

BitTorrent 类型 P2P 流媒体系统研究

郑伟平^{1,2} 齐德昱¹ 徐克付³

(华南理工大学计算机系统研究所 广州 510640)¹ (华南师范大学计算机学院 广州 510631)²

(中科院计算技术研究所信息安全研究中心 北京 100190)³

摘要 介绍提供流媒体分发能力的 BitTorrent 类型系统的研究现状,探讨此类系统对数据块选择算法与对等点选择算法的改进,以及系统重要的性能参数和设计参数。

关键词 BitTorrent 类型,流媒体,TFT 算法,对等系统

中图分类号 TP311. **文献标识码** A

Research on BitTorrent-like Peer-to-Peer Media Streaming Systems

ZHENG Wei-ping^{1,2} QI De-yu¹ XU Ke-fu³

(Research Institute of Computer Systems, South China University of Technology, Guangzhou 510640, China)¹

(College of Computer, South China Normal University, Guangzhou 510631, China)²

(Information Security Research Center, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)³

Abstract Current research progress on BitTorrent-like systems which provide media streaming capacities was presented. Then modifications towards piece selection and peer selection algorithms within these kinds of systems were fully discussed, as well as the important performance metrics and design parameters.

Keywords BitTorrent-like, Media streaming, Tit-for-tat algorithms, Peer-to-peer systems

1 引言

流媒体应用一直是互联网上深受期待的杀手锏业务,其传输技术经历了传统的 C/S 模式、内容分发网络(CDN)技术到如今的 P2P 共享。P2P 为流媒体应用注入了新的生命力,P2P 流媒体系统受到了广泛的关注。从覆盖网拓扑上来看,P2P 流媒体系统可以分为树型系统和网状(Mesh)系统两类。多数研究^[17,18]认为网状系统更加适合 P2P 网络环境,其流媒体分发性能更佳。

近年来,P2P 技术在大文件分发应用上取得了巨大的成功。BitTorrent(简记为 BT)是最为流行的 P2P 文件共享软件之一,在 P2P 文件共享领域具有举足轻重的地位(BT is the king of the P2P protocols^[1])。网状流媒体系统与 BT 有着很多相似之处,利用 BT 提供流媒体服务的研究成为流媒体领域的一个热点^[1-8]。但是,BT 的设计初衷是提供大容量文件分发,并不支持流媒体“边下边看”的应用特性,也无法达到流媒体系统的服务质量要求。于是,人们开始研究如何改进 BT 以支持流媒体传输。我们将此类利用 BT 改进协议实现的流媒体分发系统称为“BT 类型流媒体系统”。

本文探讨 BT 协议在提供流媒体服务上的优越性和局限性,按照服务提供方式把 BT 类型流媒体系统分为两类:混合式系统和纯 P2P 系统,并在第 2 节中进行介绍;第 3 节重点研究以提供流服务为目的而进行的 BT 算法改造;第 4 节对影

响 BT 类型流媒体系统的几个重要性能参数和设计参数进行讨论;最后总结全文并指出存在的问题。

2 BitTorrent 类型流媒体系统

BT 与网状 P2P 流媒体系统有着很多相似之处。两者最本质的相同之处是都采用了网状拓扑,这决定了很多其它相似特性:都采用“拉(pull)”交换方式;数据驱动(Data-driven)特性;数据扩散路径(即分发树^[17])由数据可得性、拓扑连接和网络状态动态决定;节点间需要交换数据可得性描述信息;初加入节点须获取邻居列表,等等。例如 CoolStreaming^[19]的 BM 位图相当于 BT 的 have 消息;副手(deputy)节点相当于 BT 的 tracker 节点。鉴于 BT 系统的盛行,BT 与 P2P 网状流媒体系统又高度相似,研究人员开始权衡基于 BT 提供流媒体服务的可行性。文献[15]讨论了 BT 在流媒体服务上的优势和劣势,其优势有:

1) 分片机制(segmentation):BT 内置灵活的分片机制特别适合于流媒体应用。2) 元数据(metadata):为了实现文件共享,BT 先对文件进行抽象描述,并把此元数据(torrent 文件)发布到网络上。元数据提供了描述流媒体对象相关属性(如流码率等)的手段,这些先行获得的属性信息为设计更加智能的流媒体算法提供了便利。

