第32卷 第20期 计 算 机 工 程 2006年10月
Vol.32 № 20 Computer Engineering October 2006
·多媒体技术及应用· 文章编号：1000—3428(2006)20—0218—03 文献标识码：A 中图分类号：TP37

可扩展的实时流媒体应用层组播系统设计

徐 敏[1]，李仁发[1]，乐光学[2]
(1. 湖南大学计算机与通信学院，长沙 410082；2. 怀化学院计算机科学与技术系，怀化 418000)

摘 要：分析应用层组播在流媒体分发领域的研究，设计了可扩展的实时流媒体应用层组播系统ALMLS。该系统基于非结构化Overlay，采用基于Gossip思想的节点加入算法和消息散播机制；设计了基于流媒体技术的数据缓存和获取策略；通过故障检测和恢复机制增强系统的健壮性。系统的特点是易于实现和具有良好的扩展性。

关键词：应用层组播；实时流媒体；非结构化；Gossip；消息散播

Scalable Application Layer Multicast System Design for Live Media Streaming

XU Min[1], LI Renfa[1], YUE Guangxue[2]
(1. School of Computer and Communication, Hunan University, Changsha 410082;
2. Department of Computer Science & Technology, Huaihua Institute, Huaihua 418000)

【Abstract】 This paper analyses the research of application layer multicast in media streaming distribution, and designs scalable application layer multicast system for live media streaming. The system based on unstructured Overlay adopts the algorithm for node's joining and message distribution mechanism which are both based on Gossip; It describes the strategy of data buffer and data acquisition. With the function of failure detection and recovery, the robustness of the system can be enhanced. The system can be implemented in ease and has good expansibility.

【Key words】 Application layer multicast; Live media streaming; Unstructured; Gossip; Message distribution

1 概述

传统的实时流媒体系统采用Client/Server模型，由于依赖视频源的计算能力和带宽，这种模式可扩展性差，服务器负载过大易出现系统崩溃的情况。IP组播是解决C/S模式瓶颈的有效方案，由于网络部署、管理和拥塞控制等原因没有被广泛地应用，从而提出了应用层组播的思想。应用层组播是在应用层上构建Overlay网络，客户端是Overlay中的节点，每个节点在接收数据的同时也向其它节点发送数据，实现IP组播路由器的分发功能。应用层组播无需特殊的路由器和硬件设施，也无需修改网络协议，易于流媒体应用的部署。许多研究者将应用层组播技术(ALM)应用到了媒体分发领域。

根据数据拓扑和控制拓扑构建的先后顺序，基于应用层组播的媒体分发系统主要分为3类：树优先，网状优先和隐式系统[1]。

树优先系统[2]把节点组织成一棵单数据源的组播树，当节点加入到组播树中，采用相应的算法搜索一些其它节点，与它们建立用于传输控制数据的连接。树优先Overlays通常包括一个组播节点(源节点)和许多接收节点(其它所有节点)。

在基于网状优先系统[3]中，节点之间形成网状拓扑关系，通过这种网状关系交换节点之间的控制消息。采用IP组播算法在网状Overlay上构建单源组播树。网状优先组播算法构建的overlays适合多源组播。

在隐式构建的系统中，网状拓扑和组播树同时被构建，Nice[4]就属于这种系统。Nice协议把Overlay的成员组织成一个分层的簇结构，当新节点加入和已有节点离开时，协议的基本操作是创建和维护分层的簇结构。分层的簇结构隐式地定义了流媒体数据的传输路径，因此分层结构的构建算法对于系统的扩展性是很重要的。

这几类应用层组播系统基于不同拓扑的Overlays，通过采用相应的节点加入和离开算法构建和维护Overlay。网状优先系统的控制负载是$O(N^2)$，适合于小规模的组播。树优先系统对树的深度往往没有限制，因此不适合延迟敏感的应用如实时流媒体。隐式系统具有较短的路径长度和较小的节点度数，适合延迟敏感的应用，也适合高带宽的应用。

隐式系统适于实时流媒体的分发，但是其实现是相当复杂的。张欣研等[5]设计了非结构化Overlay——DoNet，基于数据接受驱动的应用层组播，使系统的实现变得相对简单。

基于对这些系统的研究，设计了可扩展的实时流媒体应用层组播系统(ALMLS)。ALMLS采用DoNet的非结构化Overlay思想。根据非结构化Overlay的特点，新节点的加入算法和控制消息散播机制采用了Gossip思想，采用基于流媒体技术的数据缓存和获取策略。

