

面向卫星网络的 TCP 代理*

曾斌^{1,2+}, 李之棠¹, 徐帆江³

¹(华中科技大学 计算机学院,湖北 武汉 430074)

²(海军工程大学 管理工程系,湖北 武汉 430030)

³(中国科学院 软件研究所,北京 100080)

TCP Proxy for Satellite Network

ZENG Bin^{1,2+}, LI Zhi-Tang¹, XU Fan-Jiang³

¹(School of Computer, Huazhong University of Science and Technology, Wuhan 430074, China)

²(Department of Management, Navel University of Engineering, Wuhan 430030, China)

³(Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China)

+ Corresponding author: Phn: +86-27-62481088, E-mail: trueice@public.wh.hb.cn

Zeng B, Li ZT, Xu FJ. TCP proxy for satellite network. *Journal of Software*, 2007,18(7):1695–1704.
<http://www.jos.org.cn/1000-9825/18/1695.htm>

Abstract: When used with satellite links, traditional TCP suffers from a number of performance problems, especially for higher propagation delay, bit error rate, asymmetrical bandwidth and signal attenuation due to rain. A new active proxy without breaking TCP semantic is proposed to improve TCP performance of satellite network. The congestion control algorithm in the proxy sets the window size according to the measurement information on bandwidth changes from the two probe packet pairs with different priorities. A stability analysis of the hybrid proxy system is performed. The loss recovery based on selective negative acknowledgement (SNACK) can work even if the clocks is not synchronized. The results demonstrate that proxy in combination with SNACK allows TCP to fully exploit the possible throughput of the satellite network.

Key words: satellite network; TCP performance enhancing proxy; active measurement; SNACK (selective negative acknowledgement); performance analysis

摘要: 当传统 TCP 协议用于卫星网络时面临各种问题,原因包括:卫星信道的高传输延迟、较大的误码率、带宽不对称、信号衰减等。为了提高卫星网络环境下 TCP 的性能,在不破坏 TCP 端到端特性的前提下,提出了一种新的主动代理方案。拥塞控制算法通过具有不同优先级的双报文对实施主动监测,能够根据网络带宽的变化自适应地调整 TCP 窗口。并且,对代理控制系统进行了稳定性分析。针对卫星网络下的 TCP 丢包恢复,提出了一种不依赖时钟同步的选择性否定应答算法。模拟实验表明,提出的 TCP 代理能够充分利用卫星网络的带宽,从而达到获得最佳有效吞吐率的目的。

关键词: 卫星网络;TCP 性能增强代理;主动监测;选择性否定确认;性能分析

中图法分类号: TP393 文献标识码: A

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2002AA7170 (国家高技术研究发展计划(863))

Received 2005-11-04; Accepted 2006-04-03

TCP/IP 协议组构成了 Internet 的基础.因此,基于卫星的 Internet 应该能够继续提供基于 TCP 的应用.然而,TCP 协议的性能受到卫星链路的长时延和高误码率的较大影响.

TCP 通过一种确认的反馈机制来进行流量控制,并可靠地传输数据.发送端通过调节发送窗口来控制报文发送速率,即发送端一次最多发送的报文数不能超过发送窗口的大小.特别是当处于慢启动状态时,拥塞窗口初始为 1 个数据段,然后每收到一个确认消息对,拥塞窗口就加倍.卫星链路的长时延增大了 TCP 的往返时延(round trip time,简称 RTT),导致确认信息的延缓.这种缓慢的反馈会减弱流量控制,降低了避免拥塞的性能,并会影响吞吐量.另外,潜在的问题是,由于 LEO(low earth orbit)星座网络的动态拓扑导致了 RTT 的漂移,RTT 的剧烈变化将导致错误的超时和重传.因此,卫星链路没有得到充分利用,需要一个与带宽/时延同步增长的窗口来提高吞吐量.

卫星链路容易受到不同因素的影响(如干扰、衰落、阴影效应和雨衰),因此会有很高的误码率(bit error ratio,简称 BER).但 TCP 协议不会区分数据包丢失是由于传输错误还是由于拥塞造成的,丢包原因都被解释成网络拥塞.当接收到一个损坏的数据包时,即使没有拥塞发生,窗口的大小也随即变为原来的一半.而且,网络的不对称性也降低了 TCP 的性能.反向链路的容量限制会导致缺少确认的问题,积存的反馈信息将降低窗口的更新速度.而且,由于反向链路阻塞造成的确认信息的丢失,可能会引起不必要的重传.

TCP 中的另一个问题就是不同 RTT 的 TCP 连接的公平性问题.当 TCP 连接共享一个有瓶颈的链路时,有较长 RTT 的 TCP 连接会得到不公平的带宽分配.

因此,在卫星网络下进行 TCP 性能优化是十分具有挑战性的.本文提出了一种基于代理的 TCP 性能增强机制,对分割连接技术进行改进.在关口站运行上行代理,由其模拟出卫星网络环境.在不破坏 TCP 端到端语义的情况下,通过反馈控制保证代理报文队列维持在可控范围内,在丢包恢复机制中实现了 SNACK(selective negative acknowledgement)协议.而且,该算法没有对时钟同步作任何强制性要求,增强了系统的可扩展性.

1 相关工作

当前能够增强卫星网络下 TCP 性能的研究主要可以分为如下 4 类:

第 1 类是在不改变标准 TCP 协议的前提下,调整优化 TCP 的一些运行参数,例如接收方缓存(通告窗口)或时钟粒度等.这种方式只能部分改善 TCP 的性能,对于高带宽、长时延的宽带网络效果不明显^[1].