因此,对 BT 类型流媒体系统而言,一般具有如下基本特征:a) 存在描述媒体对象的元数据,并依靠此元数据发布共

到稿日期:2009-09-18 返修日期:2009-12-28 本文受粤港关键领域重点突破项目(2008A011400010),国家技术创新基金项目(08C26214411198),广东省自然科学基金博士启动项目(9451063101002238)资助。

郑伟平(1979-),男,博士生,工程师,主要研究方向为网格、P2P 计算、流媒体分发等,E-mail: newmood@tom.com;齐德昱(1959-),男,教授,博士生导师,主要研究方向为分布式系统、网络安全、网格计算等;徐克付(1977-),男,博士,主要研究方向为网络安全。

享对象;b) 分片、数据驱动以及群内交换(swarming);c) 可以在以管理为主的中心实体,一般不参与媒体数据的传输。

其劣势体现在:

1) 数据块选择:BT 采用最少共享优先(rarest first)算法,这种策略忽略了流媒体的按序播放以及数据时限等约束;2) 对等点选择:BT 选择节点的标准单一,只考虑下载速率,而对邻近程度、延迟等因素缺乏考虑。而流媒体系统则需要权衡更多的综合因素;3) QoS 支持:可以在 BT 的握手过程中加入流媒体相关的 QoS 参数协商,例如,节点的带宽、延迟、最长等待时间等。

可见,BT 类型流媒体系统只需改进 BT 部分算法,而对 BT 系统构件无须改动。BT 类型流媒体系统的组成构件示意图如图 1 所示。

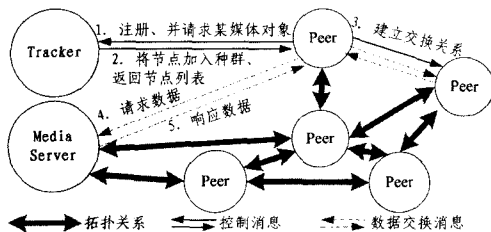


图 1 BT 类型流媒体系统结构图

节点可以选择从其它对等点下载媒体数据,也可以从媒体服务器下载数据。但是,BT 类型流媒体系统的媒体服务器是可选的。按照是否存在专门的媒体服务器,BT 类型流媒体系统可以分为混合式系统和纯 P2P 系统两类。

2.1 混合式系统

混合式系统部署有媒体服务器。BT 系统通过节点之间共享交换从媒体服务器上已下载的数据来减轻媒体服务器的负载压力;对于 BT 系统无法或无法及时提供的媒体数据,媒体服务器提供数据来源,保障了流媒体的播放质量。BASS, Toast 是典型的混合式系统,下面举例介绍混合式 BT 类型流媒体系统的工作原理。

2.1.1 BASS

BASS 系统^[14]的客户端同时从媒体服务器和其它 BT 节点上获取媒体数据。BASS 只须对 BT 部分做一个小小的修改:只请求当前播放点之后的数据块;在此前提下,BT 从其它对等点下载媒体数据,并存储到 BASS 本地数据库上。与此同时,客户端按照播放次序依次从媒体服务器上下载媒体数据块,但是跳过下列数据块:a) 由 BT 已下载到本地的块;b) 已发出请求且估计能按时到达的块。BASS 客户端工作模型如图 2 所示。

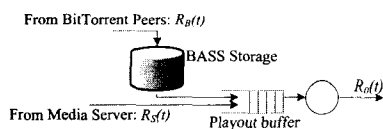


图 2 BASS 客户端模型^[14]

需要指出,由媒体服务器处获取到的块,也在 BT 的 have 消息中对外声明,并提供给其它节点共享。文献^[14]的仿真实验表明 BASS 能够减少媒体服务器大约 34% 的负载。

2.1.2 Toast

Toast^[2]的工作原理和 BASS 非常相似。Toast 系统设置有点播服务器,点播服务器使用 HTTP 协议,只提供客户请求的媒体数据块,不须向每个客户上载整个媒体对象。Toast 对 BT 客户端软件(4.4 版本,Python 语言编写)进行改写,添

加了两个功能模块:StreamWatcher 和 PiecePicker。

1) StreamWatcher:该模块负责 3 方面的工作:a) 维护节点的媒体播放位置(哪个块是当前播放块);b) 从点播服务器上下载急需且未下载的块;c) 并将这些块与 BT 模块下载的数据块一起写入本地库文件。

