

基于 FGS 的 P2P 流媒体网络编码及调度

于敬敬, 王子磊, 奚宏生

(中国科学技术大学自动化系网络传播系统与控制安徽省重点实验室, 合肥 230027)

摘 要: 为满足异质网络环境下用户对服务质量的不同需求, 利用网络编码在 P2P 网络流媒体中的性能优势, 提出基于 FGS 的 P2P 流媒体网络编码及调度方法。通过网络编码分级和节点分级管理, 使各个异质节点协作工作, 由此实现非结构化 P2P 网络中的分布式数据调度策略。实验结果表明, 该方法可以使系统中各节点的播放连续度维持在 1.0 左右, 尤其是在节点接入带宽、起始播放时延、提供的缓冲区较小的情况下, 性能明显优于传统方法。

关键词: 异质网络; P2P 网络; 流媒体; FGS 编码; 网络编码; 数据调度

Network Coding and Scheduling of P2P Streaming Media Based on Fine Granularity Scalability

YU Jing-jing, WANG Zi-lei, XI Hong-sheng

(Key Lab of Anhui Network Communication System and Control, Department of Automation,
University of Science and Technology of China, Hefei 230027, China)

【Abstract】 In order to satisfy diverse quality of service in the heterogeneous P2P streaming systems, by making use of the advantages of network coding, a P2P streaming network coding and data scheduling method based on Fine Granularity Scalability(FGS) is proposed. By network coding grading and node rank management, it makes the heterogeneous nodes operate cooperatively with each other, and achieves the distributed data schedule strategy in unstructured P2P system. Simulation results demonstrate that the proposed method is capable of maintaining playing continuity degree close to 1.0 and has noticeable advantages over conventional scheduling method, especially under particular conditions such as low access bandwidth, small buffer and playback time delay and so on.

【Key words】 heterogeneous network; P2P network; streaming media; Fine Granularity Scalability(FGS) coding; network coding; data schedule

DOI: 10.3969/j.issn.1000-3428.2012.04.029

1 概述

P2P 流媒体利用互联网上的普通主机节点资源, 实现数据共享和节点协作, 在有限服务带宽下提供具有良好扩展性和较高性价比的视频服务。在实际的 P2P 流媒体系统中, 节点通常是异质的, 即节点的计算能力、接入带宽、缓冲区水平是不同的, 且接入带宽随网络状态波动。因此, 研究异质网络环境下 P2P 流媒体的高效协作技术具有非常高的实用价值。分层视频编码技术通过将视频信息分层来适应节点差异和网络带宽变化, 是处理异质 P2P 流媒体系统的主要技术手段^[1]。然而, 分层视频编码具有严格的解码顺序要求(只有底层数据接收后, 上层数据才能够解码), 需要为此设计复杂的数据调度算法, 以保障视频服务质量^[2]。另一方面, 网络编码允许网络中间节点对传输信息进行编码处理, 可提高 P2P 流媒体系统的网络吞吐量、系统鲁棒性、负载均衡水平等^[3], 研究人员建立的多种实验系统已经证明了网络编码的这种优势^[4-5]。本文考虑网络编码在基于分层视频编码的异质 P2P 流媒体上的应用, 研究有效的网络编码方案和数据调度方法, 实现分层视频编码和网络编码的有机结合。

2 相关研究

研究者们针对不同的网络条件提出了结合分层编码和网络编码的解决方案, 以期满足异质网络环境下各节点的不同视频播放要求。文献[4]提出在 tree-based P2P 网络流媒体系统中基于网络编码的分层最优化速率分配数学模型 LION。文献[6]提出了将网络编码应用于分层速率分配的思想, 利用

