分层迁移:分层多播拥塞控制中的快速响应方法*)

杨 明 张福炎

(南京大学计算机科学系 南京210093) (南京大学软件新技术国家重点实验室 南京210093)

Layer Transfer: A Rapid Response Method in Layered Multicast Congestion Control

YANG Ming ZHANG Fu-Yan

(Department of Computer science, Nanjing University, Nanjing 210093)

(Key Laboratory for Novel Software Techniques, Nanjing University, Nanjing 210093)

Abstract There exists large Leave latency (often on the order of several seconds)in current Internet group management mechanism. This will cause layered multicast congestion control in slow response to network congestion. We give a new concept of layer transfer, which can implement rapid congestion response in layered multicast. Layer transfer will move congested layer data from one group to another, and the receivers will determine whether to re-subscribe the moved layer or not according to their current situations. Layer transfer has a key feature that it replaces slow Leave process by fast Join process, so it doesn't require any change to IGMP protocol, routers, or other multicast routing protocol. The simulation results show the new method can improve layered multicast sessions' fairness towards TCP sessions.

Keywords Layered multicast, Congestion control, Layer transfer

1 引言

随着多播(multicast)技术的发展以及多播主干(Mbone)的普及,在Internet上出现了许多基于IP多播的音频/视频会话工具和多播流应用,然而这类应用却面临Internet的异构性和缺乏拥塞控制等主要问题。缺乏拥塞控制措施的多播应用无疑会影响行为规范的TCP流,还有导致网络瘫痪的可能。基于收方驱动的分层多播协议(RLM[1])最先提出分层多播拥塞控制方法,分层多播通过多个多播组实现了多速率多播传输,既满足了不同用户间的异构性和不同需求,又避免了冗余数据传输带来的带宽浪费。同时,收方能通过多播组的Join/Leave机制进行拥塞速率控制。这些优点使得分层多播成为多播拥塞控制中的一种有效方法。

分层多播拥塞控制中对拥塞的响应是通过组成员发出Leave请求进行的,因此拥塞响应的速度取决于互联网组管理协议IGMP[2,3]中Leave的实现机制。遗憾的是,在现有的IGMP协议中,由于不保存组成员信息,多播路由器在停止转发多播数据之前必须通过多次轮询来确认是否还有其他的有效成员存在,而在轮询时间内路由器并不停止转发多播数据,因此这种较长的轮询时延会导致拥塞响应慢,造成对拥塞响应迅速的TCP流的不公平性。

尽管RLC[4]采用同步点技术和使用较大时间参数的方法可部分缓解这一问题,但过大的时间参数会减慢拥塞反应速度,而减小时间参数值会增加不同步的机会,同样导致较大的Leave时延。L. Rizzo[5]尽管提出了在IGMP协议中实现快组管理方案,但需要对现行的IGMP协议作部分修改。J. Byers[6]提出动态分层的概念来解决Leave时延过大的问题,但所有组成员必须不停地定时加入动态分层来保持接收速率。本文提出分层迁移的方法既不需要对现有的IGMP作任何修改,又避免需不停地加入动态组带来的系统开销和复杂性。

2 Leave时延问题

Leave时延是指组成员从发出Leave请求到路由器确定停止转发多播数据的时延。在第1版的IGMP协议中,主机离开多播组的过程很简单,只要停止向路由器发送IGMP报告(report)即可,路由器在多次轮询后未收到任何IGMP报告,就停止转发并向上游路由器发送剪枝(prune)请求,从而完成多播数据的转发终止过程。由于离开的主机不能立即触发轮询过程,等待多次轮询超时使得Leave时延很大。

为适应拥塞控制的需要,在第2版的IGMP协议中,增加了Leave报文来加速离开过程。当一个主机离开多播组时,向最后一跳的多播路由器发送Leave报文,Leave报文将立即触发轮询定时器,同时还增加了缩短轮询间隔时间来加速组状态的检测的措施。然而,尽管在第2版的IGMP协议中采用了发送Leave报文的措施来加速离开,但由于该路由器不保存组成员信息,在停止转发之前必须通过轮询确认是否还有其他的有效成员存在;此外,为保证可靠性,路由器还要进行多次轮询。因此,总的轮询时间为T * R,其中T为轮询超时时间(缺省值为1秒),R为鲁棒因子(典型值R≥2)。由于路由器只有在这几秒的轮询时延(Leave时延)后才可能停止多播数据的转发,这将使得拥塞的状况持续较长时间,甚至加剧网络拥塞。

