文章编号:1001 - 9081(2006)12 - 2826 - 03

基于 DHT 资源定位服务覆盖网的研究

薛 颖,王 玲,冷 华
(湖南大学 电气与信息工程学院,湖南 长沙 410082)
(xyflame@163.com)

摘 要:提出了一种结构化 P2P 覆盖网——LAOverlay,采用组的思想,通过构建两层 Hash 结构,将本地的资源尽量映射到与本地节点在物理距离上相差较近的节点上,并通过数据本地化较好地解决了资源定位方法与实际的物理结构联系较小的问题,实现了稳定、可靠、查询广泛的资源定位。

关键词:覆盖网;组;分布式哈希表
中图分类号: TP393.07 **文献标识码**:A

Overlay network for DHT-based P2P resource locating service

XUE Ying, WANG Ling, LENG Hua
(*College of Electrical and Information Engineering, Hunan University, Changsha Hunan* 410082, *China*)

Abstract: A kind of the structured P2P overlay network named LAOverlay was proposed. The thought of the group was adopted and two-layer Hash architecture was constructed to make the resource of the underlying network node be mapped in a nearby node. The problem that resources location methods had little relationship with real physics structure was resolved, and steady, reliable and inquiring extensive resources location was achieved.

Key words: overlay network; group; Distributed Hash Tables (DHT)

0 引言

对于 P2P 系统而言,能够适应网络规模是一项非常重要的指标。然而早期设计的系统,如 Gnutella 和 Napster,在这个方面都有一定的缺陷,前者使用的是不适合大规模系统的泛洪管理,后者引入了集中式的目录管理。在这样的背景下,结构化覆盖网(Structured Overlay Networks)以及分布式哈希表(Distributed Hash Table,DHT)的提出引发了 P2P 研究的热潮,由此产生了 Tapestry、Pastry、Chord[2] 以及 CAN 等一系列的 DHT 系统。在这些系统中,文件根据系统生成的标志(ID)排序。DHT 系统的核心是路由协议,系统中的 DHT 节点构成一个覆盖网,每一个查询操作都需要通过这个覆盖网找到目标节点。所以,DHT 系统的性能取决于其采用的路由协议的效率。

传统的覆盖网路由协议是根据接收到的标识,把信息路由到相应的节点,每个节点也具有一个标识符(ID),而且这个标识符通常是和它对应的文件的标识相同。同时,每个节点都维护一张路由表,记录一些节点的信息,当一个节点受到一个查询操作时,如果发现所查询的标识不在自己关联的区间内,那么该节点将会把查询发送给其路由表中它认为最靠近目标的邻居。由此产生的问题是,逻辑空间中节点的关系并不能对应实际网络中的关系,即覆盖网中相邻的节点可能在底层物理网络中相隔很远。

针对这个问题,有不少研究人员进行了探讨:文献[3]中提到的基于网络拓扑的思想,利用了边界网关协议(Border Gateway Protocol,BGP)的路由信息,但这种方法对于建立在应用层上的流媒体的传输显然不适合;文献[4]中提到的 Brocade 利用了地标的方法;文献[5]中提到的 Expressway 方法采用了两层的结构,但是不能很好地解决逻辑跳数的问题。

本文提出的结构化 P2P 覆盖网——LAOverlay,将系统以组的方式组织起来,与 Chord 以及其他的一些基于 DHT 的系统相比,数据放置实现了本地化,并且性能各方面也得到了很好的改进。该覆盖网支持流媒体应用的底层 P2P 网络协议,能够为上层的流媒体服务提供更好的支持。

1 LAOverlay 覆盖网模型结构

LAOverlay 采用了组的思想,将参与覆盖网的节点以组的方式组织在一起,组内节点采用 DHT 的构造方式,同时分配给系统中的每个节点一个全局 HashID,将系统所有的节点统一编号组成一个逻辑全局 Hash 环。如图 1 所示,Group1 中有三个节点,其 GroupNodeID(本地 HashID)分别为 1,5,7,GlobalNodeID(全局 HashID)分别为 35,11,91。下面将详细介绍其模型结构以及一些基本的操作。