2) PiecePicker:Toast 规定,紧靠当前播放点之后一段区域内的数据块由点播服务器处请求,BT 模块只负责该区域之后数据块的下载。PiecePicker 修改了 BT 数据块选择策略,提供了 3 种可选的策略:a) 按播放次序下载;b) 按 beta 分布下载,即越靠近播放点其下载概率越大;c) 前面两种策略的混合,将 BT 负责的区域分为两段,前段采用按序策略,后段用 beta 策略,但只有前段所有块都收到后才开始后段的下载。

BASS 使用传统的最少共享优先策略,而 Toast 的块选择策略更为精细。另外,文献^[14]主要提出了 BASS 的模型框架,只对该框架进行简单的仿真验证,仿真节点规模也较小(350 个节点)。而文献^[2]在 BT 客户端软件的基础上编码实现了 Toast 系统,其实验结果更有说服力,在节点播放完毕后愿意充当种子的情况下,Toast 系统能够减轻点播服务器 70% 到 90% 的传输流量。但文献^[2]的实验系统规模依然偏小(300 个节点)。文献^[6]对 BASS 进行大规模的仿真模拟(131072 个节点),并对 BASS 的缓冲策略进行了改进,提出了双缓冲区(播放缓冲区和预备缓冲区)方案;为了实现大规模模拟,文献^[6]采用了流级别(flow-level)的模拟,并对 BT 模拟器的仿真过程进行了优化。文献^[6]的实验数据指出,该系统能够节省服务器 73% 的带宽开销,且具有较好的服务体验(用户平均缓存时间小于 2 秒)。

2.2 纯 P2P 系统

目前,绝大多数的 BT 类型流媒体研究^[1,3-5,7,11,12]都假设系统采用纯 P2P 方式,研究主要包括:数据块选择算法的改进、对等点选择算法的改进以及 TFT(tit-for-tat)算法改进,其目标是通过改进使播放更顺畅、缩短启动延迟并鼓励数据共享以改善公平性等。TFT 算法主要通过阻塞/疏通(choke/unchoke)操作来选择节点,实现系统公平性,因此可将其归纳入对等点选择范畴。第 3 节详细介绍纯 P2P 系统的数据块选择算法和对等点选择算法,值得注意的是,此类算法同样适用于混合式系统。

3 面向流媒体传输的 BitTorrent 算法改进

3.1 数据块选择算法

数据块选择算法解决当系统具备数据传输条件时选择哪个(些)数据块才能使系统性能达到近似最优的问题。根据节点角色不同,可分为请求端块选择算法和服务端块选择算法两类。

3.1.1 请求端块选择算法

请求端块选择算法必须解决节点本地性能与系统全局性能的矛盾。为了改善系统播放质量,必须按照播放次序请求数据块。但按序请求极大影响了系统数据的多样性,节点之间数据互补性大大削弱,系统吞吐量将急剧下降。目前,多采用按序请求和最少共享优先的混合策略,但在实现细节上有所差别。

文献^[7]提出了一种混合块选择策略,以概率 p 使用按序选择,以概率 $1-p$ 采用最少共享优先,在该实验场景下, p 取 0.8~0.9 时实验效果最好。文献^[7]还提出根据节点播放

连续指数(continuity index),来动态调整概率 p 大小的做法。文献[12]提出 Zipf(θ)算法,对第 k 块以概率 $1/(k+1-k_0)^\theta$ 进行选择, k_0 为第一个缺少的块的索引。文献[12]仿真实验表明 Zipf(1.25)的性能优于 Portion(0.5)、Portion(0.9)、最少共享优先和按序选择等策略(Portion(p))即为文献[7]的混合策略)。文献[11]对 Zipf(θ)进行修正,规定节点处于低缓存状态时采用按序选择。BiToS^[1]将播放点之后的块划分为高优先级(HP)和低优先级(LP)两个区域,HP 包含紧靠播放点之后节点急需的块;BiToS 以概率 p 请求 HP 内的块,以概率 $1-p$ 请求 LP 的块,两个区域内均使用最少共享优先选择数据块,实验表明 p 取 0.8 时性能最优。块选择算法实质上是对数据块的优先级进行赋值的过程。文献[3]将数据块在邻居节点上出现的次数与播放紧急程度加权表示为选择优先级。文献[20]指出很多选择算法都可以表示为基于优先级的选择策略,并使用“密度相依的跳变马尔可夫过程”(density dependent jump Markov process)进行最优块选择策略分析。文献[20]指出不同系统状态的最优块选择策略并不相同,提出了由服务器通告各节点调整最优块选择策略的分布式自适应算法。

