

混合型P2P网络中的基于Gossip的动态自适应算法

张纯容, 王 忠, 周庆标, 施晓秋

(温州大学城市学院 浙江 温州 325035)

【摘要】为了提高搜索的性能,出现了混合P2P网络搜索方法。在混合P2P网络中,关键的问题在于确定资源的流行程度。针对该问题,该文提出了一种基于Gossip的动态自适应算法(DAHG),通过抛硬币操作估计节点加入和离开P2P网络时所带入和带走的文档副本数;并通过Gossips传递估计值,获得资源的流行程度。仿真结果表明,该算法具有良好的适应性,能够反映P2P网络中节点和资源的动态性,从而选择出正确的资源搜索方法,减少搜索响应时间和提高资源的命中率。

关键词 分布式哈希表; 基于Gossip的动态自适应算法; 泛洪; 混和P2P网络
中图分类号 TP393 **文献标识码** A

Gossip-Based Dynamical Adaptive Search Selection in Hybrid Peer-to-Peer Networks

ZHANG Chun-rong, WANG Zhong, ZHOU Qing-biao, and SHI Xiao-qi

(City College, Wenzhou University Wenzhou Zhejiang 325035)

Abstract In hybrid Peer-to-Peer (P2P) networks, the decision of whether to use flooding or DHT depends mainly on the popularity of desired data. Previous work either used only local information, or do not consider the dynamic factors of P2P systems. In this paper, an improved algorithm called dynamic adaptive hybrid based on Gossip (DAHG) is presented. In DAHG, a P2P ultrapeer tosses a coin when an end node joins or leaves the P2P networks, and uses a gossip-style algorithm to collect global statistics about document popularity. Therefore the dynamics of the resources is taken into consideration by DAHG, which it can be used to get the exact popularity of resources in a dynamic P2P network. Simulation shows that DAHG outperforms existing approaches and also scales well.

Key words distributed Hash tables; dynamic adaptive hybrid based on Gossip; flooding; hybrid peer-to-peer network

P2P网络是目前新一代网络业务研究与应用的热点之一,通常分为非结构化P2P网络和结构化P2P网络。典型的非结构化P2P网络有Gnutella^[1]等,它们以对等模式组织,广泛应用于文件共享系统。查询方法采用泛洪方法,这种查询机制对查询流行文档非常有效,但对于查询稀有文档却效率低下。结构化P2P网络模型作为搜索方法的另外一个可选项,采用基于分布式哈希表(DHTs)的查询机制^[2]。DHTs在没有网络失效的情况下,能保证每个查询结果都比较完美,并且能在经历一个较小的跳数后即可查询到匹配的结果;然而DHTs在发布其内容时需要耗费大量的带宽,为每个有多种属性的查询需要执行更多的复杂计算,一旦系统不稳定,维护的开销较

大。因此,尽管泛洪查询技术和DHTs查询技术都有其显著的查询优点,但对于有关P2P最好查询的设计仍然没有达到共识。

1 混和P2P查询方法

文献[3-4]提出的混合P2P搜索方法综合了非结构化网络和结构化网络模式查询的优点,它的查询原理为对任何查询首先使用有限深度的洪泛查询方式,如果没有返回结果,则将查询提交给DHT查询。这种混合的搜索方法对于流行的文档提供了廉价而快速的查询;同时对稀有的文档减少了泛洪的开销,但也对稀有的文档增加了响应时间,并且由于无用的泛洪而浪费了带宽。

收稿日期: 2008-04-02; 修回日期: 2008-07-10

基金项目: 国家自然科学基金(60573140)

作者简介: 张纯容(1973-),女,硕士,讲师,主要从事计算机网络方面的研究;王忠(1971-),男,博士,高级工程师,主要从事计算机网络方面的研究;周庆标(1974-),男,博士,副教授,主要从事计算机网络方面的研究;施晓秋(1964-),女,硕士,教授,主要从事计算机网络方面的研究。

文献[5]提出了GAB(Gossip adaptive hybrid)算法。在GAB算法中,网络中每个超级节点都维护了文档的全局副本估计值和泛洪的阈值。对任何一次查询,根据文档的全局副本估计值判断文档的流行程度,如果副本估计数量超过设定的泛洪的阈值,采用洪泛方式进行查询;如果副本估计数量小于设定的泛洪的阈值,采用DHT方式查询。该方法大大降低了搜索的响应时间,减少了带宽占用,提高了搜索效率。