3 分层迁移方法

3.1 分层迁移

*)本课题获国家自然科学基金资助(项目编号60103013),获江苏省自然科学基金资助(项目编号BK2001035)。杨 明 博士后,研究方向为多媒体传输技术、多播通信。张福炎 教授,博士生导师,研究方向为多媒体技术、数字图书馆。

分层多播拥塞控制中的速率控制是通过多播组的 Join/Leave 机制完成的。在 Join 阶段，希望加入多播组的接收者向多播路由器发送 IGMP 请求报文，路由器则依次向位于源方向上游(upstream)的多播路由器发送接枝(graft)报文，接枝报文经过的反向路径将成为连接新组成员路由树的新分支，新分支建立后，该组成员便能接收多播数据。整个 Join 过程是很快的，Join 时延大约为一个 RTT(该接收者到源的巡回时间)。

分层迁移的基本思想是利用 IGMP Join 时延小(大约为一个 RTT 时间)的特点，用快速的 Join 过程取代慢速的 Leave 过程。取代的方法是发方将某个分层的数据由一个多播组迁移到另一个组播出，收方则根据情况进行重新预订。分层迁移后，迁移分层在原频道内的传输将关闭，因此拥塞将得到立即响应。对于其他已预订了迁移分层的而未遇拥塞的用户，将通过快速的 Join 过程重新预订迁移分层。

3.2 分层迁移中的信息交换

在分层迁移方法中，发方和收方需要进行信息交换，迁移前，收方需向发方发送分层迁移请求；迁移后，发方需通知所有收方分层迁移情况。由于发方需要让所有用户了解分层迁移，因此理想信息交换方式是多播传输。由于收方也需要向发方传输请求，因此可采用一个双向共享多播树作为信息交换的通道，通过该双向共享多播树，无论发方还是收方均可自由地发送和接收信息。

信息交换的过程中，主要采用以下两种报文：

(1)Move_req 报文：由收方发出的迁移请求报文，报文说明请求迁移的分层号和该分层对应的多播组地址。

(2)Move_rep 报文：由发方发出的迁移通知报文，报文说明迁移的分层号和该分层迁移后的多播组地址。

另外，在信息交换过程中还必须考虑反馈抑制技术。当某一链路处出现拥塞时，共享这一链路的所有收方都有可能获得拥塞信号，可能在几乎相同的时间内向发方发出分层迁移请求信息，这可能导致发方出现反馈风暴(feedback implosion)，因此有必要采用适当的抑制技术。抑制方法可采用基于定时器的反馈抑制算法，即收方在获悉拥塞信号后，开启延迟定时器，在定时器未超时前，如果收到来自其他收方的迁移请求报文中的分层号等于其报文中的分层号，则放弃发送该请求迁移报文，并取消延迟定时器，否则在定时器超时后立即多播迁移请求。为实现有效抑制，可采用 J. Nonnenmacher[7]提出的定时器设置方案。每个接收者使用一个在区间$[0,T]$上的截尾指数分布定时器，其概率密度函数为：

$$f_{timer}(z)=\begin{cases}\dfrac{1}{e^{\lambda}-1}\times\dfrac{\lambda}{T}e^{\frac{\lambda}{T}z}, & 0\leqslant z\leqslant T\\ 0, & \text{otherwise}\end{cases}$$

在给定 λ 的情况下，调节 T 可控制反馈数目和反馈响应时间。T 越小响应时间越快，但随着收方数量的增多反馈数量也会变多。

3.3 分层迁移中信息交换的可靠性

分层迁移中的信息交换是采用不保证可靠性的多播传输，因此在传输中可能存在丢失，这种丢失会对分层迁移的正常运转产生很大影响，因此必须加以考虑。

(1)Move_req 报文丢失　Move_req 报文丢失可能导致发方或部分收方收不到 Move_req 报文，发方收不到 Move_req 报文，将不会进行分层迁移。为解决这一问题，需在收方设置重发定时器，定时间隔可设置为 $2*RTT+\delta$。在定时器超时前收到 Move_rep，则取消定时器。为防止出现极端的 Move_req 始终无法到达发方的情形，在发送第一个 Move_req 报文时，也向多播路由器发送 IGMP Leave 报文，这样将确保拥塞消除。

