

无线 IPTV 中 RTP-over-TCP 传输方案的设计 实现以及多 TCP 传输技术的应用

柴颖 陈荣华 叶德建

(复旦大学软件学院宽带网络与互动多媒体实验室, 上海 201203)

摘要 在无线环境下, TCP 的传输效果要好于不带拥塞控制的 UDP。为了满足“集中-分布式”无线流媒体系统进行 RTP-over-TCP 传输的需要, 设计实现了一套双 TCP 通道的传输方案, 并考察了 TCP 阻塞调用和非阻塞调用在单线程多用户调度方式下的性能。另外, 还尝试用多 TCP 传输技术来改进 TCP 在无线环境中带宽利用率低的问题, 仿真实验结果表明, 该方案可得到较好的传输效果。

关键词 无线网络电视 RTP-over-TCP TCP 阻塞/非阻塞模式 多 TCP 传输技术

中图分类号: TN919.85 **文献标识码:** A **文章编号:** 1006-8961(2007)10-1725-05

The Design and Implementation of RTP-over-TCP Transmission Mode and the Use of Multi-TCP Transmission Technology for Wireless IPTV Service

CHAI Ying, CHEN Rong-hua, YE De-jian

(Multimedia and Networking Laboratory, Software School, Fudan University, Shanghai 201203)

Abstract Under the wireless network, the performance of TCP is better than that of UDP which lack of congestion control. In order to conform the RTP-over-TCP transmission mode to our hybrid centralized-distributed server architecture, we propose a method with double-TCP channel. Besides, we compare the transmit performance of TCP blocking mode to non-blocking mode. Moreover, we try to improve the bandwidth usage of TCP under the wireless network by using multi-TCP transmission technology. The simulation results demonstrate that our method is applicable.

Keywords wireless IPTV, RTP-over-TCP, TCP blocking/non-blocking mode, multi-TCP transmission

1 引言

流媒体应用, 如网络电视 (IPTV) 对互联网络数据传输的实时性要求很高, 目前业界的标准是采用 RTSP-over-TCP 和 RTP-over-UDP 协议, 前者负责交互通信, 后者负责数据传输。在传输层, RTP (real-time transport protocol) 由于携带了大量的多媒体数据, 传输量大, 实时性要求高, 较合适采用延迟和抖动较小的 UDP (user datagram protocol) 传输协议。但是在无线应用中, 情况发生了以下变化: 首先, 无

线 IPTV 系统服务器和客户端之间的网络通常是异构的, 其中有线网络和无线网络之间的带宽不匹配情况较严重, 其各自的带宽也是时刻在变动的, 没有拥塞控制的 UDP 传输方式在这种情况下很容易出现大量的拥塞丢包; 其次, 无线信道高误码率的特点也令没有丢包补偿措施的 UDP 传输质量难以达到流媒体的要求。虽然可以在应用层配合 UDP 协议实现拥塞控制和丢包补偿措施^[1], 但是这一方面大大增加了开发难度, 导致了大量的重复工作; 另一方面不同的应用层实现也使得系统之间难以兼容。而 TCP (transfer control protocol) 协议其本身拥有的拥

基金项目: 国家自然科学基金项目 (60503044); 上海市科技启明星人才计划项目 (06QA14008)

收稿日期: 2007-06-01; 改回日期: 2007-07-31

第一作者简介: 柴颖 (1982 -) 男, 复旦大学硕士生, 主要研究方向为宽带网络与互动多媒体。E-mail: 042053008@fudan.edu.cn

塞控制机制和重传机制则正好可以克服这些缺点，而且该协议是目前互联网中应用得最广泛的标准协议。另外，由于 TCP 能够较好地穿越防火墙，因而可以很好地适应无线异构网络的各种架构形式。为此，本文将研究重点放在 RTP-over-TCP 传输方式的工程应用上。

