CORS TV:一种基于网络编码的P2P TV系统

张志明[1,2] 杜 剑[1,2] 郭 瑛[1,2] 周 晋[2] 陈 震[2,3] 李 军[2,3]
(清华大学自动化系 北京 100084)[1] (清华大学信息技术研究院 北京 100084)[2]
(清华信息科学与技术国家实验室 北京 100084)[3]

摘 要 网络编码可以实现组播的最大吞吐率,若应用于P2P TV系统,具有降低用户播放延时、提高系统有效传输率,从而提高视频质量的潜力。为了提高P2P TV系统的性能,设计并实现了一个基于随机线性网络编码的P2P TV系统——CORS TV。围绕拓扑构建和数据传输这两个关键部分,充分利用网络编码提高系统性能,CORS TV具有功能模块线程隔离的节点内部结构,集成了基于Gossip协议的拓扑构建算法,使用了最多者优先的初始播放点设置算法,并利用了基于推的数据传输方案和先到先得式的数据传输算法。在计算机集群上的实验验证了CORS TV系统设计的正确性和有效性。与已有P2P TV系统相比,该系统具有降低冗余率、提升系统的有效传输率、改善用户视频播放质量的优势。

关键词 计算机网络,P2P TV,网络编码

CORS TV:A Network Coding Based P2P TV System

ZHANG Zhi-ming[1,2] DU Jian[1,2] GUO Ying[1,2] ZHOU Jin[2] CHEN Zhen[2,3] LI Jun[2,3]
(Department of Automation, Tsinghua University, Beijing 100084, China)[1]
(Research Institute of Information Technology, Tsinghua University, Beijing 100084, China)[2]
(Tsinghua National Lab for Information Science and Technology, Beijing 100084, China)[3]

Abstract Network coding can achieve maximum throughput of multicast, which shows the potential to decrease playback delay of users in P2P TV system, to improve delivery ratio, and therefore to improve video quality of the whole system. In order to improve the performance of the P2P TV system, a system based on Random Linear Nework Coding (RLNC), called CORS TV, was designed and implemented with carefully designed topology construction and data transmission. Full taking advantage of network coding and further improving the performance of P2P TV, CORS TV has a modular architecture, and each function module runs an independent thread. The system integrates Gossip protocol for topology construction, uses most-first algorithm for determining initial playback point, and employs push based transmission scheme with First Come First Served(FCFS) transmission algorithm. The experiment results validate the effectiveness of the CORS TV design. Compared with current P2P TV systems, it has the advantages of lower redundancy ratio, higher delivery ratio and better playback quality of users.

Keywords Computer networks, P2P TV, Network coding

1 引言

近年来,随着宽带接入的普及,P2P TV得到了快速发展[1-4]。P2P(Peer-to-Peer)技术[5]可以利用正在获得服务的用户的上行带宽和计算资源为其他用户提供服务,并且随着用户规模的增长,整个系统的可用上行带宽和计算资源也随之增加。这可以弥补服务器相关资源的不足,从而降低服务提供商的运营成本,或者在服务器资源一定的情况下,扩大系统的容量规模。因为视频传输所需要的带宽较大,所以当P2P技术成熟后即被迅速应用到了视频传输领域,出现了P2P TV系统。

现有P2P TV系统仍存在不足。从用户角度来讲,主要是频道之间的切换延时较大,播放延时过大,观看同一直播内容的P2P TV用户之间会出现明显的不同步现象。播放的视频会出现马赛克、画面跳跃和播放暂停等现象。研究人员为此做了大量工作来提高这些问题[6-9]。

在P2P TV被提出、改善和部署的同时,网络编码被提出,以实现组播的最大吞吐率[10,11],并被应用到了P2P TV系统中。经过几年的发展,网络编码逐步从理论走向实用,并显示出改善P2P TV系统延时和播放质量的潜力。网络编码

到稿日期:2011-01-08 返修日期:2011-04-04 本文受NEC中国研究院资金资助。

张志明(1979-),男,博士生,主要研究方向为P2P流媒体,E-mail:zzm02@mails.tsinghua.edu.cn;杜 剑(1987-),男,硕士生,主要研究方向为P2P流媒体;郭 瑛(1987-),女,硕士生,主要研究方向为网络编码;周 晋(1977-),男,博士,主要研究方向为结构化P2P系统、P2P流媒体;陈 震(1976-),男,讲师,主要研究方向为高速网络、P2P系统、可信计算;李 军(1962-),男,研究员,主要研究方向为网络安全、网络体系结构。

已被应用于 P2P TV 系统，以缩短初始缓冲延时，提高播放质量，增强系统抵御节点动态性对系统的冲击能力等[12,13]。然而，现有基于网路编码的 P2P TV 系统的研究还不完善。例如，当系统中用户可获得的上行带宽不足时，就会出现有些编码数据包因其所在的段不能解码而被迫丢弃，造成数据包的冗余率(Redundancy Ratio)上升，有效传输率(Delivery Ratio)下降，用户的播放质量(Playback Quality)变差。其中冗余率定义为未被用来播放的已收数据包数量与收到的所有数据包数量的比值。有效传输率定义为被用来播放的数据包数量与所有原始数据包数量的比值。用户的播放质量是指用户看到的视频质量，这里使用有效传输率来计算。

本文将随机线性网络编码引入到 P2P TV 系统中，设计并实现了一个基于网络编码的 P2P TV 系统——CORS TV。CORS(Cooperatively Overlay Routing Service)[14]是一个用来提供覆盖层路由服务的原型系统。本文对 CORS 系统进行扩展，实现了一个可以提供实时流媒体服务的子系统，命名为 CORS TV。实验室计算机集群上的实验表明，与其它类似系统和未使用网络编码的系统相比，CORS TV 可以降低系统冗余率，提高系统有效传输率和播放质量。

本文第 2 节介绍相关工作；第 3 节介绍 CORS TV 系统设计；第 4 节通过计算机集群上的原型系统实验验证 CORS TV 系统的性能。

2 相关工作介绍

基于 P2P 技术的流媒体传输协议主要包含拓扑构建和数据传输两个部分[15]，现有工作主要从这两个方面改进 P2P TV 系统。拓扑的类型主要有树状拓扑和网状拓扑。实际系统中，节点的上下线会对树状传输拓扑造成严重冲击。新节点上线时，根节点需要根据整棵组播树当前的情况将该节点加入到合适的位置。节点退出时，该节点的子节点需要马上找到一个新的节点作为自己的父节点。当系统规模比较大时，组播树的维护就变得非常困难。所以现在的商业系统中没有使用树状拓扑，而都是网状拓扑。网状拓扑系统中的数据传输方案有推、拉和推拉结合等。