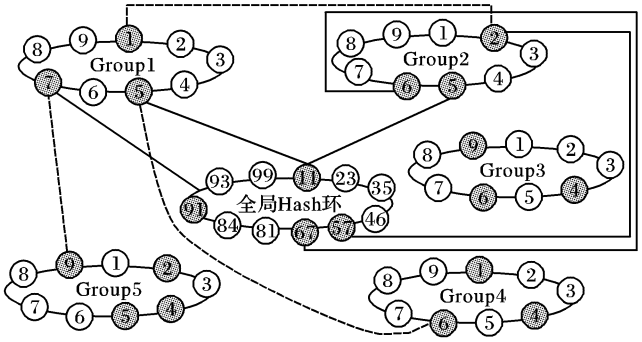


图 1 LAOverlay 的结构

由于本文中的 DHT 构造过程与一般的 DHT 构造过程相

收稿日期:2006 - 06 - 26;**修订日期**:2006 - 08 - 25

作者简介:薛颖(1982 -),女,湖南怀化人,硕士研究生,主要研究方向:计算机网络体系结构、Overlay 网络; 王玲(1963 -),女,湖南长沙人,教授,博士,主要研究方向:现代通信理论及应用、计算机网络通信; 冷华(1982 -),男,湖南长沙人,硕士研究生,主要研究方向:电力网络通讯模式.

同,在这里采用最为简单的 Chord(相关资料参见文献[2]),下面主要介绍 LAOverlay 中的一些基本概念以及基本操作。

1.1 基本概念

1.1.1 组

组由一组邻近的节点所构成。在底层物理网络拓扑中,如果节点 P、A 的距离与节点 P、B 的距离的差值在一个 K 值以内,那么说节点 A、B 是位于同一组中。这里的距离含义很广泛,可以是网络延时,链路的最小带宽等,也可以是实际的地理距离。文献[6]中提到的 Group 的定义与本文相近。每一个组都有唯一的标志,记为 GroupID。

1.1.2 组成员

在每个组中有两类成员:Leader Peers(LP)和 Regular Peers(RP)。任何一个节点只属于一个组。LP 由一些网络速度和连通能力等综合能力最佳的节点组成。每个组中有两个 LP,分别为 ILP、OLP。其中 ILP 主要负责管理组内成员的信息表,即 Local_ Table。当一个节点加入该组时,ILP 会收集其信息,如网络带宽、CPU 速度、内存、硬盘大小等。OLP 主要负责将没有得到解决的查询请求传递到其他组,相当于一个组与外界进行交互的接口,主要负责 Dir_File_Table。而 RP 的主要功能是维护依据 DHT 方式在本地存放的资源信息以及系统资源的备份信息。

1.1.3 组文件备份范围

每个组负责备份一个连续范围内的文件(Fid)信息。例如图 1 中 Group1 备份系统中文件 GlobalFileHashID 范围是(0,30)的文件信息。其备份操作见 1.2 基本操作。

1.1.4 组中信息表的定义

(1)Local_Table 表

ID	CPU	Memory	BandWidth

该表主要由每个组中网络速度和连通能力等综合能力较佳的节点 ILP 负责管理,记录组内成员的信息,用于组中 LP(包括 ILP,OLP)的选举。

(2)Dir_File_Table 表

GroupID	OLPID	Weight

该表主要由每个组中网络速度和连通能力等综合能力最佳的节点 OLP,即在 Local_Table 中排第一的节点负责管理,记录每个组的 OLP 的 ID,用于组备份操作、组外查询服务以及组合并。

(3)Group_Finger_Table 表[2]

I	Start	Successor(Start)	Successor(Start) _IP

组内所有的节点都拥有该表,主要用于组内资源定位,其中 I 表示索引值,在节点 K 上第 I 个表项的后两个域分别为:

$Start = K + 2^{I} (\mathrm{mod}\ 2^{m})$,其中 2^m 表示组内节点总数

$Successor(Start) = sucessor(Start[I])$

(4)Group_File_Table 表

GroupFileID	ID

组内所有的节点都拥有该表,主要用于记录关联信息。例如节点 GroupNodeID = 199 节点将会记录拥有文件 successor(GroupFileID) = 199 的节点的信息,例如其物理 ID 号。

1.1.5 组容量

组最大的容量节点数为 MC。MC 可自定义,当超过定义的 MC 值时,自适应完成组分裂操作;当小于定义的 MC 值时,自适应完成组合并操作(详见 1. 2 基本操作)。

