

IP 组播下的视频点播实现技术

A Video-on-Demand Technology Based IP Multicast

梁建民 祝明发

(中国科学院计算所 北京100080)

Abstract Traditional VOD provides interactive on-demand video service by allocating each users a dedicated video stream, such system can not accommodate a large number of users. In this paper, we propose a Video-on-Demand technology based on pre-buffering, channel-switch and IP multicast which is lower bandwidth requirement and can provide full interactive operation.

Keywords Video-on-demand, Interactive operation, IP multicast, Pre-buffering, Channel-switch

一、引言

与电视不同,视频点播(VOD)可以让用户自由地选择播放的节目,进行任意的交互操作,并立即得到响应。VOD是很有前途的一种宽带业务,但由于带宽的限制,VOD系统只能在小范围内应用。随着因特网的发展,ADSL、Cable Modem等宽带接入技术的出现,在广域网实现高质量的VOD业务将成为可能。

对于因特网来说,最宝贵的资源就是带宽,而对于VOD系统的实现来说,如何有效地利用信道资源也是最关键的。视频数据具有信息量特别大的特点,传统的VOD系统只支持视频服务器和用户之间的单点连接,为每个用户都提供一条单独的高速信道,对于mpeg-1和mpeg-2压缩方式,分别需要1.5Mbps和3Mbps的传输速率,带宽的占用必然会使用户的使用价格增加,同时太多的连接数也会浪费系统资源,使视频服务器能负荷的用户很有限。因特网也存在同样的问题,TCP/IP数据传送方式主要有两种,一种是单播传送(unicast delivery),一种是广播传送(broadcast delivery)。这两种方式对于一些特定的应用是不合适的,如IP会议,如果用单播传送,需要为每个用户都发送一份数据拷贝;如果用广播方式实现,则可能有大量主机收到与自己无关的数据,都会造成资源浪费,这些都促成了IP组播通信方式的发展。

IP组播的最重要特点就是可以实现一对多的通信方式,接收端是动态的,主机可随时加入或退出,发送端不需要知道接收端的地址。IP组播依托于IP协议完成,强制网络在数据传递树的分叉处进行信息包复制,而不是由发送端多次重复地发送同样的信息包。1988年,斯坦福大学实现了第一次多路通话,1989年,IP组播的详细说明问世,1992年,IETF定义和发布了一个IP组播的网络标准,用于建立组播主干网(MBONE),即在Internet上运行的单路广播和IP组播综合网络,这个平台构建在原有的Internet设备上,由支持IP组播的子网和路由器组成,同时允许IP组播通信量通过无IP组播能力的Internet部分,从而在整个Internet范围内对IP组播开放,IP组播成为人们关注的焦点。

二、IP 组播技术

IP组播技术主要包括两方面的内容:主机组模式和IP

组播路由。主机组模式就是确定具有IP组播发送、接受、组成员管理的主机,通常由一个D类地址(224.0.0.0~239.255.255.255)来标识,使用相同组播地址的所有主机构成一个主机组。主机组是开放的,任一主机均可自由加入或退出,不需要知道组内其他主机是否存在,这也符合IP会议的特点。主机组建立后,IP组播路由算法便对网络进行剪枝,将不存在的组成员的分枝剪去,形成一个暂态的IP组播树,信息就在这棵树上从源向所有的接收端流动。IP组播通信需要指定发送分组的生存期(TTL),当一台主机同时有多个网络接口时,还需要指定发送分组的网络接口。

IP地址为静态绑定在网卡上时,而组播地址为动态绑定在不同IP网中的不同网卡,路由器不需要知道每个组的主机列表,只需要知道该子网是否有加入某个组的主机。

组播数据报包含TTL域,控制数据报最大跳数,每经过一个路由器,TTL域减1,如TTL为0则丢弃数据报,不给发送者错误提示,这种机制防止数据报不必要的传送到超出指定范围但有相同组地址的主机。子网内TTL为1,如TTL大于1,由组播路由器负责转发到由目标组编号的网络,组播路由器可带有TTL控制阀,以阻止小于一定TTL的包进入子网。

