, 其它的计算机只能通过虚拟主机上网^[8]. 例如要从外部网络下载一个文件, 可以先用虚拟主机下载, 然后别的计算机再从虚拟主机上下载, 这样下载的效率就会较高. 所以首先要找出一台最适合做虚拟主机的计算机.

同样, 假设某个比较小的局域网, 它们都只有一台主机与服务器相连, 需要找出那台主机. 例如某局域网有 7 台计算机, 它们的连接如图 1 所示.

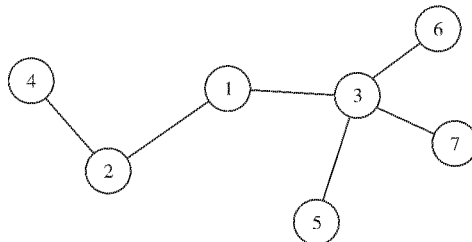


图 1 网络连接图

Fig. 1 Network connection diagram

收稿日期: 2012-11-19

基金项目: 湖北省教育厅人文社会科学研究项目(20089172)

作者简介: 高峰(1978-), 男, 湖北鄂州人, 硕士. 研究方向: 网络构建技术.

2 最佳虚拟主机查找

从图 1 可以看出,3 号计算机连接计算机数目最多,所以 3 号计算机是最佳虚拟主机.从连接矩阵上也可以清楚的识别出哪台计算机最适合做虚拟主机.图 1 对应的连接矩阵如下:

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

现在定义一个 2 维数组 $A[m][n]$,遍历矩阵的每一行(列),统计出每一行(列)的 1 的个数,个数最多的那一行(列)即为虚拟主机代表的行(列).

算法描述如下:

算法 1 FindOptimumVirtualMachine

输入:网络连接矩阵 $A[M][M]$

输出:最佳虚拟主机编号

步骤:

(1) 定义整形数组 $B[M]$ 存放各虚拟主机的连接数

(2) 定义 n 记录最佳虚拟主机编号

(3) 定义 \max 记录最大连接数

(4) For each element in B do

(5) element = 0;

(6) End for

(7) For each row in A do

(8) For each col in A do

(9) $B[\text{row}] = B[\text{row}] + A[\text{row}][\text{col}]$;

(10) End for

(11) End for

(12) $n = 0$;

(13) $\max = B[n]$;

(14) For each element in B do

(15) if element > \max then

(16) $\max = \text{element}$;

(17) $n = \text{index of element}$;

(18) End if

(19) End for

(20) Output n ;

把上述矩阵输入进去,输出结果为 3,则计算机 3 连接的计算机最多,所以计算机 3 最适合做虚拟主机.先用计算机 3 下载文件,然后再通过计

算机 3 把文件传到其它计算机上,这无疑也是最快最有效的一种方法.

2.1 连接矩阵分块的实现与比较

在大型网络中虚拟主机(对等节点)的查找和定位比较复杂,历时久,使整个网络变慢,并且随着网络规模的扩大,通过广播方式定位对等节点的方法将造成网络流量急剧增加,从而导致网络拥塞,把网络分解成若干小型网络是一种可行的办法^[9].约定连接计算机台数大于 3 的计算机为虚拟主机,和虚拟主机相连的计算机如果不是其它分块的虚拟主机那么就肯定和这台虚拟主机在同一分块内,如果是其它分块的虚拟主机就划入到其它分块,这样就能保证分块之间不会有重复.下面介绍算法:

(1)通过 2 个循环遍历连接矩阵 $A[M]$,找出那些 1 的个数大于 3 的行的下标放在一个数组 $[M]$ 上.

(2)只遍历这些以 $C[i]$ 为行标的行,如果在这一行中的某一个元素为 1,则遍历这一列元素,计算这一列的 1 的个数,存放在数组 $E[M]$ 中.

(3)判断数组 $E[M]$ 的每一个元素,如果小于等于 3,则相应元素能划入以上的分块.

