典型 P2P 流媒体模型①

蒋良军 李太君 (海南大学 信息科学技术学院 海南 海口 570228)

摘 要: 分析和比较了几种典型的 P2P 流媒体模型，介绍了 P2P 流媒体不同模型常用的数据调度算法，并指出了基于应用层多播的树模型和基于 Gossip 协议的网状模型的区别,最后指出了 P2P 流媒体的研究方向。

关键词: 流媒体；多播树协议；gossip 协议；数据调度

Review of Typical P2P Streaming Media Models

JIANG Liang-Jun, LI Tai-Jun

(College of Information Science and Technology, Hainan University, Haikou 570228, China)

Abstract: This paper analyzes and compares several typical P2P streaming media models and introduces peer selection algorithm used by different P2P streaming media models. It points out the differences between P2P streaming media service models based on multicast tree protocol and gossip protocol. Some issues are mentioned for further research.

Keywords: streaming media; multicast-tree protocol; gossip protocol; data scheme

流媒体应用中，基于传统客户端/服务器结构的媒体服务器容易成为系统瓶颈，其扩展性不高，不适合大规模数据的分发。IP 组播能减轻服务器和网络负载，但众多原因使之在短期内难以广泛实现，CDN 通过把服务和内容“推”向网络的“边缘”，也能减轻服务器和网络负载，但其昂贵的费用使得一般 ICPs（互联网内容提供商）无法承担，基于 P2P 流媒体中，每个节点都充当了服务器与客户机的功能，为整个系统贡献自己的存储和带宽资源。因此，P2P 流媒体成为研究的热点。

目前 P2P 流媒体模型主要分为两大类：一类是基于应用层多播的树模型(包括单树和多树),典型代表是采用单树的 PeerCast[1]，ZigZag[2]和采用多树的 SplitStream[3]，CoopNet[4]，P2PCast[5]；另一类是基于 Gossip 协议的无结构化(或网状)模型，典型代表是 CoopStreaming[6]，DONET[6]。

本文分析了上述典型 P2P 流媒体模型，比较了各自的优缺点，然后介绍了 P2P 流媒体不同模型常用的数据调度算法，讨论了基于应用层多播的树模型和基于 Gossip 协议的网状模型的区别。

1 基于单多播树的P2P流媒体模型

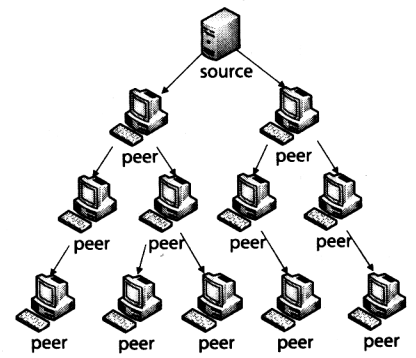

图 1 单多播树结构

单多播树即将系统中所有节点在逻辑上组成一棵

① 基金项目:海南省自然科学基金(807002);海南大学 2009 年度科研项目(hd09xm86)
收稿时间:2009-05-19

分发树，系统的数据分发流程均按照该分发树的拓扑结构由上至下分发，所有的中间节点和叶子节点都是P2P 网络中的节点，中间节点必须转发数据到其他节点，而叶子节点不需要。此外，节点维护工作

可以放在做为根节点的服务器中，或者选择树中其他一部分节点共同维护。单多播树结构如图 1 所示。

许多早期 P2P 流媒体模型采用这种结构，最简单的模型是 PeerCast[1]模型。在PeerCast 中，节点被组织成一个树状结构，树的父节点给子节点提供服务。节点的加入和离开策略都很简单，但也容易导致树的不平衡。如果节点离根节点越远，则数据的时延就越大。因此，树的深度应该尽可能短。但是每个节点的有限输出带宽限制了节点的宽度。理想的组播树是在深度和宽度之间能够有效的平衡。事实上，当所有节点的深度都为 1 的时候就退化成了传统的客户端/服务器端模型了。

