一种动态负载均衡的 P2P 应用层组播方案*)

林龙新 周 杰 张 凌 叶 昭

(华南理工大学广东省计算机网络重点实验室 广州 510641)

摘 要 基于结构化的 P2P 基础设施,给出一种动态负载均衡的应用层组播方案——DLBMS。利用 Tapestry 协议的路由和定位机制,设计了延迟优化的组播转发树结构,采用根节点复制的方法生成多棵不相交的组播转发树,根据负载的变化动态调节组播转发树数目以实现负载均衡和降低源到组成员节点的端到端延迟。通过模拟实验说明了此方案在平均控制负载和端到端平均延迟方面的有效性。

关键词 组播,应用层组播,对等网络,负载均衡

A Dynamic Load Balancing P2P Application Layer Multicast Scheme

LIN Long-Xin ZHOU Jie ZHANG Ling YE Zhao

(Guangdong Key Laboratory of Computer Network, South China Univ. of Tech , Guangzhou 510641)

Abstract This paper presents DLBMS, a scalable and dynamic load balancing application layer multicast scheme built on structured P2P substrate. DLBMS utilizes Tapestry's routing and data location service to create delay optimized multicast delivery tree, optimizes the end-to-end delay between source and group members and achieves load balancing by splitting the DLBMS multicast delivery tree into a set of disjoint multicast delivery trees and adjusting the number of multicast delivery trees dynamically according to the change of load. We compare DLBMS and Bayeux by simulation, and the results show that DLBMS has the obvious advantage in terms of end-to-end delay and load balancing.

Keywords Multicast, Application layer multicast, Peer to peer, Load balancing

组播是 Internet 中实现群组通信业务的关键技术。对于传统 IP 组播,数据分组的复制、转发以及群组的管理都由路由器完成。应用层组播(Application Layer Multicast, ALM)则利用覆盖网(Overlay Network)技术,在现有的 IP 网络上构建一个虚拟网络,将组播功能从路由器转移到端系统。与 IP 组播相比,ALM 的优势在于:组播服务部署容易;可动态适应网络条件的变化,实现服务定制等。近来,对等网络(Peer-to-Peer, P2P)技术在许多领域取得了巨大成功,成为了网络研究的焦点。P2P 网络具有自组织、扩展性好、容错能力强、完全分布式等特性,基于 P2P 网络实现组播服务是当前的研究热点,出现了众多研究成果,例如:Bayeux[1]、Scribe[2]、SplitStream[3]、CoolStreaming[4]等。这些方案有一个共性,即在已经存在的结构化或者无结构化的 P2P 基础设施之上构建组播通信业务。本文提出了一种动态负载均衡的应用层组播方案(Dynamic Load Balancing P2P Multicast Scheme, DLBMS)。它基于 Tapestry[5] 协议构建的结构化 P2P 基础网络,其主要目的是希望在 P2P 协同工作环境中实现一些实时应用。DLBMS 利用 Tapestry 的路由和定位机制来生成延迟优化的组播转发树,采用动态选择多个根节点的方法把单棵组播转发树分割成互不相交的多棵组播转发树,并根据负载的变化自动调节组播转发树数目以实现动态负载均衡和优化源到组成员节点的端到端延迟。

1 Tapestry

Tapestry 是一种著名的基于分布式哈希(Distributed Hash Table, DHT)方法的 P2P 协议,可用于形成 P2P 网络,具有良好的节点路由和资源定位性能。在 Tapestry 中,节点代表 P2P 网络中的对等主机,资源对象代表网络中共享的资源。每个节点和资源对象都对应一个唯一的标识,分别为 *NodeId* 和 *ObjectId*。*NodeId* 和 *ObjectId* 属于相同的命名空间,可表示为形如 $a_{n-1}a_{n-2}\cdots a_i\cdots a_1a_0$ 的数字序列,通过某种均匀的分布式哈希算法(如 SHA-1)生成。其中,n 代表序列的长度,a_i 为 b 位二进制数表示的整数,且 $a_i\in[0,2^b-1]$,2^b 称为基(base),$N=(2^b)^n$ 为 Tapestry 的命名空间大小,称 $a_{i-1}\cdots a_1a_0$ 为 *NodeId* 的 i 位后缀($i\geqslant 1$)。