分层编码实现加权有向图网络模型的多速率组播。文献[7]提出一种基于有向图网络多速率视频流的网络编码方法。然而上述研究采用的拓扑知识难以从实际系统中获取, 无法执行分布式的数据调度, 实用化水平有限。为满足实际的异质 P2P 流媒体分发需求, 文献[8]以分层编码为基础提出了分级网络编码(Hierarchical Network Coding, HNC), 它通过单独编码重要数据和混合编码不重要数据, 使重要数据具有更高的解码概率, 但其没有提供具体的 HNC 构造方式和参数计算方法, 无法根据资源约束进行优化。文献[2]基于邻居节点选择算法, 并结合网络编码和 SVC(Scalable Video Coding), 提出了一种自适应的 P2P 流媒体传输方案。其中的邻居选择算法是基于节点所能达到的最高质量水平或平均质量水平的, 对突发的网络状态改变没有自调节能力。文献[9]提出一种 P2P 直播流系统, 采用 SVC 将流媒体编码为多层, 每层内单独进行随机网络编码, 接收方通过下载调度模块计算所需的编码块数和感兴趣的视频层, 发送方根据接收方所请求的层和段编码后发送给接收方。但是其没有提供如何计算所需的视频层数, 调度算法也比较复杂。可见, P2P 流媒体系统的高性

基金项目: 国家自然科学基金资助项目(61074033); 中国博士后科学基金资助项目(20100470855); 高等学校博士学科点专项科研基金资助项目(20093402110019)

作者简介: 于敬敬(1985—), 女, 硕士研究生, 主研方向: P2P 流媒体技术, 网络编码; 王子磊, 副教授; 奚宏生, 教授、博士生导师
收稿日期: 2011-07-20 **E-mail:** jiyu@mail.ustc.edu.cn

能分发仍面临诸多挑战^[9]。

本文针对实际P2P流媒体系统中异质节点的数据分发问题, 提出基于FGS(Fine Granularity Scalability)^[10]的P2P流媒体网络编码及调度方法。该方法可以实现非结构化P2P网络中执行分布式的数据调度策略, 有效利用节点资源并适应网络动态性, 提高网络吞吐量。

3 FGSNC 编码方法

在P2P流媒体系统中, 提供服务的节点称为种子节点, 请求服务的节点称为下游节点。种子节点提供服务时, 对流媒体数据进行编码。本文的FGSNC编码方法先进行FGS编码^[10]和网络编码分段, 段内进行随机网络编码。该编码方法通过调整增强层比例因子, 得到图像质量可调节的FGSNC编码块。系统中各节点根据自身性能选择相应的图像质量等级, 通过分布式数据调度实现各级数据在网络中的分发, 从而满足异质节点的不同视频服务质量需求。

3.1 FGS 编码和网络编码分段

FGSNC编码方法首先采用FGS编码得到一个基本层视频流和一个增强层视频流。基本层视频流 B 采用非可分级编码方式进行编码, 其码率作为解码端可分级解码的下界; 增强层视频流 E 则是采用基于离散余弦变换(Discrete Cosine Transform, DCT)系数的位平面编码机制, 对原始图像和重构图像的帧差图像进行编码。

FGSNC编码方法对FGS编码后的基本层视频流 B 和增强层视频流 E 均采用时间分段方式(每个分段的时间长度 Δt 相同)进行网络编码分段。基本层视频流和增强层视频流均被等分为 w 段。基本层视频流为 $B = \sum_{k=1}^w B_k$, 增强层视频流为 $E = \sum_{k=1}^w E_k$ 。视频文件的总长度为:

$$S = \sum_{k=1}^w S_k, S_k = B_k + E_k, 1 \leq k \leq w$$

其中, k 为段标识。

基本层视频流分段 B_k 和一定比例的增强层视频流分段 $\beta_i E_k$ 组成段 k 分段码流:

$$S_k^i = B_k + \beta_i E_k, 1 \leq k \leq w, 0 \leq \beta_i \leq 1$$

其中, β_i 为增强层比例因子, 是实际传输的增强层比特流和总的增强层比特流之比。 β_i 值越大, 表示实际传输的增强层的比特数越多, 解码的图像质量越好。当 $\beta_i = 0$ 时, 段 k 分段码流 S_k^i 只包括段 k 基本层视频流 B_k ; 当 $0 < \beta_i < 1$ 时, S_k^i 由段 k 基本层视频流 B_k 和比例为 β_i 的段 k 增强层视频流组成; 当 $\beta_i = 1$ 时, $S_k^i = S_k$, S_k^i 由段 k 的基本层视频流和全部段 k 增强层视频流组成。

