



(12)发明专利

(10)授权公告号 CN 104506897 B

(45)授权公告日 2017.12.19

(21)申请号 201410495196.6

H04N 21/238(2011.01)

(22)申请日 2014.09.24

H04N 21/643(2011.01)

(65)同一申请的已公布的文献号

申请公布号 CN 104506897 A

(56)对比文件

CN 103096177 A,2013.05.08,

US 2008155061 A1,2008.06.26,

CN 101087403 A,2007.12.12,

黄国伟.P2P流媒体数据块的优化调度策略.

《计算机工程》.2013,第39卷(第2期),

(43)申请公布日 2015.04.08

(73)专利权人 中国人民解放军理工大学
地址 210007 江苏省南京市秦淮区海福巷1号

审查员 曹珊珊

(72)发明人 胡超 陈鸣 邢长友 李兵
余沛毅

(74)专利代理机构 南京苏高专利商标事务所
(普通合伙) 32204

代理人 李玉平

(51)Int.Cl.

H04N 21/262(2011.01)

权利要求书1页 说明书4页 附图2页

(54)发明名称

P2P直播流媒体系统中的加权随机数据块选择方法

(57)摘要

本发明公开了一种P2P直播流媒体系统中的加权随机数据块选择方法,实现对等网络中端主机带宽资源的高效利用。该方法的基本思想是:对等方在与其他对等方建立应用层连接之后,通过交互缓存映射表感知数据状态;对等方给缓存空间中的数据块分配权重,距离播放时间越远的数据块所获得的权重越大;在选择要下载的数据块时,对等方根据各个数据块的权重进行加权随机选择,并向已经拥有该数据块的对等方集合中的一个对等方发送请求。本发明具有对等网络资源利用率高、应对突发访问能力强等优点。

1. 一种P2P直播流媒体系统中的加权随机数据块选择方法,其特征在于,包括以下步骤:

A. 对等方对视频数据缓存空间大小进行初始化设置,设置各个数据块的权重大小,之后建立三个数据块集合,然后转入步骤B;三个数据块集合分别为RequestSet集合、DerivedSet集合和WaitSet集合,其中RequestSet集合表明目前已请求但尚未得到的数据块,DerivedSet集合表明已经得到的数据块集合,WaitSet集合表明尚未请求的数据块,初始化设置时,将缓存中所有数据块加入到WaitSet集合中,而其他两个集合置为空;

B. 对等方与其他对等方建立应用层连接,并定期交互缓存数据信息以感知当前对等网络中数据状态,然后转入步骤C;

C. 对等方在每次经历一个视频数据块播放完成时间之后,然后更新WaitSet集合中各个数据块的权重,并转入步骤D;

D. 对等方计算WaitSet集合中所有数据块总权重,然后得到WaitSet集合当中各个数据块被请求的概率,并转入步骤E;

E. 确定要请求的数据块,并从已拥有该数据块的其他对等方集合当中随机选择一个,然后向该对等方发送请求,并将该数据块从集合WaitSet转移到RequestSet,并转入步骤F;

F. 一旦对等方在规定的时间内仍未收到该数据块,重新选择另一个对等方,并向其发送数据块请求,而如果对等方接收到该数据块,就将该数据块从集合RequestSet转移到DerivedSet;

根据各个数据块存储区距离流媒体播放器的距离设置数据块的权重大小,其中距离播放器第*i*个数据块分配的权重为*i*;

步骤D中,WaitSet集合中所有数据块总权重为*W*,WaitSet集合当中各个数据块被请求的概率为*w/W*,其中*w*为数据块所分配的权重。

2. 如权利要求1所述的P2P直播流媒体系统中的加权随机数据块选择方法,其特征在于:步骤E中,对等方通过产生[0,1)之间的随机数确定要请求的数据块。

P2P直播流媒体系统中的加权随机数据块选择方法

技术领域

[0001] 本发明涉及一种对基于P2P直播流媒体系统资源调度的方法,一种对对等网络资源进行高效利用的方法,具体地说是一种P2P直播流媒体系统中的加权随机数据块选择方法,加权随机化数据块选择算法,即Weighted Randomized Chunk Selection Algorithm, WRCS,属于网络数据通信技术领域。