组播路由定期给子网内所有主机发送组成员资格查询,让其报告所拥有组成员资格。如果所有主机回应报文同时发回,可能造成网络拥塞,可以通过让每台主机随机延迟一段时间,当没有看到从另一台主机发来的相同组的报告时再回应。因此同一子网中,对每个活动的组地址,尽管可能有多台主机具有组成员资格,但只收到一个回应报文。

组播地址标识了一个特殊的传送会话路径,而不是一个物理地址。最简单的传送数据到多个接收点的方法是发送源维护一个包括所有参与会话者子网地址的表并分别发送数据拷贝到每个接收子网,但这种方式很浪费带宽,因为许多数据流传送时可能沿相同的路径。因此发送源没有必要知道所有相关地址,而由中间路由器完成从组播地址到主机地址的转换,多个组播路由器相互交换相邻路由信息。由于组播组是动态的,因此路由信息要不断更新。

三、基于IP 组播的VOD技术

3.1 存在的问题

梁建民 博士生,主要研究方向:基于Linux集群的视频服务。祝明发 博士生导师,主要研究方向:高性能服务器。

正如我们收看不同电视节目可以选择不同的频道,视频服务器也可以在不同的端口传输不同的视频流,我们这里称之为频段。我们把系统的数据流分为两种:视频数据流和交互数据流。视频数据流又可以分为单点视频流和多点组播视频流,单点视频流采用点对点的方式,为单个用户开辟单独的信道;多点组播视频流采用 IP 组播的方式发送视频数据,可以同时被多个用户共享;交互数据流只针对单个用户的交互需求,采用点对点方式。我们这里称传输多点组播视频流的频段为 MV 频段,传输单点视频流的频段为 SV 频段,传输交互数据流的频段为 I 频段。

如果把视频以 IP 组播形式发送,就可以实现资源共享,但这样无法给用户 VCR 交互功能。只是简单地采用同步缓存预取和 IP 组播结合方式也只能给用户有限的 VCR 交互功能,真正的 VOD 应该能使用户在任何时间,点播任何节目,完成任意的交互操作。

另一种解决办法是,为所有的用户交互操作都分配 I 频段,用户交互结束播放恢复时,VOD 服务器尽力把用户合并到一个 MV 频段中,通过同步缓存实现用户和 MV 频段同步,如果同步失败,就申请一个新的 MV 频段。这种方式所存在的问题是 MV 频段按照需求分配会使系统中 MV 频段太多,资源占用过大,整个系统性能降低,在接受新请求和提供交互操作服务方面都缺少可扩展性,因此 MV 频段的分配应该和用户的交互行为分离。其次为所有的用户交互操作都分配 I 频段也不合理,很多交互操作可以通过同步缓存预取和频段切换实现,并不需要分配 I 频段。

还有一种情况是,为使 MV 频段的共享度增加,用户的播放请求被强制等待一段时间以合并到 MV 频段,这种方法既缺少可扩展性也容易引起无法控制的用户服务质量下降。

基于以上问题,我们提出一种技术通过频段切换(MV 频段、SV 频段和 I 频段之间)和同步缓存预取实现在低带宽占用下给用户提供完全的交互功能。基本原则如下:

(1)系统 MV 频段的分配和用户的交互行为分离,由系统根据用户请求分布优化的分配。

(2)用户的播放请求到达时,如果没有合适的 MV 频段,系统给播放请求临时分配 SV 频段。

(3)系统利用同步缓存预取和 SV 频段,使播放请求合并到 MV 频段中,并释放临时的 SV 频段。

(4)用户的交互请求到达时,首先尝试用同步缓存和 MV 频段实现,无法实现则为用户分配 I 频段。

(5)如果系统没有 I 频段可以分配,用户的交互请求进入等待队列,正常播放继续。

(6)用户结束交互操作后,通过同步缓存预取使用户合并到正在进行的 MV 频段。

(7)如果(6)失败,不为用户分配新的 MV 频段,而是为用户分配临时的 SV 频段,同步缓存开始预取,等待下一次合并。

(8)系统通过频段切换结合同步缓存,给用户提供完全的交互功能。

3.2 频段的分配

系统中 SV 频段和 I 频段根据需要分配。I 频段在同步缓存和 MV 频段无法满足用户交互需要时分配,如果系统没有 I 频段可以分配,用户的交互请求将被阻塞,放入等待队列,正常播放继续。如果 I 频段可以分配,则用户进入交互模式。SV 频段在下面两种情况下分配:1)用户初始一个播放请求,