Tapestry 的路由机制:Tapestry 中每个节点 A 都有一个本地路由映射表。路由映射表包含 n 列(从第 0 列开始),每列有 2^b 个记录,每个记录为节点 A 的一个邻居节点 B 的相关信息(B 的 *NodeId*、B 的 IP 地址等)。第 j 列的第 i 个记录($0\leqslant i\leqslant 2^b-1$,$0\leqslant j\leqslant n-1$)所代表邻居节点的 *Nodel* 以"$i$"+$suffix(A,j)$为后缀,其中 $suffix(A,j)$为 A 的 j 位后缀($j\geqslant 1$);当 $j=0$ 时,*NodeId* 以"i"为后缀。在 Tapestry 中,节点之间的路由通过后缀匹配的方式实现。路由过程中,第 j($j\geqslant 1$)跳节点的 *NodeId* 和目标节点的 *NodeId* 至少有长度为 j 的后缀是相同的。通过查询第 j 跳节点路由映射表的第 $j+1$ 列的邻居节点信息,找到后缀的第 $j+1$ 个数字和目标节点 *NodeId* 相同的邻居节点作为第 $j+1$ 跳节点。这种路由机制保证经过最大 $\log_{2^b}N$ 跳之后任何节点都能够被找到。

Tapestry 的资源发布和定位机制:在 Tapestry 中,每个资源对象都和一个或多个称为"定位根"的节点相关联。为了

*)基金项目:国家 973 计划项目(2003CB314805);国家 CNGI 项目(CNGI-04-13-2T);2005 年粤港关键领域重点突破项目"IPv6 核心路由器研发与产品化"。林龙新 博士研究生,主要研究方向:应用层组播,对等网络。周 杰 副教授,博士。张 凌 教授,博士。叶 昭 博士研究生。

在P2P网络中发布资源对象O,存贮O的服务器S通过Tapestry路由机制以定位根为目标节点发布"位置消息"。此消息包含〈*ObjectId*,*ServerId*〉映射。从S到定位根的路径上除S之外的所有节点都保存此映射信息。在资源对象查询和定位过程中,查询节点向目标资源对象关联的定位根节点路由查询消息,如果此路径中的一个中间节点恰好含有该资源对象的位置信息,那么,查询消息将被直接转发到对应的服务器S。否则,将一步步靠近定位根。查询消息到达定位根节点可确保找到其位置信息,从而准确定位目标资源对象。

2 DLBMS组播树和动态负载均衡策略

DLBMS是一种源指定应用层组播,建立在Tapestry之上。在会话管理方面采用和Bayeux同样的方法:在DLBMS中,一个组播会话用二元组〈*SessionName*,*UID*〉唯一标识。组播源将此二元组转化为160比特的*ObjectId*(即*GroupId*),创建一个以此*ObjectId*为名字的文件,其中包含组播会话的相关信息。它被看成是Tapestry的资源对象。组播源通过Tapestry的资源发布机制把该资源对象发布到对应的定位根。

2.1 DLBMS组播转发树

在源指定的ALM方案中,源到组成员节点的转发路径的长度在某种程度上决定了源到组成员节点的延迟,对于实时应用,尽可能使组播转发树的深度在满足度约束的条件下越小越好。现有的P2P组播方案,数据转发树中包含了大量的非组成员节点,而这些节点的存在增加了源到组成员节点的数据传输延迟,并且因为这些节点不是组成员节点,在对等网络中,这些节点随时都可能离开网络,这样会影响组播转发树结构的稳定。因此,DLBMS所构建的组播转发树不再包含非组成员转发节点。在DLBMS的组播转发树中,节点需要保存的主要状态信息有:该节点在组播转发树中所处的逻辑"层"(源处于最高层,即第0层),该节点的孩子节点信息,该节点的父亲节点信息。组成员节点的加入(JOIN过程)伴随着组播转发树的生成,算法如下:

①假定源节点为S,S是组播转发树的根。节点M(设其*NodeId*为$a_{n-1}a_{n-2}\cdots a_1a_0$)要求加入组播组。

②M根据*GroupId*通过Tapestry的资源对象定位机制获得S的IP地址信息,直接向S发送JOIN请求消息。S根据自身在组播转发树所处的逻辑"层"值来确定初始后缀对比序列(层值为i,那么初始后缀对比序列为$a_ia_{i-1}\cdots a_1a_0$)。S先查看M的*NodeId*的a_0值,并检查自己的孩子节点列表中是否有*NodeId*的第1个数字和a_0相同的节点。如果没有,S就将M作为其孩子节点,并将M作为组播转发树层1中a_0域的代表节点,然后回复成功的JOIN响应消息,JOIN过程结束。否则,S向M回复重新启动JOIN过程的消息,其中包含*NodeId*的第1个数字和a_0相同的S的孩子节点C的相关信息。M继续向C发送JOIN请求消息,C将查看M的*NodeId*的a_1值来进行类似操作。

③如上的JOIN过程最多经过n次迭代,M最终会被放置到组播转发树的一个合适位置。

例如,在一个Tapestry网络中,设base等于2,节点的*NodeId*表示为$a_2a_1a_0$,$a_i\in\{0,1\}$。假设节点010为源,节点000,101,100,001为组成员,按照如上算法先后加入到组播组。最后形成的组播转发树如图1所示。

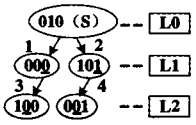

图1 DLMS组播转发树

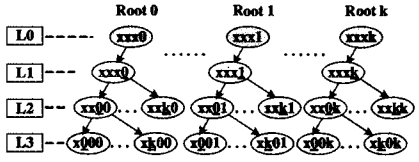

图2 基本树分割产生的组播森林

图1中,S为源,椭圆表示节点,带箭头的连线表示树的边,边上的数字表示节点加入的次序。L0表示第0层,L1表示第1层。第1层包含节点000和101,其中带下划线的0、1表示不同的域。000作为树中第1层"0"域的代表节点,101作为第1层"1"域的代表节点。源的不同子树中包含的节点的*NodeId*有相同后缀。DLBMS组播转发树的最大深度小于等于n。树所包含的节点只有源S和组成员节点,不再包含非组成员转发节点。由于每个节点的孩子数最大为base,因此节点的最大出度不超过base。组播树中相邻节点通过定期交互消息来保持联系,当节点离开的时候,它需要向其相邻节点发送LEAVE消息引发相应过程来维持树结构的稳定,具体细节在此不再叙述。

2.2 DLBMS动态负载均衡策略

在DLBMS中,组播转发树的根节点需要处理所有组成员的加入请求并转发所有数据分组,这将是大规模扩展的瓶颈并易造成单点失效。为此,DLBMS采用动态树分割的方法来消除这种影响,基本思路如下:

① 把DLBMS产生的单棵组播转发树分割成互不相交的多个部分,每部分对应一棵新的DLBMS组播转发树,这些组播转发树组成组播森林。每棵树的根节点是Tapestry网络中的非组成员节点。源把需要组播的资源对象复制到每棵组播转发树的根节点。不同的根节点负责自己所在的组播转发树的控制信息处理和数据分组分发。

② 组播过程中组成员是动态变化的,可能会造成节点在同一棵组播转发树的某个分支聚集的情况。DLBMS把满足一定聚集条件的分支分离出新的组播转发树,从而进一步优化源到组成员节点的传输延迟和降低根节点的平均控制负载。

③ 根节点之间以网状(mesh)结构组织在一起,彼此都拥用一份相同的资源对象和状态信息副本,通过信息交换以确保此网状结构的稳定。DLBMS树分割策略分为两步:第一步为基本树分割,第二步为动态树分割。

基本树分割算法

在DLBMS单树中,源S(即根节点)的所有孩子节点的*NodeId*的a_0项互不相同,除根节点外,其它节点被自然地被划分成多个不相交的分支。把这些分支所包含的节点组成的集合定义为$S_0,S_1,\cdots,S_i,\cdots,S_{2^b-1}$,($2^b=base$),$S_i$包含的节点的*NodeId*有相同的$a_0$值。基本树分割算法如下:

① 源S从本地路由映射表的第0列中选取*NodeId*和S的1位后缀不同的2^b-1个节点,S与这2^b-1个2^b节点组成个初始根节点。

② S 把要组播的资源对象 O 复制到其余的 2^b-1 个根节点。

③ 所有根节点通过 Tapestry 提供的资源对象发布机制向 Tapestry 网络中发布对象 O。

④ 组成员通过 Tapestry 的路由和定位机制找到和自己的 *NodeId* 具有相同 a_0 值的根节点，然后运用 DLBMS 的 JOIN 过程，把自己加入到相应的组播转发树中。

图 2 给出了基本树分割算法生成的 2^b 棵组播转发树(图中 $k=2^b-1$)。图 2 中，椭圆代表节点，椭圆内的数字代表该节点的 *NodeId*。灰色背景的节点是根节点。

根据 DLBMS 的 JOIN 过程，DLBMS 产生的每棵组播转发树的根结点只有一个孩子节点。这一性质在讨论根节点失效时起重要作用。

动态树分割

组成员节点的动态变化会造成在同一棵组播树中出现节点局部聚集的情况，即在树的某些分支组成员节点相对密集，而另外一些分支又很稀疏。如果把满足一定密集程度的分支单独分离出来，组成新的组播树，可以进一步优化源到组成员节点的传输延迟，同时降低根节点处理控制信息的开销。

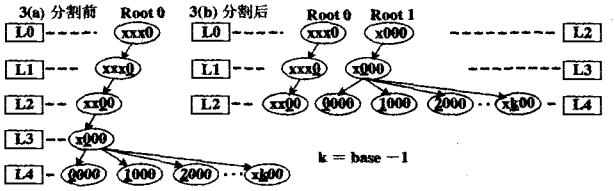

图 3　动态树分割过程

设在 DLBMS 的组播森林中某棵组播转发树节点的分布情况如图 3(a)所示，组成员节点大量聚集在节点 x000 所在的子树中。若把 x000 所在的子树从组播转发树中分离出来，组成新的组播转发树(如图 3(b)所示)，就可以进一步降低根节点平均控制负载和源到组成员节点的传输延迟。

配置参数 max_root_number 代表组播森林容许包含的组播转发树的最大数目。max_member_number 表示每棵组播转发树所包含的最大节点数目。只有当组播森林中组播转发树数目小于 max_root_number，某棵组播转发树所包含的组成员节点数目大于 max_member_number 且该组播转播转发树某个节点的出度大于等于给定的临界值 m 时启动动态树分割算法。算法描述如下：

① 组成员加入到某棵组播转发树后，相关节点检查自己的出度，当出度大于等于 m 时它向对应根节点发送消息，其中包含节点 *NodeId*、所处的层、出度、所辖成员数等参数，供树分割算法进行分割决策。

② 如果当前组播转发树满足前述的动态树分割条件，根节点选择出度大于等于 m 且“层”值最小的节点所在的分支进行分割，称这样的节点为“分割节点”。

③根节点向分割节点 P 发送 TREE_PARTITON_Req 消息，要求 P 选择新的组播转发树的根。

④设 P 的层值为 l。P 从本地路由映射表中选取和自己的 l 位后缀相同的非组成员且非根节点作为新的组播转发树的根，并把此节点的相关信息通过 TREE_PARTITION_Rsp 消息发送给当前的根节点。

⑤ 当前根节点自己或者依靠已存在的其他根节点把需要组播的资源对象复制到新的根节点上，新的根节点把资源对象发布到 Tapestry 网络。

⑥ 把 P 所在的子树迁移到新的根节点上。

图 3 是一个动态树分割的例子。动态树分割后，设分割节点 P 的层值为 l。当新的组成员 M 加入到被分割出来的组播转发树时，其 JOIN 过程中 M 和根节点的后缀对比序列的长度等于 l。

如前所述的基本树分割和动态树分割产生的组播森林的每棵组播转发树的根节点只有一个孩子，这些根节点之间组织成网状结构，彼此都拥用一份相同的资源对象和状态信息副本。每个根节点都保存“根节点网”的信息，并且把这些信息定期通知给各自的孩子节点。