第 2 类是对 TCP 协议的控制机制进行修改.例如针对无线网络的 NewReno^[2],Peach^[3],SCTP(stream control transmission protocol)^[4]和 TCP WESTWOOD^[5],通过修改 TCP 的快速重传算法等机制来增强其在无线条件下的性能,它们的某些特性适用于卫星网络下的报文传输.上述研究都证明了在无线条件下,区分误码和拥塞产生的报文丢失对提高 TCP 性能具有较大的作用.

第 3 类是专门针对卫星环境设计的传输协议,包括 STP(satellite transport protocol)^[6],XSTP(extended satellite transport protocol)^[7],SCPS-TP(TCP for space communications)^[8],TCP Hybla^[9]等.STP 和 XSTP 的主要改进是通过引入链路衰减探测机制来区分误码和拥塞产生的报文丢失,TCP 发送端对于传输出错情况只是采取重传,而不发生拥塞控制行为,从而提高了 TCP 性能.这类方式都可称作 TCP 的“端修改”,需要对 TCP 进行全面修改,难以推广和部署.

为了绕开对 TCP 进行“端修改”所带来的问题,第 4 类方式利用了“分割连接”技术^[10],将原先的 TCP 端到端连接分为卫星传输部分和非卫星传输部分:非卫星传输部分主要是指发送端到上行代理的通信链路和下行代理到接收端的通信链路,它采用标准的 TCP 协议;而卫星传输部分是由上、下行代理和卫星节点组成的卫星网络,它采用针对卫星网络进行了优化的 TCP 协议.当上行代理收到 TCP 源发来的报文后迅速回传应答消息,然后由它代表 TCP 源与接收方通信.这种方式虽然能够大幅度提高 TCP 在长时延网络环境下的性能,但违背了 TCP 端到端的特性.另外,上行代理需要保存每一个连接流的数据报直至收到接收端的应答消息,而卫星网络巨大的带宽延迟积(bandwidth delay product,简称 BDP)会给代理的这种缓冲管理带来困难.近几年,这方面的研究越来越受到重视,例如 SaTPEP^[11]以及 PEPsal^[12]都利用了“分割连接”来提高 TCP 的性能.其中:SaTPEP 还需要对 TCP

源、目的端进行修改,不易于部署;而 PEPsal 虽然不需要对 TCP 端用户进行修改,但与 SaTPEP 一样,假设上、下行代理之间只有一颗 GEO(geostationary earth orbit)卫星,不存在拥塞情况.而且,由于它们破坏了 TCP 端到端语义,导致一些应用或协议(例如 IPsec)无法使用.

2 流量控制

2.1 算法描述

为了提高卫星链路的带宽利用率,并最大限度地利用卫星链路的超大 BDP,理想的 TCP 行为应该是:1) 当网络发生拥塞时,及时减小发送速率,拥塞消除时迅速恢复;2) 对于传输出错情况,只采用重传,避免进入拥塞控制状态.本文的目的就是通过 TCP 代理调节 TCP 发送端的接收窗口(receive window,简称 RWND),使卫星网络的 TCP 连接逼近上述两个理想状态.

卫星网络中物理链路的高误码率造成传输层的丢包率较大,而且,视距丢失和高频段的阴影效果都会造成长时间的通信中断(burst error).在复杂的太空条件下,TCP 会反复进入不必要的“慢启动”状态,而当发生突发性通信中断后,传统的“指数回退”方式使得 TCP 不能迅速恢复正常的发送速率,浪费了 BDP.

为此,我们在代理上采用了新的拥塞控制方案,上行代理按照一定的速率($\tau=OWD/d$, d 为整数, OWD 为单向延迟,其算法见第 4 节)连续发送一对高优先级监测报文和一对低优先级监测报文,长度都为 40 字节.它主要有以下 4 个方面的特点:

(1) 与其他采用优先级探测报文的协议不同,如 Peach^[3],是由上行代理而不是由端用户发送探测报文,从而增强了透明性,易于部署;而且,发送的是带优先级的探测报文对,不仅能够及时发现链路异常情况,利用报文对背靠背两个报文之间发送间隔与接收间隔的不同,还可以估算出网络可用带宽(B_{aw})^[13].

(2) 占用网络带宽少,所占卫星链路容量的比例为($d \times 160/BDP$),由于卫星网络的 BDP 一般为 10MB 以上,所以探测报文占用资源较少.

(3) 准确、可靠地得到网络拥塞情况.由于这两对报文除了优先级(type of service,简称 TOS)不同以外,长度和传输速率都相同,所以,链路层误码率对它们造成的丢包率也都相同.当发生网络拥塞时,卫星路由器会首先丢弃低优先级的报文.所以,根据这两种报文对的丢包率,可以推测出网络的拥塞情况.下行代理以滑动时窗 λ_w 为单位统计到达的低优先级报文数(N_{lp})和高优先级报文数(N_{hp}),每隔一个测量周期回传包含(N_{lp},N_{hp})的监测应答消息.这样,上行代理根据 $\delta=N_{lp}/N_{hp}$ 的值就可以推测出一个测量周期内的网络拥塞情况.我们设置 δ 的下限为 δ_l ,上限为 δ_u ,当 $\delta < \delta_l$ 时,可推测网络发生拥塞,需要降低发送速率;当 $\delta > \delta_u$ 时,网络空闲,应该提高发送速率.