笔者在将 RTP-over-TCP 的传输方式应用在自主研发的“集中-分布式”架构的无线 IPTV 系统中时，发现协议标准^[2,3]中只提到在某些环境下可能采用 TCP 协议传输 RTP 数据，并且很可能与其他数据，如 RTCP (RTP control protocol) 数据或 RTSP (real-time streaming protocol) 数据混合在一起，实际实现中，目前主流的流媒体系统也都以单 TCP 的方式来混合传输 (interleave) RTSP 和 RTP 数据。这种方式将 RTSP 和 RTP 模块紧密地绑定在了一起，无法应用在笔者开发的系统中 (该系统的服务器在架构上采用了半集中半分布式的形式，即以一个主控服务器负责与用户的交互工作，若干个播放服务器负责数据的传输，由于这些服务器各自需要工作在不同的机器上，因而无法共享 TCP 链接)。针对这种情况，本文提出了一个双 TCP 通道的方案，以适应分布式的服务器架构。

另外，本文所针对的服务器采用了多用户单线程轮询的服务调度方式，在这种情况下，应用程序可以选择 TCP 阻塞调用或非阻塞调用这两种方式来传输数据，为此有必要考察一下两者的性能优劣。

当然，对于无线 IPTV 应用来说，TCP 也有自身的缺点，例如，TCP 协议原本是为有线环境设计的，由于无法区分拥塞丢包和差错丢包，因而在无线环境中其网络利用率低于 UDP^[4]。已有一些研究给出了相应的解决方案^[5,6]，本文则从工程应用的角度，尝试采用多 TCP 传输技术来改善其网络利用率。

2 双 TCP 通道传输方案

该方案在笔者开发的流媒体系统中的主控服务器和客户端、播放服务器和客户端之间各建立一个 TCP 通道，前者用来传输 RTSP 消息，后者用来传输 RTP 数据。该方案的核心是一个 TCP 通道的认证协议，这个协议在传统的 RTSP 交互协议基础上进行了扩展，加入了 RTP-over-TCP 的连接过程和必要的认证措施。协议的参与者有：客户端 (C)、主控服

务器 (M)、播放服务器 (P) 三方。三方之间的交互、连接和认证过程被分成以下 4 步：

- (1) C 与 M 建立 RTSP-on-TCP 通道；
- (2) C 和 M 之间进行 RTSP 交互；
- (3) C 与 P 建立 RTP-on-TCP 通道；
- (4) C 和 P 之间进行合法性认证。

图 1 给出了具体的协议过程，其中 RTSP 的 4 步交互分别为“协议协商 (describe)”、“音频通道协商 (setup audio)”、“视频通道协商 (setup video)”和“播放 (play)”，会话标识码 (session ID) 会在“协议协商”的应答中传给客户端，播放服务器会根据客户端的地址和标识号进行验证。为了认证协议的安全性和健壮性，本文还制定了 3 个超时机制，分别用来应对客户端或网络的崩溃、无效的 RTP/TCP 连接，以及验证信息在服务器内部丢失的情况。实践证明这些机制可以有效地抵御第 3 方的恶意攻击，以及各种认证失败的情况。

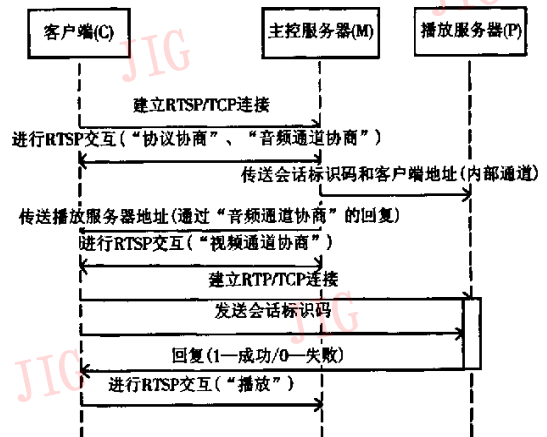


图 1 双 TCP 认证协议交互时序图

Fig. 1 The process of Double-TCP channel authorization protocol

3 TCP 阻塞模式和非阻塞模式的比较

由于播放服务器采用了单线程调度的方式，为了不让单个用户阻塞线程，具体实现中可以选择将所有的 TCP 发送调用改为非阻塞方式，也可以用 select 模型来配合 TCP 阻塞调用。本节将通过实验来分析 TCP 阻塞调用和非阻塞调用之间的性能差异。