根节点失效处理

组播转发树的根节点是 Tapestry 网络中的节点，这些根节点可能会失效(如根节点退出 Tapestry 网络)。本节给出处理根节点失效的机制，具体策略为：

① 失效根节点检测：通过“根节点网”定期检测根节点的存活情况，或通过组成员汇报的方式来检测根节点是否失效。

② 临时替代根节点：当某个根节点失效时，其孩子节点根据自身保存的“根节点网”信息，先指定另外一个有效的根节点作为其父节点，即“临时替代根节点”，以避免由于根节点失效而导致的组播会话中断。

③ 新组播转发树的构造：临时替代根节点把失效根节点的孩子节点作为动态树分割算法中的分割节点。按照动态树分割的算法选择一个新的根节点以替代失效根节点，并重新构造组播转发树。

图 4 给出一个根节点 xxx0 失效后的处理过程，节点 xxxk 为临时替代根节点，yyy0 为新构造的组播转发树的根节点。

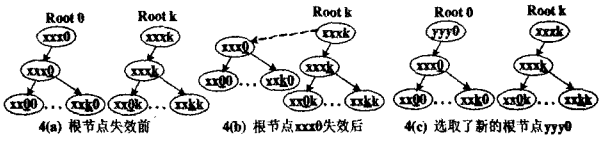

图 4　根节点恢复

3　性能评价

3.1　评价度量

为了评价 DLBMS 的有效性，就以下两个性能参数将其分别与 Bayeux 方案进行比较：

① 根节点的平均控制负载(Root Average Control Load，RACL)：在组成员加入组播组过程中，每根节点需要处理的控制消息的平均数目。

② 平均相对延迟损耗(Average Relative Delay Penalty，ARDP)：源到每个组成员节点的相对传输延迟损耗(RDP)[6]的平均值。在应用层组播中，源到一个组成员节点的组播通信延迟通常要大于它们之间的单播延迟。RDP 描述源到组成员节点的实际传输延迟和它们之间的单播延迟的比值。

3.2　仿真实验设计

参考相关 P2P 和应用层组播仿真器源代码，用 C++语言设计实现了一个可仿真 Bayeux 和 DLBMS 协议的离散事件模拟仿真器。用 GT-ITM[7] 网络拓扑发生器产生由 20000 个节点组成的 Transit-Stub 类型的随机网络拓扑，节点的单播路由按照最短路径算法设计。Tapestry 网的命名空间大小为 16384 (4^7)，base 为 4。从 20000 个节点中随机选取 16384

个节点作为 Tapestry 节点。

实验一：DLBMS 有效性测试。根据不同组播转发树的数目测试 DLBMS 在 RACL 和 ARDP 等方面的有效性。为保证可以分割出足够多的组播转发树，设定组播组的规模为 4000 个组成员，max_member_number 的值为 80，节点出度临界值 m 为 2。通过调节参数 max_root_number 来限定每次产生的根节点数目(组播组规模和 m 值的选取保证可以产生足够多的分割节点)。

实验二：DLBMS 多树、Bayeux 以及 DLBMS 单树在 RACL 和 ARDP 等方面的性能比较。组播组规模的最小值为 200，以 200 的步长递增，最大值 3000。设置参数 max_root_number 值为 20，max_member_number 值为 200，节点出度临界值 m 为 3。