(4) 迅速发现网络通信中断和恢复时刻.如果上行代理在一个 RTT 时间后还没有得到任何应答消息,则推测网络发生突发性中断,立即堵塞所有活动 TCP 连接($rwnd_i=0$),并保持发送监测报文对;而下行代理如果在 OWD(one way delay)时间内没有收到任何报文,也认为网络发生中断,提高速率发送监测应答消息,域(N_{lp},N_{hp})设为(0,0),上行代理一旦收到应答消息,可以判断中断恢复,即尽快发送新报文.

一旦发现网络发生拥塞/中断后,上行代理需要调节应答消息中接收窗口($rwnd_i$)的值,其计算公式如下:

$$rwnd_i = \min(W_i, rwnd_i) \quad (1)$$

$$W_i(t) = \begin{cases} W_i(t-1) - 1, & \delta < \delta_l \\ W_i(t-1) + \Delta W_i, & \delta > \delta_u \\ W_i(t-1) + \Delta q, & \delta_l < \delta < \delta_u \end{cases} \quad (2)$$

$$\Delta W_i = (\hat{B}_{aw} \hat{T}_{rtt} + q_x) \left(1 - \frac{W_i}{n \sum_{i=1}^n W_i} \right) \quad (3)$$

其中: $q_x = \sum W_i - BDP$; $\Delta q = k(q_x - q(t))/n$, n 为 t 时刻活动连接数目. ΔW_i 的这种设置,使具有较小接收窗口的连接得到较大的增额,达到公平分配带宽的目的. ΔW_i 之和为($BDP + q_x$), q_x 为上行代理缓冲队列的控制长度.由于卫星网络

中BDP的值是动态变化的,TCP发送端发送的报文数与链路能够接收的报文数也就不断产生波动,所以,上行代理除了SOCKET缓存以外,还需要设置缓冲队列来吸纳这部分差值.假设上行代理处理速度足够快,可以看出 q_x 需要满足公式: $q_x \geq \Delta RTT_{\max} \cdot B_{\max}$.通过引入 Δq 这个反馈机制,可以减少可用带宽和往返时延的测量误差所造成的影响.

```
/* CONNECTION */
```

```
Send two probe packet pairs of different priority with rate  $r_{probe}$ ;
```

```
If (PROBE_ACK_RECV) OR (DATA_ACK_RECV)
```

```
/*Congestion Decision*,  $N_{ac}$  denotes the subset formed by active connections of the proxy/
```

```
For (split-conn: $i \in N_{ac}$ )
```

```
  If ( $\delta < \delta_l$ )
```

```
/*Decrease  $rwnd_i$ */
```

```
   $dwnd_i = -1 * MSS$ ;
```

```
  Else If ( $\delta_l < \delta < \delta_u$ )
```

```
/* Adjust  $rwnd_i$ */
```

```
   $dwnd_i = (q_x - q_i) / n$ 
```

```
  Else If ( $\delta > \delta_u$ )
```

```
/*Increase  $rwnd_i$ */
```

$$dwnd_i = (\hat{B}_{aw} \hat{T}_{rtt} + q_x) \left(1 - \frac{W_i}{n \sum_{i=1}^n W_i} \right)$$

```
  End If;
```

```
End If;
```

```
End For;
```

```
 $rwnd_i += dwnd_i$ ;
```

```
end if;
```

```
/* INTERMIT*/
```

```
If (NO_ACK_IN_Td)
```

```
For (split-conn: $i \in N_{ac}$ )  $rwnd_i = 0$ ; end for;
```

```
  Transmit probe packet pairs;
```

```
If (PROBE_ACK_RECV) OR (DATA_ACK_RECV)
```

```
  Goto CONNECTION;
```

```
Else
```

```
  Goto INTERMIT;
```

```
End if
```

2.2 稳定性分析

设 $W(t)$ 为在时刻 t 所有发送方发送窗口之和, T_f 为发送方至上行代理的时延, T_b 为报文由上行代理至接收方, 其应答报文再返回上行代理的时延, $T = 2 \times T_f + T_b$, $T_b \gg T_f$. t 时刻发送方发送速率之和为

$$r(t) = \frac{W(t)}{T + q(t - T_b - T_f) / b} \quad (4)$$

b 为卫星网络带宽, $T + q(t - T_b - T_f) / b$ 为 RTT 加上上行代理缓冲时间的往返时延. 上行代理缓冲队列的变化率为其接收速率和与卫星网络带宽之差:

$$\dot{q}(t) = r(t - T_f) - b = \frac{W(t - T_f)}{T + q(t - T)/b} - b = \frac{W(t - T_f) - bT - q(t - T)}{T + q(t - T)/b} \quad (5)$$

定理 1. 公式(2)构造的代理系统为渐进稳定系统,平衡点为 $q_x + (\hat{b}T - bT)$ (网络容量的测量误差).

证明:由公式(2)可得出 $W(t) = \hat{b}\hat{T} + q_x$. \hat{b} 为卫星链路的测量带宽,带入式(5)得:

$$\dot{q}(t) = \frac{\hat{b}\hat{T} + q_x - bT - q(t - T)}{T + q(t - T)/b} \quad (6)$$

该时延差分方程的平衡点为 $q^* = q_x + \hat{b}\hat{T} - bT$, 带入式(6)得:

$$\dot{q}(t) = \frac{q^* - q(t - T)}{T + q(t - T)/b}.$$

定义 $T^* = T + q^*/b$ 为对应平衡点的往返时延,式(6)可线性化为

$$\dot{q}(t) = -\frac{1}{T^*} (q(t - T) - q^*).$$

由于 $\frac{T}{T^*} < 1 < \frac{\pi}{2}$, 根据 Nyquist 判据可得系统渐进稳定.