分段码流 S_k^i 可以在 $[B_k, B_k + E_k]$ 范围内调节码流大小, 且 S_k^i 可以通过增强层再次截断, 分解出更小的分段码流 S_k^j :

$$S_k^j = B_k + \beta_j E_k, 0 \leq \beta_j \leq 1, \beta_i \beta_j = \beta_j$$

通过调整增强层比例因子, 可以得到大小不等的分段码流, 分段码流的大小决定了最终图像播放质量的优劣。

3.2 节点的网络编解码

随机网络编码不需要任何网络拓扑信息, 具有很高的实用性^[5]。视频流经上述FGS编码以及网络编码分段后得到分段码流 S_k^i , 对分段码流进行段内随机网络编码。

将 S_k^i 分为 N 块, 用符号表示为 $R_{k,1}, R_{k,2}, \dots, R_{k,N}$, 即段 k 基本层分段 B_k 和增强层分段 $\beta_i E_k$ 均被等分为 N 块, 表示为 $B_{k,1}, B_{k,2}, \dots, B_{k,N}$ 和 $\beta_i E_{k,1}, \beta_i E_{k,2}, \dots, \beta_i E_{k,N}$, 则 $R_{k,l} = B_{k,l} + \beta_i E_{k,l}$,

$1 \leq l \leq N$, 分段码流 S_k^i 可表示为 $S_k^i = B_k + \beta_i E_k = \sum_{l=1}^N R_{k,l}$ 。

种子节点 D 为其下游节点 A 提供数据服务时, 在有限域 $GF(2^8)$ 中随机选择 m 个随机数 $\alpha_1, \alpha_2, \dots, \alpha_m, m \leq N$, 种子节点从其缓存的段 k 的所有数据块中随机选择 m 个数据块 $[R_{k,1}, R_{k,2}, \dots, R_{k,m}]$ 生成编码块 $C_{k,1}^i$:

$$C_{k,1}^i = \sum_{l=1}^m \alpha_l R_{k,l} = \sum_{l=1}^m \alpha_l (B_{k,l} + \beta_i E_{k,l})$$

其中, m/N 为编码密度。种子节点 D 将段标识 k , 编码向量 α 和编码块 $C_{k,1}^i$ 传输给节点 A 。同理, 如果 A 再向 D 请求段 k 的一个数据块, D 生成一个新编码块 $C_{k,2}^i = \sum_{l=1}^m \alpha_{2l} R_{k,l}$ 。 $B_{k,l}$ 和 $\beta_i E_{k,l}$ 共享编码向量, 编码向量维数与增强层比例因子 β_i 值无关。

进一步, 若节点 A 收到了节点 H 对一个数据块请求, A 根据 H 的性能判断是否需要增强层再次截断。若需要, A 对其缓存的数据块进行随机网络解码, 得到分段码流 S_k^i , 对 S_k^i 进行截断, 得到:

$$S_k^j = B_k + \beta_j E_k, \beta_i \beta_j = \beta_j, \beta_i \geq \beta_j$$

之后对 S_k^j 进行随机网络编码, 并传送给下游节点 H ; 若不需要, A 根据其缓存数据块生成一个新的数据块 $\Omega_{k,1}^i$, 并传送给下游节点 H 。其中, $\Omega_{k,1}^i = \alpha_{\Omega 1} C_{k,1}^i + \alpha_{\Omega 2} C_{k,2}^i$, $C_{k,1}^i$ 和 $C_{k,2}^i$ 为节点 A 缓存的数据块。节点收到编码数据块后存储到自己的缓冲区中, 如果收到段 k 的数据块个数达到 N , 且这 N 个数据块所对应的编码向量线性无关, 则采用Gauss消元法进行随机网络解码, 得到 $R_{k,1}, R_{k,2}, \dots, R_{k,N}$, 进而得到分段码流 S_k^i 。对分段码流进行FGS解码, 即可得到可播放的视频流。

FGSNC编码方法可以提供连续分级编码, 且允许中间节点参与编码。分段码流 S_k^i 较大的编码块可以通过再次截断得到较小的编码块。拥有较大分段码流的种子节点可以为需要分段码流较小的下游节点提供服务。这样, FGSNC编码方法为异质网络环境下P2P流媒体调度提供了方便。