网络简化图如图 2 所示.

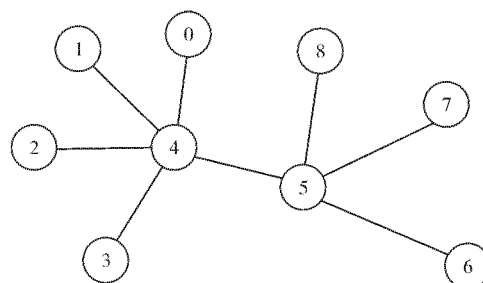


图 2 大型网络简化图

Fig. 2 Large network simplified diagram

图 2 相应的连接矩阵如下:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

算法描述如下:

算法 2 Partition

输入:网络连接矩阵 $A[M][M]$

输出:网络分块

步骤:

(1) 定义整形数组 $B[M]$ 存放各虚拟主机的连接数

(2) 定义整形数组 $E[M][M]$ 存放分块

(3) 定义整形变量 count 记录每列 1 的个数, 初始值为 0

(4) For each element in B do

(5) element = 0;

(6) End for

(7) For each row in A do

(8) For each col in A do

(9) $B[\text{row}] = B[\text{row}] + A[\text{row}][\text{col}]$;

(10) End for

(11) End for

(12) For each element in B do

(13) if element > 2 then

(14) 定义 $i = \text{index of element}$

(15) For each e in $A[i]$ do

(16) if $e = 1$ then

(17) 定义 $j = \text{column of } e$;

(18) For each row in A do

(19) Count += $A[\text{row}][j]$;

(20) End for

(21) if count < 3 then

(22) $E[i][j] = 1$;

(23) End if

(24) End if

(25) End for

(26) End if

(27) End for

(28) Output E

把上述矩阵输入, 输出结果如下:

4:0 1 2 3

5:6 7 8

结果表示矩阵分为两块, 第一块有 0, 1, 2, 3, 4, 其中 4 为虚拟主机; 第 2 块有 5, 6, 7, 8, 其中 5 为虚拟主机. 与预期的结果相符.

以上程序只能找到虚拟主机以及与虚拟主机相连的计算机, 即第一种子, 在有些情况下可能会导致分块不全, 或者有的块无法分入. 例子如图 3 所示.

把图 3 相应的连接矩阵输入得到的结果为:

4:0 1 2 3

5:6 7 8

结果与图 2 的结果一样, 但是图 3 中的 9 号计算机就会“落空”, 不属于任何分块, 把这样的计

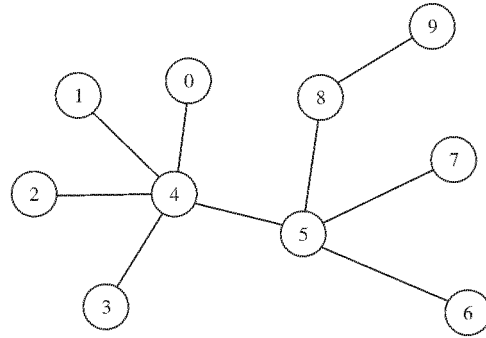


图 3 较复杂的大型网络连接图

Fig. 3 Complex large network simplified diagram

算机叫做第 2 种子. 找到第 2 种子的方法为遍历每个分块矩阵的第 1 种子, 把那些不属于其它分块的计算机划入这个分块以内, 就像图 3 中可以把 9 号计算机加入 5 号机为虚拟主机的分块中. 就可以找到第 2 种子, 然后依此循环则能找到第 3 种子, 第 4 种子……

这样一直将全部的计算机遍历完毕, 直到所有的计算机都有自己的分块.

整个网络全部被划分为小的局域网.