定理 2. 引入缓冲反馈机制后,公式(2)构造的代理系统全局渐进稳定.

证明:引入缓冲反馈机制的目的是消除可用带宽和往返时延的测量误差,为此我们采用了一个积分控制器.

$$\dot{w}(t) = -k(q(t - T_b - T_f) - q_x).$$

这时,代理系统可由以下差分方程组表达:

$$\begin{cases} \dot{q}(t) = \frac{w(t - T_f) - bT - q(t - T)}{T + q(t - T)/b} \\ \dot{w}(t) = -k(q(t - T_b - T_f) - q_x) \end{cases}.$$

该方程组的平衡点为 $q^* = q_x, w^* = bT + q_x$. 由于 $T^* = T + q_x/b$, 系统可表达为

$$\begin{cases} \dot{q}(t) = \frac{w(t - T_f) - w^* - (q(t - T) - q^*)}{T^* + (q(t - T) - q^*)/b} \\ \dot{w}(t) = -k(q(t - T_b - T_f) - q^*) \end{cases}.$$

设 $\tilde{q} = q(t - T) - q^*, \tilde{w} = w(t - T_f) - w^*$, 由于 $T \gg T_f$, 上述非线性系统可简化成

$$\begin{cases} \dot{\tilde{q}}(t) = \frac{\tilde{w} - \tilde{q}}{T^* + \tilde{q}/b} \\ \dot{\tilde{w}}(t) = -k\tilde{q} \end{cases}.$$

针对往返时延,我们引入了一个虚拟时间变量 $i, dt = (T^* + \tilde{q}/b)di$, 则有,

$$\begin{cases} \frac{d\tilde{q}}{di} = \tilde{w} - \tilde{q} \\ \frac{d\tilde{w}}{di} = -kT^*\tilde{q} - \frac{k}{b}\tilde{q}^2 \end{cases} \quad (7)$$

我们的目的是证明在 $w, q > 0$ 的区域内,式(7)全局渐进稳定.在区域边界上有:当 $w=0$ 且 $\tilde{q} > 0$ 时, $\tilde{w} = 0$; 当 $q=0$ 且 $\tilde{w} < -q_x$ 时, $\tilde{q} = 0$. 我们定义 Lyapunov 后选函数如下:

$$V(\tilde{q}, \tilde{w}) = \tilde{q}^2(\tilde{q} + A) + \frac{Bb}{2k}\tilde{w}^2,$$

则有: $\frac{dV}{di} = \tilde{q}(3\tilde{q} + 2A)(\tilde{w} - \tilde{q}) + B\tilde{w}\tilde{q}(-bT^* - \tilde{q}) = -\tilde{q}^2(3\tilde{q} + 2A) + \tilde{q}\tilde{w}(2A - BbT^*) + \tilde{q}^2\tilde{w}(3 - B)$.

设置 $B=3, A=3bT^*/2$, 有,

$$\tilde{q} + A = q - q_x + 3(bT + q_x)/2 = q + q_x/2 + 3bT/2 > 0,$$

$$3\tilde{q} + 2A = 3(q - q_x) + 3(bT + q_x) = 3(q + bT) > 0.$$

因此有: $V(\tilde{q}, \tilde{w}) = \tilde{q}^2(q + \frac{q_x + 3bT}{2}) + \frac{3b}{2k}\tilde{w}^2 \geq 0, \frac{dV}{dt} = -3\tilde{q}^2(q + bT) \leq 0.$

根据 Lyapunov 稳定性定理,本系统为全局渐进稳定.

3 丢失恢复

当发生报文丢失时,标准的 TCP 协议在一个 RTT 时间内只能进行一次重传,这在卫星网络中是不太适合的.例如,如果单向时延为 500ms,数据发送率为 1Mbps,一个报文平均有 1 000 字节,误码率 BER 为 $1e-5$,那么,平均一个 RTT 内会丢失 10 个报文,这种丢失恢复机制在卫星网络环境下效率较低.

TCP 重传增强协议有多种,根据文献[14],SCPS-TP 的性能最佳. SCPS-TP 对 TCP 协议的一个重要改进就是增加了 SNACK 机制. SNACK 允许接收方“显性”地通知发送方丢失的数据段(“空洞”).但在 SNACK 中没有规定时钟/重传算法,需要根据具体应用来实现.为了进行可靠恢复(robust recovery),接收方应给发送方足够的时间,使其收到 SNACK 后重传丢失报文,并经过链路到达接收方.而且 SNACK 还需要一定延迟,保证“乱序”数据段能够到达.

为了能够利用 SNACK 实现有效的报文重传,下行代理必须能够精确估算出上行代理与它的单向延迟(OWD),利用 OWD 及其偏差,下行代理设置“接收超时值”,当“空洞”报文超时后触发第 1 个 SNACK 消息,如果第一次重传失败,下行代理需要申请后续重传,这时,需要估算出往返时延 RTT 来设置“重传定时器”.由于下行代理通过自己的时钟就可以计算出 RTT,所以,后续重传可以直接利用标准 TCP 的超时算法.但首次重传计算 OWD 时,需要上、下行代理时钟同步.由于卫星网络代理地理分布分散,而且卫星链路时延以及时延变化量非常大,这使得上/下行代理采用时钟同步协议(network time protocol,简称 NTP)开销太大,所以,实现 SNACK 的关键是如何建立不依赖时钟同步的 SNACK 首次重传机制.如图 1 所示.