1.2 基本操作

1.2.1 节点加入

当一个节点请求加入时采用泛洪的方式将信息发布出去,但并不是所有接收到该信息的节点都返回确认信息,只有 OLP 返回确认信息。在 TTL 时间内,节点将选择反应时间最快的 OLP 所在的组,并加入进去,同时执行备份操作。

1.2.2 节点退出

当节点退出时,会通知 ILP 以及 OLP。组内修改参见 Chord 方法中的节点退出,组外需修改负责备份该节点拥有的文件信息。

1.2.3 备份操作

当新节点加入时组 $GroupID_1$,拥有一文件,其本地 HashID 为 $GroupFileID_1$,系统会根据其文件的 $GlobalFileID_1$ 计算出负责备份文件的组号 $GroupID_2$,并要求组 $GroupID_2$ 中 $GroupNodeID_2 = successor(GroupFileID_1)$ 的节点保存该文件信息。当 $GroupID_3$ 中的一节点要求系统查询文件 $GroupFileID_1$ 时,若组内没有满足条件的节点,系统就会根据计算得到负责备份文件的组号 $GroupID_2$,查询 $GroupID_2$ 中 $GroupNodeID_2 = successor(GroupFileID_1)$ 的节点得到文件的信息。如图 1 所示,Group1 中节点 7 将备份文件 9 的信息。

1.2.4 查询服务

查询服务分为两类,一类为组内查询,一类为组外查询。若组 $GroupID_1 = A$ 中节点 $GroupNodeID_1 = Aa$ 要查询文件($GroupFileID_1 = Ak$, $GlobalFileID_1 = Akk$)时,首先,进行组内查询,即节点 Aa 将在组 A 中利用 Chord 算法,查找 $GroupNodeID_n = successor(GroupFileID_1) = successor(Ak)$ 的节点,该节点拥有文件 Ak 的信息。若组内没有满足条件的节点,则进行组外查询,首先节点 Aa 计算负责备份 $GlobalFileID_1$ 的组号 $GroupID_m$,然后请求 $GroupID_m$ 查询 $GroupNodeID_m = successor(GroupFileID_1) = successor(Ak)$ 的节点,并通知该节点返回节点信息给节点 Aa。若经过以上两步都没有得到查询信息,则返回空信息。下面是查询服务的伪代码:

```
Searching Group service (GroupNodeID, GroupFileID)  //组内查询
  { GroupNodeID_n = Successor_Chord(GroupFileID_1)
    Communicate (GroupNodeID, GroupNodeID_n)
      { If (Search_Group_File_Table (GroupFileID))
          TargetID = Return_Group_File_Table(GroupFileID)
        Else
          GroupID_m = Calculate_Group(GlobalFileID)
          OLPID = Search_Dir_File_Table(GroupID_m)
          TargetID = OLPID
      }
    Communicate (GroupNodeID, TargetID)
  }
Searching Global service(GroupNodeID, GroupFileID)   //组外查询
  { GroupNodeID_n = Successor_Chord(GroupFileID_1)
    Communicate (OLPID, GroupNodeID_n)
    TargetID = Return_Group_File_Table(GroupFileID)
    Communicate (GroupNodeID, TargetID)
  }
```

1.2.5 组合并

当组内的成员较少时,采取组合并的操作。即查询本组中的 Dir_File_Table,选择权值最低的组加入进去,并将本组

中所有负责管理服务的节点的表通知欲加入的组中相应的节点,并修改相应的表的内容。

1.2.6 组分裂

当组内的成员超过 MC(组最大的容量节点数)时,采取组分裂的操作。选择组内网络速度和连通能力等综合能力次佳的节点组成另一个 ILP 及 OLP,并进行表的初始化。

1.3 实例

假设如图 1 所示,系统由 5 个组构成,图 2 显示了 Group1 以及 Group5 部分表的结构。在这个例子中显示了当 $m = 5$ 时的节点标识符环:标识符为 1,4,7,12,15,20,27 的那些节点对应于 Group1 中实际的网络节点;标识符为 2,6,10,19,23 的那些节点对应于 Group5 中实际的网络节点;其余的节点并不对应于实际的网络节点。图 2(a) ~ 图 2(c)中括号前记录的是 Group1 的相应信息,括号内记录的是 Group5 中的相应信息。