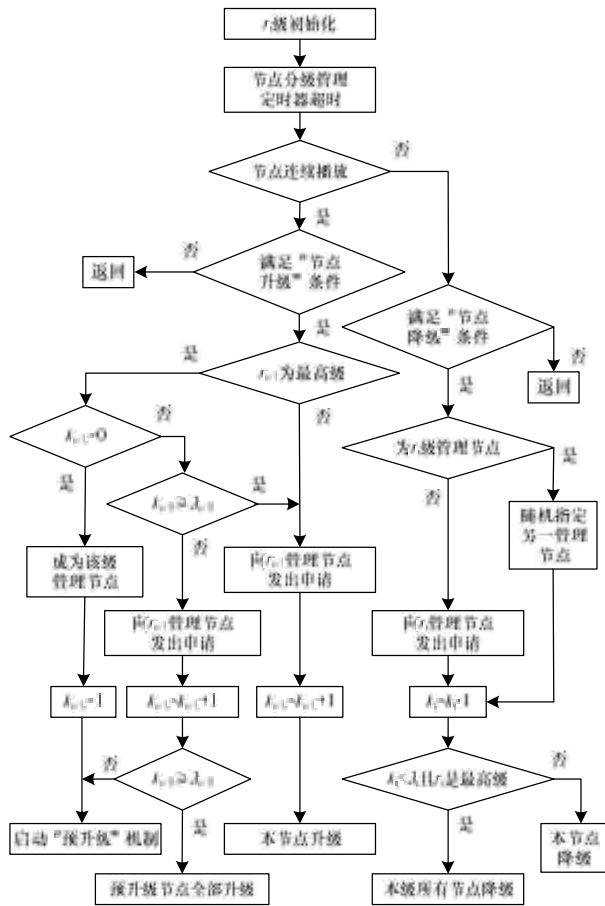
4 数据调度算法

在实际的P2P流媒体系统中, 各Peer节点的接入带宽、起始播放时延、提供的缓冲区等资源是不同的。若对系统中的所有节点不做区分地进行相同的调度, 则性能较好的节点不能获得相应的服务质量, 而性能较差的节点则会出现不能连续播放的情况。为了解决这个问题, 本文提出“节点分级管理”机制, 对异质节点进行分级化处理。

4.1 节点分级管理

“节点分级管理”机制分为节点初始化和节点动态调整2个阶段。该机制首先初始化系统节点到相同的级别, 然后通过“节点升级”或“节点降级”动态调整节点所处的级。节点分级后, 对高级别节点传输较大分段码流的编码块, 低级别节点传输较小分段码流的编码块。考虑FGSNC编码特性, 低级节点可使用同级或较高级节点的编码数据, 最高级节点只能使用同级节点以及服务器的编码数据。为保证最高级节点的播放质量, 在节点升级过程中, 采用“节点预升级”机制。为了保证节点播放的平滑性和连续性, 在节点动态调整阶段, “节点升级”和“节点降级”只能逐级进行, 不允许节点一次升或降2级及2级以上。

图1为 r_i 级节点分级管理流程。在初始化阶段, 所有节点初始化到 r_1 级, r_1 级节点对应的增强层比例因子为 β_1 , 则 r_1 级节点的段 k 分段码流长度为 $S_k^1 = B_k + \beta_1 E_k$ 。

图1 r_i 级节点分级管理流程

若 r_i 级的某节点在定时器间隔内可以连续播放, 节点判断自身是否满足升级条件。该升级条件包括三方面内容:

- (1) $\beta_i < 1$, 申请升级的节点所处的级别小于最高级。
- (2) Peer 节点缓冲区填充率大于给定阈值 δ 。节点缓冲区填充率为节点缓冲区中数据块的个数与缓冲区大小的比率, 其中, 缓冲区中的数据块为已经从邻居节点获取但还未得到播放的数据块; 缓冲区大小为播放启动延迟与流速率的乘积。
- (3) 解码率: 在定时器时间间隔内实现完全解码的数据段个数大于应该播放的数据段个数。

若系统中 r_i 级某 Peer 节点满足升级条件, 但系统中申请升级到 r_{i+1} 级的节点个数 k_{i+1} 小于给定阈值 λ_{i+1} , 则该节点必须等待一段时间, 直到 $k_{i+1} \geq \lambda_{i+1}$ 才可以得到升级, 这一过程称为“节点预升级”。满足升级条件的 r_i 级节点, 请求升级到 r_{i+1} 级的处理过程分为以下 4 种情况:

(1) 若 r_{i+1} 级是当前最高级且 r_{i+1} 级节点的分级计数 $k_{i+1}=0$, 该节点是第 1 个申请升级到 r_{i+1} 级的 r_i 级节点, 它将成为“ r_{i+1} 级管理节点”, 并记录申请升级到 r_{i+1} 级的节点个数 k_{i+1} , 给定初值 $k_{i+1}=1$ 。节点进入“预升级”状态, 等待后续 r_i 级节点的升级申请, 直到 $k_{i+1} \geq \lambda_{i+1}$, 所有申请节点升级到 r_{i+1} 级。

(2) 若 r_{i+1} 级是当前最高级且 r_{i+1} 级节点的分级计数 $k_{i+1} \geq \lambda_{i+1} - 1$, 已有节点工作在 r_{i+1} 级, 那么该节点只需要向“ r_{i+1} 级管理节点”发出升级申请。“ r_{i+1} 级管理节点”的“分级节点计数”加 1, 即 $k_{i+1} = k_{i+1} + 1$, 该节点立即升级到 r_{i+1} 级。

(3) 若 r_{i+1} 级是当前最高级且 r_{i+1} 级节点的分级计数

$\lambda_{i+1} > k_{i+1} \geq 1$, 则该节点只需向“ r_{i+1} 级管理节点”发出升级申请, “ r_{i+1} 级管理节点”的“分级节点计数”加 1, 即 $k_{i+1} = k_{i+1} + 1$, 该节点进入“节点预升级”状态。

(4) 若 r_{i+1} 级不是当前最高级, 该节点只需向“ r_{i+1} 级管理节点”发出升级请求。“ r_{i+1} 级管理节点”的“分级节点计数”加 1, 即 $k_{i+1} = k_{i+1} + 1$, 该节点升级到 r_{i+1} 级。

若在一段时间内 r_i 级某节点不能连续播放, 则对其降级。节点首先判断自身是否达到了降级条件, 该条件为: 该节点在定时器时间间隔内出现“断续播放”的次数 d_i 大于等于 D_i , D_i 为给定常数。

若 r_i 级某节点满足降级条件且是“ r_i 级管理节点”, 则该节点需随机指定该级内另一节点为管理节点, 新管理节点记录该级内节点个数 k_i , $k_i = k_i - 1$; 若该节点不是“ r_i 级管理节点”, 则向“ r_i 级管理节点”发出“降级请求”, r_i 级节点个数 k_i 自动减 1, 即 $k_i = k_i - 1$ 。

若 $k_i < \lambda_i$ 且 r_i 级为当前最高级, 则 r_i 级所有节点降低到 r_{i-1} 级; 否则, 仅该节点降级。

4.2 分布式数据调度

基于网络编码的 P2P 流媒体调度方法有 Pull 和 Push 两种模式: 在 Pull 模式下, 节点通过主动请求尽力保障自身的数据播放需求; Push 模式属于被动调用方式, 视频播放质量由邻居节点的数据推送来保障。本文在传统 P2P 流媒体调度算法基础上, 增加“节点分级管理”机制, 并结合 FGSNC 编码特性, 实现异质网络环境下 P2P 流媒体的分布式调度。其在节点邻居选择、节点加入和退出、数据调度等方面与传统方法不同。

节点选择邻居节点时只能从同级或较高级节点中选取, 最高级节点只能选择同级节点或者服务器作为邻居节点。若有多个候选节点可供选择, 则从候选节点中随机选择邻居节点。节点升级或降低后立即更新邻居节点表。根据节点的级别, 调度相应的 FGSNC 编码块。

节点加入系统时初始化为 r_1 级, 启动“节点分级管理”动态调整自身所处的级别; 节点退出系统的操作同“节点降级”。节点可以记录自身工作所处的稳定级别以及平均的分段码流长度, 保留历史记录以便下次播放时可以快速初始化到最佳的级别。