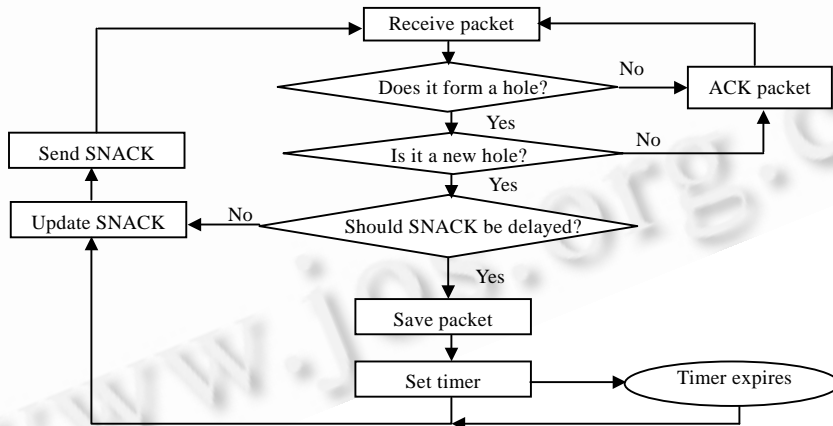


Fig.1 Flow of control in SNACK downlink proxy

图 1 下行代理 SNACK 流程图

设 T_i 为上行代理传送报文 i 的实际发送时刻, T'_i 为下行代理估算的报文 i 的发送时刻,由于上/下行代理时钟不同步,所以, T_i 与 T'_i 的差可能非常大,理想的定时器超时值应该不受这个差值的影响.我们假设上行代理每隔 M 秒传输一个报文,下行代理收到报文 0 时把时钟相应设置为 0.参照下行代理时钟,上行代理发送报文 i 的时刻为

$$\tilde{T}_i = iM \quad (i \geq 0) \tag{8}$$

由于时钟不同步,参照下行代理时钟,报文 i 的实际发送时刻为

$$T_i = i(M - \phi) + \Delta \quad (i \geq 0) \tag{9}$$

Δ 为上/下行代理时钟的起始差值, ϕ 为上行代理时钟的增加速率与下行代理时钟增加速率的差值.由于上/下行代理之间的卫星链路具有随机的时延变化值,所以缺乏有效的方法来对 Δ 和 ϕ 进行同步.设 A_i 为下行代理测量出的报文 i 的到达时刻.

$$A_i = T_i + d_i = i(M - \phi) + \Delta + d_i (i \geq 0) \quad (10)$$

d_i 为上行代理至下行代理的单向时延.为了能够计算出报文 $(i+1)$ 的超时值,下行代理需要估算出其到达时刻.首先需要计算单向时延 d_i 的值.我们借鉴了TCP中计算RTT的平滑算法.设 D_i 为估算的单向时延 \tilde{d}_i ,

$$D_i = \lambda_D D_{i-1} + (1 - \lambda_D) \tilde{d}_i, D_0 = \tilde{d}_0 \quad (11)$$

其中: $\tilde{d}_i = A_i - \tilde{T}_i = A_i - iM = \Delta + d_i - i\phi$; λ_D 为修正因子,决定以前的单向时延的权值.对 D_i 求解,则有,

$$D_i = \Delta - \left(i - \sum_{k=1}^i \lambda_D^k \right) \phi + \lambda_D^i d_0 + (1 - \lambda_D) \sum_{k=1}^i \lambda_D^{i-k} d_k; i \geq 1 \quad (12)$$

下行代理利用 D_i 估算报文 $i+1$ 的到达时刻:

$$\tilde{A}_{i+1} = \tilde{T}_{i+1} + D_i = (i+1)M + D_i \quad (13)$$

该估算值的偏差量为

$$\begin{aligned} e_{i+1} &= \tilde{A}_{i+1} - A_{i+1} = D_i + (i+1)\phi - \Delta - d_{i+1} \\ &= \sum_{k=0}^i \lambda_D^k \phi + \lambda_D^i d_0 + (1 - \lambda_D) \sum_{k=1}^i \lambda_D^{i-k} d_k - d_{i+1} \end{aligned} \quad (14)$$

式(14)的第1项为时钟不同步的影响量;其他项与时延有关,不受时钟影响.如果 d_i 保持不变,其他项为0.式(14)第1项的绝对值在 $i \rightarrow \infty$ 时,接近 $|\phi/(1 - \lambda_D)|$.由于 ϕ 为连续两个报文时间间隔的时钟漂移,因此其值应该非常小.以一个很宽松的条件为例,连续两个报文的时间间隔为毫秒级,时钟频率的精度为 10^{-3} , ϕ 则为微秒级, $(1 - \lambda_D)$ 的实际值一般设为0.1.所以在我们的算法中,时钟不同步带来的影响非常小.