受到 P2P 文件共享系统的启发，CoolStreaming[2]将树状拓扑替换为基于 Gossip 协议的网状拓扑，来增强系统应对节点动态性的能力。系统中，节点之间没有父子关系，只有邻居关系，每个节点以闲聊的方式与自己的邻居交换各自的信息。因为每个节点的邻居都比较多，所以节点的加入和退出对邻居造成的冲击会小得多。每个节点都主动向邻居请求自己所缺的数据，将数据拉过来，而不是像树状网络中那样由父节点主动推送给自己。这种方案有效地降低了节点的动态性对系统的覆盖层网络造成的冲击，但它却增加了数据的传输延时，并使得数据传输问题变得非常复杂。已有文献证明[2]，该类系统中的数据传输问题是一个 NP 难问题，因此研究人员提出了各种启发式传输算法[15]来优化数据的传输。针对传输延时大的问题，研究人员提出了推拉结合的数据传输方案[16]，将视频数据包序列划分为若干子流，如果某个子流中的一个数据包通过拉的方式传输，则该子流的后续数据包就使用推的方式发送给相应的邻居节点，直到有丢包、断网等异常情况发生为止。

网络编码具有降低数据传输问题的难度，进而改善 P2P TV 系统播放质量和延时等性能的潜力。它的主要思想是中间节点不只进行存储转发，也参与编码过程。Y. -h. Chu 等[17]已经证明，使用简单的线性网络编码，就可以实现组播的最大吞吐率。Y. -h. Chu 等[18]进一步提出了有限域[1]上的随机线性网络编码，使得网络编码向实际应用迈出了重要的一步。C. Gkantsidis 等[19,20]首先将随机线性网络编码应用于文件下载系统，其实验结果表明，网络编码可以有效提高系统性能。M. Wang 等[12]将随机线性网络编码应用于 P2P TV 系统，使用了基于拉的数据传输方案。计算机集群上的实验表明，当系统上行带宽容量略超过所需上行带宽容量时，网络编码可以提升系统的有效传输率和播放质量。其他研究人员也做了相关研究[21,22]，并引入了 MDC 来解决节点的异构性问题。M. Wang 等[13]进一步用推的传输方案代替了拉的传输方案，实现了一个原型系统 R^2。计算机集群上的实验表明，网络编码可以提升系统中节点的播放质量，提高系统鲁棒性，更好地抵御节点加入和退出对系统造成的冲击，减少服务器的带宽开销，缩短节点的初始缓冲时间等。与此同时，其它研究人员也做了类似研究[23-25]。

现有基于网络编码的 P2P TV 系统还存在不足。Lava[12]专注于公平比较，且传输延时偏大。R^2[13]在使用随机线性网络编码时，每个段中所包含的数据包过多，导致编码系数的带宽开销过大。虽然可以使用伪随机种子来代替编码系数，减少带宽开销，但这种方案需要每个节点收齐段中所有数据包，并解码后才能使用，大大增加了传输延时，失去了推送方案的优势。而且 R^2 中所有节点的播放点同步限制了系统的规模。因为数据包的传输需要时间，数据包传输的跳数越多，所花费的时间越长。当传输的时间超过缓存的长度后，系统规模也就达到了极限。受控推[23]的缺点是，节点在发送数据包为邻居服务时过于保守，不能有效利用节点的上行带宽，且增加了传输时延，使得推送模式的优势无法有效显示出来。SonicStream[24,25]的缺点是，系统的多种参数设置得有失合理性，如邻居数量太少(3 到 5 个)、缓存的长度过短(5s)等。在正常的参数设置下，系统性能如何，还有待进一步检验。在现有工作中，网络编码传输算法的设计都过于简单，虽然网络编码可以简化传输问题的难度，但传输算法仍需要精心设计。例如，随机线性网络编码需要将连续的数据包序列划分为一系列的段，并以段为单位进行编码传输，每个节点只有收齐段中原有数量的线性独立数据包后才能解码。当系统上行带宽不足时，会有大量段没有收齐，不能解码，导致已收到的数据包被迫丢弃，造成数据包冗余率上升和系统有效传输率下降。

3 CORS TV 系统设计

CORS TV 系统将随机线性网络编码引入到 P2P TV 系统，并采用推式数据传输方案，使得系统获得更低的传输时延。系统的核心设计思想是充分发挥网络编码的优势，避免或尽量降低其缺点对系统造成的冲击。系统设计主要包括系

1) 又称伽罗瓦(Galois)域，即包含有限多个元素的域，这里指的是从 0 到 2^n-1 这 2^n 个数所组成的数域，对域内的元素进行这个域上所定义的加、减、乘和求逆运算之后的值仍属于这个域。

统结构设计、拓扑构建算法设计、初始播放点设置算法设计、数据传输方案设计和数据传输算法设计。在设计系统之前，先介绍一下 CORS TV 系统所使用的随机线性网络编码。

3.1 随机线性网络编码

随机线性网络编码应用于 P2P TV 时，需要将原始的视频数据包序列划分为一系列的段，每段包含 n 个原始数据包 $[X_1, X_2, \cdots, X_n]$，每个数据包 X_i 包含 k 个字节。当有限域的大小为 2^8 时，X_i 就可以看作 k 维的向量。当有限域的大小为 2^4 时，X_i 可以看作 $2*k$ 维的向量。之后，系统以段为单位进行编解码，以数据包为单位进行传输。

当服务器需要为某个邻居 p 发送一个段中的数据包时，编码模块就在预先确定的有限域范围内随机生成一组编码系数向量 $C_j^p=[\alpha_{1j}^p, \alpha_{2j}^p, \cdots, \alpha_{nj}^p]$。对该段中的数据包做如下的线性组合，生成一个编码数据包：

$$Y_j=\sum_{i=1}^{n}\alpha_{ij}^p * X_i$$

编码系数作为包头与编码后的数据一起发送出去。这些编码系数会带来一定的带宽开销，开销的大小与有限域的大小和段中数据包的个数 n 成正比，与原始数据包的长度 k 成反比。

