

基于 IPv6、簇和超节点的 P2P 路由模型研究

曹 慧, 董健全

(上海大学计算机工程与科学学院, 上海 200072)

摘要: 针对当前 P2P 网络模型存在逻辑拓扑与物理拓扑失配和没有充分考虑节点异构性的缺点, 利用 IPv6 特有的层次化地理布局, 实现逻辑拓扑与物理拓扑的有效结合, 应用簇减少节点频繁加入和退出所带来的网络抖动问题, 充分考虑节点之间的性能差异, 让性能好的节点承担更多的任务, 利用节点成功的资源访问记录进一步提高路由效率。通过模拟实验和对相关数据的分析, 证明该项研究能使路由延迟减少, 网络更稳定, 更易于管理和维护。

关键词: P2P 网络; IPv6 协议; 簇; 超节点

Research on P2P Routing Model Based on IPv6, Cluster and Super-peer

CAO Hui, DONG Jian-quan

(College of Computer Engineering and Science, Shanghai University, Shanghai 200072)

【Abstract】 In view of the mismatch between P2P overlay network topology and physical network topology, and neglecting of the heterogeneity of peers, this paper applies IPv6 hierarchical structure to make overlay network being congruent to physical network, introduces cluster to decrease churn of the network due to frequent joining and leaving of the peers takes into account the heterogeneity of peers chooses peers with great capability to take more responsibility in routing; and adopts successful resource lookup history to improve routing efficiency. Simulation experiments and analysis of involved data show that the research can make routing delay decrease, network more stable and easy to manage and maintain.

【Key words】 P2P network; IPv6; cluster; super-peer

1 概述

P2P技术通过有效地组织网络中的节点来管理其提供的资源。经典的结构化P2P网络Chord, CAN, Viceroy都是基于分布式的哈希表(DHT), 它们把节点组织成结构化的覆盖网络, 克服了非结构模型的可扩展性差和存在单点失效的不足, 并且查找消息大多能够在 $O(\ln n)$ 逻辑跳数内到达目标节点, 其中 n 是系统中节点个数。但是, 这种构建在IP层之上的逻辑覆盖网络与物理拓扑即IP层相分离^[1], 很可能逻辑网络的一跳在物理网络却是跨洲的一跳, 从而导致了路由延时过大、路由效率低下等问题。另外, 所有的结构化P2P都直接或间接地假设系统中的节点在网络带宽、CPU性能、在线时间等是相似的, 使得性能差的节点成为路由瓶颈, 导致了在路由效率方面实践低于理论。

Viceroy^[2]与Chord类似, 它把所有节点组织成一个环, 并充分利用了蝴蝶图的强大路由能力。在Viceroy中, 每个节点都有2个值: 标号 id [0, 1]和层号 L , 所有节点构成一个大环, 同层节点又构成一个小环。一个位于 L 层的节点有7个指针指向它的邻居, 它们分别是前驱和后继节点, 同层小环上的前后2个相邻节点以及 $L+1$ 层的左右2个节点和 $L-1$ 层的上行节点。Viceroy的路由包括3步: 由上行指针上升到第1层的某个节点, 再通过下行指针到达一个离目标最近的节点, 最后利用大环或者同层环指针到达目标节点。Viceroy的每次查询需要 $O(\ln n)$ 步。

本文在对结构化 P2P——Viceroy 研究的基础上提出了 Viceroy in IPv6 network(以下简称 Viceroy6), 它充分考虑网络

中节点的性能差异, 让性能好的节点参与到 Viceroy6 的主干网络中, 并且充分利用 IPv6 层次化地址结构来解决逻辑网络和物理网络的失配问题, 使得具有相同 IPv6 前缀的节点聚集在一起形成簇, 并充分利用历史查询数据, 来减少路由延迟和提高路由效率。