为了进一步消除时延变化带来的影响,报文 $(i+1)$ 的超时值为其到达时刻的估算值加上一个修正量 $(g_i, g_{i-1}, g_{i-2}, \dots)$,它们分别代表报文 $(i+1)$ 的实际到达时刻与估算值的似然偏差.我们利用 $|g_i|$ 的平滑值 G_i 来设置超时值.

$$G_i = \lambda_g + (1 - \lambda_g) |g_i| \quad (15)$$

λ_g 同样为介于0和1之间的常数,报文 $i+1$ 的超时值为

$$TO_{i+1} = \tilde{A}_{i+1} + \alpha G_i, \alpha > 1 \quad (16)$$

由于卫星链路具有较大的丢包率,所以在修正 D_i 和 G_i 时需要考虑报文丢失的情况.为此,应该进一步计算接收报文的统计数,这里借鉴了TCP修正超时值的平滑算法.

$$\begin{aligned} D_{new} &= \lambda_D D_{old} + (1 - \lambda_D) \tilde{d}_{new} \\ G_{new} &= \lambda_g + (1 - \lambda_g) | \tilde{A}_{new} - A_{new} | \\ \tilde{A}_i &= iM + D_{new}, TO_i = \tilde{A}_i + \alpha G_{new} \end{aligned} \quad (17)$$

4 性能分析

为了验证本文提出的TCP协议(下称SatMP)的有效性,我们以NS-2为实验平台,建立了LEO(low earth orbit)低轨道卫星链路模型,网络拓扑结构如图2所示.在这种结构中,上行链路的单程传输时延为10ms,带宽为1.5Mb/s,星际链路(inter-satellite link,简称ISL)传输延迟为10ms,带宽为25Mb/s.从用户到卫星网关是时延1ms、带宽10Mb/s的信道.参数配置类似于Iridium系统结构.这个模型同样反映出了卫星网络环境的长时延、高带宽的特性.同时,在网络拓扑上体现出了LEO卫星网络环境与GEO卫星网络环境的基本区别:卫星数量的增加、星际链路的存在、卫星信道传输延迟的减小等.这些已经足以研究监测算法在LEO卫星网络中的性能.

由图3可以看出,SatMP的吞吐率和链路利用率超出了常规TCP.

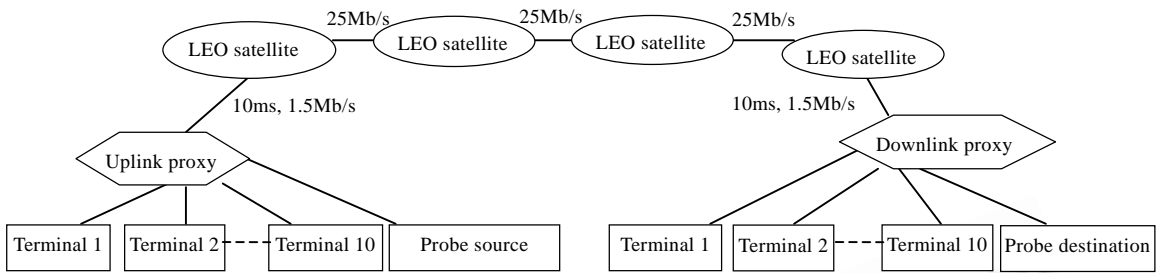


Fig.2 Simulation model of LEO
图 2 LEO 的仿真模型

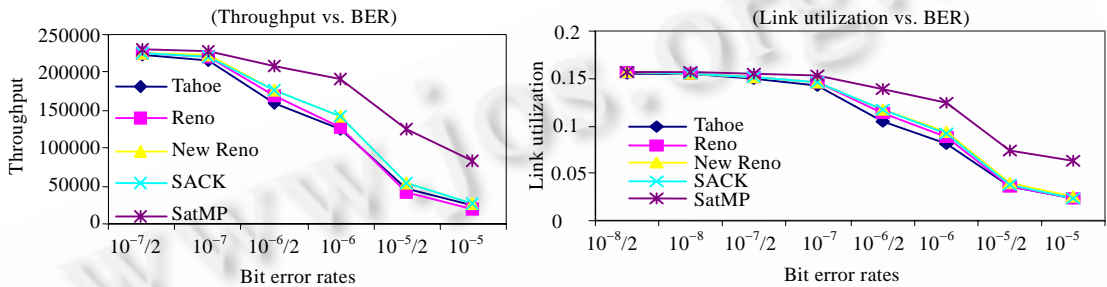


Fig.3 Throughputs and link utilization for Tahoe, Reno, NewReno, SACK, and SatMP
图 3 Tahoe,Reno,NewReno,SACK 和 SatMP 的吞吐量与链路利用率

在图 4 中,随着 BER 的增加,与 SatMP 相比,常规 TCP 拥塞窗口的抖动更加剧烈,而且 SatMP 的拥塞窗口比常规 TCP 大得多。

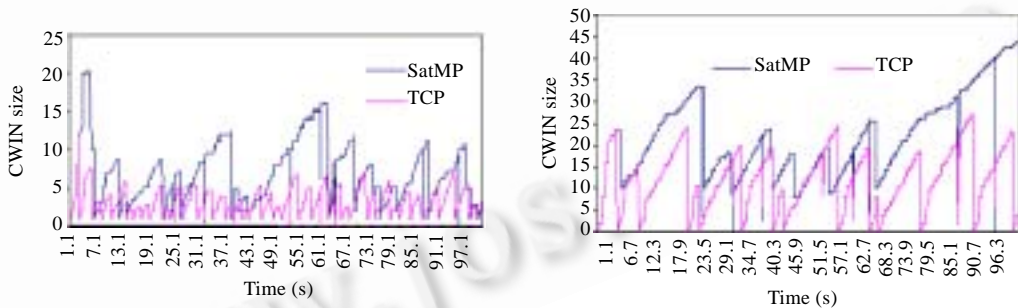


Fig.4 Congestion window vs. time of flow 1—Left denotes BER 10^{-5} and right denotes BER 10^{-6}
图 4 仿真模型中流 1 发送端拥塞窗口的变化——左图 BER= 10^{-5} ,右图 BER= 10^{-6}

