

基于分段连接机制实现卫星 TCP 性能的改善

Improvement TCP Performance via Satellite Link based on Splitting Connection Mechanism

吴 结 高随祥 (中国科学院研究生院 100049)

摘要: TCP 经由卫星链路传输时性能较低,这是卫星通信系统设计和应用中的一个需要解决的重要问题。分段连接机制是改善卫星 TCP 性能的一种有效方式。在对分段连接的研究中,它的整体系统性能和语义维持问题被大量关注,但在如何实现却少有论述。为了填补这一空白,提出了分段连接的一种实现方法,并进行了测试,测试结果表明此种实现遵守了分段连接的语义,并且可以明显改善卫星链路 TCP 的性能。

关键词: 卫星通信 TCP 性能 协议 网关

1 引言

TCP 协议是面向连接的传输层协议,向有关应用提供可靠的数据传输。随着通信事业的发展,大量卫星通信系统承载了大量的网络应用,其中基于 TCP 的应用大约占了应用总量的 85% 以上。由于 TCP 协议主要是以有线通信系统为应用环境设计的,没有考虑到卫星通信系统的特有属性,造成经由卫星链路传输 TCP 性能低下。如何提高卫星系统 TCP 的性能引起了人们广泛的重视。目前提高卫星系统中 TCP 性能的方法主要有端对端方式和分割连接方式两种。端对端方式着重于 TCP 协议本身的改进,而分段连接机制则考虑对不同物理链路分割为不同的段,在每个段内采用不同的优化方式进行优化。

2 卫星链路对 TCP 性能的影响

卫星通信系统对 TCP 性能影响,主要有以下 3 个方面原因:

2.1 传播延迟

影响卫星网络延迟的主要因素是传播延时。以最为常用的同步卫星系统为例,其单向传播延迟为 250 - 280ms,双向传播延迟为 500 - 560ms [2]。由于 TCP 采用滑动窗口机制, TCP 的最大吞吐率决定于最大窗口尺寸和往返时间(Round Trip Time, RTT)时间的比值。在双向卫星系统中,标准 TCP(窗口的最大尺寸为

65535) 的最大吞吐量为 $65535 * 8 / 0.55 = 953$ (Kbps);其次,在 TCP 慢启动/拥塞控制阶段,均是以 RTT 为单位进行拥塞窗口调整,较大的 RTT 直接导致了在窗口的调整速度较低,限制了吞吐能力^[1]。

2.2 信道差错率

卫星信道的比特差错率(BER)大约为 $10^{-2} - 10^{-7}$ 之间,这远远高于高速有线媒质(如光纤),另外一些随机因素(如雨衰,太阳黑子,微波干扰等)使得信道出现突发错误。由于 TCP 无法区分报文的丢失是由网络拥塞造成的还是链路误码造成的^[5],在误码出现造成报文丢失时, TCP 会误认为发生拥塞,进行拥塞避免,从而降低报文发送速率。另外,ACK 分组的丢失使吞吐量进一步恶化^[1]。

2.3 信道不对称

考虑到大量 TCP 传输本身的较大非对称性(如从 Web 服务器到远端主机),许多卫星系统在前向和反向信道间有较大的带宽不对称性,采用速度较慢的反向信道以节省卫星带宽。但严重非对称的配置会对 TCP 产生显著的影响。因为 TCP 的流量控制的基础是自同步机制^[5],ACK 的接收速率决定了数据报文的发送速率。反向信道的拥塞会造成对 TCP 在前向信道报文发送的抑制,对吞吐率产生负面影响。

3 提高卫星 TCP 性能的常用方法

卫星通信系统 TCP 的性能问题引起了人们的重

视,目前有三种方式^[4]来提高经由卫星链路 TCP 的性能:

端到端方式:是指发送端和接收端的协议栈采用协议改进和增加协议选项的方式,通过对 TCP 协议本身的修改进行 TCP 性能的提升。

所谓的卫星协议网关(以下简称协议网关)是通过打断发送端与接收端的 TCP 连接,建立分段连接,在卫星链路中采用扩展的 TCP 或其他专门针对卫星链路设计的传输协议,以提高 TCP 在卫星链路中传输效率的网络设备。同时,协议网关与主机的仍以标准 TCP