(2)Move_rep 报文丢失　Move_rep 报文丢失将会导致收方不知道迁移分层所在的频道，收方将无法进行迁移分层的预订，这将严重影响数据的正常接收，还会造成传输频道与空闲频道的混乱，导致数据接收的失败。因此，Move_rep 报文的传输必须确保让所有的用户均收到，这是一个可靠多播问题，并且这种可靠性必须在小于 IGMP Leave 的时延内达到，否则分层迁移就没有意义。显然，要实现这种快速的100%的可靠多播是很困难的。我们知道，如果这种可靠性无法保证，后果是用户不知道迁移分层所在频道，但如果预先将每个分层的迁移频道固定下来，这样即使收不到 Move_rep 报文也能知道分层的迁移频道，也就能重新加以预订，为此，可采用如下方案：

假定分层总数为 n，申请 $2n+1$ 个多播组地址 $G_i(i=0,1,\cdots,n,\cdots,2n)$，令 $G_i,i\in[l..n]$ 与 $G_{n+i},i\in[l..n]$ 互为分层 i 的对偶迁移频道，即当第 i 层数据在 G_i 中传输，迁移后数据改在 G_{n+i} 中传输；或者当第 i 层数据在 G_{n+i} 中传输，迁移后在 G_i 中传输。这样，当分层迁移后，若用户没有收到 Move_rep 报文，但由于分层迁移后原频道已停止发送数据，用户根据该分层100%的丢失率可确信该分层已进行迁移，这样他可发出 IGMP Leave 报文离开该频道，同时发出 IGMP Join 报文预订其刚离开频道的对偶迁移频道。

3.4 迁移的基本过程

为方便说明整个迁移过程，假定发方 S 将待传的数据流分为 n 个分层 $L_i(i=1,2,\cdots,n)$，其中 L_1 为基层，其余各层为增强层。同时，申请 $2n+1$ 个多播组地址 $G_i(i=0,1,\cdots,2n)$，其中 G_0 为迁移信息交换频道；$G_i(i=1,2,\cdots,n)$ 为初始分层播发送频道，发送数据流时，在一个组多播传输一个数据分层 $L_i\rightarrow G_i(i=1,2,\cdots,n)$；$G_i(i=n+1,\cdots,2n)$ 为相应的对偶空闲频道。

分层迁移过程是一个发方和收方在信息交换的基础上共同完成的，基本的迁移过程如图1所示，说明如下：

①收方在获悉拥塞信号后，通过信息交换频道 G_0 以多播的形式发送回包含其当前最高层号 i 和对应组地址 G_i 的迁移请求，即 Move_req 报文。

②发方在获悉迁移请求后，在 G_0 频道内播送迁移通知，即 Move_rep 报文，通知用户第 i 层数据 L_i 即将迁移到对偶频道 G_{n+i} 中传输。

③发方停止在 G_i 中传送第 i 层数据 L_i，同时开始在 G_{n+i} 中传输第 i 层数据 L_i。

④收方在收到迁移通知 Move_rep 后，根据以下情况确定是否预订迁移分层：

◇当前最大层号＜迁移层号：不预订 G_{n+i} 中的第 i 层数据。

◇当前最大层号＝迁移层号：分为两种情况：(a)有拥塞信号，则不预订迁移层数据；(b)无拥塞信号，则立即取消 G_i 的预订同时预订 G_{n+i} 中的第 i 层数据。

◇当前最大层号＞迁移层号：立即取消 G_i 的预订同时预订 G_{n+i} 中的第 i 层数据。

⑤收方未收到迁移通知 Move_rep，但觉察分层数据已发生迁移时，立即取消 G_i 的预订并预订 G_{n+i} 中的第 i 层数

据。

4 分层迁移在分层多播拥塞控制中的实现

在分层多播拥塞控制中使用分层迁移方法，发方将在获悉分层迁移后立即停止当前频道的分层传输并迁移该分层，拥塞将很快消除。特别是在网络极度拥塞的情况下，发方可以同时进行多个分层迁移，可很快解除网络拥塞，避免网络拥塞情况的恶化。在分层多播拥塞控制中使用分层迁移需要考虑拥塞信号和稳定性问题。