如图 2 所示，两台 PC 机被一个千兆网络连接

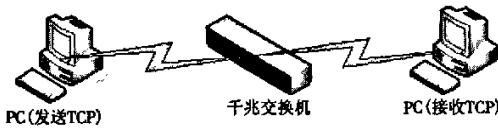


图 2 实验示意图

Fig. 2 The experiment topology

起来, TCP 发送和接收程序分别跑在这两台机器上, 并保证没有其他的应用程序在运行。发送程序直接在内存中建立伪数据包, 并不停地发送, 尽量占用带宽, 而接收程序则尽可能快地将数据接收下来。表 1 给出了具体的软硬件配置情况。

表 1 实验配置情况

Tab. 1 The configuration of experiment

配置项	配置	说明
交换机 型号	Cisco 千兆交换机	
发送端 平台	Red Hat Advance Server	
网卡	INTER @ PRO/1000 GT	
系统缓存	262142 Bytes	操作系统为每个 TCP 连接提供的缓存大小
发送方法	fp 式尽力发送	
包大小	800 Bytes	程序每次调用系统函数所传人的数据量
调度方式	在两个模式下都采用单线程 select 轮询。在非阻塞模式下, 如果每条 TCP 连接的系统缓存都满了, 则程序在进行下次轮询前睡眠 10ms	
接收端 平台	Windows XP SP2	
网卡	INTER @ PRO/1000 GT	
接收缓存	50 000 Bytes	程序每次调用系统函数所传人的内存块大小
调度方式	阻塞模式, select 模型	接收端总是以阻塞模式来工作

表 2 是通过网络监控程序所获得的数据, 从表 2 可以看出, 虽然在单用户情况下, 阻塞传输的效率比非阻塞传输的效率, 但在给多用户提供服务时, 非阻塞模式的传输能力要远好于阻塞模式。

究其原因, 一方面, 在阻塞模式下, 由于虽然发送端采用了 select 机制预先判断了是否有系统缓冲

表 2 TCP 阻塞模式和非阻塞模式的传输性能比较

Tab. 2 The transmit performance comparison between TCP Blocking mode and TCP Non-blocking mode

测试项	网络占用率 (%)
阻塞模式, 1 TCP	70
阻塞模式, 2 TCP	22
阻塞模式, 3 TCP	10
阻塞模式, 10 TCP	4
非阻塞模式, 1 TCP	20
非阻塞模式, 2 TCP	40
非阻塞模式, 3 TCP	50
非阻塞模式, 10 TCP	60
非阻塞模式, 40 TCP	60

区剩余空间, 但是对于空间的大小仍无法得知, 仍然会造成发送阻塞, 反而使其他 TCP 通道空等, 并且并发数越多, 这种情况越严重; 另一方面, 由于采用非阻塞模式, 使发送端不会在某个 TCP 连接的系统缓冲区耗尽的情况下白白浪费时间, 虽然可能在 TCP 通道已经就绪的情况下未能及时将应用层数据送入, 从而造成带宽浪费, 但是由于带宽是各个 TCP 通道共享的, 因而并发数越多, 带宽的利用率越高。

通过实验比较和分析可以看出, 在大规模多用户服务中, 服务器如果采用单线程调度方式, 则 TCP 非阻塞模式要优于阻塞模式。

4 多 TCP 传输技术

TCP 传输协议在无线网络环境下的一个主要问题就是无法区分网络拥塞丢包和差错丢包。由于无线网络高误码率的特性会导致许多差错丢包, 而 TCP 又将这些丢包一律看成是因网络拥塞造成的, 从而错误地降低了发送速率, 由于其 AIMD (additive increase and multiplicative decrease) 的策略, 需要用“原窗口大小 \times RTT (roundtrip time) / 2”的时间来恢复到原速率, 从而使得无线网络的带宽无法充分被利用。这对于本身就带宽不足的无线网络流媒体服务来说, 无疑是雪上加霜。许多研究针对这一问题提出了一些解决方案^[5,6], 但都是着眼于如何帮助 TCP 区分丢包的原因, 由于要更改 TCP 协议实现或网络节点的参与, 因此在实际开发中难以应用。