节点 p 收到 $m(m\leqslant n)$ 个数据包后，就可以为自己的邻居服务，发送该段内的数据包。当节点 p 给邻居 q 发送数据包时，假设节点 p 已经收到该段中 d 个编码数据包，则编码模块就随机生成一组编码系数 $C_l^q=[\beta_{1l}^q, \beta_{2l}^q, \cdots, \beta_{dl}^q]$。对该段中数据包进行线性组合，生成一个新的编码数据包：

$$Z_l=\sum_{j=1}^{d}(\beta_{jl}^q * Y_j)=\sum_{j=1}^{d}(\beta_{jl}^q * \sum_{i=1}^{n}(\alpha_{ij}^p * X_i))$$

$$=\sum_{i=1}^{n}((\sum_{j=1}^{m}(\beta_{jl}^q * \alpha_{ij}^p)) * X_i)$$

式中，各个 X_i 前的系数为新的全局编码系数，这些系数可以通过新生成的 $1\times d$ 维系数向量 $[\beta_{1l}, \beta_{2l}, \cdots, \beta_{dl}]$ 和嵌在编码数据包 Y_j 中的 $d\times n$ 维系数矩阵相乘计算出来。

节点一旦收到了某个段中 n 个线性独立的编码数据包，就可以使用高斯-约当消去法进行解码，解出原始的 n 个数据包。原始数据组成数据矩阵 X，n 个线性独立编码数据包中的编码数据组成数据矩阵 B，嵌在编码数据包中的系数组成系数矩阵 A，这三者的关系是：

$$B=AX$$

使用高斯-约当消去法求出 A 的逆矩阵，就可以解出原始数据：

$$X=A^{-1}B$$

因为网络编码算法的所有操作都是在有限域内进行的，所以不会引起数据溢出问题。本文中有限域的大小取为 $2^8=256$。已有实验表明，这一大小可以使得由编码所引起的冗余率非常低[26,27]。为了降低有限域上加、减、乘和除等基本运算的开销，CORS TV 系统中采用了查表的方式，以空间换取时间。

为了降低编码系数的带宽开销，数据包应该越大越好。但不能大于 1500 字节，否则就会在 IP 层分片。本文中原始数据包的长度设为 1250 字节。段的大小被设置为 1s 的原始数据包数量。即当视频码率为 500kbps 时，每个段的数据包数量为 50 个。在这样的编码参数设置下，编码系数所引入的额外带宽开销为 $(50*8)/(1250*8)=4\%$。如果有限域的大小为 2^4，则带宽开销为 $(50*4)/(1250*8)=2\%$。

为了描述方便，在此给出几个与随机线性网络编码相关的定义：

1)“激进度”(Aggressiveness)α：定义为 m 与 n 的比值 m/n。

2)一个段的拥有者：这个节点收到了该段中至少 m 个线性独立的编码数据包。

3)一个节点的某个段准备好为邻居提供服务：这个节点是该段的拥有者。

4)一个段是空的：这个段没有包含或收到任何数据包。

5)一个段是满的或一个节点收齐了某个段中的数据包：一个节点收到了该段原有数量(n 个)的线性独立编码数据包。

3.2 系统结构设计

图 1 是 CORS TV 系统的结构图，图 2 是系统的程序流程图。客户端和服务器使用了同一份代码，程序在运行过程中通过判断相应的配置参数来区分服务器和客户端。

在 CORS TV 系统中，服务器和客户端统称为节点，每个节点都具有相同的内部结构。CORS TV 系统对节点的各种功能进行了划分，并设计了相应的功能模块，这些模块包括通信模块、调度模块、邻居列表模块、缓存状态查询模块、传输模块、缓存及管理模块、数据读取或保存模块、系统状态监控模块以及网络编码和解码模块。内部主要功能模块都具有单独的线程，作为核心的调度模块通过线程间通信调度这些模块，实现拓扑的构建和数据的传输。

3.2.1 通信模块

通信模块有 3 个通信端口，使用 UDP 协议传输数据包和控制包。其中两个端口用来发送，一个端口用来接收。控制包和数据包使用不同的端口分别发送，以减少相互之间的干扰。而同一个端口接收所有的数据包和控制包，以简化系统的调度。

3.2.2 调度模块

调度模块是系统的核心，使用 *select*() 函数实现了一个定时器，来调度各模块。*select*() 函数在监视接收端口时，它的超时时长被设为当前时间与下一个任务执行时间的差值。*select*() 函数的返回原因有 3 种：第一种是该端口收到了包，第二种是达到了超时时长，第三种是发生了异常。当 *select*() 函数返回后，首先检查是否有包收到。如果有，则收取缓存中的包，交给相应的模块处理。然后检查任务队列是否有任务需要执行。如果有，则执行相应的任务，否则就直接进入下一个周期。为了避免干扰，各功能模块都单独启动了一个线程，调度模块与其它各模块线程之间使用信号加变量的方式进行通信，并使用互斥锁来保护相应的变量。图 1 中的虚线箭头表示线程间通信，实线箭头表示函数调用。当 *select*() 函数在监视接收端口时，可能会有新任务加入到任务队列，且执行时间早于 *select*() 函数的超时返回时间，但这一新任务可能会等更长的时间，直到 *select*() 超时返回后才得到执行。为了解决这一问题，提高定时器的精度，CORS TV 系统在设置 *select*() 函数的超时时长时规定超时时长不能大于 10ms。

系统中需要定时器调度的任务类型分为两种：一种是需要周期性处理的任务，另一种是一次性任务。对于服务器来说，需要周期性处理的任务包括拓扑维护、缓存状态广播、数

据包的传输调度、数据的读取以及系统状态的保存。对于客户端来说，数据的读取改为数据的保存，另外需要增加一个周期性任务，向服务器汇报在线状态。图 1 中的 6 个圆圈标示了 6 个任务所涉及的功能模块。一次性任务主要是数据包的发送，每个数据包的发送都作为一个任务插入定时器任务列表，尽量使得数据包均匀发送，避免突发流量。各任务的细节会在后续相关部分详细介绍。