播放缓冲区中存在 3 种数据分段: 完全接收数据段, 尚未完全接收数据段, 尚未开始传输数据段。“节点升级”或“节点降级”后, 采用“缓冲机制”保证节点播放质量的平滑过渡。节点从 r_i 级升级到 r_{i+1} 级后, 对于节点中“尚未完全接收”的数据段, 仍传输 r_i 级节点对应的增强层大小 $S_k^i = B_k + \beta_i E_k$ 。对于“尚未开始传输”的数据块, 传输码流大小为 S_k^{i+1} 的编码块, 其中, $S_k^{i+1} = B_k + \beta_i \beta_{i(i+1)} E_k$, $0 \leq \beta_{i(i+1)} < 1$ 。节点从 r_i 级降到 r_{i-1} 级后, 分段码流长度立即减少到 $S_k^{i-1} = B_k + \beta_{i-1} E_k$, 以保证节点的播放连续度。

5 仿真实验与结果分析

为了评价节点的播放质量, 引入播放连续度性能指标。播放连续度定义为 Peer 节点在一段时间内实际播放(即在播放期限之前达到 Peer 节点, 并实现完全解码)的数据段个数与应该播放的数据段个数的比率。播放连续度的取值范围为 $[0, 1]$, 值越大, 表示节点的播放质量越好。

实验采用FALPS^[11]仿真平台, 参照R2^[5]和customR2^[12]的实验结论, 设置100个节点作为仿真对象, 每个数据段播放时长为4s, 其数据分为128块, 编码密度为8%。FGSNC仿真中, 流媒体采用FGSNC编码, 基本层采用H.264视频压缩编码方案, 增强层采用一个FGS层。节点分级 r_1 、 r_2 、 r_3 、 r_4 对应的增强层比例因子分别为 $\beta_1=0.2$ 、 $\beta_2=0.5$ 、 $\beta_3=0.7$ 、 $\beta_4=1.0$, 最高级节点阈值 $\lambda_i=10$, 节点缓冲区填充率阈值 $\delta=0.6$, 定时器间隔为 $\tau_i=1$ s, $D_i=2$ 。节点从同级或更高级别节点中选择6个节点作为自己的邻居节点, 服务器选择最高级节点中的 $\lambda_i=10$ 个节点直接发送数据。R2采用随机网络编码, 每个节点从中选择6个节点作为自己的邻居节点, 服务器节点选择其中的16个节点直接发送数据。每次仿真实验运行600s终止, 重复10次, 结果取平均值, 将FGSNC方法与R2的实验结果进行比较。

设置节点播放启动延迟为10s, 流速率为64Kb/s, 服务器带宽为1Mb/s, 取带宽范围为[40Kb/s, 60Kb/s]、[80Kb/s, 100Kb/s]、[180Kb/s, 200Kb/s]、[290Kb/s, 310Kb/s]、[400Kb/s, 420Kb/s]的节点各20个, 计算各个带宽范围内节点的平均播放连续度, 并分别以带宽范围的中心带宽值即50Kb/s、90Kb/s、190Kb/s、300Kb/s、410Kb/s表示该带宽范围, 如图2所示。可以看出, 在节点带宽高于300Kb/s时, R2和FGSNC的播放连续度都维持在1.0的水平上。但当节点带宽较低时, R2的节点播放连续度明显低于FGSNC, FGSNC仍可使节点播放连续度维持在1.0左右。FGSNC方法的节点分级管理机制可对异质节点分级化处理, 并通过调整增强层比例因子 β 值截断部分增强层比特流, 使性能较差的节点可以连续播放。

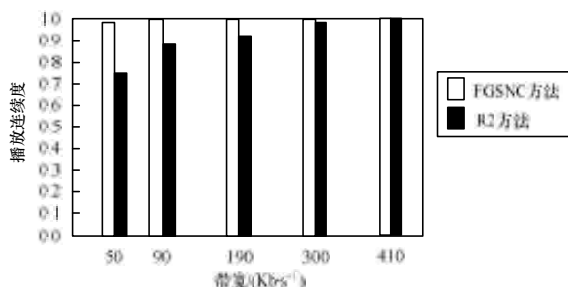


图2 不同带宽下的播放连续度比较

设置服务器带宽为1Mb/s, 节点上、下行带宽均在[80Kb/s, 100Kb/s]内随机分布, 流速率为64Kb/s, 节点播放时延为4s、8s、12s、16s、20s的节点各20个, 计算100个节点的平均播放连续度。图3给出了不同节点起始时延下, 采用R2和FGSNC方法的节点播放连续度的比较。