GroupID	OLPIP	Weight
1(1)	XXX.XXX.XXX.XXX	1(2)
2(2)	XXX.XXX.XXX.XXX	2(4)
3(3)	XXX.XXX.XXX.XXX	4(5)
4(4)	XXX.XXX.XXX.XXX	5(3)
5(5)	XXX.XXX.XXX.XXX	3(1)

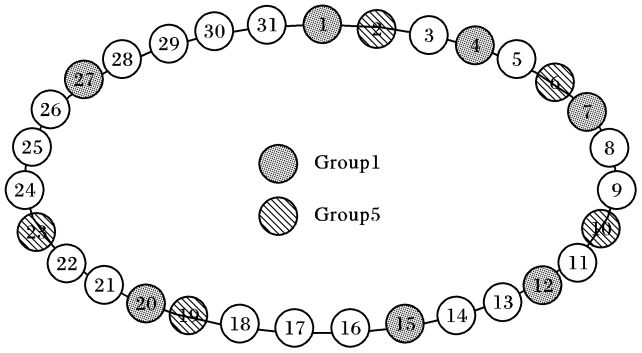
(a) Dir_File_Table(Group1,Group5)

I	Start	Successor(Start)	Successor(Start)_IP
0(0)	2(2)	4(6)	XXX.XXX.XXX.XXX
1(1)	3(3)	4(6)	XXX.XXX.XXX.XXX
2(2)	5(5)	7(6)	XXX.XXX.XXX.XXX
3(3)	9(9)	12(10)	XXX.XXX.XXX.XXX
4(4)	17(17)	20(19)	XXX.XXX.XXX.XXX

(b) Group_Finger_Table(节点1, 节点10)

GroupFileID	IP
13(14)	XXX.XXX.XXX.XXX
14(17)	XXX.XXX.XXX.XXX

(c) Group_File_Table(节点15, 节点19)



(d) 全局Hash环

图 2 LAOverlay 中表的结构

组内查询:假设 Group1 中的节点 1 想查找文件 GroupFileID = 14,首先它将在 Group_Finger_Table 表中查找 Start 域最接近 14 但在节点 14 之前的节点,得到节点 9 以及 9 所在的 Successor(Start)域 12 的 IP 地址,节点 12 看到 14 落在它和它的后继节点(15)之间,于是返回 15 的 IP 地址。然后节点 1 将与节点 15 进行通信,要求节点 15 查找 Group_File_Table 表,如果表中 GroupFileID 有文件 14 的信息,则将拥有文件 14 的目标节点 IP 返回给节点 1,最后节点 1 与目标节点通信,进行文件的传输。这种情况下,组内查询与一般的 Chord 查询方法一致,但是如果 Group_File_Table 表中没有文件 14 的信息,这就需要进行组外查询了。

组外查询:假设 Group$n$(n = 1,2,3,4,5)备份管理的文件 GlobalFileID 范围为(100n,100(n + 1)),文件 14 的 GlobalFileID = 445。首先计算得到负责文件 14 的备份组为 Group4,然后节点 1 要求本组的 OLP(节点 10)查找 Dir_File_Table 表,得到 Group4 的 OLPIP 地址,接下来请求 Group4 的 OLP 按照 Chord 的查询方法开始查找 GroupNodeID = successor(GroupFileID)的节点,如组内查询一样,得到 GroupNodeID = 19,于是通知节点 19 查询 Group_File_Table 表中是否拥有文件 14 的信息,若有则将目标节点的 IP 通过两个组的 OLP 返回给节点 1,最后节点 1 与目标节点通信,进行文件的传输。

若经过组内、组外查询都没有满足条件的节点返回,则查询返回空值。

2 LAOverlay 性能分析

现有 P2P 覆盖网环境模型和数据查询定位算法的评比方式主要包括:数据的存储代价、数据的搜索效率、数据定位复杂度和系统可扩展性等多个方面。

2.1 数据的存储代价

系统中的存储代价主要包括两个方面:路由表和信息备份数。假设 Chord 和 LAOverlay 的系统有同样数目的节点,用 $P_{system} = 2^N$ 表示系统的节点数目,用 $P_{number} = 2^{N-M}$ 表示在 LAOverlay 系统中组的数目。假设在所有的情况下每个组的节点数相同都为 $G_{number} = 2^M$,则在 Chord 系统中每个节点需要维护的路由表的大小为 N;而在 LAOverlay 系统中每个实体节点需要维护的路由表的大小为 M,很显然 $N > M$。