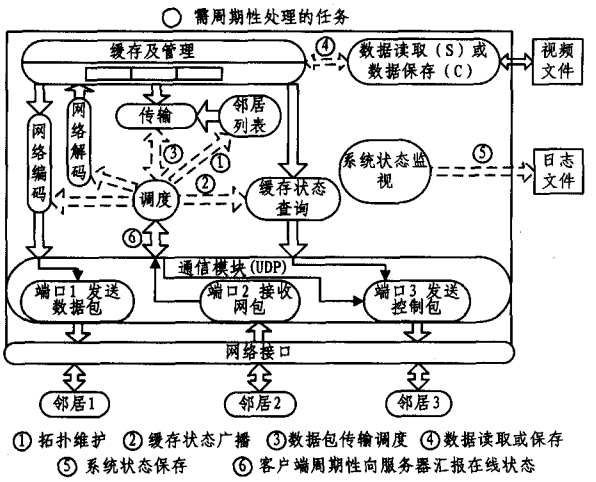


图 1　CORS TV 系统结构图

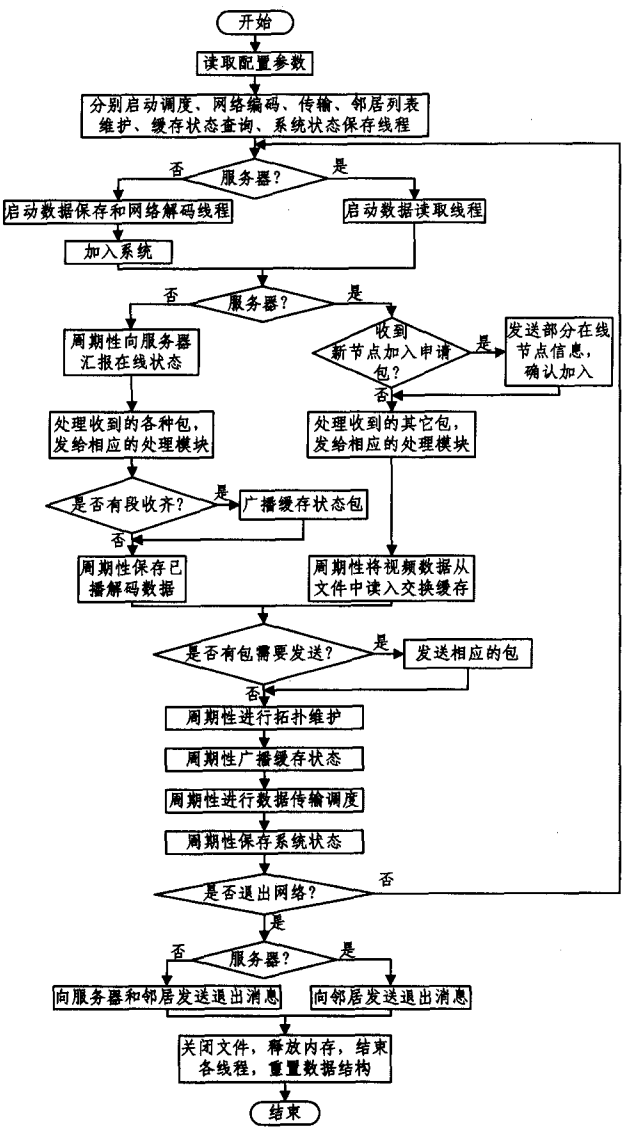


图 2　CORS TV 程序流程图

3.2.3　邻居列表模块

邻居列表模块负责节点的加入和邻居关系的维护。对于客户端来说，该模块维护着两个列表，分别是成员列表和邻居列表。每个节点只与邻居列表中的节点进行数据传输和周期性的信息交换，成员列表中的节点只是作为备用节点，当邻居节点的数量不足时再与之建立邻居关系。如果当前邻居数量低于邻居数量限制的下限，则该模块就会发起一个任务，寻找新的邻居。该模块首先检查成员列表中是否有备用节点。如果没有，就向服务器或邻居节点请求新的在线节点信息，并尝试与之建立邻居关系。

对于服务器来说，除了上述两个列表外，还维护着一个在线节点信息列表，该列表记录了所有在线节点的信息。当有服务器收到新节点加入申请包时，就从这些在线节点中随机选取部分节点信息发送给新加入的节点。

3.2.4　缓存状态查询模块

缓存状态查询模块周期性地查询本地的缓存状态，并将缓存状态信息通过缓存状态包发送给邻居节点。当节点收到邻居广播的缓存状态包后，就将其保存到邻居列表模块中，供传输模块使用。

3.2.5　传输模块

传输模块负责数据包的传输调度。它根据本节点和邻居节点的缓存状态，决定该选取哪些数据包，以及该将这些数据包发给哪些邻居节点。

3.2.6　缓存及管理模块

缓存及管理模块负责缓存收到的数据包，并提供一些基本的操作接口，供其它模块使用，如 delete()，insert()，read() 和 checkstatus()等。

缓存使用循环式队列实现，并以段为单位保存收到的编码数据包。

3.2.7　数据读取或保存模块

数据读取或保存模块负责数据的读取或保存。对于服务器来说，该模块周期性地将视频数据读到缓存中，并总是保持还未播放的缓存为满的状态；对于客户端来说，该模块周期性地将缓存中已经播放过的数据保存到文件中。

3.2.8　系统状态监控模块

系统状态监控模块负责监控整个节点的运行状态。该模块记录了节点所有的操作、传输和状态信息，并将其保存到文件里，供实验结束后对系统进行调试、优化和分析。后续的系统评估都需要分析这些日志文件。

3.2.9　网络编码和解码模块

与非编码系统相比，CORS TV 系统增加了网络编码和解码模块，它们分别负责对数据包进行编码和解码。在对某个段进行编码时，网络编码模块读取缓存中相应段的所有数据包进行组合编码，生成新的编码数据包，交给传输模块。当节点收到一个编码数据包后，解码模块就读取相应段中所有的数据包与新收到的数据包一起进行渐进式解码。如果新到达的数据包与已收到的数据包线性相关，则将该新到达的数据包丢弃，否则将渐进式解码的结果存到缓存中。为了保证解码模块能及时解码已收到的数据包，CORS TV 在编码和解码模块间加了一个互斥锁，以保证解码模块的优先权。