GAB算法中关键的问题就是搜集文档的全局副本数量,它使用Gossip收集关于可用文档和关键字全局范围内的文档副本数量,其基本思想为:当某个超级节点发现1个新文档,在本地范围内不存在有关该文档的记录,则该超级节点就执行1个抛硬币的操作,并记录第1次出现正面的扔硬币次数,称为CT,即为这个超级节点对该文档的全局副本数量的估计值。然后,超级节点对其存在的所有文档的CT值通过Gossips传递给所有其他超级节点,并接收来自其他超级节点的所有文档的CT值。在Gossips期间,每个超级节点对每个文档计算其最大的副本估计值(maxCT),如果一个文档副本数量多,则其maxCT就会比稀有文档的maxCT大,拥有该文档的超级节点数^[6]近似等于 $2^{\max CT}$ 。

2 GAB算法存在的问题

GAB算法在对文档的全局估计副本做统计时,并没有考虑到P2P网络的动态性。GAB算法中每个超级节点只是对每个新加入的文档,进行抛硬币产生1个全局的副本估计值;并通过Gossips将估计值传给其他超级节点,比较CT值,获得1个全局最大值(maxCT)。因此GAB算法产生的文档全局估计副本值将随着节点的不断加入而增加,即单调递增。但当有节点离开P2P网络时,GAB算法对文档副本的估计值却不会减小。如果网络中的节点(包括超级节点和叶节点)状态保持不变,GAB算法能够对全局范围内的副本数量做出正确的估计。然而大多数P2P网络中的节点是动态变化的,新节点总是不断地加入,部分节点不断地离开P2P网络。文献[7]的研究表明,在Gnutella中,每小时内系统中有一半的节点被更新。因此,在动态变化的P2P网络中,GAB算法不能对全局的副本数量做出较准确的估计,从而降低了搜索的效率。

3 DAHG算法

本文提出的基于Gossip的动态自适应算法的关

键之处是节点加入和离开P2P网络时对文档副本数量的估计,其主要思路为:当节点加入P2P网络时,超级节点对节点带入的任何新文档进行抛硬币操作,估计带入的文档全局副本数量,并通过Gossips将估计值传给其他超级节点,比较CT值,获得1个全局最大值(maxCT);当有节点离开P2P网络时,超级节点对节点带走的超级节点上文档的最后副本也进行抛硬币操作,得到LCT(文档注销时抛硬币过程中正面第1次出现时扔硬币的次数),并通过Gossips将值传给其他超级节点,比较LCT值,获得1个全局最大值(maxLCT)。在可动态获得的全局范围内,现存拥有文档副本的超级节点数量近似为 $2^{\max CT} - 2^{\max LCT}$ 。当进行搜索时,如果文档副本数超过了阈值,则使用泛洪的搜索方法;否则使用DHT搜索方法。整个算法分为文档注册流程、文档注销流程、分布式信息传递流程和查询方法选择流程。

3.1 文档注册流程

在本文的算法中,每个超级节点使用格式($title_i, num_i$)记录本地范围内节点存在的文档信息, $title_i$ 表示为第*i*个文档, num_i 表示第*i*个文档即 $title_i$ 所对应的数量。超级节点根据文档的相关信息,使用($keyword_i, num_i$)记录本地文档信息的关键词及其副本数量, num_i 表示关键字 $keyword_i$ 所对应的数量。

当有新的叶子节点加入时,超级节点要求叶子节点注册可共享的文件信息,文档的注册流程如下:

(1) 如果节点注册的文档为超级节点上没有的新文档,则在超级节点中增加一条新的文档记录($title_i, 1$),并添加多个相应的关键词($keyword_i, 1$)记录。如果某个关键字 $keyword$ 已经存在,则将其所对应的数量 num_i 加1。

(2) 当节点注册的文档在超级节点上已经存在,则将相应的文档信息记录以及关键词信息记录对应条目中的数量 num_i 加1,此时不再执行下一步操作。

(3) 对注册的新文档 $title_i$ 和每个新关键词 $keyword_i$,超级节点上都需要各扔1次硬币来产生对应的CT,并将信息($title_i, CT_i$)和($keyword_i, CT_i$)记录到超级节点的数据库中。