2 基于Gossip的消息散播协议

基于Gossip的消息散播协议由于其良好的扩展性和可靠性，变得越来越流行。基于Gossip的散播协议的基本思想是：一个节点负责向一组随机选择的节点发送消息；收到消息的节点也做同样的事情，直到所有的节点收到消息为止。为了

基金项目：中国网上教育平台试点工程基金资助项目(计高技[2000]2034)

作者简介：徐 敏(1980－)，男，硕士生，主研方向：P2P计算和网络；李仁发，教授、博导；乐光学，教授

收稿日期：2005-11-07 **E-mail：**i_am_xumin@163.com

保证这个过程是收敛的，每个消息都会有 1 个TTL，没经过 1 次转发，TTL就减 1，当TTL减少到 0 时，消息就不再转发了。这种随机的方式将导致消息的冗余，但正是这种冗余保证了节点崩溃或网络丢包严重的情况下系统的可靠性。传统的基于Gossip的散播协议假设每个节点从其它所有节点中随机地、平等地选择一组节点发送消息，要求每个节点掌握其它所有节点的信息。因此，每个节点需要动态地维护全局信息，这不仅要消耗大量的内存，而且在高动态的网络下，是很难实现的。Ayalvadi J等人在传统的基于Gossip的散播协议上提出了可扩展的组成员协议SCAMP[6]，基于P2P Overlay，以完全分布式的方式运作。SCAMP把所有节点分成多个分组，分组的组员是互相交叉的。每个节点从它的组员中随机选择一些节点发送消息，收到消息的节点以同样的方式转发消息，直到TTL减少到 0。组的大小由节点自动调节，每个节点负载的增加与组大小的增加成对数关系。本文设计的ALMLS将采用SCAMP的思想散播控制消息。

3 设计方案

3.1 系统的体系结构

为了描述方便，视频源称为源节点，客户端称节点。如果节点 f 与 g 建立了关系，那么 f 和 g 是伙伴关系。每个节点维护着伙伴状态表，伙伴关系管理模块更新伙伴节点的状态和维护伙伴关系 ;缓存控制模块的作用是控制数据的缓存，包括更新缓存和提供缓存的情况。数据块调度模块是由缓存控制模块调度的，从伙伴中获取数据或向伙伴发送数据。网络接口层负责与其它节点建立连接和传输数据。图 1 是系统的体系结构，从 4 个方面介绍系统的设计思想：新节点的加入，伙伴关系的维护，数据的缓存和获取策略，故障的检测和恢复机制。

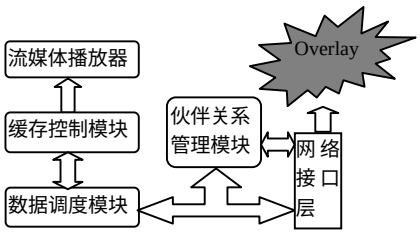


图 1 系统的体系结构

3.2 新节点的加入

当节点 N 加入到流媒体组播系统中，首先必须与一些节点建立伙伴关系，从伙伴中选择第 1 个获取数据的节点。只有满足以下 2 个条件的节点才能成为 N 的伙伴：

(1)非饱和节点，即当前伙伴个数小于最大伙伴个数的节点。节点的最大伙伴个数 MaxPtNum＝f(outbw)，outbw 表示该节点外向链路带宽。

(2)低延迟节点，即与 N 的传输延迟小于某个阈值 DEL 的节点。采用 RTT/2 表示节点之间的传输延迟，DEL 的值随着 Overlay 规模的增大做适当的调整。

下面描述新节点的加入算法：

(1)新节点加入后，首先与源节点建立联系，源节点给它分配一个唯一的标识。如果源节点满足条件，与源节点建立伙伴关系，则加入过程结束。否则，从源节点的伙伴中随机选出一些节点联系，满足条件的节点成为 N 的伙伴；

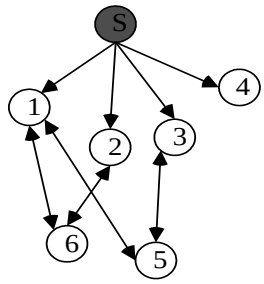
(2)从源节点的伙伴中随机选择一个节点作为代理节点，获取代理节点伙伴的列表，与这些节点联系，如果满足条件就与其建立伙伴关系。如果源节点的伙伴都被选过了，则选择伙伴的伙伴作为代理节点，依次类推；