由于本系统采用了组备份的方法,因此每个节点都需要维护 1 张 Group_File_Table 表,虽然增加了系统的存储代价,但是换来的是更稳定、可靠、查询更广泛的资源定位。

2.2 搜索效率

衡量覆盖网的一个关键参数就是如何能够定位到最近的节点。我们都知道 Chord 覆盖网定位的一个最大问题就是:逻辑空间中节点的关系并不能对应实际网络中的关系,即覆盖网中相邻的节点可能在底层物理网络中相隔很远。但是在 LAOverlay 中,通过分组的形式,将数据放置在本地的节点上,很好地解决了这个问题。

2.3 数据定位复杂度

Chord 中由于采用了指取表,所以每次定位都使得到达目标节点的距离减少了一半,因此其复杂度为 $O(\log_2 2^N) = O(N)$。LAOverlay 系统中由于组内采用的也是 Chord 方法,因此其复杂度为 $O(\log_2 2^M) = O(M)$。由于组外查找中得到备份组号后采用的也是 Chord 方法,因此其复杂度也为 $O(\log_2 2^M) = O(M)$。因此 LAOverlay 系统的复杂度在 $[O(M), 2O(M)]$ 之间。

2.4 系统可扩展性

当 Chord 系统内节点增加时,导致其时间和空间复杂度随节点数目增长呈级数增长,当节点数目由 2^{10} 增长到 2^{30} 时,其每个节点的数据存储量要增长 3 倍,然而 LAOverlay 每个节点的数据存储量基本保持不变,具有较好的可扩展性。

3 结语

提出了基于 DHT 资源定位服务覆盖网——LAOverlay,具有高效的查询服务、自组织能力以及可扩展性,与 Chord 及其他基于 DHT 的系统相比,性能各方面得到了很好的改善,并且解决了基于DHT覆盖网中逻辑空间中节点的关系不能对

(下转第 2876 页)

面的计算步骤：

步骤 1：对模式 1 进行运动搜索，获得运动矢量。

步骤 2：计算该宏块是否为均匀区域。

步骤 3：如果运动矢量为零或者该宏块为均匀区域，则判断该宏块采用模式 1；否则进入步骤 4。

步骤 4：分别对该宏块内的四个 8×8 块进行小模式合并运算；如果可以合并则对最佳块模式进行运动搜索，算法结束，否则进入步骤 5。

步骤 5：分别对各 8×8 块内的 4×4 块进行小模式合并运算，如果可以合并则对新合并的模式进行运动搜索，算法结束；否则判定模式 7 为最佳模式。

3 实验结果

对本文提出的快速算法进行了实验仿真。计算机配置为：P4 1.6GHz，256M 内存；实验环境：编码器为 JM8.6，帧速率为 30 帧/s，运动矢量搜索范围为(－16，16)，码率控制关闭，熵编码 CAVLC，编码帧数为 100 帧，视频序列编码的 GOP 格式为 IPPPP…。实验结果如表 4～表 6 和图 4 所示。

表 4 快速算法与原算法决策的模式分布比例

模式	原算法(%)	快速算法(%)
模式 0	0.37	0.55
模式 1	54.68	53.78
模式 2	11.47	7.10
模式 3	15.81	12.43
模式 4	11.41	21.97
模式 5	2.55	1.08
模式 6	2.90	1.72
模式 7	0.81	1.37

表 5 Mobile 的参考帧分布比例

参考帧	原算法(%)	快速算法(%)
参考帧 1	47.15	47.43
参考帧 2	14.58	14.76
参考帧 3	15.17	15.53
参考帧 4	12.10	11.32
参考帧 5	11.00	10.95