2 节点性能评价

本文计算节点的性能值利用4项参数: CPU性能(C), 网络带宽(B), 存储能力(M)和在线时间(T), 那么节点 p 的性能计算公式为

$$capability(p) = \sqrt{C^2 + M^2 + B^2 + T^2} \quad (1)$$

当 A, B 两个节点交互时, A, B 都要获得对方的4个参数, C, M, T 是一方通过发送请求给另一方得到, 其中, T 是节点上次的在线时间, 双方在交互时可直接获得另一方的 B 。某节点从另一节点得到的参数真实性如何, 涉及到节点的信誉度问题, 本文采用文献[3]中关于信誉度评价的策略, 即在 a 节点评价 b 节点的信誉度时, a 从本地信任值表中选择其最信任的 m 个节点, 同时在网络中搜索 n 个不在 a 本地信任值表中的节点, 查询这些节点对 b 的评价, 综合考虑 $m+n$ 个节点对 b 的评价, 从而得到 a 对 b 的最终评价, 即 R_{ab} 。

为了记录节点交互时对方节点的性能值, 每个节点要维护一张节点交互表。节点 A 的交互见表1。

基金项目: 上海市科委发展基金资助项目(7A05722)

作者简介: 曹 慧(1981-), 女, 硕士研究生, 主研方向: P2P 网络, 数据库技术, 人工智能; 董健全, 副教授

收稿日期: 2007-11-28 **E-mail:** caohui1028@163.com

表1 节点A的交互表

节点ID	交互时间	成功与否	CPU性能	存储能力	接入带宽	在线时间	IP
ID1	Time1	0或1	C1	M1	B1	T1	IP1
...

表1为节点A与其他节点在某个时间 t_c 内的交互记录,这个时间可由具体应用或时间和空间条件而定。交互是否成功是由节点A来判断的。当节点A要计算节点B的性能时,假设当前表中节点A与节点B中交互成功次数为 s ,失败次数为 f ,则

$$capability(B)_A = R_{ab} \left(\frac{s}{s+f} \times \sum_{i=1}^s capability(i) + \frac{f}{s+f} \times \sum_{j=1}^f capability(j) \right) \quad (2)$$

其中, R_{ab} 为节点A对节点B的信誉度; $capability(i)$ 和 $capability(j)$ 为采用式(1)计算得到的性能值。

当节点A的交互表中没有节点B的交互记录时,节点A要从自己的交互表中选择 t 个与节点B有过交互的节点(通过表中的IP直接发送询问请求),通过这 t 个节点间接地得到节点B的性能值:

$$capability(B)_A = \left(\sum_{i=1}^t (R_{ai} \times capability(B)_i) \right) / t \quad (3)$$

其中, R_{ai} 为A对选择的 t 个节点中第 i 个节点的信誉度; $capability(B)_i$ 为选择的 t 个节点中第 i 个节点按照式(2)计算得到的对节点B的性能评价。

3 Viceroy6

IPv6的IP地址由表示特定网络的网络前缀和表示主机或服务器的接口标识2部分构成,IPv6的分配体现了很强的层次性,特定相同前缀长度的节点都处于相同的自治域内(例如fec0:0:0:1::1234/64,前64位构成地址前缀,这64位前缀代表一个自治域,具有相同此前缀的主机都处于此自治域中),Viceroy6就充分利用了IPv6地址特有的层次化地理布局。

3.1 拓扑结构

Viceroy6的拓扑结构如图1所示(此图没有画出上行和环指针),从图1(1)中可以看出其拓扑结构从整体上与Viceroy是一致的,但是图1(1)中的每个“节点”其实是一个虚拟节点,从图1(2)中看出其实它是一个簇(cluster,类似于非结构化的^[4]),每个簇中有若干个超节点和普通节点。处于同一自治域(IPv6地址中前 c 位相同)的节点首先组成一个簇,每个簇作为一个“节点”加入Viceroy6中,簇中的超节点是按照前述节点性能的计算方法选出性能优异的节点,负责此簇的组织路由等工作。当第1个节点加入到簇中时,簇便建立,并且此节点成为超节点,它的 id 也成为此簇的 id ^[5]。