背景技术

[0002] 基于因特网传输直播电视节目已经成为一种流行的多媒体服务。为了取得更好的性能,服务提供商在直播多媒体流系统中使用了内容分发网(Content Delivery Network, CDN)和P2P(Peer-to-Peer)技术。特别是,P2P技术根据应用层的连接关系将用户构建为一个覆盖网络,进行了一定程度的组织和管理,从而能够利用了这些用户的带宽资源和数据资源,极大地降低了服务器所承担的数据上传负载。当前在因特网中大规模部署的直播多媒体流系统,如Coolstreaming^[1]、PPLive^[2]等,都采用了P2P技术。

[0003] 从技术上看,P2P技术能够为服务器分担大量数据传输任务的前提是对等网络中存在足够多的对等方感兴趣的内容资源。为此,对等方通过向对等方发送连接请求,建立邻居关系,从而能从对等方而不是从服务器得到所需的内容。一般而言,对等方之间数据的差异性越大,对等方之间彼此感兴趣数据块的数量越多,从P2P网络并行下载潜力也就越大。例如,Cohen首次提出并成功地运用于BitTorrent中的最稀优先(rarest first, RF)算法^[3],促进了稀缺资源在对等方之间的扩散,有效增加了对等方之间数据块的差异,加快了内容分发的速度。

[0004] 尽管基于P2P的直播流系统的数据块分发方式与BitTorrent有相似之处,但由于直播流数据块对时延敏感,所有数据块都必须在被播放之前到达对等方处,导致两者在数据块分发方式上存在着实质性差异,简单地照搬现有的相关算法将导致直播流系统的效率低下。因为一旦某个数据块出现超时,直播流对等方只能丢弃数据块转而播放后续的视频数据,降低了用户收看节目的体验质量。直播流的实时性使新产生的数据块最为稀缺,从而导致对等方都会集中优先请求新产生的数据块,无法顾及数据块播放的紧迫程度,导致丢包率增加。然而,如果仅根据数据块的播放期限来选择数据块,又会降低对等方之间数据块差异,无法发挥P2P机制应有的作用。此外,突发访问是直播流系统中需要解决的一个重要问题。当某些热门节目开始播放的时候,会有大量用户在非常短的时间内访问该频道。如果系统不能很好处理这些用户的数据块请求,将会出现严重的性能问题。

[0005] 为此,通过设计合理的数据块选择算法对于提高对等网络资源的利用率,改善用户体验质量具有非常重要的作用。

[0006] 参考文献

[0007] [1]X. Zhang, J. Liu, B. Li, et al. Coolstreaming/DONet: a data-driven overlay network for efficient live media streaming. IEEE INFOCOM, Miami, USA, 2005: 2102-2111.

[0008] [2]PPLive. <http://www.pptv.com>.

[0009] [3]B. Cohen. Incentives build robustness in BitTorrent [C]. In Workshop on Economics of Peer-to-Peer Systems, Berkeley, CA, USA, June 2003.

发明内容

[0010] 发明目的:针对目前基于P2P直播流媒体系统中的数据块选择算法难以高效利用对等网络资源问题,提出一种以提高系统带宽利用,改善系统服务质量为目标的数据块选择方法。

[0011] 技术方案:一种P2P直播流媒体系统中的加权随机数据块选择方法,着重描述了对等方在选择要请求的数据块时所采用的方法,该方法包括以下步骤:

[0012] A. 对等方对视频数据缓存空间大小进行初始化设置,并根据各个数据块存储区距离流媒体播放器的距离设置权重大小,其中距离播放器第 i 个数据块分配的权重为 i ,之后建立三个数据块集合,其中RequestSet集合表明目前已请求但尚未得到的数据块,DerivedSet集合表明已经得到的数据块集合,WaitSet集合表明尚未请求的数据块,并将缓存中所有数据块加入到WaitSet集合中,而其他两个集合置为空,然后转入步骤B;