(3)继续(2)的操作，当新节点成为饱和节点，停止循环，假设循环的最多次数为 C1；

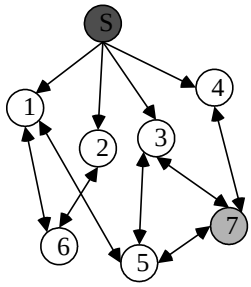
(4)如果 N 找到了至少一个伙伴，说明 N 已经成功加入。反之，回到(1)。假设此循环的最多次数为 C2；

(5)N 选择拥有最新数据的伙伴作为第 1 个获取流媒体的节点。

与源节点建立伙伴关系后，加入过程就结束了。因为源节点一直会存在并拥有最新的数据，所以就没有必要从其它节点获取数据了。图 2 表示加入新节点 7 前后拓扑结构的变化。关于C_1*C_2的取值和算法的复杂度将在对系统的分析一节中说明。



(a)加入新节点 7 前



(b)加入新节点 7 后

图 2 节点 7 加入前后 Overlay 的变化

3.3 伙伴关系的维护

伙伴状态表开始为空，在节点加入的过程中，表的成员不断增加，如表 1 所示。使用消息驱动机制来更新和维护伙伴关系。采用 SCAMP 协议向其它节点散播消息。下面几种情况将促使节点发送消息：(1)伙伴之间定时地交换状态信息；(2)增加新的伙伴，发送伙伴增加的消息；(3)发现某个节点不存在时，发送某节点离开的消息；(4)伙伴状态的成员过少时，发送增加新伙伴的消息；(5)节点正常离开时，发送离开的消息。

表 1 伙伴状态表

Identifier	节点的标识符
MaxPtNum	最多伙伴个数 MaxPtNum＝f(outbw)
CurrPtNum	当前伙伴个数
Delay	传输延迟 delay=RTT/2
Realtime	节点的实时性realtime＝$nData_{max}$，$nData_{max}$是节点缓存的最新数据块的标识

在表 2 中，MSG_ID 代表节点号，NODE_ID 代表消息号，MSG_TYPE 代表消息类型，MSG_CON 代表消息的内容。采用 SCAMP 协议分发消息，会导致节点重复收到消息，通过 NODE_ID 和 MSG_ID 的组合来判断消息是否重复接收，

如果是的，就丢弃。MSG_TYPE e 不同，MSG_CON 的格式不同。根据 MSG_TYPE 来解析 MSG_CON 的内容。

表 2 消息的格式

MSG_ID	NODE_ID	MSG_TYPE	MSG_CON

消息驱动机制：节点 n 收到节点 m 的消息后，更新伙伴状态表。如果 n 收到 m 离开的消息，则从表中删除 m 的信息；如果是增加一个伙伴的消息，n 检查与 m 的传输延迟。一般情况下，如果 m 的伙伴增多，会导致现有带宽的减少，n 与 m 之间的传输延迟可能增加。如果延迟过大，则取消它们的伙伴关系。节点 n 将从伙伴节点的伙伴中选择一个满足条件的作为新的伙伴。

3.4 数据的缓存和获取策略

节点是数据的接收者也是发送者。采用缓存策略一方面是减少多媒体的抖动，另一方面是实现节点作为发送者的功能。流媒体数据被分成长度一致的数据块，每个数据块有唯一的标识。每个节点都会缓存一些数据块。设置一个数据块视图表，其作用是标识节点缓存了哪些数据块。某个节点想从伙伴中获取一些数据块，首先查询该伙伴的数据块视图表，如果数据块存在，则获取。系统提供的是实时流媒体，所以只需缓存适当多的数据。假设每个节点会缓存它正在播放的数据块前的 1min 数据，如果一个数据块代表 1s 的流媒体，那么就是缓存 60 个数据块。在播放的同时，节点会不断检查伙伴的数据块视图表，如果有新的数据块，就获取。每个节点的缓存大小并不一致，一般而言，实时性好的节点缓存的数据相对较少。如果节点想获取的数据只有一个伙伴拥有，就别无选择了。如果多个伙伴拥有，就选择传输延迟最小的节点。