4.1 拥塞信号

拥塞信号在分层迁移方法中起着重要作用，拥塞信号将触发分层迁移请求，而在目前端到端的拥塞控制方案中，路由器并不提供任何拥塞信号，拥塞信号只有通过收方根据接收包的丢失来隐含说明。由于网络情况和预订层数的差异，会导致不同的组成员有不同的丢失率，如何确定拥塞信号与丢失率间的关系就成为一个涉及公平性和拥塞响应的关键问题。基于分层多播应用多为不需完全可靠性的应用，若针对每个丢失就立即取消预订的分层，无疑会严重影响传输质量分层；另外，不同分层也存在不同的丢失容许度。因此，可以考虑成员预订层数越高，越容易产生拥塞信号。具体的实现方法可采用 X. Li[8]提出的分层拥塞敏感度概念。拥塞信号由分层拥塞敏感度确定，而分层拥塞敏感度包括丢失率门限 l_r 和持续时间门限 t_r 两个参数，当下列两个条件满足时便认为产生拥塞信号：

①包丢失率＞丢失率门限 l_r；进入拥塞验证状态

②拥塞验证状态持续时间＞持续时间门限 t_r

丢失率门限 l_r 和持续时间 t_r 和各自当前预订层数的多少有关，预订层数越多，对拥塞就越敏感。可设 $l_r=L_b/r(i)$，$t_r=T_b/r(i)$，其中 L_b，T_b 分别为第1分层（基层）的丢失率门限和持续时间门限，而 $r(i)$ 为收方 r 当前的预订层数。

4.2 分层迁移的稳定性

拥塞消除的过程中，可能会出现这样的情况：当刚完成某个分层迁移后，又有收方发出对该分层进行迁移的请求，使得分层在两个对偶频道间频繁地迁移。频繁的分层迁移会影响传输的效果。为使分层迁移保持一定的稳定性，发方可引入一个迁移保持定时器，分层迁移完成后，立即启动迁移保持定时器，在定时器超时之前，对该分层的迁移请求暂时不作响应，使得分层在迁移后有一定的保持时间。迁移保持定时器超时后，对该分层的迁移请求作正常响应。

5 仿真结果

为说明分层迁移方法的有效性，我们通过网络仿真软件 ns 比较了分层迁移对拥塞的响应以及对 TCP 流公平性的影响。

图1是用于该仿真实验的网络拓扑，S1和 S2是流发送方，分别发送分层多播流和 TCP 流，分层多播流采用相等速率体制，每层速率为200Kb/s；TCP 则采用接收方收到每个报文就发送一个 ACK 的实现，发送窗口为50，包大小统一为1000字节。R1和 R2是路由器，采用 FIFO/droptail 调度策略，队列最大长度为30个包。H1和 H2为接收方，分别接收多播流和 TCP 流。

在实验中我们分别比较了具有较大 Leave 时延分层多播流和采用分层迁移多播流与 TCP 流的公平性问题。实验中不同流的公平性是采用瓶颈链路 R1-R2 上的有效吞吐率（Goodput）来说明的。图3为拥有较大 Leave 时延多播流与 TCP 流的比较图，其中 Leave 时延为2～4秒内的随机值。在实验初始阶段（0～10秒），由于有较小的 RTT 值，TCP 流在带宽的竞争中占据有利位置，多播流的有效吞吐率则随着分层数的增加在逐渐增大。随着丢失的出现（t＝12秒），网络开始出现拥塞，随着 TCP 对丢失的快速响应，其发送速率急剧下降，而多播流由于有较大的 Leave 时延，对丢失的响应比 TCP 慢得多，使得多播流在资源的竞争中占据了主导地位，多播流的有效吞吐率也在常时间内处于很高的水平（0.8），TCP 流的有效吞吐率则仅为0.2左右，无疑导致 TCP 流在资源共享中的极大不公平性。随着多播流对拥塞的响应，由于其加入实验间隔的增大，TCP 流才逐渐恢复正常。