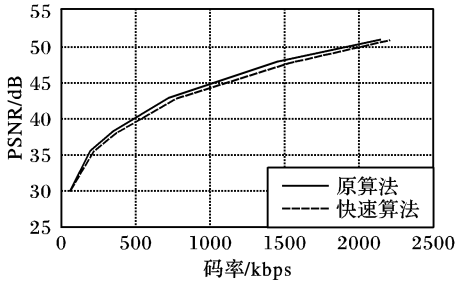


图 4 原算法与快速算法的 RD 曲线对比

由表 4 可以看出，快速算法和原算法模式抉择后各模式的分布比例很相近，特别是大模式的分布比例非常接近，但小模式的比例有些差异，这是因为本算法主要是从大模式的角度来提高编码速度，对小模式的判决可能有一定的误差；图 4 表明快速算法与原算法的 RD 曲线十分接近；表 5 表明快速算法与原算法模式决策后参考帧的分布情况几乎完全一样；由表 6 可以看出快速算法在不同量化系数时均可以保证 PSNR 降低很小(平均不超过 0.03dB)、码率提高不大(平均不超过 4%)的同时提高编码效率，编码时间平均减少 60% 以上。

表 6 快速算法与原算法编码时间和 PSNR 的对比

QP	图像序列	时间变化(%)	PSNR/dB	码率(%)
8	Foreman	－51.63	－0.02	＋2.96
	Grandma	－64.38	－0.02	＋1.13
	Table tennis	－46.51	－0.01	＋2.93
	Mobile	－46.08	－0.01	＋3.77
18	Foreman	－53.93	－0.03	＋2.86
	Grandma	－65.97	－0.01	＋2.39
	Table tennis	－54.28	－0.02	＋3.44
	Mobile	－48.00	－0.02	＋2.49
28	Foreman	－62.43	－0.01	＋3.40
	Grandma	－69.68	－0.01	＋2.92
	Table tennis	－59.16	－0.04	＋3.58
	Mobile	－52.41	－0.02	＋4.13
38	Foreman	－66.67	＋0.01	＋2.97
	Grandma	－69.53	－0.01	＋1.12
	Table tennis	－62.71	－0.02	＋4.23
	Mobile	－63.47	－0.02	＋3.07

本文的算法是一种很有效的模式抉择快速算法，可以在对编码质量产生很小影响的情况下提高编码效率。

参考文献：

[1] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)[S]. JVT of ISO/IEC MPEG and ITU-T VCEG, 2003. 131－152.

[2] RICHARDSON I EG. H.264/MPEG-4 Part 10 White Paper[Z]. 2003.170－172.

[3] JM Reference Software. JM8.6[CP/DK]. http://iphome.hhi.de/suehring/tml/, 2005－10.

[4] LIM KP, WU S, WU DJ, *et al.* Fast INTER Mode Selection[A]. JVT I020, 9thJVT meeting[C]. 2003.2－7.

[5] PURI A, HANG H, SCHILLING D. Interframe Coding with Variable Block Size Motion Compensation[A]. Proceedingsof GLOBECOM'87[C]. 1987. 65－69.

[6] YU-KUANG TU, JAR-FERR YANG. Fast variable-size block motion estimation using merging procedure with an adaptive threshold[A]. Proceedings of ICME[C]. 2003.789－792.

(上接第 2828 页)

应实际网络中的关系的问题，将资源本地化，提高了系统的性能。该覆盖网适用于媒体分发以及应用层流媒体，具有广阔的应用前景。

参考文献：

[1] 中国计算机学会学术工作委员会. 中国计算机科学技术发展报告(2004)[R]. 北京：清华大学出版社，2004.121－122.

[2] STOICA I, MORRIS R, KARGER D, *et al.* Chord: a scalable peer-to-peer lookup service for Internet applications[A]. Proceedings of ACM SIGCOMM[C]. 2001.

[3] KRISHNAMURTHY B, WANG J. On network-aware clustering of Web clients[A]. Proceedings of ACM SIGCOMM[C]. 2000.

[4] ZHAO BY, DUAN Y, HUANG L, *et al.* Brocade: landmark routing on overlay networks[A]. Proceedings of First International Workshop on Peer-to-Peer Systems[C]. 2002.

[5] XU ZC, MAHALINGAM M, KARLSSON M. Turning Heterogeneity into an Advantage in Overlay Routing[A]. Proceedings of INFOCOM[C]. 2003.

[6] ZHANG X, LIU J, ZHANG Q, *et al.* Measure: a group-based network performance measurement service for peer-to-peer applications[A]. GLOBECOM[C]. 2002.