3.3 实验结果与分析

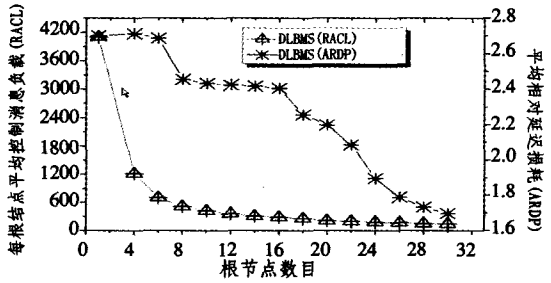

图 5 DLBMS 有效性实验

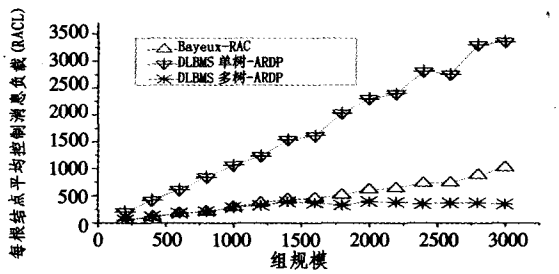

图 6 RACL 对比实验

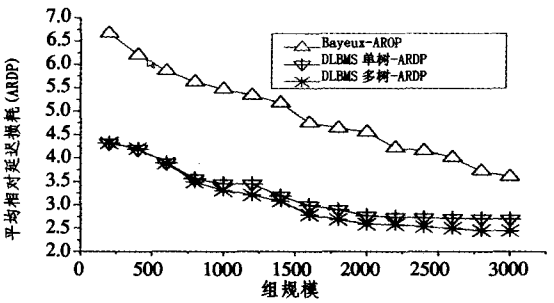

图 7 ARDP 对比实验

实验一测试结果如图 5 所示，横坐标表示测试中根节点数目，左侧纵坐标表示 RACL 的值，右侧纵坐标表示 ARDP 的值。当根节点数目为 1 时，即为 DLBMS 单棵组播转发树的情形；当根节点数目为 4(即 base)时，为 DLBMS 的基本树分割情形；当根节点数目大于 4 时为 DLBMS 动态树分割情形。实验表明，在组播组规模确定的情况下，随着根节点数目的增加，RACL 明显降低，ARDP 也逐步降低，这表明 DLBMS 在负载分担和优化延迟方面是有效的。

实验二中对于 RACL 的对比测试结果如图 6 所示。横坐标代表组播组的规模，纵坐标表示 RACL 的值。实验表明，DLBMS 在组播组规模较小的情况下(组成员数小于 1500)，它和 Bayeux 静态多根方案每个根节点所承担的控制负载基本接近。这是因为此时 DLBMS 还处于基本树分割状态，所产生的分担根节点数目和 Bayeux 相同。当组播组规模逐渐增大时，DLBMS 每个根节点平均控制负载要明显低于 Bayeux，这是因为此时动态树分割算法发挥作用，由于 DLBMS 单树情形只有一棵组播转发树，因此其在 RACL 方面性能最差。实验二中 ARDP 的对比测试结果如图 7 所示，对于 ARDP，DLBMS 多树优于 DLBMS 单树并明显优于 Bayeux。

结束语 基于 Tapestry 基础架构，给出动态负载均衡的应用层组播方案——DLBMS。不同于其他的 P2P 组播方案，DLBMS 利用了 Tapestry 协议特性构建了不再包含非组成员节点的延迟优化的组播转发树，通过动态选择 Tapestry 网络中的非组成员节点作为组播转发树的根节点，根据组播组规模的变化情况自动调整组播转发树的数目来实现控制负载的动态平衡和优化源到组成员节点的端到端延迟，仿真结果表明了它的有效性。所给方案适合于在 P2P 环境中开展对延迟有较高要求的组播应用。在 P2P 网络中引入组播技术将对改善当前 P2P 环境下一些业务造成的网络拥塞有一定帮助。下一步的工作重点是把节点的转发能力和带宽条件进行综合考虑，以构造更有效更鲁棒的组播转发结构并转化为系统在实际网络中验证。

参 考 文 献

1 Zhuang S Q, Zhao B Y, Joseph A D, et al. Bayeux: An architecture for scalable and fault-tolerant wide-area data dissemination. In: Proceedings of the 11th NOSSDAV, New York, 2001. 11～20

2 Castro M, Druschel P, Kermarrec A, et al. Scribe: a large-scale and decentralized application-level multicast infrastructure. Selected Areas in Communications, 2002, 20(8): 1489～1499

3 Castro M, Druschel P, Kermarrec A, et al. SplitStream: High-bandwidth Content Distribution in Cooperative Environments. In: Proceedings of SOSP, 2003

4 Zhang X, Liu J, Li B, Yum T-S P. DONet/CoolStreaming: A Data-driven Overlay Network for Live Media Streaming. In: Proceedings of IEEE INFOCOM, 2005

5 Zhao B Y, Huang L, Stribling J, et al. Tapestry: a resilient global scale overlay for service deployment. IEEE Journal on Selected Areas in Communications. 2004, 22(1): 41～53

6 Chu Y, Rao S G, Seshan S, et al. A case for end system multicast. ACM Sigmetrics, 2000. 1～12

7 Zegura E W, Calvert K L, Bhattacharjee S. How to model an internetwork. INFOCOM '96, IEEE, 1996. 594～602