系统中没有合适的 MV 频段。2)用户从交互模式切换到播放模式,暂时无法合并到合适的 MV 频段。系统为用户分配临时 SV 频段的同时,用户开始缓存 MV 频段数据,SV 频段持续到用户缓存中数据包含 MV 频段的播放点,然后用户就可以释放 SV 频段,合并到 MV 频段中。和正常播放相比,用户交互操作只会持续很短的时间,I 频段的分配不会影响系统性能。

MV 频段的分配应和用户的交互行为分离,分配策略可以分为两种:一种由服务器按照某种规则生成不同的频段进行视频流 IP 组播,用户加入相应的 MV 频段接收所需的视频流,服务器分配频段和用户的请求无关,这种称为“推”方式;另一种是服务器根据用户的请求分配频段,如果几个请求的目标是同一个视频对象,则可以合并成为一个批请求,这种称为“拉”方式。

由于用户的请求随视频内容、时间的不同都是在变化的,“推”方式的平均主义无法充分地利用系统资源。“拉”方式把几个相同视频对象请求合并成为一个批请求,节省了信道资源,但这样意味着单个用户的交互操作也会影响到同一组播频段的所有用户,所以我们必须使这种共享对用户是透明的,也就是说,用户交互操作和正常播放应该采用不同的频段,这也是我们单独设置 I 频段的原因。

MV 频段的分配应该是动态的,即系统中对于同一视频的 IP 组播间隔是根据用户的请求自适应变化的,对特定视频流的请求增加时,系统会为这个视频产生更多的 MV 频段,对特定视频流的请求减少时,MV 频段也相应减少,在用户请求的视频流比较集中时,系统效率较高。

假定为在频段 MV1 开始 W_i 分钟后到达的用户分配新的频段 MV2,在 W_i 分钟之间到达用户由 SV 频段提供电影缺少的部分以合并到频段 MV1 中,则 SV 频段持续的时间等于用户播放请求到达和 MV2 频段开始的时间差。如果 W_i 太长,则需要的 MV 频段数目变多;如果 W_i 太小,则需要的 SV 频段数目变多,应该有一个优化的 W_i 使 MV 频段和 SV 频段的数目和最小。

这里我们设每个用户的同步缓存可以容纳 B 分钟的视频节目,设 T_i 是电影的平均长度, N_e 是系统的不同电影数目, $N_{m,i}$ 和 $N_{s,i}$ 分别是不同电影的 MV 频段和 SV 频段的个数,其中 i 是指第 i 部电影,这里有 $i=1,2,\dots,N_e$,用户点播电影 i 的概率为 V_i ,显然有 $\sum_{i=1}^{N_e} V_i = 1$ 。假设 VOD 系统用户的请求速率为 K 个请求/分钟,则第 i 部电影的请求率是 $\lambda_i = KV_i$ 个请求/分钟。设 $N_{c,i}$ 是第 i 部电影的 MV 频段和 SV 频段的总数,有 $N_{c,i} \triangleq N_{m,i} + N_{s,i}$ 。我们需要合适的 W_i 使系统 MV 频段和 SV 频段的数目和最小。

$$W_i \leq B \text{ 时有 } \min_{W_i} (N_{c,i} \triangleq \sum_{i=1}^{N_e} N_{c,i}) \quad (1)$$