2.2 算法测试

测试算法:

算法 3 Test

输入: 网络连接矩阵 $A[M][M]$

输出: 最佳虚拟主机编号与邻接主机

步骤:

(1) 定义一个邻接链表, 存放各虚拟主机之间的邻接关系

(2) 将矩阵 $A[M][M]$ 转换成邻接链表表示 List[N], 其中 Link 中存放了虚拟主机与其它主机之间的邻接关系. List[N]. Link. Num 表示连接主机个数.

(3) 定义 max 记录最大连接数

(4) 定义 n 保存最佳虚拟主机编号

(5) For each element in list do

(6) If List[N]. Link. Num > max then

(7) Max = List[N]. Link. Num

(8) $n = N$

(9) End if

(10) Output List[N]

(11) For each element in List[N]. Link do

(12) Output element (13) End for

通过测试, 证明算法是可行的. 通过对 BT 网络的分块选择策略的分析, 虚拟主机在网络中分布的越均匀网络性能就越好, 文献[5]中的片段选择算法和文献[7]中的基于 Seed 的内容分发算法

也是基于同样的目的提出了改进的选择算法。

3 结 语

在对 BitTorrent 的虚拟主机算法进行了分析后,发现虚拟主机在网络节点中分布不均,影响了下载的效率。采用主机分块算法,该算法的选择策略可以让虚拟主机在网络中较为均匀分布。并从节点的物理位置分布和平均下载时间等几个方面进行了仿真测试,结果证明加入主机分块算法的系统降低了下载时间,提供了下载的整体效率。

参考文献:

- [1] Handurukande S B, Kermarrec A M, Fessant L F, et al. Peer sharing behavior in the eDonkey network, and implications for the design of server-less file sharing systems[J]. SIGOPS Oper Syst Rev, 2006, 40(4):359-371.
- [2] Bram Cohen. Incentives Build Robustness in BitTorrent[EB/OL]. <http://bitconjurer.org/BitTorrent/bittorrentecon.pdf>/2003-05-22.
- [3] Karzonov A. Determining the maximal flow in a network by the method of preflows[J]. Soviet Math Doklady, 1974(15):434-437.
- [4] 熊伟,谢冬青,刘洁. 一种结构化 P2P 协议中的负载均衡方法[J]. 微电子学与计算机, 2008(10):76-79.
- [5] 陆晨. 对等网络模型的研究及应用[D]. 合肥:合肥工业大学, 2003.
- [6] 杨少军, BT 型的 P2P 网络数据传播的数理机制与优化管理[D]. 广州:广东工业大学, 2012.
- [7] 刘宏亮, BitTorrent 核心算法研究与改进[D]. 北京:北京交通大学, 2008.
- [8] 杨祝林, BitTorrent 系统中文件传输算法与优化[D]. 长沙:湖南大学, 2008.
- [9] 孙鹏. 对等网络集群下载模式的研究及应用[D]. 郑州:郑州大学, 2004.

Optimization algorithm of host block of network communication using bit torrent

GAO Feng¹, CHENG Jin-gang¹, CHEN Deng², ZHU Hong³

(1. The WIT Administrative Office, Wuhan Institute of Technology, Wuhan 430205, China;

2. School of Computer Science and Technology, Huazhong University of Science & Technology, Wuhan 430074, China;

3. School of Law & Business, Wuhan Institute of Technology, Wuhan 430205, China)

Abstract: The algorithm of BitTorrent in theory and the simulation environment was studied. Firstly, the file transfer protocol of P2P in BitTorrent internet was analyzed and the function of the BitTorrent client and server was introduced. Secondly, the key algorithm of BT download process which was about how to find best webhosting service was discussed, which using the connection matrix imitating abstract data model through the method of one indicating adjacent and zero indicating nonadjacent. Thirdly, the large-scale network to network partition method was put forward, and the efficiency of BitTorrent was improved and the network communication was optimized. Finally, by setting up test environment and testing block matrix algorithm, it is proved that the block's uniformity effectively improves download speed of node file, shortens download time and improves the efficiency of BitTorrent.

Key words: BitTorrent; virtual host; host block

本文编辑:陈小平