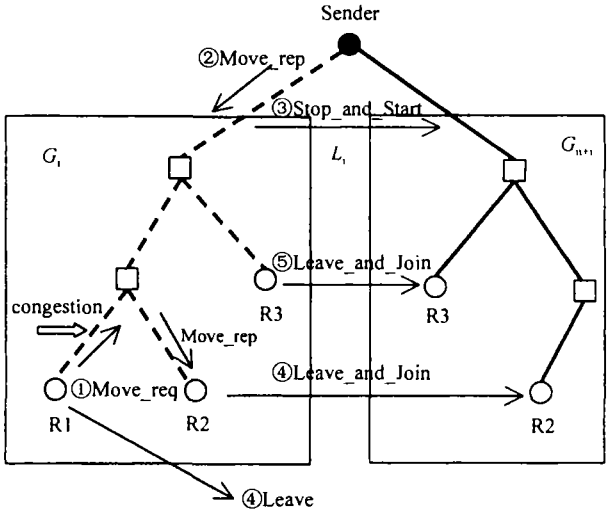

图1 分层迁移的基本过程

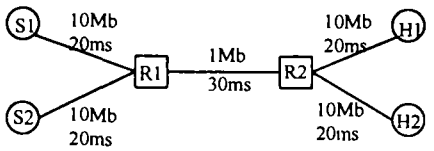

图2 用于仿真的网络拓扑结构

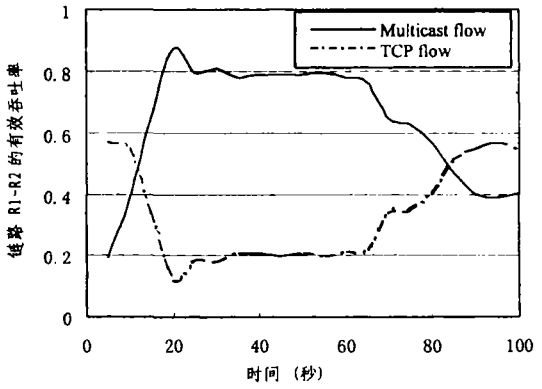

图3 具有较大 Leave 时延多播流与 TCP 流的吞吐率

图4为采用分层迁移多播流与 TCP 流的比较图。初始阶段的情况与图3相似，多播流在 t＝19秒时检测到网络拥塞，由于分层迁移能对拥塞实施快速响应，拥塞得以快速消除，导

（下转第82页）

免其他客户同时访问或更新这个文件以致破坏文件的一致性;③ 返回文件的元数据;4、应用服务器取得元数据之后,以Block I/O 的方式直接向存储池中相应的存储设备发出Block I/O 请求;5、存储设备完成相应的读写操作,读写数据块并返回给应用服务器;6、应用服务器上的文件系统将得到的数据块组合成文件,并通过 NFS/CIFS 提供给 USN File Client 使用。

基于 SAN 的 USN 融合了 NAS 和 SAN 技术的优点但又克服了各自的缺点:在客户机看来,由于访问方式采用的是NAS 的文件 I/O 的方式,整个存储系统相当于一个 NAS 文件服务器,因此具有 NAS 的一些优点:如可以提供异构环境下的文件访问和文件共享,适应复杂的网络环境等。另外,文件系统和存储子系统之间采用 SAN 的结构,因而在可扩展性、可用性、资源整合、集中管理等众多方面具有 SAN 的优点。和基于 NAS 的 USN 不同,在这种体系结构中,控制信息和数据信息具有不同的通道,因此避免了可能的性能瓶颈。虽然整个系统只有一个元数据服务器,但由于只是处理元数据,因此不会对整个系统的性能构成太大的影响。当然,也可以采用冗余措施来降低元数据服务器的负载或提高系统的可用性。

除提供 File I/O 服务,USN 还可以利用 iSCSI 技术提供Block I/O。其原理和基于 NAS 的统一存储网络相同,具体的数据读写过程如图3、5所示。