ZigZag[2]能够有效构造组播树，它定义了一整套完整的树的构造规则。ZigZag 中 peer 之间的关系有两种：一种是逻辑关系——簇管理机构，一种是真正负责数据分发的物理关系——组播树。ZigZag 利用层次簇思想，转发节点只与少量固定节点数目的节点联系，每个节点平均维护负载为 O(K)，保证树的深度维持在 O(logN)，N 为系统中节点的数量，簇的管理和数据分发由不同节点完成，从而使节点所带子节点数目最多为 O(K^2)，与参与节点数目无关，且节点退出只影响局部节点，不影响根节点。此外，ZigZag 还拥有许多优良特性：节点数目 N 可以任意多；控制协议的开销低；新节点加入快速，簇的维护成本低。

上述单多播树的 P2P 流媒体模型的缺点表现在：首先，每个节点仅与它的父节点相连，一旦父节点瘫痪或与之相连的网络中断，那么这个节点与它所有的后续节点将无法得到数据，重新恢复需要一定得时间，对于实时流媒体来说，将造成难以估量的损失。其次，中间节点至少需要转发两份数据到它的子节点，而叶子节点不需要转发数据，在实际情况下，叶子节点的增长速度明显比中间节点快，这样对于各节点而言，显得很不公平。再者，带宽的理想模式应该是随着单播树的深度而递减，受带宽限制，一个远离媒体源的叶子节点，即使它的父节点有足够带宽也可能收不到理想的数据。

2　基于多个多播树P2P流媒体模型

在单播模型中，系统只有部分节点参与负责数据分发，所有叶子节点资源都没有利用。多树模型通过引入以源节点为根节点的多个分发树，每个分发树通过多描述编码 MDC 只分发源的一层数据来克服单树模型的上述两个问题。多个多播树结构如图 2 所示。多树模型的目标是将同一个节点放在不同分发树的多个位置，如图中节点 A，B，C，D，E，F 放在两个分发树的不同位置。这些位置可以随机选择或是采取某种确定性的算法来实现。多个多播树模型的典型代表有 CoopNet[3]，SplitStream[4]，P2PCast[5]。

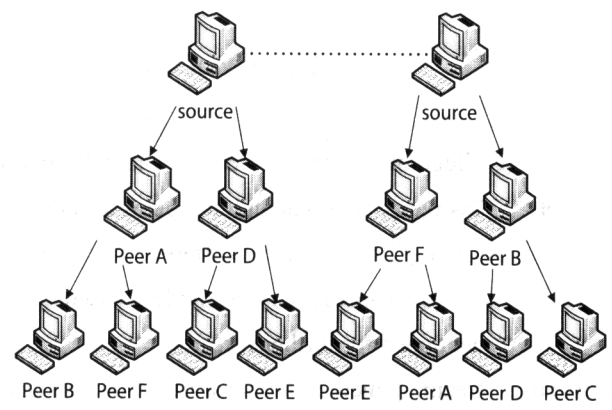

图 2　多个多播树结构

CoopNet[3]是 Microsoft Research 的第一个多树直播视频流媒体网络系统。CoopNet 使用客户端的多树来减轻流媒体服务器的负载，并且帮助服务器克服瞬间冲击的问题。CoopNet 受益于一个比分布式更高效的中心的管理，有一个指定的工作站负责管理节点的加入和离开。工作站把多播树的整个结构存储在内存中。当一个节点开始接受现场直播的流媒体时，这个节点与工作站接洽加入的操作。工作站从保存在内存的多播树中找到一个合适的位置，把这个节点的父节点返回给这个节点。

SplitStream[4]是 Microsoft Research 提出的第二个方案，它与 CoopNet 不同的是它旨在提供纯粹的对等网络服务。SplitStream 的关键思想是把媒体数据流分成 K 个独立的码流，可以通过 MDC[7]实现，然后为每个码流构造一个组播树，形成一个“森林”，每个节点可以根据带宽情况选择接收其中的几个码流。树的构造的主要困难是要求每个节点只在某棵树中为中间节点，而在其他树中都为叶子节点。SplitStream 的实现依赖于 Pastry[8]和 Scribe[9]。Pastry 是一个类似 Tapstry，Chord，CAN 的可扩展的，自