[0013] B. 对等方与部分同时收看直播节目的其他对等方建立应用层连接,并定期交互缓存数据信息以感知当前对等网络中数据状态,然后转入步骤C;

[0014] C. 对等方在每次经历一个视频数据块播放完成时间之后,将该被播放的数据块从集合RequestSet、DerivedSet或WaitSet中清除,并将新产生的数据块添加到集合WaitSet当中,然后更新WaitSet当中各个数据块的权重,并转入步骤D;

[0015] D. 对等方计算WaitSet集合中所有数据块总权重 W (总权重为各个数据块的权重之和),然后得到WaitSet集合当中各个数据块被请求的概率为 w/W ,其中 w 为数据块所分配的权重,并转入步骤E;

[0016] E. 对等方通过产生 $[0, 1)$ 之间的随机数确定要请求的数据块,并从已拥有该数据块的其他对等方集合当中随机选择一个,然后向该对等方发送请求,并将该数据块从集合WaitSet转移到RequestSet,并转入步骤F;

[0017] F. 一旦对等方在1秒钟之内仍未收到该数据块,重新选择另一个对等方,并向其发送数据块请求,而如果对等方接收到该数据块,就将该数据块从集合RequestSet转移到DerivedSet。

[0018] 本发明相对现有技术具有以下优点:

[0019] 1. 能高效利用对等网络资源:本发明提出的WRCS方法能够通过在对等方之间进行负载均衡,从而充分利用对等网络的闲散资源,达到直播流媒体系统服务质量的提升。

[0020] 2. 应对突发访问的能力强:WRCS方法能够通过将一些突发的稀缺数据块请求丢弃,从而达到隐式的访问控制,增强了直播流媒体系统应对突发访问的能力,同时也间接提高对等网络数据的差异性,以有利于对等网络资源的利用。

附图说明

[0021] 图1为本发明实施例运行的网络环境;

[0022] 图2为本发明实施例对等方进行数据块选择处理的流程图。

具体实施方式

[0023] 下面结合具体实施例,进一步阐明本发明,应理解这些实施例仅用于说明本发明而并不用于限制本发明的范围,在阅读了本发明之后,本领域技术人员对本发明的各种等价形式的修改均落于本申请所附权利要求所限定的范围。

[0024] 首先需要给出本发明提供的直播流媒体系统资源调度方法所需要的环境,如图1所示:在若干台Intel-Linux架构的PC机上安装并运行具有本发明WRCS数据块选择方法的软件,同时在另外一台Intel-Linux架构的PC机上安装并运行定期产生新数据块的流媒体服务器,将这些PC机的100/1000Mbps以太网卡连接到接入因特网的局域网交换机上,使之与因特网上的其他PC机之间能够相互进行通信。

[0025] 运行本发明提供的基于P2P直播流媒体系统数据块选择方法的PC机系统配置如下:在Intel-Linux架构的PC机上安装并运行基于本发明的WRCS软件,PC机的100/1000Mbps以太网卡与网络交换机相连。这些PC机硬件的主频4.0GHz及以上的Core双核CPU的PC机,内存 ≥ 2 GB,硬盘120GB,运行Red Hat 6.1操作系统。

[0026] 图2给出了本发明基于WRCS方法的对等方数据块选择方法的处理流程,该流程开始于步骤S101,对等方对数据缓存区和各个数据块的权重进行初始化,根据各个数据块存储区距离流媒体播放器的距离设置权重大小,其中距离播放器第 i 个数据块分配的权重为 i ,并建立三个数据块集合RequestSet、DerivedSet和WaitSet,其中RequestSet集合表明目前已请求但尚未得到的数据块,DerivedSet集合表明已经得到的数据块集合,WaitSet集合表明尚未请求的数据块,并将缓存中所有数据块加入到WaitSet集合中,而其他两个集合置为空,然后转S102。