由以上的假设可以得出平均有 $1 + \lambda W_i$ 个用户共享一个 MV 频段,其中 1 是指首先到达并开始一个 MV 频段的用户,其它 λW_i 个用户则首先由 SV 频段提供服务,因此 SV 频段的请求率为 $\frac{\lambda W_i}{(1 + \lambda W_i) \lambda_i}$ 。由于每个 SV 频段的平均持续时间为 $W_i/2$ 分钟,我们可以得到公式(2):

$$N_{s,i} = \lambda_i \left(\frac{\lambda W_i}{1 + \lambda W_i} \right) \frac{W_i}{2} \quad (2)$$

由以上的假设可以得出 MV 频段分配的时间为每隔

W_{i+1}/λ 分钟,我们得到公式(3):

$$N_{m,i} = T_b / (W_i + 1/\lambda) \quad (3)$$

由公式(1)、(2)、(3)求 W_i 的最小值,得到下面的公式(4),也就是使系统 MV 频段和 SV 频段的数目和最小的 W_i 。

$$\min_{W_i} = (\sqrt{1 + 2\lambda T_b} - 1) / \lambda \quad (4)$$

3.3 用户的同步缓存

同步缓存可以看作是一个在可用的视频序列上的滑动窗口,是系统的重要组成部分。每个用户从交互模式恢复到正常播放时需要合并到一个 MV 频段,有可能播放点在时间上和原来的 MV 频段有偏移,比如向前跳转 n 分钟的操作要求一个比原来的 MV 频段向前 n 分钟的 MV 频段,有可能这样的 MV 频段不存在,系统将尝试通过正在传送的 MV 频段和同步缓存产生一个虚拟的 MV 频段。首先系统找到正在播放的 MV 频段中和虚拟 MV 频段偏移量最小的,用这个 MV 频段的内容填充同步缓存,而虚拟 MV 频段在所需的时间偏移后从缓存中读取内容。

同步缓存是环形的,有两个操作位置,一个用于填充视频数据,一个用于读取视频数据。只要同步缓存有足够的空间,系统就可以利用同步缓存生成虚拟的 MV 频段实现资源共享。

同步缓存可以在服务器端实现,由用户部分共享,也可以在用户的接收装置上单独配置。部分共享的同步缓存对于合并操作是具有排它性的,有可能导致阻塞和 VCR 操作的延迟,用户端缓存则可以提供更好服务质量保证和 VCR 操作的可扩展性。虽然部分共享缓存是更便宜的选择,但分布管理非共享的缓存更适合于大规模的应用,服务提供者扩展时不需要另外的缓存,带宽,处理器,缓存的花费分摊到用户中。

3.4 合并操作

为使系统不会因为非共享模式导致性能下降,用户从交互操作恢复播放时应尽力合并到一个在播放的 MV 频段,合并操作的基本功能就是寻找合适的目标频段。这里我们称用户正在读取帧的位置称为播放点,在同步缓存中用户已经播放过的帧称为过去帧,在同步缓存中用户还没有播放过的帧称为未来帧,在同步缓存中用户刚刚预取的帧称为最新帧。

合并时,有可能目标频段的播放点已经在同步缓存中,如暂停、停止、慢放等操作,这种情况合并操作可以立刻实现。当用户已经合并到目标频段时,有两种处理方式。第一,同步缓存丢弃在目标频段播放点之前的所有帧,以便未来帧和目标频段同步。第二,也有可能同步缓存丢弃目标频段传送的帧,因为它已经在同步缓存中了,同时向前移动播放点位置,丢弃帧留出的缓存空间可以保存过去帧。如果有多个频段都适合合并操作,只能由同步缓存随意选择一个,如果所选频段的播放位置接近用户的播放点,大部分预取的帧将被丢弃,留出更多的空间容纳过去帧。如果目标频段播放点接近同步缓存存储的最新帧,大部分帧在合并操作后将继续保留。

第二种合并发生在目标频段的播放点还没有被预取,由于同步缓存保存的未来帧很少,合并操作的持续时间对应于目标频段的播放点和同步缓存中最新帧的偏移量。合并操作由三个步骤组成:1)通过丢弃过去帧留出足够的同步缓存空间。2)给用户分配 SV 频段同时从目标频段预取帧。3)当目标频段的播放点已经预取到同步缓存中时,用户合并到目标频段并释放 SV 频段。