3.3　拓扑构建算法设计

CORS TV 系统的拓扑构建采用了 Gossip 协议。Gossip

协议是分布式系统常用的通信协议，它不需要中心服务器，因此具有很好的可扩展性。在 Gossip 协议中，每个节点将消息发送给邻居节点，邻居节点再根据需要决定是否转发该消息。

CORS TV 系统中的 Gossip 协议包括新节点加入、节点的邻居关系维护和节点的退出等 3 部分。

3.3.1 新节点加入

按照时间顺序，节点的加入过程主要包括以下几步：

1)节点向服务器发送加入网络的申请包，申请加入；

2)服务器给该节点返回当前部分在线节点的信息列表，确认加入；

3)该节点向这些在线节点发送连接请求包，尝试建立邻居关系；

4)待收到在线节点的连接确认信息包后，将该节点加入邻居列表，即与该节点建立了邻居关系；否则，如果收到该在线节点的拒绝则建立连接请求包，或超时后未收到任何回应，则放弃与该节点建立邻居关系；

5)待该节点连接的邻居数量达到了邻居数量限制的下限后，该节点就确定自己的初始播放点，即完成了加入过程。对于未尝试建立邻居关系的其它在线节点，则将它们的相关信息保存在成员列表中备用。

3.3.2 节点的邻居关系维护

在系统正常运行阶段，节点需要周期性地或根据需要维护邻居关系，以便保持合理的邻居数量，获得满意的数据包速率。

1)节点周期性地向服务器汇报自己的在线状态，防止服务器将自己错误地从在线列表中删除。否则，后续新加入的节点就无法获得本节点信息，影响新节点的加入速度。

2)节点周期性地向自己的邻居发送心跳信息，来声明自己的存在。因本系统需要周期性地向邻居广播缓存状态信息，所以使用缓存状态包代替心跳信息包。

3)如果一个节点收到一个新节点的建立邻居关系包，该节点就检查自己的邻居数量。如果小于邻居数量限制的上限，就向该新节点发送确认建立邻居关系包，否则就拒绝该新节点的请求。

4)如果节点收到某个邻居节点的下线信息包，或在一段时间内没有收到该邻居的心跳信息包，则将其从邻居列表中删除。

5)节点周期性地检查与每个邻居的数据传输情况。如果在一个周期内，本节点与某个邻居之间没有传输任何数据，或传输数据量过少，则与该邻居断开邻居关系。为了降低邻居调整对系统的冲击，CORS TV 规定检查的周期为 60s，每次最多只能调整一个邻居关系。

6)如果本节点的邻居节点的数量少于邻居数量限制的下限，则该节点就发起任务，寻找新的邻居节点。

3.3.3 节点的退出

正常情况下，每个节点在离开之前都会主动向所有的邻居发送下线信息包，断开与所有邻居的连接关系，并向服务器注销。但节点还可能会因为死机、断网和程序被强制退出等原因非正常退出，这种情况下，它的邻居关系维护机制会在一定的时间内将其从邻居列表中删除，服务器也会将其从在线节点列表中删除，从而保证邻居信息和在线节点信息的有效性。

3.4 初始播放点设置算法设计

初始播放点的设置非常关键，它直接影响到后续数据包的传输和邻居关系的维护。如果设置得不合理，会使得本节点缓存与邻居节点缓存的重叠部分变得很短，影响本节点可获得的数据包速率，进而影响邻居关系。这会对整个系统的性能产生影响，所以将其单独列出来。这一功能由调度模块完成。

本文采用最多者优先的方法来设置各节点的初始播放点，即如果邻居中有一个节点是服务器，则将本节点的初始播放点设置为与服务器相同；如果邻居中没有服务器，则计算邻居缓存中各个段或数据包的拥有者数量，设置拥有者最多的那个段或数据包作为本节点的初始播放点，这样使得节点可以获得尽可能高的数据包速率。

3.5 数据传输方案设计

CORS TV 的数据传输使用随机推送的方案。每个节点周期性地调用数据传输算法，随机选择部分邻居节点作为这个调度周期待服务的邻居节点。传输算法根据收到的邻居缓存状态包和本节点的缓存状态，选取本节点拥有而邻居节点未收齐的段，并发送这些段中的数据包给相应的邻居节点。在发送数据包的过程中，如果收到了邻居节点新的缓存状态包，发现该邻居已经收齐了正在发送的数据包所在的段，就及时终止这些数据包的发送。

3.5.1 随机推送的传输方案

CORS TV 中，随机推送的调度周期设为 500ms，即每 500ms 调用一次传输算法，来决定该向哪些邻居推送数据，以及该推送哪些段中的数据包。为了充分利用每个节点的上行带宽，每次调度所推送的数据包数量上限为视频码率的 2.5 倍。因为传输协议使用的是 UDP 协议，所以超出上行带宽容量的那部分数据包会被马上丢弃，不会增加其它数据包的传输时间。

周期性地传输调度会造成数据包的集中发送，引起突发流量，使得节点的上行带宽瞬间被占满，导致大量数据包被丢弃。而在下次传输调度之前，又会出现较长时间的空档，没有数据包可供传输，节点的上行带宽得不到充分利用。所以，CORS TV 在传输算法的调度完成之后，将每个数据包的发送作为任务插入定时器任务队列，使得应该发送的数据包尽量均匀地分散在接下来的一个传输周期内发出去。

3.5.2 缓存状态信息的及时更新

缓存状态包的格式如图 3 所示。因为 CORS TV 使用了随机推送的数据传输方案，所以需要在段中数据包收齐后及时通知邻居节点停止发送，减少数据包的冗余。为了实现这一目的，只要节点缓存中的任意一个段收齐，节点就向邻居广播自己的缓存状态，所以缓存信息包在此也作为命令包来使用。除此以外，系统还会周期性地广播自己的缓存信息，及时向邻居更新自己的缓存状态。这是为了防止节点长时间无法收齐任何一个段中数据包，使得缓存状态包的广播频率过低。因为缓存状态包也作为邻居关系维护的心跳信息包使用，所以这可以避免邻居错误地将自己从邻居列表中删除。