组织的 P2P 的基础架构。

P2PCast[5]模型也是把媒体数据分成 K 个独立码流，然后为每个码流构造一个组播树，形成一个“森林”，这和 SplitStream 模型很相似，但 P2PCast 模型不依赖 Pastry 和 Scribe。其次，P2PCast 模型对用户的可用上行带宽采取了更多的控制。P2PCast 模型在树的构造上采用分布式，这也不同于 CoopNet 采取集中式的方法。

上述多播树的缺点是需要同时维护多个组播树，这会导致开销过大，另外必须保证多路径传送时的数据同步，再者这种设计也很难优化。

3 基于Gossip网状P2P流媒体模型

在基于应用层多播的树模型中，都需要显式的定义节点之间的关系，而在基于 Gossip 协议的网状模型中，节点之间不需要构造复杂的拓扑关系。Gossip 算法中，节点随机的给系统中的部分节点发送消息，每个接收到消息的节点继续向其他节点发送消息，重复这个过程，直到消息被发送给系统中的所有节点。正是这种消息转发的共享性和灵活性，基于 Gossip 的算法成为目前流行的在 P2P 流媒体系统中分发消息的算法。网状模型结构如图 3 所示。基于 Gossip 协议的网状 P2P 流媒体模型典型代表有 CoopStreaming[6]，DONET[6]。

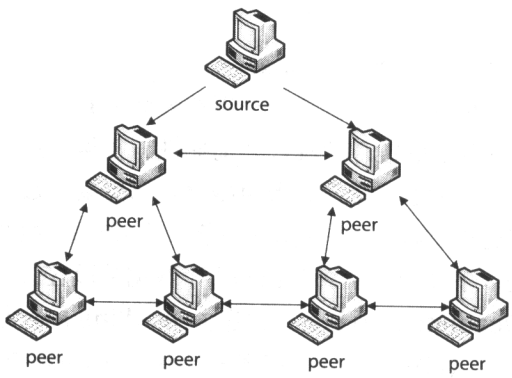

图 3 网状模型结构

CoopStreaming[6]模型是一个融合了 C/S 架构和 P2P 架构的混合式模型，在 CoopStreaming 中，P2P 并不是用来取代 C/S 模式，而是作为一种补充，每个用户节点都有两种方式获得媒体数据：从服务器直接获取或者通过其他节点获取。CoopStreaming 每个节点维护一个伙伴列表 PartnerList，节点可以从服务器或者伙伴节点获取媒体数据。其实，服务器作为一个特殊的节点是所有节点的伙伴。节点和伙伴节点不断的交换各自的缓存信息，然后根据伙伴的缓存信息，通过一定的数据调度算法从伙伴节点获取媒体数据。节点的伙伴列表并不是固定的，节点在运行过程中会不断的优化伙伴列表。CoopStreaming 的服务器除了直接提供数据服务之外，还负责维护所有节点的列表 PeerList，管理网络中所有节点，包括新节点的加入，节点的正常和异常退出。

在 DONet[6]中，节点的伙伴及伙伴之间数据的传输方向并不固定，每个节点既是数据的接收者，也是数据的提供者。伙伴之间根据各自的缓存的数据情况进行数据交换，所以节点和伙伴需要相互知道所缓存的数据的内容。在 DONet 中，视频数据被分割成相同大小的片断，用一个缓存映射 BM(buffer map)来表示节点中是否拥有某个片断的数据。节点和伙伴通过不断交换 BM 来了解相互间的缓存情况。

在基于 Gossip 协议的 P2P 流媒体系统中，每个节点动态的和其他节点交换数据，因此，这种系统通常需要比较大的缓存，另外系统的启动延时相对比较大。但是，因为每个节点的数据来源并不依赖于某个特定的父节点，所以系统有更强的健壮性。表 1 比较了 P2P 流媒体模型典型代表的优缺点。