在注册流程完成后,超级节点中所管理的范围内文档信息集为:

$$Titles = \{(title_i, num_i) | i \in \{1, N_{titles}\}\}$$

关键字集合为:

$$Keywords = \{(keyword_i, num_i) | i \in \{1, N_{keys}\}\}$$

式中 N_{titles} 为全局范围内文档资源的总数; N_{keys} 为

全局范围内关键词的总数。如果本地没有文档或者关键词, 则对应的 num_i 置为0。

对全局范围内拥有资源的超级节点数量的估计值为:

$$\text{TitleSynopsis} = \{(\text{title}_i, \text{CT}_i) \mid i \in \{1, N_{\text{titles}}\}\}$$

$$\text{KeySynopsis} = \{(\text{keyword}_i, \text{CT}_i) \mid i \in \{1, N_{\text{keys}}\}\}$$

同样的, 如果本地没有某个文档或关键词时, 在需要的时候可将对应的CT置为0。

3.2 文档注销流程

文档的注销有主动注销和被动注销两种方式。主动注销是节点离开P2P网络时, 主动将自己的文档信息从其所属的超级节点注销掉, 称为优雅注销流程。被动注销是节点离开网络时, 并不直接通知其所属超级节点。对这种离开行为, 超级节点需要采取探测方式来获知叶子节点的状态。文档主动注销算法的流程如下:

(1) 当叶子节点准备离开时, 要向超级节点申请共享文件的注销。

(2) 超级节点更新其数据库, 将对应条目的数量减1(包括 $(\text{title}_i, \text{num}_i)$ 和多个 $(\text{keyword}_i, \text{num}_i)$)。

(3) 如果 num_i 大于0, 表明超级节点中仍有该文档的副本, 直接返回; 否则扔硬币, 产生LCT, 并将信息 $(\text{title}_i, \text{LCT}_i)$ 和多个 $(\text{keyword}_i, \text{LCT}_i)$ 也记录到数据库中。

当节点突然离开P2P网络, 而没有向其所属超级节点进行注销, 此时需要超级节点主动探测叶子节点的状态, 即定期发送探测报文检测叶子节点的活动性。如果在规定的时间内节点没有响应, 则认为节点已经离开, 执行主动注销流程的步骤(2)和(3)。

任何发生了注销流程的节点上产生以下2个集合为:

$$\text{LTitleSynopsis} = \{(\text{title}_i, \text{LCT}_i) \mid i \in \{1, N_{\text{titles}}\}\}$$

$$\text{LKeySynopsis} = \{(\text{keyword}_i, \text{LCT}_i) \mid i \in \{1, N_{\text{keys}}\}\}$$

3.3 分布式信息传递流程

分布式信息传递的过程是利用Gossip方法^[8], 将注册流程和注销流程中获得的CT值、LCT值传递出去, 并接收来自于其他节点传递过来的估计值。对新收到的估计值, 如果某CT或者LCT值比其自身保存的数值小, 不予理睬; 否则, 超级节点需要更新本地的估计值, 并将最新的数值用Gossip传递给其他相邻的超级节点。

经过若干轮的Gossip后, 网络中的每个节点都将获得如下信息:

$$\text{TitleSynopsis} = \{(\text{title}_i, \max \text{CT}_i) \mid i \in \{1, N_{\text{titles}}\}\}$$

$$\text{KeySynopsis} = \{(\text{keyword}_i, \max \text{CT}_i) \mid i \in \{1, N_{\text{keys}}\}\}$$

$$\text{LTitleSynopsis} = \{(\text{title}_i, \max \text{LCT}_i) \mid i \in \{1, N_{\text{titles}}\}\}$$

$$\text{LKeySynopsis} = \{(\text{keyword}_i, \max \text{LCT}_i) \mid i \in \{1, N_{\text{keys}}\}\}$$

所以, 对于文档 title_i , 全局范围内拥有该资源的超级节点数量为 $2^{\max \text{CT}_i} - 2^{\max \text{LCT}_i}$; 对于关键词 keyword_i , 全局范围内拥有该资源的超级节点数量为 $2^{\max \text{CT}_i} - 2^{\max \text{LCT}_i}$ 。