合并操作可能由于以下原因失败:第一,找不到一个合适

的 MV 频段,播放通过 SV 频段完成。第二,用户可能初始化另一个 VCR 操作,打断合并操作。两种情况下,都会改变播放点在同步缓存中的位置。

3.5 VCR 操作实现分析

VCR 操作包括快进/后退、暂停/恢复、随机定位等。VCR 交互操作包括以下内容:(1)通过缓存预取结合频段切换实现 VCR 操作;(2)当同步缓存无法满足用户交互请求时,分配 I 频段来完成交互操作。如果没有 I 频段可分配则用户交互请求进入等待队列;(3)用户退出交互模式时合并到合适的 MV 频段。这里我们称 MV 频段正在传送帧的位置为组播位置,如果 MV 频段的组播位置和用户的当前播放点相同,则称该组播位置为该 MV 频段的播放点。

3.5.1 快进/后退、随机定位操作 假设用户正在播放模式,播放点在 t_0 ,当用户快进/后退时间 d 时,播放点到了 t_1 ,如果同步缓存大于 $t_{ff} = t_1 - t_0$,则只需要在同步缓存中修改一下用户的播放点,否则需要在多个视频流中选择一个组播位置和用户当前播放点最接近的 MV 频段,用户合并到该频段。随机定位和快进/后退操作类似。

3.5.2 暂停/恢复 设每个用户的同步缓存可以容纳 B_i 分钟的视频节目,由 MV1 频段为用户提供视频流 i ,当暂停发生时,MV1 频段填充同步缓存,在暂停结束后,用户将合并到一个最接近的频段 MV2 中。假设暂停持续的时间为 d ,MV2 频段的播放点和用户暂停前的播放点的时间偏移为 t ,可能有以下几种情况:

(1) $t \leq B_i, d < t$,即 $d < t \leq B_i$,即当用户从暂停恢复时同步缓存还没有被改变,用户可以简单地从同步缓存中读取数据。 $d \geq t$,意味着 MV2 频段的组播位置在暂停操作终止前已经到达播放点,用户只能首先清除同步缓存,从 MV1 频段中分离, MV2 频段开始填充同步缓存,分配 SV 频段为用户服务,直到用户通过同步缓存和 MV2 频段实现同步,释放 SV 频段,合并到 MV2 频段。

(2) $t > B_i, d < t$,暂停操作在 MV2 频段的组播位置到达播放点前结束,系统暂时为用户分配一个 SV 频段,SV 频段持续到 MV2 频段的组播位置到达播放点。 $d \geq t$,暂停操作在 MV2 频段的组播位置到达播放点之后结束,用户首先清除同步缓存,从 MV1 频段中分离, MV2 频段开始填充同步缓存,分配 SV 频段为用户服务,直到用户通过同步缓存和 MV2 频段实现同步,释放 SV 频段,合并到 MV2 频段。

结束语 基于用户端的同步缓存使 VOD 系统具有更好的可扩展性;频段切换和同步缓存预取等技术的应用,在给用户提供完全的交互功能的同时可以有效减少系统资源占用;MV 频段根据用户请求率动态分配,和用户的交互行为分离,选择合适的组播间隔 W_i ,系统性能得到进一步优化。

现代的社会是网络高速发展的社会,宽带网开始走入千家万户,IP 组播技术的研究也趋向成熟,为 VOD 在因特网的应用提供了良好的基础。基于 IP 组播的 VOD 技术方兴未艾,许多领域还有待我们进一步研究。

参考文献

- 1 Hua K A, Sheu S. Patching: A Multicast Technique For True Video-on-Demand Services. ACM Multimedia, Sept. 1998
- 2 Shenoy P J, Goyal P, Vin H M. Issues in Multimedia Server Design. ACM Computing Surveys, Dec. 1995
- 3 Dan A, Sitaram D, Shahabuddin P. Dynamic Batching Policies for an On-Demand Video Server. In Multimedia Systems, 1996, 4: 112 ~ 121
- 4 梁建民,等.基于分布式网络和推的视频点播实现技术.计算机应用研究,2000. 4