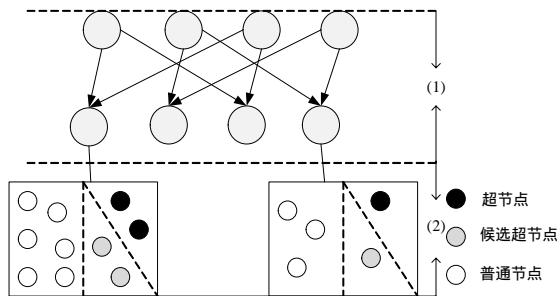


图1 Viceroy6的拓扑结构

另外,所有节点无论是超节点还是普通节点,在查找资源并成功得到回复后,要记下资源地址信息(key_id, IP),如

下所示,若是普通节点,则要保存到超节点,即超节点要保存一个资源地址信息表:

Key_id	IP	交互次数	最近一次交互时间
--------	----	------	----------

每次交互成功后才记录此次交互,若是第1次交互则插入记录,否则只要更新记录即可,即交互次数加1,最近一次交互时间为本次交互时间。资源地址信息表由每个簇内的超节点保存某个时间 t_s 内簇内各节点交互成功的资源地址记录, t_s 与上文的节点交互表的 t_c 相似。“交互次数”可以记录资源的流行度,可以对流行资源增加副本。“最近一次交互时间”主要用于资源地址记录的删除,最近未使用的记录先删除。当簇内某节点查找某资源时,它首先查看其超节点的资源地址信息中是否有要查找的资源地址,若有则直接进行交互即可,既节省资源又提高效率。

3.2 节点层数和标识符的计算

节点标识符由value1和value2两部分拼接而成,value1为IPv6地址中前 c 位的Hash函数值,value2为IPv6地址中余下的 $128-c$ 位的Hash函数(与上面函数相同)值,value2拼接在value1之后得到的值作为节点的标识符,这样拼接后得到的节点标识符,前 c' 位相同的节点加入系统后处于同一个簇中,即具有相同前缀部分的节点被映射到邻近的逻辑空间中,实现了逻辑拓扑与物理拓扑的有效吻合。簇的 id 是建立簇的节点的 id 。

节点 s 加入后若存在与其标识符前 c' 位相同的簇,则其层数即为其簇中节点的层数,否则,节点 s 为簇中第1个节点,它通过计算与其后继簇的距离得到当前网络规模估计值 n_0 ,然后从 $1, 2, \dots, \lfloor bn_0 \rfloor$ 之间随机选择一整数作为其level,即

$$(1) n_0 = 1/d(s, \text{SUCC}(s));$$

$$(2) \text{从 } 1, 2, \dots, \lfloor bn_0 \rfloor \text{ 之间选择一整数作为 level.}$$

3.3 路由表维护及路由算法

在Viceroy6中,超节点的路由表和普通节点是不同的,因为只有每个簇的超节点在Viceroy6中起路由作用,普通节点通过某个超节点与其他簇中的节点进行交互,所以所有超节点都维护着所在簇的 id ,大环的前驱和后继邻居,同层环的前后2个邻居以及上下层的3个蝴蝶邻居,还有其在簇中的普通节点的信息;而普通节点除了维护层之外,只需知道它所在簇的某个超节点即可。

Viceroy6的查找算法包含3个阶段:在上升阶段查找发起节点找到所在簇的某个超节点,看它的普通节点信息或资源地址信息中是否包括要查找的关键字,若包括则直接返回此信息给发起节点,发起节点可通过此信息直接和存放资源的节点交互,否则超节点要通过上行指针上升到第1层,然后通过下行指针向下查找;在下降阶段,当到达 i 层的某个簇,距离目标最多为 $1/2^{i-1}$,如果距离目标大于 $1/2^i$,则沿右下指针下降,否则沿左下指针下降,直到某个簇没有下行指针或超过了目标;第3阶段顺时针或逆时针方向遍历层环或大环。整个路由过程都是在各个簇之间进行,直到找到目标簇,目标簇中的超节点会在本簇内进行查找,找到后此存放目标资源的节点才和发起查询的节点直接交互。