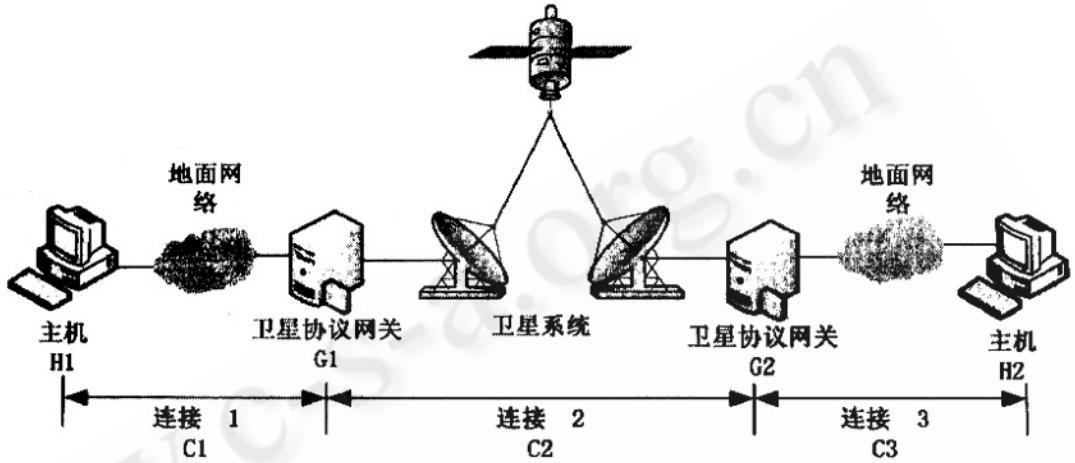


图 1 分段 TCP 连接示意图

分段连接机制^[2,4]:在本方式中,充分考虑到卫星链路与地面链路的不同,将发送端和接收端的 TCP 连接分为三段,在不同的段中可以采用不同传输协议。这三个段的连接构成了一个对应用层透明的端对端连接。通过分段可以针对不同传输特性进行协议的优化。此种方式是我们讨论的主要内容。类似的,对于移动通信系统中,采用 I-TCP 的分段方式解决移动通信系统中 TCP 的效率问题。

链路层方式^[1]:是指对链路层的改进,主要针对卫星设备的性能提升,目的是通过采用新的调制方式和较高前向纠错编码方式 FEC (forward error correction),以获取链路中更好的误码率,减少因为链路传输误码造成的报文丢失。

3.1 分段连接简述

分段连接改善卫星 TCP 性能的设计思路是这样的,如果在端到端的链路中由一些传输特性有很大不同的物理链路构成,那么将这些不同物理链路分割为了不同段,在每个段内采用不同的优化方式进行优化,从而使得 TCP 的端到端总体性能得到较大的提高。分段连接机制是通过卫星协议网关实现的。

建立连接,保证了对端到端应用的完全透明。如上图示,我们可以看到协议网关在位置在卫星系统与地面系统连接处,来连接两类完全不同的物理链路。

在图 2 中,为协议网关的协议栈示意图,图中我们

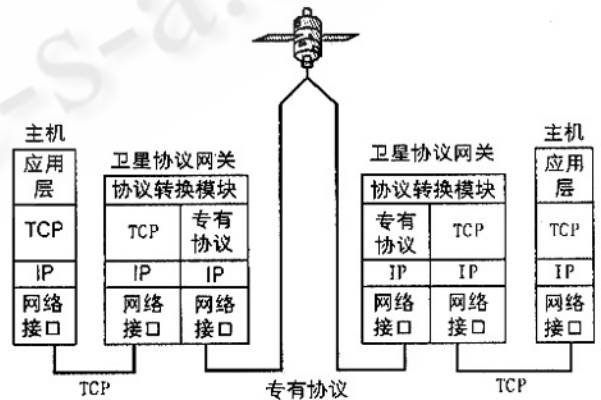


图 2 卫星协议网关协议栈示意图

可以看到专有协议与标准 TCP 协议在栈中的对等位置,通过这样设计首先可以将卫星链路的缺陷隔离在有缆链路之外,实现对 TCP 性能的改善;其次,对应用

完全透明,主机上的 TCP/IP 协议栈和应用不需要针对协议网关进行任何额外的开发工作。

在协议网关 (G_1, G_2) 之间采用的专用协议可以是 TCP 协议的扩展版本,也可以是专门针对卫星系统传输特点设计的传输层协议,由于此类协议考虑到卫星系统的特点,因而可以执行正确的拥塞控制,获取较高的传输率,从而更有效解决卫星链路的效率问题。

3.2 分段连接优缺点分析

分段连接有如下的优点:

- 将端对端 TCP 连接分割成三部分,从而卫星链路中的问题不会与地面链路产生关联,因而,分段连接方法支持采用局部解决方案来解决卫星链路中局部的问题,如:高延时,信道误码^[4]。

- TCP 连接分成三个独立的部分之后,在 C_2 连接上可以将许多最近提出的、有关提高卫星连接中 TCP 性能的 RFC 建议尽快使用;或者采用专门为卫星链路设计的传输协议^[4];

- 分段连接对应用完全透明,所有主机的 TCP/IP 协议栈和应用不需要为分段连接进行任何修改和特殊设计。

- 部署容易。协议网关放置在卫星链路和地面链路之间,对原有的体系结构没有大的修改,实施非常方便^[5]。

- 性能提升效果明显。通过采用分段连接,端对端 TCP 的性能得到明显的提升,在 C_2 连接上采用不同的传输协议,会有不同的性能提高效果。

分段连接的缺点^[4]如下所述:

- 它打破了 TCP 端对端语法,
- 分段连接无法兼容 IPSec。

4 协议网关的设计与实现方法

对于分段连接的理论研究比较多,但对分段连接实现的论述却比较少,在这里我们提出基于协议网关的分段连接的实现方法。

4.1 协议网关的特性

- 实现分隔 TCP 的最基本机制:将主机 H_1 与 H_2 之间的 TCP 连接分割为三个连接,并在数据传输过程中保持三个连接有效;

- 协议网关对发送端和接收端的主机完全透明,主机 H_1 和 H_2 上只能看到正常的到对端的 TCP 连接;

- 维持端对端 TCP 提供的数据可靠性。在网关间采用可靠的,面向连接的传输协议;虽然有的分段连接实现中在网关间采用非面向连接的数据传输,但这样无法保证端对端的数据可靠性,所以采用面向连接的协议是非常重要的;

- 对于连接建立和中止的报文 (SYN 和 FIN 报文),协议网关均采用中继转发的方式,在没有收到对端的 ACK 之前不发出伪应答。

- 采用如第二部分所述的针对卫星链路性能增强的 TCP 版本来实现 C_2 连接,这些扩展包括:窗口比例因子选项 (RFC1323),选择确认选项 (SACK) (RFC2018),加大的初始窗口尺寸 (RFC2414)。

4.2 协议网关的实现

下面讨论协议网关的基本工作方式:发送端与接收端的连接是如何建立的,数据报文是如何进行传送的。整个过程可分为连接建立阶段,数据传送阶段,连接终止阶段。为了描述方便,我们依据图一进行讨论,当主机 H_1 与 H_2 之间连接建立 TCP 连接时,被协议网关 G_1 和 G_2 分割为三个连接,分别定义为 C_1, C_2, C_3 。在 C_1, C_3 段采用标准的 TCP 协议;在 C_2 段中,可以采用 TCP 协议的扩展版本,或者采用专门针对卫星系统传输特点设计的传输层协议。我们假设由 H_1 发起到 H_2 的连接,向 H_2 传送数据,最后由 H_1 终止连接。具体过程参考图 3。

4.3 协议网关的实现细节

(1) 报文的截取。在分段连接的示意图中,我们可以看到,协议网关需要截获标准的 TCP 的报文,对报文进行缓存,处理后再发往网络中。我们采用 linux 实现协议网关,使用 linux 的 netfilter 机制实现对报文的截取。

(2) 网络地址转换。对于每个协议网关收到主机的报文都需要进行地址转换,在收到由对端协议网关发出的报文时也需要进行反向的地址换,以便能够在 C_1 和 C_3 段都保持标准的 TCP 连接。