[0027] 在步骤S102中,对等方与其他对等方建立应用层连接,相互之间定期交互数据缓存信息来感知其他对等方的数据拥有情况,然后转S103。

[0028] 在步骤S103中,对等方计算WaitSet集合中的总权重 W 以及各个数据块的请求概率 w/W ,其中 w 为数据块所分配的权重,并通过产生 $[0, 1)$ 之间的随机数确定要请求的数据块(这里每个数据块的概率都已经得到了,而且这些概率之和为1,将这些数据块的概率排列形成多个概率区间,产生的随机数落在哪个区间就选择对应的数据块),然后转S104;

[0029] 在步骤S104中,对等方从拥有该数据块(步骤103中要请求的数据块)的其他对等方中随机选择一个并向其发送请求,同时修改RequestSet和WaitSet集合,然后转S105以及S106。

[0030] 在步骤S105中,对等方等待一个视频数据块播放时间之后转回S103。

[0031] 在步骤S106中,对等方判断在1秒钟之内是否接收到数据块,如果接收到转S108,否则转S107。

[0032] 在步骤S107中,对等方重新选择另外一个对等方并向其发送数据块请求,然后转回S106。

[0033] 在步骤S108中,对等方接收到数据块之后对DerivedSet和RequestSet这两个集合进行更新,将该数据块从集合RequestSet转移到DerivedSet。

[0034] 本方法可以通过中断方式退出。

[0035] 本实施例给出了在某个ISP中部署若干PC机运行基于本发明WRCS方法的软件对直

播流媒体系统数据块进行选择 and 调度,以掌握本发明的运行状况,为制定合理的数据块选择方法提供科学依据。

[0036] 假定各个PC机经100/1000Mbps速率的以太网链路 with 因特网相连。在若干PC上运行基于本发明的WRCS软件,通过收集和统计各个运行PRSR软件的视频数据块到达信息,就能分析出直播流媒体系统中对等网络的资源利用情况以及系统的服务质量是否得到提高。