为了充分利用这部分损失的带宽, 本文提出了

一种更为简单的方案,即采用多 TCP 通道并行传输的技术,将数据通过几个 TCP 通道中的任意一个(当前处于可发送状态)发送给客户端,当一个 TCP 通道发生丢包重传时,另外的 TCP 通道会提高速率填补剩余的带宽。

为了验证理论分析的正确性,笔者对其做了实验仿真(如图 3 所示)。

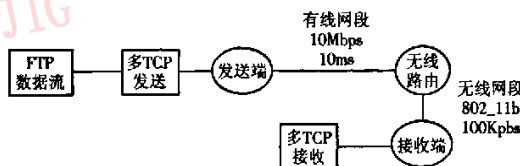


图 3 NS2 多 TCP 传输仿真拓扑图

Fig. 3 "NS2" simulation topology

图 4~图 9 给出了实验仿真无线网络段的统计结果(图中图例代表无线网络误码率)。从图 4 可以看出,随着 TCP 数量的增加,用户带宽的利用率有明显的提高,并且误码率越高,多 TCP 的优势越明显。从图中还可以看出,2 到 3 个 TCP 性价比最高,更多的 TCP 连接对带宽利用率的提升效果并不明显。但是在图 5 中,这种提高表现得不太稳定,而且当数量达到一定程度时,带宽的利用率反而有所下降,初步分析其原因可能是由于低带宽环境下,因多 TCP 的相互竞争而导致 TCP 的传输速率变得不稳定所致。

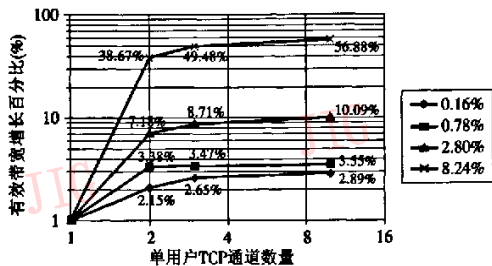


图 4 多 TCP 有效带宽增幅(1Mbps)

(该图横纵坐标经过对数处理,横坐标以 2 为底,纵坐标以 10 为底)

Fig. 4 The increment of bandwidth usage (1Mbps)

图 6 和图 7 给出了多 TCP 在 IP 层的丢包情况,从图中可以看出,带宽对多 TCP 的丢包率没什么影响,而且只有在低误码率的环境下,丢包率才随 TCP 数量的增加而上升,在高误码率的环境下,TCP 数量的增加并没有导致明显的丢包率上升。

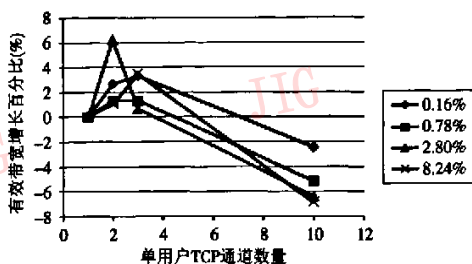


图 5 多 TCP 有效带宽增幅(100Kbps)

Fig. 5 The increment of bandwidth usage (100Kbps)

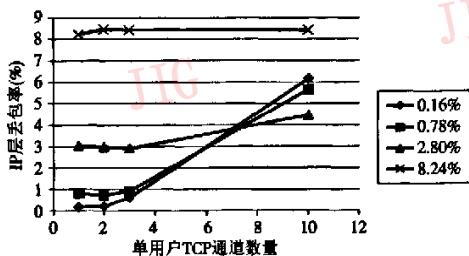


图 6 多 TCP 的 IP 层丢包率(1Mbps)

Fig. 6 The IP-packet loss rate (1Mbps)