总结 融合 NAS 和 SAN 技术的统一存储网以其优异的性能吸引了越来越多的注意力。目前的研究热点就是将NAS/SAN 各自的优点整合到一个系统中,使得单一系统同时具有 NAS/SAN 两者的优点而克服各自的缺点。首先应该具备 SAN 的优点:在服务器和存储设备之间采用高速的Block I/O 方式或具有 Block I/O 的性能;存储设备与存储设备之间采用 LAN-free 的方式备份数据;可以对存储资源进行集中管理,利用存储虚拟化技术,对存储空间进行整合,形成一个统一的存储池……;其次要具备 NAS 的优点:能够通过局域网(IP 网络)、以标准的 NFS 和 CIFS 协议在异构的环境下实现文件级的访问与共享;适应复杂的网络环境;构建、维护、管理、使用简单方便;降低总拥有成本。所以理想中的存储网络就应该统一在现有的 IP 网络的基础上,使得任何平台下的客户在得到授权的情况下都可以在任何时候,从任何地点访问放在异处的信息。

参 考 文 献

1 IBM redbook. IP Storage Networking: IBM NAS & iSCSI Solutions

2 Satran J. et al. iSCSI. http: // search. ietf. org/internet-drafts/draft-ietf-ips-iscsi-09. txt

3 (美)Marc Farley 著,孙功星等译. SAN 存储区域网络. 机械工业出版社

4 谢希仁编著. 计算机网络. 电子工业出版社

5 Brocade whitepaper: comparing the Storage Area Network and Network Attached Storage

6 Winter corporation whitepaper: Scalable Networked Storage: the convergence of Network Attached Storage and Storage Area Network with the Highroad

7 HTACHI whitepaper: Planning for Network Network Attached Storage and Storage Area Network Convergence

(上接第71页)

致 TCP 流能随即增加发送速率。因此 TCP 流的有效吞吐率随着速率的增加也相应增加,而多播流则随着分层的取消其有效吞吐率也相应降低。在 t=60时,多播流和 TCP 流已趋于稳定状态。对比图3和图4,可以看出采用分层迁移的多播流较有较大 Leave 时延多播流更能给 TCP 流提供好的公平性,还有较快的聚合速度。

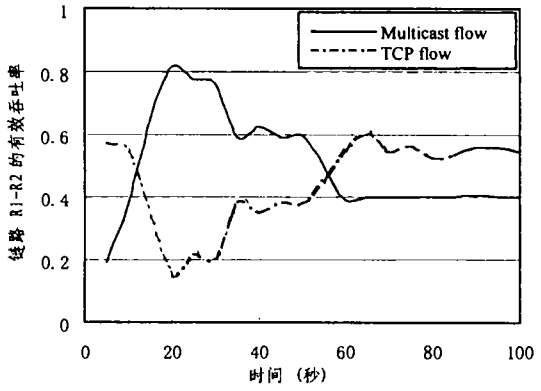

图4 采用分层迁移多播流与 TCP 流的吞吐率

结束语 针对 IGMP Leave 时延过大导致分层多播拥塞控制方案存在拥塞响应慢的问题,本文提出了分层迁移的概念和方法。分层迁移方法用快速的 Join 过程替代慢速的 Leave 过程,不仅解决了拥塞响应慢的问题,而且无需对现有的 IGMP 协议、路由器以及多播路由协议作任何修改。仿真结果不仅显示了分层迁移方法的可行性和有效性,还表明它能有效改善分层多播流对 TCP 流的公平性。

参 考 文 献

1 McCanne S, Jacobson V, Vetterli M. Receiver-driven layered multicast. In Proc. ACM SIGCOMM, Stanford, CA, August 1996. 117～130

2 Deering S. Host extensions for IP multicasting. RFC 1112, August 1989

3 Fenner W. Internet group management protocol, version 2. IETF RFC 2236, January 1997

4 Vicisano L, Rizzo L, Crowcroft J. TCP-like congestion control for layered multicast congestion control scheme. In SIGCOMM 2000, Stockholm, August 2000

5 Rizzo L. Fast group management in IGMP. In Hipparch Workshop, London, June 1998. 32～41

6 Byers j, et al. FLID-DL: Congestion for layered multicast. In: Proc. 2nd Int'l Wkshp. Networked Group Commun., Palo Alto, CA, Nov. 2000

7 Nonnenmacher J, Biersack E W. Optimal multicast feedback. In: Proc. IEEE Infocom, San Francisco, USA, March 1998

8 Li X, Paul S, Ammar M. Multi-Session Rate Control for layered Video Multicast. In: Proc. of Symp. on Multimedia Computing and Networking (MMCN'99), (San Jose, CA), Jan. 1999