[0037] 本发明未涉及部分均与现有技术相同 or 可采用现有技术加以实现。

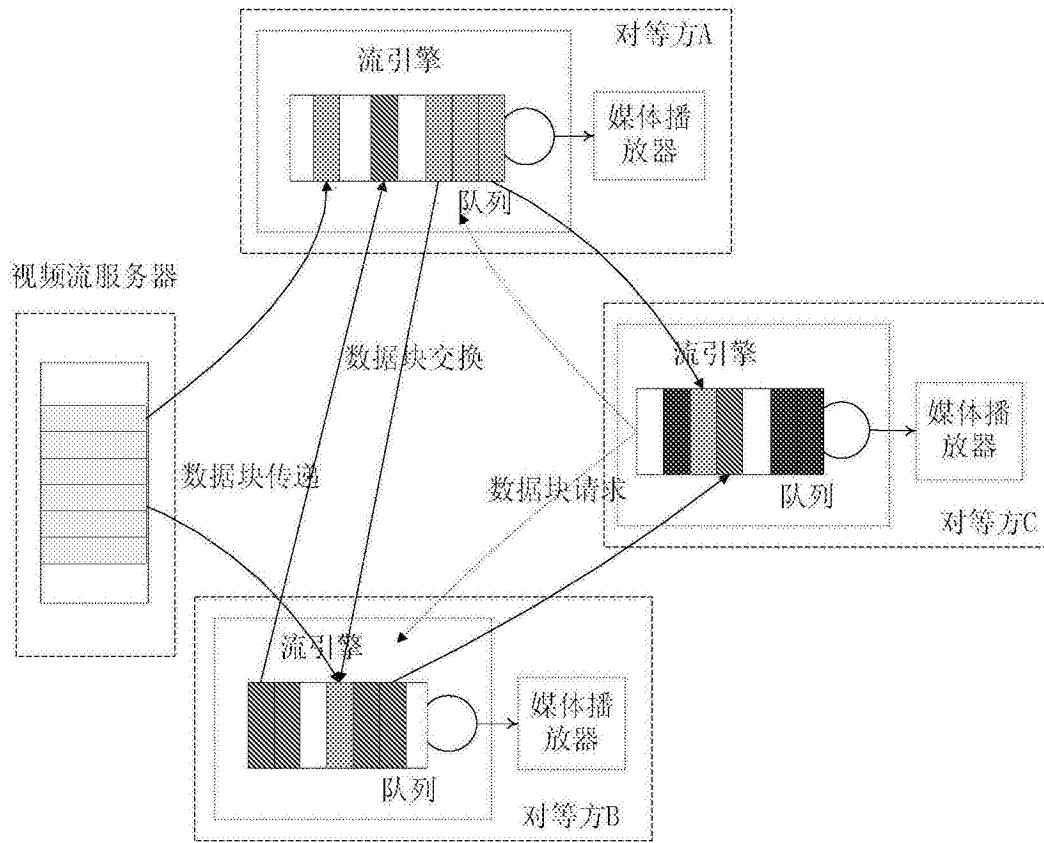


图1

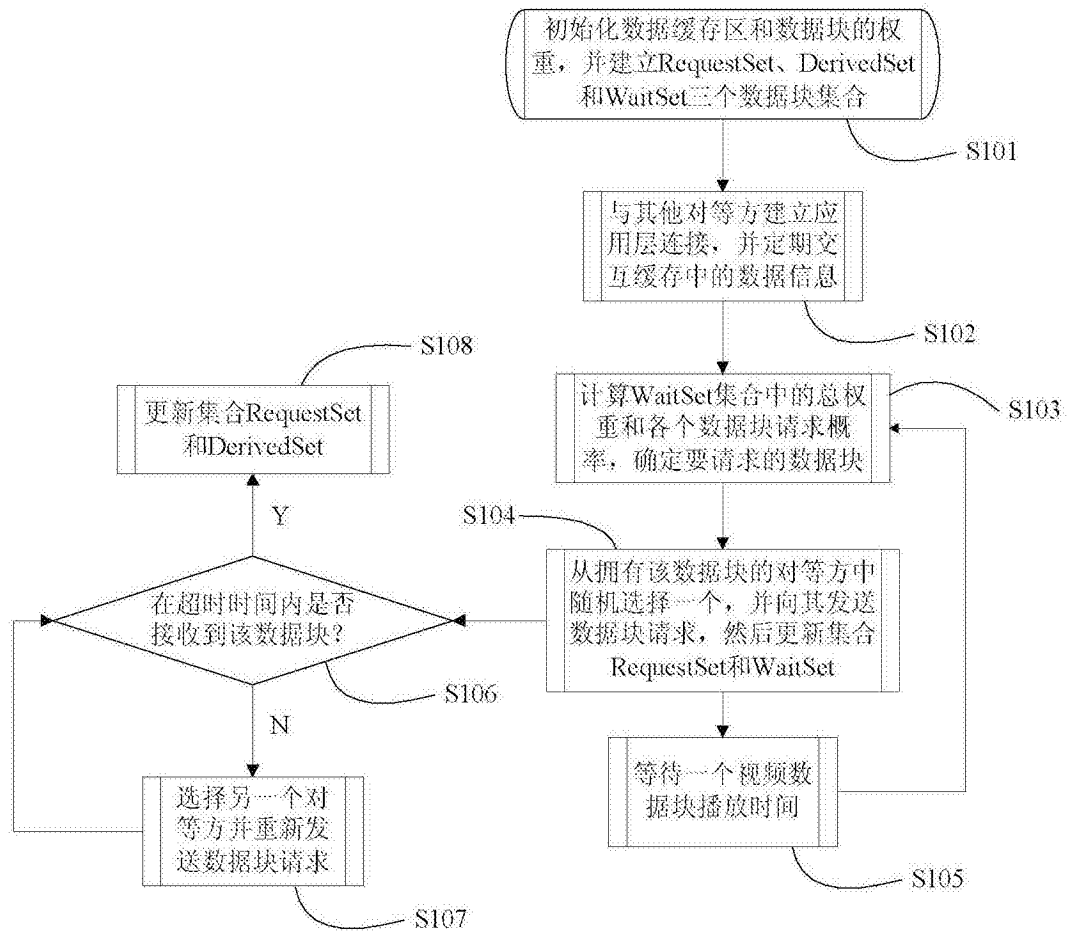


图2