表 1 P2P 流媒体模型典型代表的优缺点比较

	优点	缺点
PeerCast	节点加入和离开策略都很简单	容易导致树的不平衡
ZigZag	节点所带子节点数与参与节点数无关，且节点退出只影响局部节点，不影响根节点	稳健性不高
SplitStream	部署 MDC，部分解决了可靠性问题，底层实现基于分布式散列表覆盖网络	依赖 Pastry 和 Scribe 实现，引入了冗余编码，使网络效率降低
CoopNet	使用客户端的多播减轻流媒体服务器负载，帮助服务器克服瞬间冲击问题，受益于比分布方式更高效的集中管理	需同时维护多个组播树，导致开销过大
P2PCast	部署 MDC，对用户可用上行带宽采取了更多控制，树构造上采用分布式，不依赖 Pastry 和 Scribe 实现	MDC 编码效率较低
CoopStreaming	不依赖某个节点，保证系统可靠性，取得网络效率	将服务器作为整个分发系统数据源，给服务器带宽造成巨大负担
DONET	不依赖某个节点，保证了系统可靠性，取得了网络效率	纯分布式模式容易造成对整个系统网络带宽冲击

4　P2P流媒体不同模型的数据调度算法

P2P 流媒体模型设计的好坏依赖于合理的数据调度算法。在已有的 P2P 流媒体服务体系中，较常用的数据调度算法有补丁流调度算法，周期补丁流调度算法和 DONet 中使用的数据调度算法。补丁调度算法和周期补丁调度算法即利用组播媒体流同时服务多个用户。补丁算法能够大量节约服务器的资源、响应更多用户、缩短用户等待时间。补丁算法更适用基于应用层多播的树模型(包括单树和多树)，而 DONet 中使用的数据调度算法适用于基于 Gossip 协议的网状模型。下面简单介绍补丁流调度算法，周期补丁流调度算法和 DONet 中使用的数据调度算法。

4.1 补丁流调度算法

补丁算法(PATCHING)最早是由 Hua 等在 1998 年提出的[10,11]。该算法给节目的第一个请求建立一个共享组播流，后面的对同一节目的请求加入该共享流。同时系统再生成一单播媒体流补偿损失的数据，该单播流称为补丁流。共享流(又称常规流)，包含整个节目内容。补丁流与共享流的时间间隔称为补丁窗口，当补丁窗口过大时系统生成新的共享组播流。算法示意图如图 4 所示。

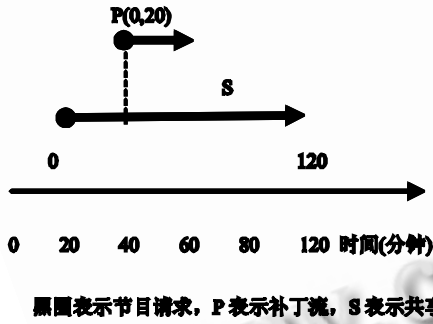

图 4　补丁算法示意图

图 4 表示在共享流播放到第 20 分钟时有新的请求到来，服务器让它加入共享流 S，并用补丁流 P(0,20)把前 20 分钟的节目内容发送给该用户。补丁调度算法要求用户具备同时接收两个通道媒体流的能力，共享流先缓存到磁盘缓冲区中，补丁流立即播放。补丁调度算法在带宽资源充足的情况下基本不需要用户等待，同时通过共享组播流减少了服务器发出的视频流数目，明显提高了系统资源利用率。

4.2 周期补丁流调度算法

由于补丁调度算法不能避免为一个媒体节目生成过多的组播流，清华大学向哲博士于 2001 年提出了周期补丁流调度算法以改进这一问题[12,13]。图 5 为周期补丁算法的流程图。