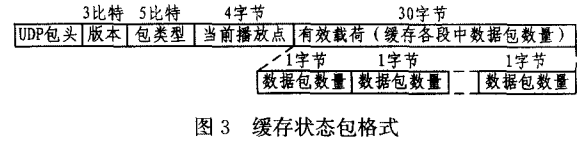


图 3 缓存状态包格式

对于没有网络编码的 P2P TV 系统来说，缓存状态的格式用位图表示，即一个比特表示一个数据包的有无。在对比实验中，它的缓存状态包的平均广播周期与基于网络编码的

P2P TV 系统的平均广播周期相同。这一平均周期被定义为 bminterval，本文中取 bminterval 为 300ms。

3.6 数据传输算法设计

数据传输算法是为了解决该向哪些邻居节点服务，以及该选取哪些段中的数据包为邻居节点服务的问题。因为网络编码对连续的数据包序列进行了分段，所以对于基于网络编码的 P2P TV 系统（NC System，简称编码系统）和无网络编码的 P2P TV 系统（NoNC System，简称非编码系统）来说，传输算法的细节略有差异。

3.6.1 NC 与 NoNC 传输算法对比

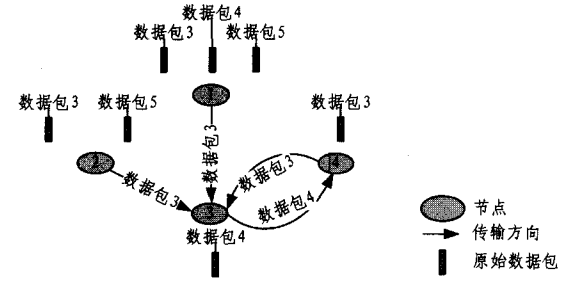
在 P2P TV 系统中，常用的数据传输算法主要有以下几种。

1）贪婪式（Greedy）：离播放点较近的数据包或段拥有更高的优先级。

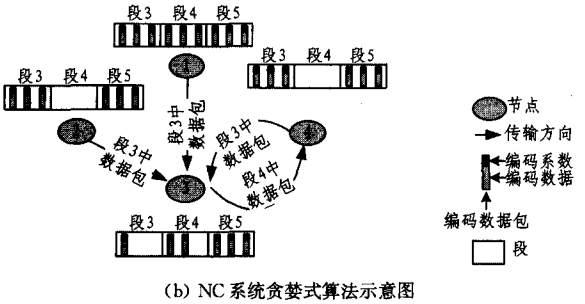
2）最少者优先式（Rarest-first）：在邻居中拥有者最少的那些数据包或段拥有更高的优先级。

3）随机式（Random）：所有数据包或段的优先级是随机确定的。

图 4 是非编码系统和编码系统的数据传输算法对比图。以贪婪式算法为例，编码系统中的传输算法调度的粒度是段，非编码系统中传输算法的调度粒度是包。每个段内包含 3 个原始数据包，激进度设置为 $\alpha=m/n=2/3$。在图 4(b)中，节点 3 的段 3 中只有一个数据包，小于 m，所以节点 3 不是段 3 的拥有者，节点 3 不能向邻居节点发送段 3 中的数据包。但节点 3 是段 4 的拥有者，虽说它没有收齐段 4 中的数据包，但该段中数据包数量大于 m。从图 4(a)中可以看到，对于非编码系统来说，数据包的拥有者的判断非常简单。



(a) NoNC 系统贪婪式算法示意图



(b) NC 系统贪婪式算法示意图

图 4 NoNC 系统与 NC 系统的贪婪式算法对比

从图 4 中也可以看出，基于推送的编码系统相对于相应的非编码系统来说，编码系统能获得较低的数据包冗余率。假设邻居之间的带宽容量都是 1，即一个数据包传输调度周期内只能发送一个数据包。在非编码系统中，一个数据包传输调度周期内，一个节点最多收到 2 个冗余数据包，即冗余率为 $2/(1+2)=0.67$。在编码系统中，一个数据包传输调度周期，一个节点的每个段中最多收到 2 个冗余数据包，冗余率为 $2/(3+2)=0.4$。而且，随着段中数据包数量的增加，冗余率会进一步降低。当段中数据包数量 $n=50$ 时，冗余率最大为 $2/(50+2)=0.038$，而非编码系统的冗余率上限仍然是 0.67，因此编码系统的冗余率比较低。较低的冗余率使得编码系统可以更好地利用节点的上行带宽容量。

3.6.2 先到先得式（FCFS）数据传输算法

虽然网络编码可以简化 P2P TV 数据传输问题的难度[28]，但编码系统的传输算法仍然需要精心设计。因为在基于网络编码的 P2P TV 系统中，所有的段都有自己的播放点。如果某个段在到达播放点时仍然无法收齐，则该段不得不被跳过，该段中已经收到的编码数据包因无法解码而被迫丢弃。这会增加 P2P TV 系统的冗余率，浪费节点的上行带宽。当节点可获得的上行带宽不足时，网络编码的这一缺点就会显现出来。

为了提高网络编码数据包的有效传输率，我们提出了先到先得式（First Come First Served，简称 FCFS）数据传输算法。该传输算法使用两个手段来获得好的有效传输率。一个是尽量减少因无法解码而被无谓丢弃的数据包数量。这通过集中获取的方法实现，即首先服务邻居节点缓存中的未收齐非空段，并主动跳过来不及收齐的空段。另一个是充分利用每个节点的上行带宽容量，获得尽可能高的上行带宽。为了实现这一目标，应该使得那些未被放弃的空段在收齐之后，在邻居的缓存中保持比较好的多样性，便于邻居之间互相帮助。

FCFS 算法包含 3 个步骤。第一步，它首先选择部分邻居节点作为本次调度的服务对象。本文中每个节点的邻居数量为 10 到 15 个，每次取 20% 的邻居作为自己的服务对象。第二步，决定邻居的缓存中各个段的相对优先级。最后，根据本节点上行带宽的情况，将优先级最高的那些数据包推送给相应的邻居节点。在 FCFS 算法中，最关键的是第二步，即如何决定各个段的相对优先级。

如图 5 所示，缓存以当前播放点为界，被分为交换缓存和过时缓存。交换缓存用来存放待播放的段，过时缓存用来存放已经播放过的段。它们的长度分别为 30s 和 10s。