3.4 系统维护

在Viceroy6中,每个簇包含一个或若干个超节点,这些超节点共同管理它们所在的簇,它们维护的信息是相同的,都维护着簇中普通节点信息和资源地址信息,它们之间要周期性地交互各自的信息以保证信息的一致性,而簇中的普通

节点只要知道一个超节点即可。当某个节点 P 要加入到系统中时,它首先要确定是否存在与其标识符前 c' 位相同的簇,若存在,则根据 P 的性能如何而确定加入到此簇中的状态。当簇中节点数过多时,需要增加超节点来分担查询和路由任务,即簇中每个超节点负责的节点数超过某个阈值 η 时,便要增加一个超节点,其中, η 为经验值,要根据节点性能和实际应用情况而定。每个簇除了包含若干个超节点外,还包含若干个候选超节点,它们是超节点的备份。超节点都是从候选节点中选择的,簇中有几个超节点,就有几个候选超节点,当某个候选超节点被选择为超节点后,其他超节点就要从普通节点中选择性能好的节点作为候选超节点。若不存在与其标识符前 c' 位相同的簇,则此节点为簇中第 1 个节点,即簇的建立者,此簇的 id 即为此节点的 id ,随后通过查找算法更新 P 的路由表即可,流程如图 2 所示。

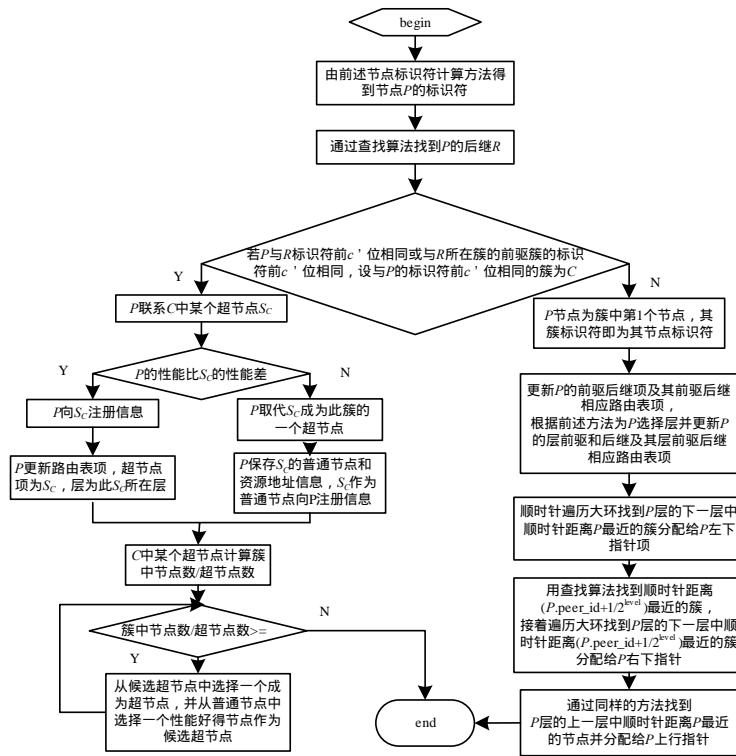


图 2 节点加入算法

当某个节点 Q 离开时,若此节点不是超节点,则其离开后只要超节点收回其负责的资源并把其信息从普通节点信息中删除即可。若离开节点是超节点,则其他超节点从候选超节点选择一个作为新的超节点,同时要从普通节点中选择一个节点成为候选超节点。

4 仿真与性能评价

本文模拟程序用 Java 编写,在节点数为 10 000~20 000 的网络中模拟了 Viceroy 和 Viceroy6 两种模型,实验结果显示了 Viceroy6 在路由效率、减少网络抖动和管理与维护方面的优势。其中,图 3 和图 4 为网络规模节点数为 10 000、域的个数从 1~1 024 的情况下 2 种模型的路由延迟和路由跳数的数据。实验中设定域内每一跳的延迟为 10 ms,域间每一跳的延迟为 100 ms,设 η 值为 8。每次查找时,随机选择一个作为查找起点的节点标识符和一个作为查找目标的数据标识符。