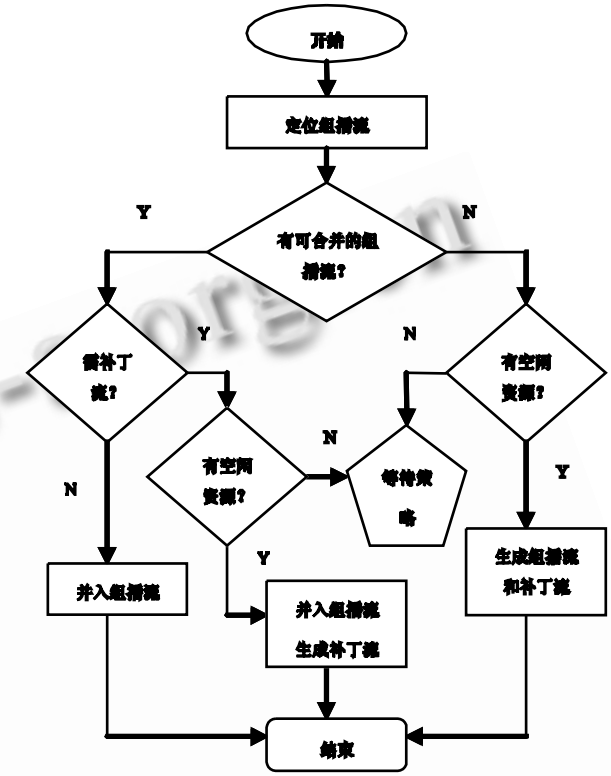

图 5　周期补丁算法流程图

4.3 DONet 中的数据传输策略

DONet[6]中，节点用一个缓存映射(Buffer Map, BM)表示自己的缓存状况，并周期性与邻居节点交换 BM 了解彼此缓存情况，节点根据自己想要的数据和 BM 来选择邻居节点，并向这些节点请求缺失的数据。调度受以下 2 个条件约束：(1)每个数据片段应该在播放的 Deadline 之前到达，使错过 Deadline 的片断尽可能地少；(2)每个节点的带宽情况不同。因为系统中，如果某个片段的提供者越少，就越难满足 Deadline 请求。因此，DONet 统计每个片段的提供者数量，并采用稀少数据优先算法，先获取只有一个提供者的数据片段，然后获取有 2 个提供者的数据片段，依此类推。获取有多个提供者的片断时，在保证 Deadline 的前提下，选择带宽最大的提供者。DONet 数据传输过程需要经过 3 个步骤：(1)节点 A 向节点 B 发送 BM 消息；(2)节点 B 根据节点 A 的 BM 消息发送数据请求；(3)节点 A 根据收到的请求给节点 B 发送相

应数据。调度完成后，同一个提供者的数据片段被表示成 BM 的形式传给响应的提供者，提供者通过一个实时的传输协议传输数据，在 DONet 中部署了 TFRC (TCP-Friendly Rate Control)协议。因为源节点作为数据的提供者，拥有 BM 中所有的数据片段，一般来说依据 DONet 的调度算法，数据源的负载应该不会过于拥塞。这也正是 DONet 成为第一个成功运行的大规模直播系统的原因，但这种调度算法在视频传输延迟和系统控制开销上比较大。

5 P2P流媒体模型比较

如表 1 中比较，单树结构的模型取得了网络带宽的有效性，然而不管树如何构建，一个显著的问题是单树结构中离服务器较近节点的离开都不可避免地会影响到后续的子节点，也就是在树结构模型中部分地牺牲了可靠性。SplitStream 中部署 MDC，部分地解决了可靠性的问题，但引入了冗余的编码，用网络效率来换取可靠性。在基于 Gossip 协议的模型中，DONet 通过 Gossip 协议获取系统中其它节点的信息建立连接，并互相交互数据。因为不依赖于某个节点，DONet 在保证了系统的可靠性的同时取得了网络效率，但牺牲了延时，而且其纯分布式的模式容易造成对整个系统网络带宽的冲击。在 CoopStreaming 的模型中，将服务器同时作为整个分发系统的数据源，无疑给服务器的带宽造成了巨大的负担，同时分层编码 FGS 的加入会使得数据调度算法变得较为复杂。