我们采用了丢包概率为 10^{-4} 的 Bernoulli 丢包模式.从图 5 可以看出,SatMP 的上行代理维持了一个相对稳定的缓冲区长度,一般都能够能在 1s 内恢复到平衡点。

卫星链路除了长延迟和位传输误码率较高以外,在 EHF(extremely high frequency)以上频段工作的卫星链路通道(上/下行)还存在着持续性链路衰减问题.由于 NS-2 中缺乏对卫星链路衰减的支持,我们在链路模型中加入了一个两状态马尔可夫链衰减模型,如图 6 所示。

S_0 状态没有衰减, S_1 状态我们引入一个固定时间间隔的链路衰减,这种简化的开关模型可以基本模拟卫星链路的衰减以及由其引起的虚假超时现象^[15].该马尔可夫链状态迁移矩阵如下:

$$Q = \begin{pmatrix} 1 - P_{err} & P_{err} \\ 1 & 0 \end{pmatrix}.$$

马尔可夫链衰减模型需要 3 个参数: P_{err} 、衰减时间、状态迁移间隔.在我们的仿真实验中, P_{err} 取不同的数值,衰减时间固定为 50s;状态迁移间隔为 1s;上/下行链路带宽增大为 10Mbps;星间链路带宽增大为 155Mb/s;报文长度增大为 5 760 字节.

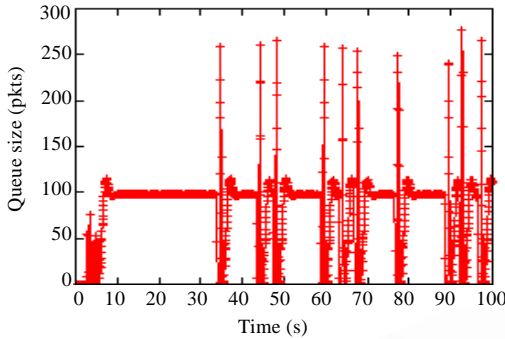


Fig.5 Change of queue size in uplink proxy
图 5 上行代理队列长度的变化情况

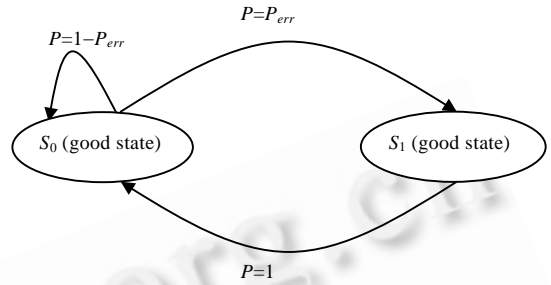


Fig.6 Two state error model
图 6 两状态错误模型

图 7 为不同 P_{err} 情况下 SatMP 与标准 TCP 有效吞吐率的比较,可以看出,SatMP 的性能极大地优于标准 TCP,接近于理想状态下的有效吞吐率.