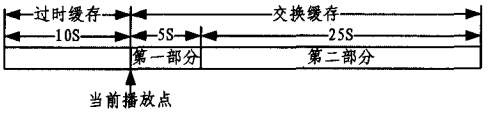


图 5 FCFS 算法对缓存的划分示意图

为了决定每个段的相对优先级，每个节点的交换缓存被划分为两部分。如图 5 所示，离播放点较近的那部分称为第一部分，另一部分称为第二部分。第一部分中的空段可能会在到达播放点前无法收齐，所以应该被主动放弃。剩下的待获取段被分为 3 类：1）第一部分中的未收齐非空段；2）第二部分中的未收齐非空段；3）第二部分中的空段。

这 3 类段的优先级别依次降低。类内各段之间的相对优先级分别使用以下方法确定：

1)贪婪式方法被用来决定第一类段的相对优先级。已有研究表明,贪婪式方法可以使得系统获得更好的有效传输率[29]。

2)最少者优先方法被用来决定第二类段之间的相对优先级,可以使得系统具有更好的可扩展性[29]。

3)随机方法被用来确定第三类段内部各段之间的相对优先级,以便获得较好的段的多样性。

交换缓存中,两部分的界限通过实验确定。为了增强每个节点中所拥有的段的随机性和多样性,第二部分应该越长越好,但不能超过交换缓存的长度。但第一部分缓存不能太短,否则第一类段可能没有足够的时间被填满。根据实验发现,第一部分设置为 5s 比较合适。

4 实验验证和结果分析

为了验证系统设计的有效性,我们在实验室的计算机集群上进行了实验验证,63 台主机上各运行了 1 个节点。

实验中,首先比较了 CORS TV 系统与其它基于推送传输方案的编码系统。这些编码系统使用的传输算法分别是贪婪式算法、最少者优先算法和随机算法,这些算法分别在整个交换缓存上使用贪婪式方法、最少者优先方法和随机方法来决定各段之间的相对优先级。因为现有的网络编码传输算法在一定的可获得带宽条件下都可以归结为这 3 种算法,所以将这 3 种算法代替已有的工作作为比较的对象。例如,Lava[12] 和 R^2[13] 的传输算法在用户可获得的上行带宽不足时,分别退化为贪婪式算法和随机算法。

然后对比了 CORS TV 系统和基于其它 3 种传输算法的非编码系统,展示了系统的有效传输率和冗余率性能。其中,非编码系统也采用了推式数据传输方案。

4.1 实验环境及参数设置

通过配置 CORS TV 系统内置的带宽控制模块,设置 3 种接入类型的节点,它们的上行带宽容量分别为 128kbps,384kbps 和 1000kbps。调节这 3 类节点的比例得到不同的带宽资源指数(Bandwidth Resource Index,BRI),来模拟 P2P 系统的瓶颈。BRI 定义为系统中所有节点的上行带宽容量与系统中所有节点所需上行带宽容量之间的比值,即

$$BRI=(\sum_{n=1}^{N}u_i)/(N*r_s)$$

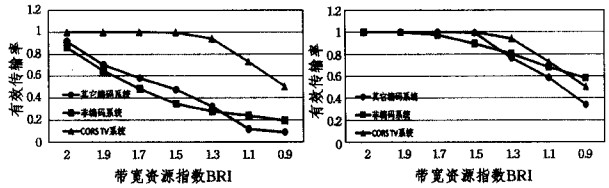
式中,N 为系统中节点数量,u_i 为节点 i 的上行带宽容量,r_s 为视频码率。

激进度的设置通过实验确定。如果激进度设置过高,会导致传输延时过长[12]。如果激进度设置过低,会引起较高的冗余率。CORS TV 中根据实验设置其为 0.2,即 $m=10$ 个数据包。

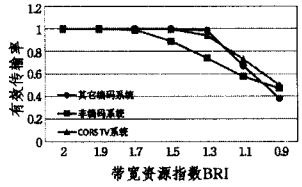
4.2 实验结果分析

图 6 是系统的有效传输率性能对比图。首先看一下 CORS TV 系统与其它编码系统的对比结果。从图中可以看到,当系统可获得的上行带宽不足时,CORS TV 系统与基于最少者优先和随机算法的编码系统相比,其有效传输率最多可以提高 15%。而且,CORS TV 系统与基于贪婪式算法的编码系统对比时,最多可以将系统有效传输率提升 60%。由图 7 可以看出,系统有效传输率的提升主要是由系统冗余率的降低获得的。

图 6 和图 7 也对比了 CORS TV 系统和无编码系统的有效传输率和冗余率性能。从图中可以看出,CORS TV 系统几乎总是可以降低非编码系统的数据包冗余率,同时在上行带宽不足时,可有效地提高系统的有效传输率。这是因为 CORS TV 系统有效降低了因无法解码而被丢弃的数据包数量,同时充分利用了系统中节点的上行带宽。

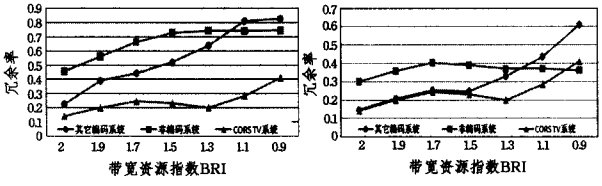


(a) 与基于贪婪式算法的系统对比　(b) 与基于最少者优先算法的系统对比

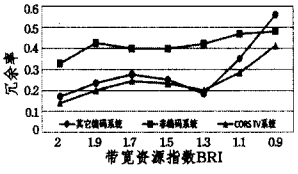


(c) 与基于随机算法的系统对比

图 6　CORS TV 系统与其它 P2P TV 系统在不同的带宽资源指数下的平均有效传输率对比



(a) 与基于贪婪式算法的系统对比　(b) 与基于最少者优先算法的系统对比



(c) 与基于随机算法的系统对比

图 7　CORS TV 系统与其它 P2P TV 系统在不同的带宽资源指数下的平均冗余率对比

结束语　P2P TV 技术的发展呈现了百花齐放的局面,如何提升系统的服务质量,改善用户的使用感受,争取到更多的用户,是 P2P TV 服务提供商所面临的挑战之一。本文将网络编码引入到 P2P TV 系统,从系统结构、拓扑构建、初始播放点设置、传输方案和传输算法等几个方面对系统进行了设计,并实现了一个基于随机线性网络编码的 P2P TV 系统:CORS TV。在实验室计算机集群上的实验结果表明,与现有类似系统和未使用网络编码的系统相比,CORS TV 降低了系统冗余率,提高了系统的有效传输率,从而改善了用户的播放质量。未来工作中,CORS TV 将被进一步完善为一个可供用户使用的、完整的 P2P TV 系统。