在图 3 中,随着域个数的增加,Viceroy 和 Viceroy6 的

平均路由跳数没有明显变化,Viceroy 的平均路由跳数大约为 50,而 Viceroy6 为 24,这是因为路由跳数只是逻辑拓扑上的跳数,即使域的个数增加也是物理网络上的,所以不会影响逻辑路由跳数。但是,从图 3 中明显看出,Viceroy6 的平均路由跳数大约是 Viceroy 的一半,这是因为 Viceroy6 充分利用了簇和节点成功的资源查找记录,使得网络规模减小,且重复路由减少,所以减少了路由跳数,提高了路由效率。

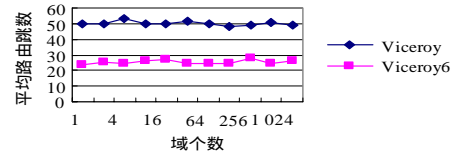


图 3 Viceroy 与 Viceroy6 平均路由跳数的比较

在图 4 中,随着域个数的增加,Viceroy6 与 Viceroy 的平均路由延迟都大幅增加,而在域个数到达各自的平均路由跳数大小时,即 Viceroy 在域个数大约为 50 时路由延迟达到最大,大约为 5 s,在 Viceroy6 中,域个数为 24 时路由延迟达到最大,大约为 2.4 s,几乎每一跳都是跨域的,路由延迟达到最大,而从这一点开始路由延迟开始较平缓,几乎不再发生变化。从图 4 中仍可明显看出,当域的个数较小时,Viceroy6 与 Viceroy 的平均路由延迟相差不多,但随着域个数的增加,Viceroy6 的平均路由延迟明显小于 Viceroy,这是因为 Viceroy6 充分利用了 IPv6 地址特有的层次化地理布局来实现逻辑拓扑与物理拓扑的吻合,使得逻辑上邻近的节点物理上也较为邻近,同时利用簇也减少了宏观上的网络规模,从而减少了路由跳数和路由延迟。

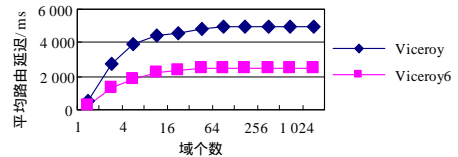


图 4 Viceroy 与 Viceroy6 平均路由延迟的比较

5 结束语

本文所述模型是结构化 P2P 中嵌入集中式的拓扑结构,与 P2P 其他网络结构模式相比,这种混合模式路由效率更高,网络更稳定,负载更平衡,更易管理,但本文中的一些参数如 η , c , t_c , t_s 都是经验值,有待进一步研究,同时怎样有效地利用查找成功的历史记录也是值得研究的内容。

参考文献

- [1] 熊继平. 基于 IPv6 地址聚类性的改进型 DHT 网络[J]. 小型微型计算机系统, 2006, 27(8): 1421-1425.
- [2] Malkhi D, Naor M, Ratajczak D. Viceroy: A Scalable and Dynamic Edulation of the Butterfly[C]//Proceedings of the 21st Annual Symposium on Principles of Distributed Computing. California, USA: [s. n.], 2002: 183-192.
- [3] 宋金龙. 一种 P2P 网络安全的信誉度模型设计[J]. 计算机应用, 2006, 26(4): 833-835.
- [4] 姜守恒. 基于超节点的 Chord 系统[J]. 小型微型计算机系统, 2007, 28(2): 266-270.
- [5] Lv Ertao, Duan Zhenhua, Qi Jianjun, et al. Incorporating Clusters into Hybrid P2P[J]. Digital Society, 2007, 20(1): 17-21.