3.1.2 服务端块选择算法

服务端的块选择算法主要优化媒体对象的扩散效率。文献[8]提出的智能种子(smart seed)算法在节点所请求的块中,优先服务该种子累计上传次数最少的数据块。文献[9]对智能种子算法进行了改进,提出了服务端的 PFS 块选择算法。PFS 不仅考虑了数据块的累计上传次数 τ ,还考虑了当前一轮时间内对该块的请求数 γ ,选择 γ/τ 比率最高的块作为上传数据。文献[8,9]所提的服务端块选择算法主要面向 BT 文件共享,但这些策略对 BT 类型流媒体系统同样具有借鉴意义。文献[11]在服务端选择从未上传过的块传输给新疏通的节点,这种做法保证新加入节点拥有其它节点所需的数据,增强其在 TFT 算法中的竞争力,也减少了新节点的启动时延。文献[3]在源端采用两阶段推-拉结合的块选择策略,在每个播放周期里,持续地把从未上传过的块主动推给源端上传数据最少的节点,直到所有块均已上传过最少一次;之后源端进入正常的拉模式,等待节点的请求。

3.2 对等点选择算法

传统 BT 采取基于速率的对等点选择算法,优先向下载速率最快的节点上传数据。BT4.0.0 版本开始考虑时间因素,节点优先选择最近疏通的节点,在同等条件下才考虑下载速率。文献[15]认为 BT 类型流媒体系统的对等点选择算法应当综合考虑节点带宽和延迟等因素。BT 从 tracker 节点处得到初始的邻居节点,这种缺乏位置感知能力的选择方法导致了 ISP 网络之间流量的剧增。文献[10]提出 BT 在构建覆盖网时应当尽量选择同一 ISP 网络的节点作为邻居,维护逻辑网络和物理网络的一致性。此策略同样适合于 BT 类型流媒体系统,选择位置邻近的节点还能够减少传输时延,进一步提高系统性能。

传统基于速率的 TFT 算法并不适合流媒体系统。文献[3]使用基于块的 TFT 算法,并验证了该算法的有效性;节点记录从各邻居节点 i 处下载的数据块数 D_i ,以及向其上传过的数据块数 U_i ,各邻居按照 $(D_i - U_i)$ 的大小进行排序,优先选择差值最大的节点,这种做法提高了新节点被疏通的概率。文献[5]提出了一种随机 TFT 算法,即在开始阶段随机选择节点进行疏通。基于块的 TFT 或者随机 TFT 都是照顾新节

点的做法,有利于节点尽早参与数据交换,减小了启动时延。

数据块的紧急程度也会影响对等点的选择。文献[11]优先选择请求紧急数据块(即该块位于播放窗口之内)的节点。文献[4]还提出了抢占式的节点选择策略,在服务节点上传带宽不足的情况下,可以剥夺正在下载预取(pre-fetching)数据的节点,而连接下载紧急块的节点。

4 性能与分析

本节讨论影响 BT 类型流媒体系统的重要性能参数和设计参数,并讨论它们之间存在的关系。

BT 类型流媒体系统的主要性能参数包括:

1) 启动延迟(Start-up Delay)SD:指节点加入系统并发出第一次数据请求后到系统开始播放之间的时间间隔。

2) 搜索延迟(seeK Delay)KD:支持 VCR 的点播系统上用户发出改变播放点指令后到开始播放之间的时间间隔。预取锚块(anchor point)可以有效地减小搜索延迟^[4]。

3) 连续指数(Continuity Index)CI:节点下载的能及时到达(即未超过播放时限)的数据块数与下载的总块数的比率。

4) 播放顺畅度(Playback Continuity)PC:PC 可表示为^[7]:

$$PC = (T - SD_{\min}) / L$$

其中, T 为节点从请求第一块数据到播完整个媒体对象所需时间, $SD_{\min}$ 为最短启动延迟, L 为媒体对象本身的时间长度, $PC \geq 1$ 。CI 与 PC 从两个不同的侧面反映了节点的播放质量。

5) 服务器负载 SL:混合式系统中衡量 BT 协助方式有效性的参数,SL 越小表示系统成本越低,节省带宽越多。

在系统设计和实现中下列参数对系统性能至关重要:

1) 播放窗口大小 W :文献[4]从减小 KD 的角度出发,根据用户播放跳转的统计数字,建议把 W 设置为 300 秒的长度。但 W 过长将降低 SD 的性能。 W 可表示为^[5]:

$$W = SD \times r / C$$

其中, r 为媒体的播放速率, C 为数据块长度。

2) 启动延迟:启动延迟也是设计参数。为了改善播放质量,须考虑开始播放的最佳时刻,既不过长等待又使播放质量(如 CI, PC)尽量好。文献[12]提出 LTA 准则,在满足下式时开始播放:

$$(Lk/K)/T \geq r(K-k)/K$$

其中, L 为文件总长度, k 为缓冲区中首段连续区域的块数, K 为文件总块数, T 是节点加入以来的时长, r 为播放速率。

3) 做种时长(Seed Time)ST:BT 系统中节点完成下载后继续留在系统中充当种子的时间长度。文献[16]基于随机流模型对 BT 类型流媒体进行建模分析,指出在系统稳态的情况下下载时间期望值随着 ST 的增大而变小,而与节点到达率无关。

结束语 本文关注 BT 协议在实现流媒体服务方面的适用性,并对现有 BT 数据块选择和对等点选择改进算法的研究进行总结和归纳。目前,BT 类型流媒体系统的研究还存在如下几个问题:

1) 网络安全、带宽不对称、NAT/防火墙等问题是实现实用 BT 类型流媒体系统所亟需解决的难题,这些问题都需要进一步研究。篇幅所限,本文并未就这些问题一一展开论述。例如,文献[13]在观察 BT 系统性能的基础上,发现 NAT/防

(下转第 149 页)

结束语 Web 服务的组合一般有两种方式: 编制 (Orchestration) 和编排 (Choreography)^[11], 但不管是哪一种, 都必须找到一定数量的整体能满足服务请求的元服务。在目前的 SOA 体系中这一工作基本上是由服务提供者 (或其代理) 完成的, 这样一种架构不能满足服务提供者主动推销服务的需要。本文将 Agent 作为 Web 服务的代理, 利用 Agent 的智能性和自治性提出了一个面向 Web 服务组合的主动服务聚合初步框架, 并通过仿真实验的性能分析很好地说明了该框架的可行性。

使用该方法进行服务聚合的基础是环境本体上的“意图-行为-实现”机制, 但基于该机制的服务能力与服务请求的匹配本质是服务与请求的功能匹配, 并没有考虑服务的 QoS 因素, 下一步的研究工作将会考虑这一点。

参考文献

- [1] Berners-Lee T, Cailliau R, Luotonen A, et al. The World-Wide Web[J]. Communications of the ACM, 1994, 37(8): 76-82
- [2] García-Sánchez F, Valencia-García R, Martínez-Béjar R, et al. An ontology, intelligent agent-based framework for the provision of semantic web services[J]. Expert Systems with Applications, 2009, 36(2): 3167-3187
- [3] 岳昆, 王晓玲, 周傲英. Web 服务核心支撑技术: 研究综述[J]. 软件学报, 2004, 15(3): 428-442
- [4] McIlraith S, Son T C, Zeng H. Semantic web services[J]. IEEE

Intelligent Systems, 2001, 16(2): 46-53

- [5] Gibbins N, Harris S, Shadbolt N. Agent-based Semantic Web Services[C]//Budapest, Hungary. Proceedings of the 12th international conference on World Wide Web, New York USA: ACM, 2003: 710-717
- [6] Charif Y. Dynamic Service Choreography based on Introspective Agents' Coordination[D]. English, Paris VI (version of April the 10th), 2008. <http://www-poleia.lip6.fr/~charif/Papers/YasmineCHARIF-PhDSummary.pdf>
- [7] 郑丽伟, 金芝. 需求驱动的主动网构实体聚合[J]. 软件学报, 2008, 19(5): 1083-1098
- [8] Sandholm T. Automated mechanism design: A new application area for search algorithms[C]//Kinsale, Ireland. Principles and Practice of Constraint Programming-CP 2003. Heidelberg: Springer, 2004: 19-36
- [9] Wang Puwei, Jin Zhi, Liu Lin, et al. Building toward Capability Specifications of Web Services Based on an Environment Ontology[J]. IEEE Transactions on Knowledge and Data Engineering, 2008, 20(4): 547-561
- [10] 叶荣华, 金芝, 等. 一种需求驱动的自主 Web 服务聚集方法[J]. 软件学报 (已录用)
- [11] Peltz C. Web Services Orchestration and Choreography[J]. IEEE Computer, 2003, 36(10): 46-52

(上接第 125 页)