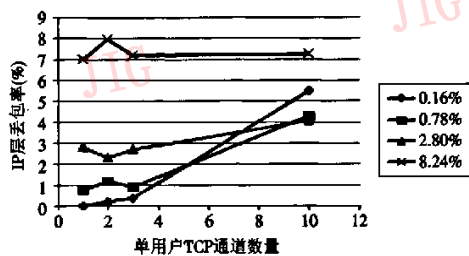


图 7 多 TCP 的 IP 层丢包率(100Kbps)

Fig. 7 The IP-packet loss rate (100Kbps)

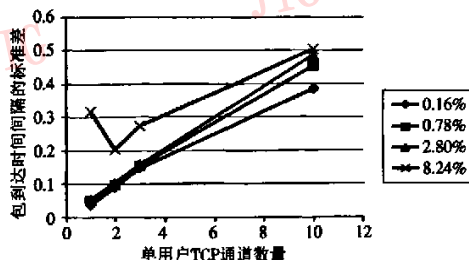


图 8 多 TCP 数据包到达时间的抖动(1Mbps)

Fig. 8 The jitter of packet arrival time (1Mbps)

多 TCP 的相互竞争还会引起数据流抖动的加剧,这在图 8 和图 9 中得到了验证。值得注意的是,在高误码率且带宽较为充足的情况下,双 TCP 并行传输比单 TCP 传输的抖动反而降了 1/3。

综上所述,多 TCP 通道并行传输模式能有效地减少带宽损失,尤其在高误码率、带宽较为充足的无线环境中会有更好的表现。

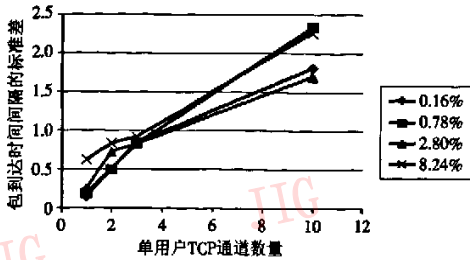


图 9 多 TCP 数据包到达时间的抖动(100Kbps)

Fig. 9 The jitter of packet arrival time under (100Kbps)

5 结 论

本文为在分布式流媒体服务器上的 RTP-over-TCP 数据传输设计了一套双 TCP 解决方案,同时通过实验分析了 TCP 阻塞模式和非阻塞模式在单线程多用户程序中的传输性能,并通过多 TCP 传输技术改善了 TCP 在无线网络的传输效率。

后续工作将对 TCP 在低带宽下的行为进行进一步的分析和改进;另外,还将研究 TCP 非阻塞模式在应用层负载为带速率控制的各种媒体流(恒定

速率码流、可变速率码流、经速率平滑后的可变速率码流)情况下的性能表现,并据此改进相关的调度算法。

参考文献 (References)

- 1 Kohler E, Handley M, Floyd S. Datagram Congestion Control Protocol(DCCP)[S]. IETF, RFC4340.
- 2 Schulzrinne H, Columbia U, Rao A, *et al.* Real Time Streaming Protocol[S]. IETF, RFC2326.
- 3 Schulzrinne H, Casner S. RTP: A Transport Protocol for Real-time Applications[S]. IETF, RFC3550.
- 4 Zhang Jing-yuan, Xie Jian-ying, Wang Ming-zhong. Performance analysis of TCP congestion control in wireless network [J]. Communications Technology, 2002, (3): 21 ~ 23. [张敬敏, 谢剑英, 王明中. 无线网络中 TCP 拥塞控制算法的性能分析[J]. 通信技术, 2002, (3): 21 ~ 23]
- 5 Bakre A, Badrinath B. R. I-TCP: Indirect TCP for mobile hosts [A]. In: Proceedings of the IEEE Computer Society 15th International Conference on Distributed Computing Systems [C], Vancouver, Canada, 1995.
- 6 Ding Wen-qing, Jamalipour A. A new explicit loss notification with acknowledgment for wireless TCP[A]. In: proceedings of 12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications(PIMRC'01) [C], San Diego, CA, USA, 2001, 1: B65-B69.