3.5 故障的检测和恢复机制

节点的离开分为 2 种情况：

(1)正常的离开：节点离开前会向其伙伴发送一个离开的消息；

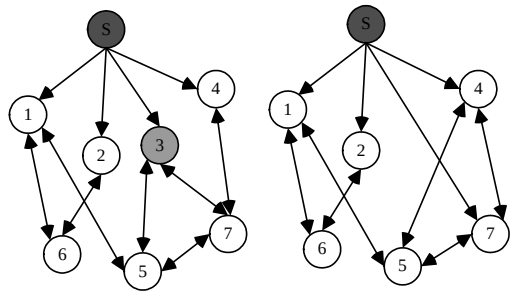
(2)非正常的离开：节点出现故障，突然掉线。在情况(2)下，它不会发送消息。

故障的检测包括 2 种方式：

(1)定时地检测伙伴的状况，包括是否存在和传输延迟；

(2)定时检测伙伴的个数。

故障的恢复：如果发现某个伙伴已经不存在或者传输延迟过大，取消伙伴关系，从自己的伙伴节点的伙伴中随机选择一个满足条件的节点作为新伙伴。如果发现伙伴个数过少，采用前面的方法选出一些新伙伴。



(a)节点3离开前　　(b)节点3离开后

图 3 节点 3 离开前后 Overlay 的变化

当节点发现某个伙伴不存在时，它将采用 SCAMP 协议向其他节点分发某个节点离开的消息。系统能较快地检测出节点故障，并采用有效的恢复机制。图 3 中说明了节点 3 离开前后 Overlay 拓扑结构的变化。

4 系统分析

伸展度(Stretch)是衡量组播系统性能的最重要参数之一。Stretch＝组播Overlay中从源节点到节点的路径长度/单播方式中源节点到节点的路径长度[1]。所有节点伸展度的平均值表示组播系统的平均路径长度，用来衡量系统的平均延迟。文献[5]证明非结构化Overlay伸展度的复杂度是O(logN)，表明ALMLS的平均延迟与Overlay的规模成对数关系，一方面保证了流媒体的实时性，另一方面体现了系统在传输延迟上的扩展性。

新节点加入算法的复杂度为 O(C1*C2)。C1 根据新加入节点的 MaxPtNum 确定，C2＝logmn(n 是所有节点的个数，m 是一个常数)，C2 是依据节点的平均伸展度确定的。这表明加入算法是可扩展的。新节点加入过程中选择的节点满足低延迟和带宽可用的条件，说明该算法是有效的。

ALMLS采用Gossip方式[6]散播消息，每个节点的消息负载与伙伴的个数成对数关系。整个系统的消息负载是O(logN)，N是所有节点的个数。因此ALMLS的控制消息负载具有扩展性。

5 小结

结合应用层组播的思想和流媒体技术，描述了 ALMLS 的设计方案。ALMLS 基于非结构化 Overlay，采用基于 Gossip 思想的节点加入算法和消息分发机制。利用故障检测和恢复机制增强系统的健壮性。系统的特点是易于实现和良好的扩展性。

目前，该系统正处于研发阶段，包括基于分布式环境 PlanetLab 的仿真实验以及原型系统的开发，我们希望将 P2P 应用层组播思想与流媒体技术结合，为中国网上教育平台的远程实时多媒体教学提供一种有效的、低成本的部署方案。

参考文献

1 Banerjee S, Bhattacharjee B. Comparative Study of Application Layer Multicast Protocols[EB/OL]. http://www.cs.wisc.edu/~suman/pubs/compare.ps.gz.

2 Zhang B, Jamin S, Zhang L. Host Multicast: A Framework for Delivering Multicast to End Users[C]. Proceedings of IEEE INFORM'02, New York, 2002-06: 1366-1375.

3 Yang Hua, Rao S G, Zhang H. A Case for End System Multicast[C]. Proceedings of ACM SIGMETRICS, 2000: 1-12.

4 Banerjee S, Bhattacharjee B, Kommareddy C. Scalable Application Layer Multicast[C]. Proceedings of ACM SIGCOMM, Pittsburgh, PA, 2002-08: 205-220.

5 Zhang X, Liu J, Li B, et al. DONet/CoolStreaming: A Data-driven Overlay Network for Live Media Streaming[C]. Proceedings of IEEE INFOCOM, Miami, FL, USA, 2005-03: 2102-2111.

6 Ganesh A J, Kermarrec A M, Massoulie L. Peer-to-Peer Membership Management for Gossip-based Protocols[J]. IEEE Transactions on Computers, 2003, 52(2).