(3) 缓冲区管理。采用分段连接模式后,在协议网关中采用双向链表进行报文缓冲区管理,保证数据在没有确认之前保留有可用的副本,进行报文的重发。

(4) 连接管理。采用多线程的模型进行连接的管理,当主线程接收到新的连接请求后,为该连接创建一个子线程,自线程负责完成端对端的连接和数据传输。

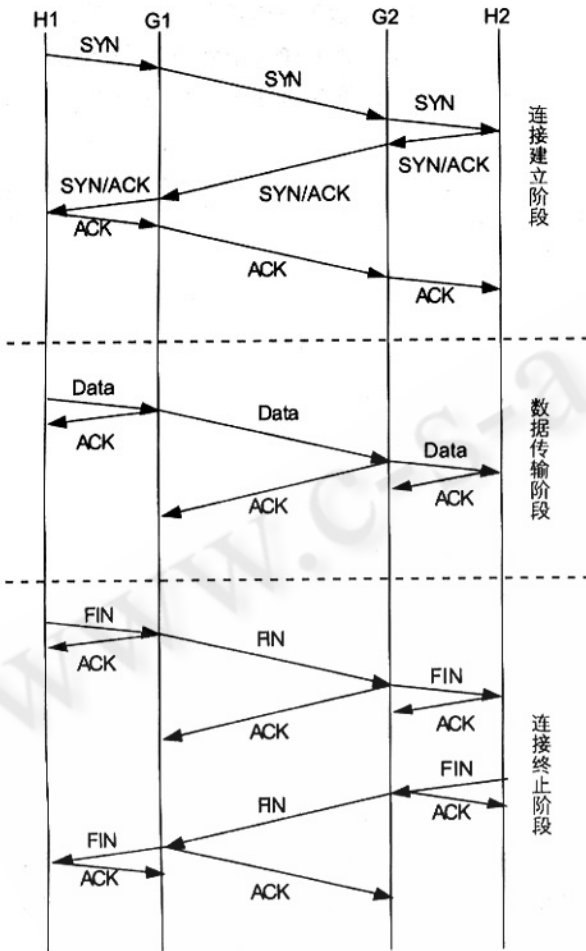


图 3 卫星协议网关工作示意

5 协议网关的性能测试与分析

我们采用如图四示的连接图,进行协议网关的测试。主机 H₁ 和 H₂ 中采用的是 Windows 2000 Professional (sp3) 和 Windows Server 2000 (sp3), 运行普通的 TCP/IP 协议栈(TCP Reno)。链路模拟器可以设置不同的速率,时延和误码率。路由器为 Cisco 公司的 2611,其端口最大的吞吐率为 8Mbps,与链路模拟器的连接采用 v.35 接口。协议网关 G₁ 和 G₂ 的实现如前所述。

为了考察单 TCP 连接的吞吐量,我们采用 ftp 应用

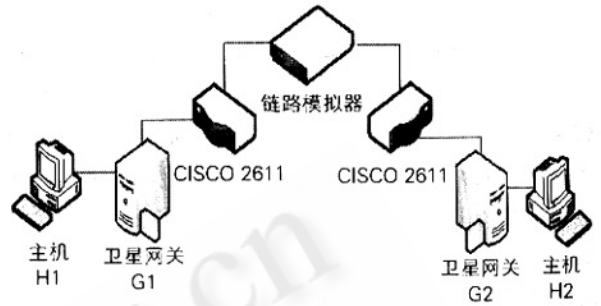


图 4 测试系统示意图

进行测试。H₁ 中安装 CuteFtp Pro, H₂ 中采用的 Ftp Server 为 Serv-U Ftp Server。采用不同速率和时延以及误码率进行测试,测试曲线如下:

(1) 在 RTT = 250ms 时,采用标准 TCP 和分段连接按照不同的带宽进行 FTP 下载测试,(下载文件大小: 10Mbytes),在图 5 中,我们可以看到采用分段连接的带宽利用率可以达到 75% - 85%,而采用标准 TCP 的带宽利用率,由于窗口尺寸的限制,随着带宽的增加,越来越低。

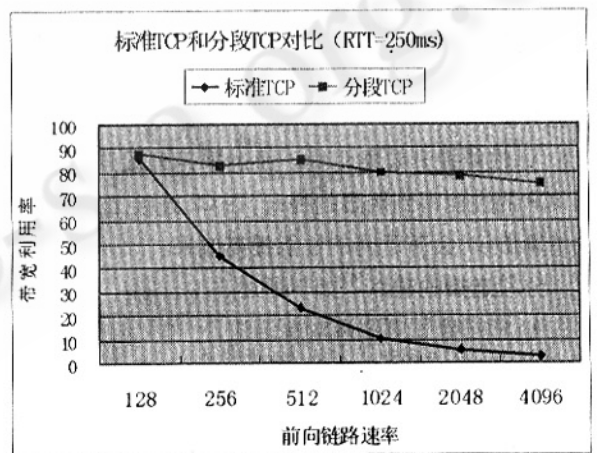


图 5 标准 TCP 和分段连接对比(RTT = 250ms)

(2) 在 RTT = 560ms 时,采用标准 TCP 和分段连接按照不同的带宽进行 FTP 下载测试,(下载文件大小: 10Mbytes),在图 6 中,我们可以看到采用分段连接的带宽利用率基本与 RTT = 250ms 时相同,而采用标准 TCP 的带宽利用率,由于 RTT 时间的增加,较 RTT =