参 考 文 献

[1] Chu Y-H, Rao S G, Seshan S, et al. A Case for End System Multicast [J]. IEEE Journal on Selected Areas in Communications (JSAC), 2002, 20(8): 1456-1471

[2] Zhang X, Liu J, Li B, et al. CoolStreaming/DONet: A Data-driven Overlay Network for Peer-to-Peer Live Media Streaming [C]// Proc of the IEEE INFOCOM. Miami, FL, USA: IEEE Press, 2005: 2102-2111

[3] PPLive [OL]. http://www.pplive.com/en/

[4] PPStream [OL]. http://www.ppstream.com/

[5] Rowstron A, Druschel P. Pastry: Scalable, Decentralized Object Location, and Routing for Large-scale Peer-to-Peer Systems [C]// Proc of the IFIP/ACM International Conference on Distributed Systems Platforms(Middleware). Heidelberg, Germany: November, 2001: 329-350

[6] Ren D, Li Y-T H, Chan S-H G. On Reducing Mesh Delay for Peer-to-Peer Live Streaming [C]// Proc of the IEEE INFOCOM. Phoenix, AZ, USA, 2008

[7] Liu Y. On the Minimum Delay Peer-to-Peer Video Streaming: how realtime can it be? [C]// Proc of the ACM Multimedia. Augsburg, Germany, 2007

[8] Bonaldy T, Massoulié L, Mathieuy F, et al. Epidemic Live Streaming: Optimal Performance Trade-Offs [C]// Proc of the ACM SIGMETRICS. Annapolis, MD, USA, 2008

[9] Li B, Xie S, Qu Y, et al. Inside the New Coolstreaming: Principles, Measurements and Performance Implications [C]// Proc of the IEEE INFOCOM. Phoenix, AZ, USA, 2008

[10] Ahlswede R, Cai N, Li S-Y R, et al. Network Information Flow [J]. IEEE Transactions on Information Theory, 2000, 46(4): 1204-1216

[11] M′edard R K M. Beyond Routing: An Algebraic Approach to Network Coding [C]// Proc of the IEEE INFOCOM. New York, NY, USA, 2002: 122-130

[12] Wang M, Li B. Lava: A Reality Check of Network Coding in Peer-to-Peer Live Streaming [C]// Proc of the IEEE INFOCOM. Anchorage, Alaska, USA: IEEE Press, 2007: 1082-1090

[13] Wang M, Li B. R^2: Random Push with Random Network Coding in Live Peer-to-Peer Streaming [J]. IEEE Journal on Selected Areas in Communications, Special Issue on Advances in Peer-to-Peer Streaming Systems, 2007, 25(9): 1655-1666

[14] Tang L, Chen Z, Yin H, et al. CORS: A Cooperative Overlay Routing Service to Enhance Interactive Multimedia Communications [J]. Journal of Visual Communication and Image Representation, 2010, 21(2): 107-119

[15] Zhang M, Zhang Y X Q, Sun L, et al. Optimizing the Throughput of Data-driven Peer-to-Peer Streaming [J]. IEEE Transactions on Parallel and Distributed Systems, 2009, 20(1): 97-110

[16] Zhang M, Luo J-G, Zhao L, et al. A Peer-to-Peer Network for Live Media Streaming-Using a Push-Pull Approach [C]// Proc of the ACM Multimedia. Singapore, 2005

[17] Li S-Y R, Yeung R W, Cai N. Linear Network Coding [J]. IEEE Transactions on Information Theory, 2003, 49(2): 371-381

[18] Ho T, Koetter R, M′edard M, et al. The Benefits of Coding over Routing in a Randomized Setting [C]// Proc of the IEEE International Symposium on Information Theory. Yokohama, Japan, 2003: 442

[19] Gkantsidis C, Rodriguez P R. Network Coding for Large Scale Content Distribution [C]// Proc of the IEEE INFOCOM. Miami, FL, USA, 2005: 2235-2245

[20] Gkantsidis C, Miller J, Rodriguez P. Anatomy of a P2P Content Distribution System with Network Coding [C]// Proc of the International Workshop on Peer-to-Peer Systems. Santa Barbara, CA, USA, 2006

[21] Liu H, Tu X, Xie J. Network Coding For P 2 P Live Media Streaming [C]// Proc of the IFIP International Conference on Network and Parallel Computing. Shanghai, China: IEEE Press, 2008: 392-398

[22] Liu Y, Peng Y, Dou W, et al. Network Coding for Peer-to-Peer Live Media Streaming [C]// Proc of the the Fifth International Conference on Grid and Cooperative Computing. Changsha, Hunan, China, Oct. 2006: 149-155

[23] 李亚龙. 基于网络编码的 P2P 直播数据传输策略研究与实现 [D]. 成都: 电子科技大学, 2009

[24] Chen X, Ren N, Zhang X, et al. SonicStream: a Network Coding Based Live P2P Media Streaming System with Rich User Experiences [J]. Journal of Communication and Networks, 2008, 10(4): 430-436

[25] 陈小刚. 基于网络编码的 P2P 直播流媒体系统的研究与实现 [D]. 上海: 复旦大学, 2008

[26] Ma G, Xu Y, Lin M, et al. A Content Distribution System Based on Sparse Linear Network Coding [C]// Proc of the Third Workshop on Network Coding, Theory, and Applications. San Diego, California, USA, January 2007

[27] Chou P A, Wu Y. Network Coding for the Internet and Wireless Networks [J]. IEEE Signal Processing Magazine, 2007, 24(5): 77-85

[28] Li Z, Li B. Network Coding in Undirected Networks [C]// Proc of the the 38th Annual Conference on Information Science and Systems (CISS 2004). Princeton, New Jersey, USA, March 2004: 257-262

[29] Zhou Y, Chiu D M, Lui J C S. A Simple Model for Analyzing P2P Streaming Protocols [C]// Proc of the IEEE International Conference on Network Protocols. Beijing, China: IEEE Press, 2007: 226-235