3.4 搜索算法

当节点获得全局范围的 $\max \text{CT}$ 和 $\max \text{LCT}$ 以后, 可以为查询方式的选择做出准确的判断。定义如下符号:

$$r = \sum_{S_{\text{keyword}} \subseteq \text{title}_i, (\text{title}_i, \max \text{CT}_i) \in \text{TitleSynopsis}} (2^{\max \text{CT}_i} - 2^{\max \text{LCT}_i})$$

式中 r 为拥有包含所有关键词的文档的超级节点数量。

搜索方法的选择算法如下: 1) 用户输入搜索的关键字集 S_{keyword} 。2) 检查本地的文档信息, 如果完全匹配的数量大于等于期望的数量, 结束搜索, 并将结果反馈给用户; 否则执行下一步。3) 计算 r 。

4) 如果 r 不小于预设的阈值时, 进行泛洪: (1) 如果洪泛返回了期望的结果, 结束, 并把结果反馈给用户; (2) 如果洪泛超时, 对用户输入的每个Keyword进行DHT搜索。5) 如果 r 小于预设的阈值时, 则有:

(1) 如果 $S_{\text{keyword}} \cap \{(\text{keyword}_i, \text{CT}_i) \in \text{KeySynopsis}\} = \emptyset$, 则选择DHT; (2) 如果 $S_{\text{keyword}} \cap \{(\text{keyword}_i, \text{CT}_i) \in \text{KeySynopsis}\} \neq \emptyset$, 表示文档很少, 此时采用低优先级的洪泛进行搜索。

4 算法仿真

本文采用自行编写的仿真程序(large scale P2P simulator, LSPS)进行了仿真, 网络拓扑数据由BRITE^[9]产生, P2P网络中节点的加入和离开行为根据文献[10]中所观察到的P2P行为模型产生。网络中每个节点中资源的分布采用了Zipf分布。

资源副本数量欢迎程度的估计结果比较分别如图1、图2所示, 显示了在30 000个和60 000个节点的网络中算法的估计结果。在图中, 横坐标表示文件资源的编号, 纵坐标表示文件资源在整个P2P网络中的副本数量。从图中可以看出, 在2种网络中, DAHG的估计准确性明显高于GAB。在副本数量较少或者中等的情况下, GAB和DAHG都能够估计出副本的数量, 从而提高DHT和泛洪的准确性。对于副本数

量较多的资源，其动态变化的数量比较多，GAB的估计出现了明显的偏差；而DAHG的估计则准确得多。因此，在设置了恰当的副本欢迎程度的阈值后，DAHG能够提高估计值的准确性，从而提高洪泛的准确性。

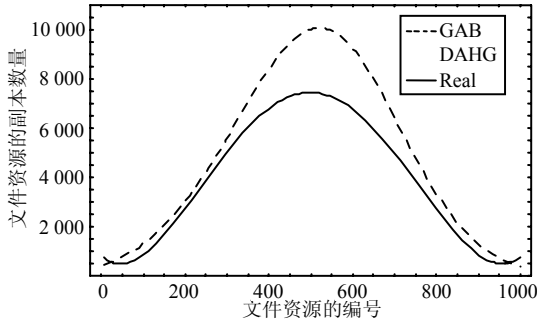


图1 30 000个节点的P2P网络拓扑中的模拟结果

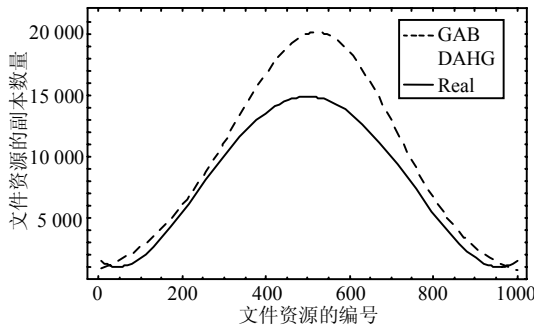


图2 60 000个节点的P2P网络拓扑中的模拟结果

GAB和DAHG算法估计结果对时间依赖性的分析比较分别如图3、图4所示。网络拓扑为有30 000个节点的P2P网络，图中给出了4个统计周期中GAB和DAHG对副本数量的估计。从图中可以看出，GAB随着时间的推移，其估计的准确性将越来越差；特别是对动态变化较多的资源来说，其估计的结果基本不能反映其真实的情况，即不能用于决定DHT和泛洪的选择。