基于应用层多播的树模型(包括单树和多树)和基于 Gossip 协议的无结构化(或网状)模型的比较[14]如表 2 所示。

表 2 三种 P2P 流媒体模型的比较

模型结构	可靠性	控制开销	服务规模	管理复杂度	网络适应性	数据传输率
单树结构	较差	较低	任意	低	差	较低
多树结构	高	适中	大中型	高	较好	适中
网状结构	较高	高	中小型	一般	很好	高

表 2 表明在可靠性，网络适应性，和数据传输速率方面，单树结构明显不如多树结构和网状结构，但在服务规模方面，单树具有优势。网状结构在网络适应性，控制开销和数据传输速率上高于树状结构，但是它只适应中小型服务规模。

6 结语

从以上 P2P 流媒体典型模型的分析中，可以看出，各种模型虽然有很多各自的优点，但其自身几乎都存在一定的问题，模型中数据的调度算法，节点的加入和退出算法还需要进一步优化，以适应不断变化的网络要求。现有的 P2P 流媒体服务体系尚未成熟，处于研究阶段，需要研究并解决如下问题：P2P 应用中的网络穿透问题(NAT)，P2P 网络资源的搜索，应用层组播的安全性问题，激励机制，QoS 保障机制。伴随着 3G 商用，即将产生一个巨大的无线多媒体应用市场。P2P 技术如何在无线环境下应用于更加动态和不可靠的网络，这对于 P2P 技术的广泛应用也是很大挑战。

参考文献

1 Deshpande H, Bawa M, Garcia-Molina H. Streaming Live Media over a Peer-to-Peer Network, Stanford University. 2001,(8).

2 Duc A, Tran KA, Hua TD. ZigZag: An Efficient Peer-to-Peer Scheme for Media Streaming. Proc.of IEEE INFOCOM'03, 2003.1283－1292.

3 Castro M, Druschel P, Kermarrec A M, et al. Splitstream: High-bandwidth Content Distribution in a Cooperative Environment. Kaashoek F, Stoica I. eds. IPTPS 2003, LNCS 2735, 2003.292－303.

4 Padmanabhan V, Wang H, Chou P, et al. Distributing Streaming Media Content using Cooperative Networking. Proc. of the 12th International Workshop on Network and Operating System Support for Digital Audio and video, Miami, Florida 2002.

5 Nicolosi A, Annapureddy S. P2PCast: A Peer-to-Peer Multicast Scheme for Streaming Data. IRIS Student Workshop, MIT, 2003.1－13.

6 Zhang XY, Liu JC, Li B, Yum TSP. CoolStreaming/DONet: A Data-drivern Overlay Network for Peer-to-Peer Live Media Streaming. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. 2005,(3):2102－2111.

7 Goyal V. Multiple Description Coding: Compression Meets the Network. IEEE Signal Processing Magazine, 2001,(9):74－93.

8 Druschel RA. Pastry: scalable, distributed object location

(下转第 188 页)

(上接第 213 页)

and routing for large-scale peer-to-peer systems. IFIP/ACM International Conference on Distributed Systems Platforms(Middleware), Heidelberg, Germany, 2001,(9):329 − 350.

9 Castro M, Druschel P, Kermarrec A M, Rowstron A. SCRIBE: a large-scaleand decentralized application-level multicast infrastructure. IEEE JSAC, 2002,20(8): 100 − 110.

10 Hua KA, Cai Y, Sheu S. Patching: a multicast technique for true video-on-demand services. Proc. of the 6th ACM International Conference on Multimedia (MULTIMEDIA'98). 1998.191 − 200.

11 夏绍春.视频点播系统流调度算法研究[硕士学位论文].长沙:湖南大学, 2004.

12 向哲,钟玉琢,冼伟铨.一种基于周期合并策略的流调度算法.软件学报, 2001,12(8):1183 − 1189.

13 郑琳.视频点播系统中流媒体调度算法的研究[硕士学位论文].葫芦岛:辽宁工程技术大学, 2005.

14 沈磊.基于超级节点的高性能 P2P 流媒体技术的研究与实现[硕士学位论文].南京:南京航空航天大学, 2007.