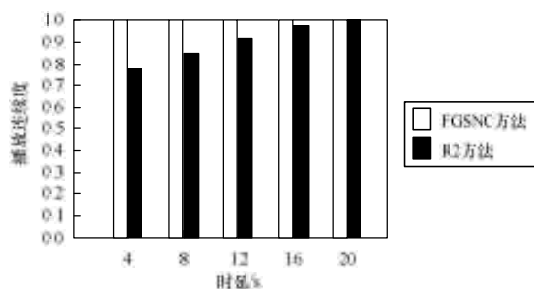


图3 不同起始时延下的播放连续度比较

由实验结果可以看出, 在节点起始时延高于16s时, R2和FGSNC的播放连续度都维持在1.0的水平上。但节点起始

时延小于12s时, FGSNC仍可使节点播放连续度维持在1.0左右, 明显优于R2。FGSNC将节点初始化到一个相同级别, 通过设置合适 β_i 值, 可以在不影响观看质量的情况下减少用户的起始等待时间。节点的缓冲区填充率反映了Peer节点实际获得的有效下载速率, 一般情况下, 缓冲区填充率越大, Peer节点的播放质量越好。在FGSNC中, 仅当节点的缓冲区填充率大于给定阈值时, 节点才有机会升级。在节点缓冲区较小时, FGSNC仍可以保证一定的节点缓冲区填充率。

6 结束语

本文提出了一种基于FGS的P2P流媒体网络编码及数据调度方法, 该方法结合了FGS编码和网络编码的优势, 首先对流媒体进行FGSNC编码, 通过网络编码分级和“节点分级管理”, 使分级化的异质节点协同工作, 实现了非结构化P2P网络中的分布式数据调度策略, 可以使异质节点得到各自较优的播放质量。

参考文献

- [1] Cui Yi, Klara N. Layered Peer-to-Peer Streaming[C]//Proc. of the 13th International Workshop on Network and Operating Systems Support for Digital Audio and Video. Monterey, USA: [s. n.], 2003.
- [2] Nguyen A T, Li Baochun, Eliassen F. Chameleon: Adaptive Peer-to-Peer Streaming with Network Coding[C]//Proc. of IEEE INFOCOM'10. San Diego, USA: [s. n.], 2010: 1-9.
- [3] Yeung R W. Information Theory and Network Coding[M]. New York, USA: Springer, 2008.
- [4] Zhao Jin, Yang Fan, Zhang Qian, et al. LION: Layered Overlay Multicast with Network Coding[J]. IEEE Transactions on Multimedia, 2006, 8(5): 1021-1032.
- [5] Wang Mea, Li Baochun. R2: Random Push with Random Network Coding in Live Peer-to-Peer Streaming[J]. IEEE Journal on Selected Areas in Communications, 2007, 25(9): 1655-1666.
- [6] Sundaram N, Ramanathan P, Banerjee S. Multirate Media Streaming Using Network Coding[C]//Proc. of the 43rd Allerton Conference on Communication, Control, and Computing. Monicello, USA: [s. n.], 2005.
- [7] Xu Chenguang, Xu Yinlong, Zhan Cheng, et al. On Network Coding Based Multirate Video Streaming in Directed Networks[C]//Proc. of 2007 International Performance, Computing and Communications Conference. New Orleans, Louisiana, USA: IEEE Press, 2007.
- [8] Nguyen K, Thinh N, Senching C. Peer-to-Peer Streaming with Hierarchical Network Coding[C]//Proc. of the 27th International Conference on Distributed Computing Systems. Minneapolis, Minnesota, USA: [s. n.], 2007: 396-399.
- [9] Mirshokraie S, Hefeeda M. Live Peer-to-Peer Streaming with Scalable Video Coding and Networking Coding[C]//Proc. of the 1st ACM SIGMM Conference on Multimedia Systems. Phoenix, Arizona, USA: ACM Press, 2010: 123-132.
- [10] Li Weiping. Overview of Fine Granularity Scalability in MPEG-4 Video Standard[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2002, 11(3): 301-317.
- [11] 武广柱, 王劲林. FALPS: 大规模P2P系统网络仿真平台[J]. 计算机工程与应用, 2008, 44(11): 9-12.
- [12] 王 晖, 陈伟涛, 刘亚杰. 基于网络编码的P2P流媒体推拉结合数据调度方法 customR2[J]. 计算机应用, 2010, 30(2): 285-288.