250ms 时测试的带宽利用率更低。

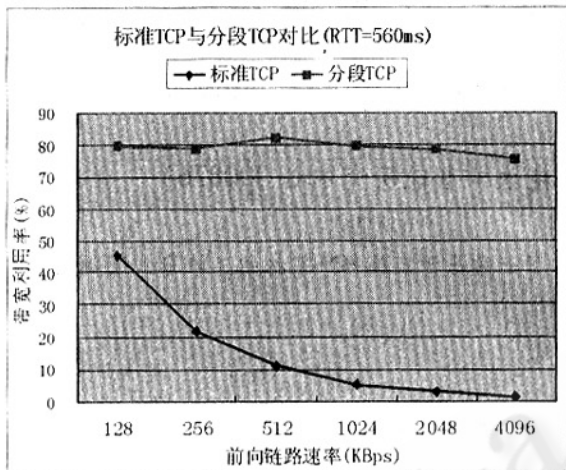


图 6 标准 TCP 和分段连接对比 (RTT = 560ms)

(3) 在 RTT = 250ms 时,采用分段连接按照不同的误码率进行 FTP 下载测试, (下载文件大小: 10Mbytes), 在图 7 中,我们可以看到,采用分段连接可以在误码率为 10^{-7} 的情况下仍能保证带宽的利用率 > 50%, 其次,无论采用何种的误码率进行测试,误码率对高带宽影响要比低带宽影响要大的多。

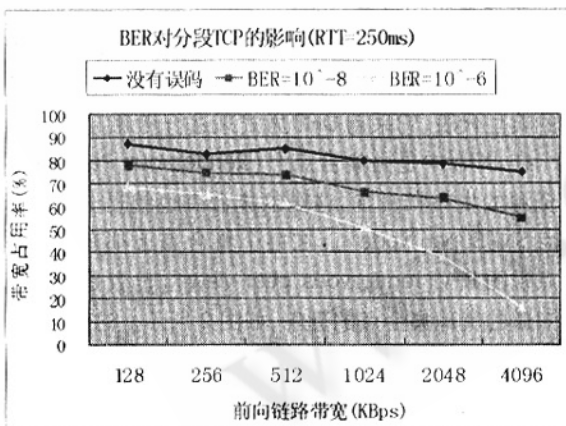


图 7 不同误码对分段连接的影响 (RTT = 250ms)

通过以上测试,我们可以看到,① 采用分段连接可以非常明显的提高 TCP 端对端的性能,② 在有误码的情况下仍有比较高的性能。③ 主机中的协议栈和

软件应用不需要变化。

通过以上测试,我们可以看到实现的协议网关有如下的优点:

- 可以很好地实现分段连接的功能。采用协议网关的分段连接可以非常明显的提高 TCP 端对端的性能,使 TCP 链路的带宽利用率可以达到 75% 以上。
- 在有误码的情况下仍有比较高的性能。可以在误码率为 10^{-7} 的情况下仍能保证带宽的利用率 > 50%
- 主机中的协议栈和软件应用不需要变化。

6 结论

本文分析了 TCP 经由卫星链路传输效率低下的原因,并对两种解决方式——端对端 TCP 协议优化/扩充方式和分段连接的方式进行了探讨。着重讨论了分段连接机制,文中提出了分段连接机制的一种实现方法,并对采用此种实现的协议网关进行了性能测试。测试结果表明,该实现可以很好的实现分段连接机制,可以明显地提高卫星 TCP 的性能。

参考文献

- 1 Mark Allman. Improving TCP Performance over Satellite Channels. Master's thesis, Ohio University, June 1997.
- 2 Henderson, T. R. and R. H. Katz, Transport Protocols for Internet - Compatible Satellite Networks, IEEE Journal on Selected Areas in Communications, Vol. 17, No. 2, pp. 345 - 359, February 1999.
- 3 M. Allman, D. Glover and L. Sanchez, Enhancing TCP Over satellite Channels using Standard Mechanisms, Request For comments, 1999, RFC 2488.
- 4 J. Border, M. Kojo, J. Griner, G. Montenegro, and Z. Shelby, Performance Enhancing Proxies Intended to Mitigate Link - Related Degradations, Internet Society, Reston, VA, RFC 3135, June 2001.
- 5 W. Richard Stevens. TCP/IP 详解 (第一卷) 协议, 北京大学出版社。