本文提出的DAHG算法能够很好地适应节点的动态变化。随着时间的推移，DAHG算法依然能够准确地判断出节点的欢迎程度，因此能够正确地选择搜索算法。

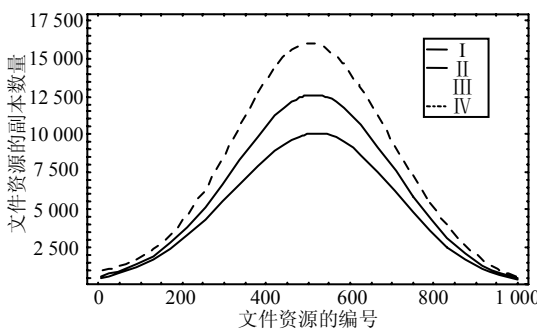


图3 GAB算法随时间变化情况的模拟结果

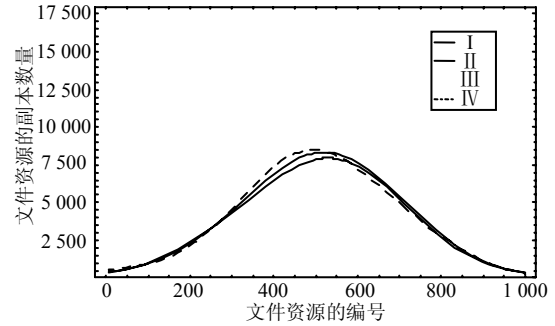


图4 DAHG算法随时间变化情况的模拟结果

5 结束语

在混合P2P网络中，关键的问题是确定资源的欢迎程度。针对该问题，本文提出的基于Gossip的动态自适应算法DAHG，能够反映P2P网络的动态性，即当有节点离开或者加入时，DAHG能够反应资源的动态变化，从而获得资源的欢迎程度。仿真结果表明，与GAB算法相比，本文提出的算法具有更好的适应性，从而能有效地提高搜索的性能。

参 考 文 献

- [1] The Gnutella Homepage[J/OL]. [2003-11-05]. <http://gnutella.wego.com>.
- [2] BALAKRISHNAM H, KAASHOEK M F, KARGER D, et al. Looking up data in P2P system[J]. Communications of the ACM, 2003, 46(2): 43-48.
- [3] LOO B T, HUEBSCH R, STOICA I, et al. The case for a hybrid P2P search infrastructure[C]//IPTPS. Berlin: Springer, 2005.
- [4] LOO B T, HELLERSTEIN J M, HUEBSCH R, et al. Enhancing P2P file-sharing with an internet-scale query processor[C]//VLDB. Toronto, Canada: [s.n.], 2004.
- [5] ZAHARIA M, KESHAV S. Gossip-based search selection in hybrid peer-to-peer networks[C]//IPTPS'06, Santa Barbara, USA: [s.n.], 2006.
- [6] FLAJOLET P, MARTIN G N. Probabilistic counting algorithms for database applications[J]. Computer and System Sciences, 1985, 31(2): 182-209.
- [7] SAROIU S, GUMMADI P K, GRIBBLE S D. A measurement study of peer-to-peer file sharing systems [C]//Multimedia Computing and Networking(MMCN'02). San Jose, USA: [s.n.], 2002.
- [8] KEMPE D, DOBRA A, GEHRKE J. Gossip-based computation of aggregation information[J]. Proc IEEE FOCS, 2003: 482-491.
- [9] MEDINA A, LAKHINA A, MATTA I, et al. An approach to universal topology generation[C]//The International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS'01). Cincinnati, USA: [s.n.], 2001.
- [10] CHU J, LABONTE K, LEVINE B N. Availability and locality measurements of peer-to-peer file systems[J/OL]. [2002-03-15]. <http://citeseer.ist.psu.edu/551697.html>.

编辑 黄 莘