火墙对系统性能的约束, 提出了一种基于代理的 P2P-CDN 混合式体系框架。

2) 目前 BT 类型流媒体系统的研究还缺乏实际部署的检验, 现有研究多数采用仿真、建模分析等手段, 而实际性能分析的研究极为少见。文献[2]编码实现了一个测试平台, 但规模极小, 用来验证具有海量访问需求的流媒体系统显得缺乏说服力。未来在实际环境中检验系统性能是研究工作的重点。

3) 未来 BT 文件共享系统和 BT 类型流媒体系统将共同存在, 如何实现两者的互通和联动是一个有趣的研究课题。

参考文献

- [1] Vlavianos A, Iliofotou M, Faloutsos M. BiToS: Enhancing BitTorrent for Supporting Streaming Applications[C]//9th IEEE Global Internet Symposium, Barcelona, Spain, 2006
- [2] Choe Y R, Schuff D L, Dyaberi J M, et al. Improving VoD server efficiency with bittorrent[C]//Proc. of the 15th international conference on Multimedia, Augsburg, Germany, 2007
- [3] Cai Q C, Zhang X J. An Effective Approach for Live Media Streaming Based on BitTorrent[J]. Journal of Communications, 2008, 3(6): 11-18
- [4] Janardhan V, Schulzrinne H. Peer assisted VoD for set-top box based IP network[C]//Workshop of Proc. of ACM SIGCOMM, August 2007
- [5] Shah P, Paris J-F. Peer-to-Peer Multimedia Streaming Using BitTorrent[C]//IEEE IPCCC'07, 2007
- [6] LaFortune R, Carothers C D, Smith W D, et al. Simulating Large-Scale P2P Assisted Video Streaming[C]//HICSS'09, 2009: 1-10
- [7] Hwang K-W, Misra V, Rubenstein D. Stored Media Streaming in BitTorrent-like P2P networks[M]. Columbia University, New York, Number cucs-024-08, April 2008
- [8] Bharambe A, Herley C, Padmanabhan V. Analyzing and Improving a BitTorrent Network's Performance Mechanisms[C]//

IEEE Infocom, Barcelona, Spain, 2006

- [9] Esposito F, Matta I, Michiardi P, et al. Seed Scheduling for Peer-to-Peer Networks[C]//IEEE NCA09, Cambridge, MA USA, 2009
- [10] Bindal R, Medved J. Improving Traffic Locality in BitTorrent via Biased Neighbor Selection[C]//Proc. of ICDCS, 2006
- [11] Carlsson N, Eager D L, Mahanti A. Peer-assisted On-demand Video Streaming with Selfish Peers[C]//Proc. IFIP/TC6 Networking'09, Aachen, Germany, 2009: 586-599
- [12] Carlsson N, Eager D L. Peer-assisted On-demand Streaming of Stored Media using BitTorrent-like Protocols[C]//Proc. IFIP/TC6 Networking'07, Atlanta, GA, 2007
- [13] Skevik K-A, Goebel V, Plagemann T. Analysis of BitTorrent and its use for the Design of a P2P based Streaming Protocol for a Hybrid CDN[R]. University of Oslo, June 2004
- [14] Dana C, Li D, Harrison D, et al. BASS: BitTorrent Assisted Streaming System for Video-on-Demand[C]//IEEE MMSP, Oct. 2005
- [15] Erman D. Extending BitTorrent for Streaming Applications[C]//Proc. of the 4th Euro-FGI workshop on "New Trends in Modeling, Quantitative Methods and Measurements", 2007
- [16] Parvez N, Williamson C, Mahanti A, et al. Analysis of BitTorrent-like Protocols for On-demand Stored Media Streaming[C]//Proc. ACM SIGMETRICS'08, Annapolis, 2008: 301-312
- [17] Magharei N, Rejaie R, Guo Y. Mesh or Multiple-Tree: A Comparative Study of Live P2P Streaming Approaches[C]//INFOCOM, 2007: 1424-1432
- [18] Seibert J, Zage D, Fahmy S, et al. Experimental Comparison of Peer-to-Peer Streaming Overlays: An Application Perspective[C]//Proc. of INFOCOM, 2007
- [19] Zhang X, Liu J, Li B, et al. CoolStreaming/DONet: A Data-Driven Overlay Network for Efficient Live Media Streaming[C]//Proc. IEEE INFOCOM'05, Miami, FL, USA, March 2005
- [20] Zhao Q, Lui JCS, Chiu DM. Exploring the Optimal Chunk Selection Policy for Data-Driven P2P Streaming Systems[C]//The 9th International Conference on Peer-to-Peer Computing, Sept. 2009