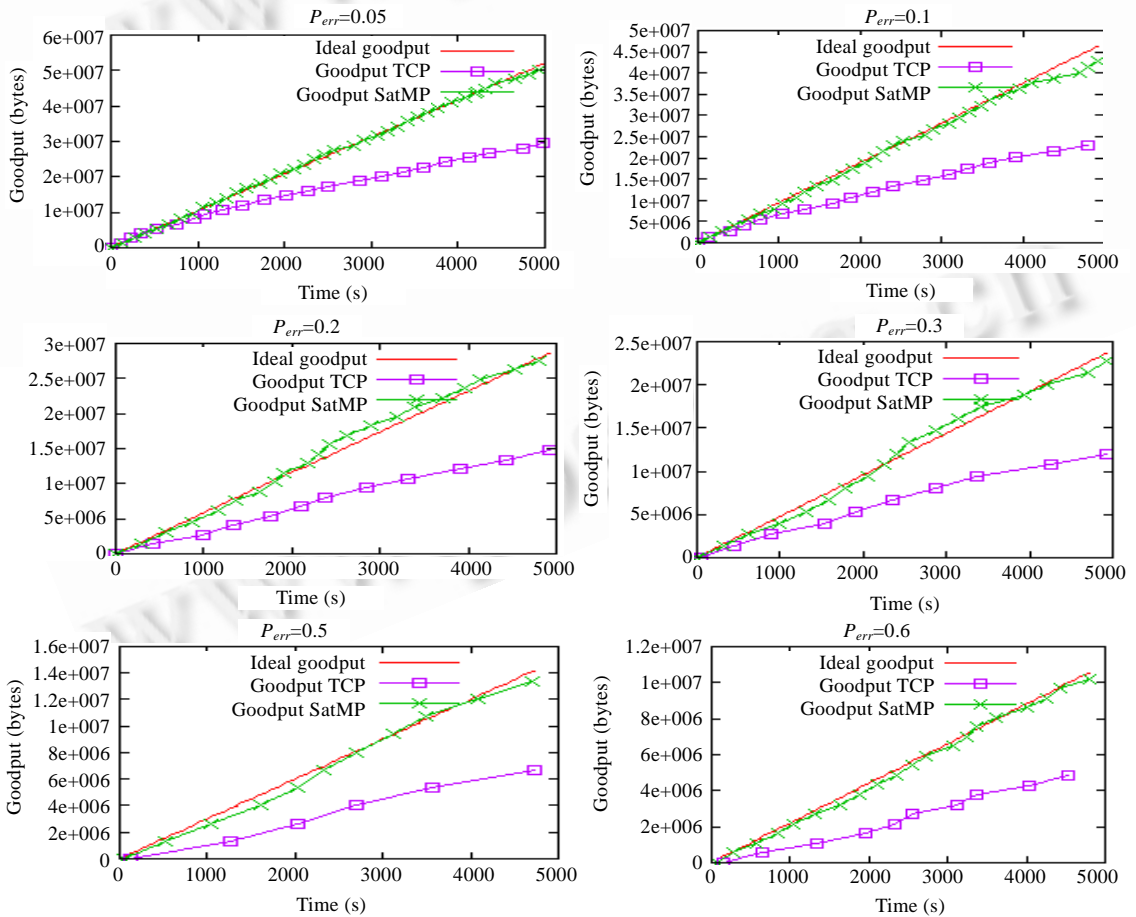


Fig.7 Goodput of flow 1 at different P_{err}

图 7 不同 P_{err} 情况下流 1 的有效吞吐率

5 结 论

为了提高卫星网络下 TCP 的性能,我们提出了一种新的 TCP 性能增强代理——SatMP. SatMP 的流量控制算法采用了主动报文监测技术,能够根据卫星网络状态的改变进行自适应调整,并利用反馈控制使代理缓冲区维持在期望范围内.SatMP 的丢包恢复机制采用了选择性否定确认机制,其实现算法不依赖时钟同步,特别适用于地理位置分散的地面卫星站.SatMP 不破坏 TCP 语义,不需要改动端用户的 TCP 协议栈,部署方便.理论分析和大量的仿真结果表明:SatMP 对于卫星网络下 TCP 的有效吞吐率、带宽利用率都有较大的提高.

References:

- [1] Emmelmann M. Effects of advertised receive buffer size and timer granularity on TCP performance over erroneous links in a LEO satellite network. In: Wang CL, ed. Proc. of the IEEE Global Telecommunications Conf. Taipei: IEEE Press, 2002. 2955–2958.
- [2] Houyou AM, Holzer R, Meer HD, Heindl M. Performance of transport layer protocols in LEO pico-satellite constellations. Technical Report, MIP-0502, Passau: University of Passau, 2005. 2–8.
- [3] Akyildiz F, Morabito G, Palazzo S. TCP-Peach: A new congestion control scheme for satellite IP networks. IEEE/ACM Trans. on Networking, 2001(9):307–321.
- [4] Fu SJ, Ivancic W. SCTP over satellite networks. In: Victor O, Wei K, eds. Proc. of the IEEE Computer Communications Workshop (CCW 2003). Dana Point: IEEE Press, 2003. 112–116.
- [5] Casetti C, Gerla M, Mascolo S, Sanadidi M, Wang R. TCP westwood: Bandwidth estimation for enhanced transport over wireless links. In: Basagni S, Sivalingam K, eds. Proc. of the MOBICOM 2001, Vol. 4. Rome: IEEE Press, 2001. 54–62.
- [6] Henderson T, Katz R. Transport protocols for Internet-compatible satellite networks. IEEE Journal on Selected Areas in Communications, 1999,17(2):326–344.
- [7] Elaasar ME. XSTP: eXtended satellite transport protocol [MS. Thesis]. Ottawa: Carleton University, 2003.
- [8] Space communications protocol specification transport protocol (SCPS-TP). Recommendation for space data systems standards. CCSDS 714.0-B-1, Blue Book. Washington: CCSDS, 1999.
- [9] Caini C, Firrincieli R. TCP hybla: A TCP enhancement for heterogeneous networks. Int'l Journal of Satellite Journal, 2004,22(5): 547–566.
- [10] Luglio M, Sanadidi MY, Gerla M, Stepanek J. On-Board satellite “Split TCP” proxy. IEEE Journal on Selected Areas in Communications, 2004,22(2):362–370.
- [11] Velenis D, Kalogeras D, Maglaris S. SaTPEP: A TCP performance enhancing proxy for satellite links. In: Gregori E, Conti M, Campbell AT, Omidyar C, Zukerman M, eds. Proc. of the 2nd Int'l IFIP-TC6 Networking Conf. Pisa: Springer-Verlag, 2002. 1233–1238.
- [12] Lacamera D. PEPsal 0.1.3. 2005. http://sourceforge.net/project/showfiles.php?group_id=143401
- [13] Hu NN, Steenkiste P. Evaluation and characterization of available bandwidth probing techniques. IEEE Journal on Selected Areas in Communications, 2003,21(6):879–894.
- [14] Loreti P, Luglio M, Kapoor R, Stepanek J, Gerla M, Vatalaro F. LEO satellite systems performance with TCP-IP applications. In: Edmonds A, Yenser G, Ferrari G, eds. Proc. of the IEEE Military Communications Conf. New York: IEEE Press, 2001. 28–31.
- [15] Zhu J, Roy S. Improving link layer performance on satellite channels with shadowing via delayed two-copy selective repeat ARQ. IEEE Journal on Selected Areas in Communications, 2004,22(3):472–481.



曾斌(1970 -),男,江西南丰人,博士,主要研究领域为卫星通信,网络仿真,网络监测.



徐帆江(1973 -),男,博士生,主要研究领域为天基综合信息网,计算机仿真.



李之棠(1951 -),男,教授,博士生导师,主要研究领域为计算机网络,网络安全,计算机系统结构,并行处理.