移动环境下支持最小速率保证的端端可靠传输机制*)

A Reliable End-to-End Transport Mechanism Supporting Minimum Rate Guarantees over Mobile Environmets

李方敏[1,2] 叶澄清[2] 李仁发[1]

(湘潭工学院计算机系 湘潭411201)[1] (浙江大学计算机系 杭州310027)[2]

Abstract With the development of wireles communication, people are not content with the traditional Internet access method, and present mobile Internet. Many TCP-based applications, such as mutual Web applications, need a mechanism supporting bandwidth reservation and minimum delay, but the inherent flaws of wireless channel, high bit error rate, delay variation and handover, degrade gravely TCP performance. On the basis of analysis and comparison between wireless and wireline communication, we improve snoop protocol, and use token bucket to mark the traffic injecting into network. The snoop agent in base station uses RIO to differentiate the packets according to their marks, which not only improve TCP performance over the wireless environment, but also support minimum rate guarantees.

Keywords Wireless communication, Snoop protocol, Minimum rate guarantees, RIO, Token marking

1. 引言

随着无线通信技术的发展，人们不再满足在固定地点从 Internet 检索、传输信息，希望能在任何时候、任何地点能方便地访问 Internet，因此提出了移动 Internet，它是传统 Internet 的有效延伸和扩展。

不管在有线还是无线情况下，IETF 的集成服务提供两种类型的服务，一类是保证服务[1]，另一类是适应服务[2]，前者对延时、丢包有严格的要求，一般应用于音频、视频会议等应用，而后者一般有最少带宽需求，但对延时要求较松，一般应用于文件传输以及 Web 应用，本文主要探讨在异构无线移动环境下对自适应服务的支持。

经过近三十年的发展，TCP/IP 协议已能有效地适应有线传输，特别是 TCP 基于慢速启动、拥挤避免、快速重传、快速恢复的拥挤控制和丢包恢复机制。但在无线移动环境下，无线信道的突发性位错误、较大的延时变化，以及移动主机的越区切换，所有这些都使得传统的 TCP 拥挤控制和丢包恢复机制并不能有效适应无线移动环境，特别是对最小带宽保证的支持。

本文将首先讨论在无线移动环境下 TCP 协议存在的问题，然后探讨了其解决方法，在此基础上我们结合令牌机制[3]在网络边界对注入网络的 TCP 流量进行标记，然后由基站的代理根据优先级标记对数据包进行相应的处理，从而支持最小速率保证服务。

2. 无线环境下 TCP 需解决的问题

相对固定通信而言，由于无线信道内在的缺点，如信道出错概率高、延时变化大、带宽不对称等，要有效支持文献[2]中的服务则 TCP 协议需解决以下四个方面的问题：

2.1 丢包问题

在有线环境下，TCP 默认丢包是由于网络拥挤造成的，因此发送方一旦检测到包丢失，则减少其发送窗口，但在无线移动环境下，无线信道突发性的位错误导致包损坏而丢弃，另外，移动主机的越区切换也可能导致短期的突发性丢包，如果 TCP 发送者将以上的丢包误认为拥挤所致，则将减少传输窗口，导致粗粒度的超时，进一步导致无线信道的空闲，大大降低端端性能，因此，必须采用一种机制，将非拥挤产生的丢包与拥挤丢包区别开来，对前者只将丢弃的包重传，无需减小 TCP 的传输窗口，而对拥挤丢包，将执行 TCP 的拥挤控制。

2.2 不对称性问题

文[4]对 TCP 性能的不对称性定义如下：如果 TCP 的吞吐率不是完全由下行链路的带宽和流量特性决定，还依赖反方向的链路特性，则该网络环境具有不对称性。

在无线移动环境下，不对称性涉及很多种情况，我们将主要考虑带宽不对称和延时变化。假设我们使用

*)本文受国家自然科学基金资助(编号：69974031)。李方敏 副教授，博士研究生，研究方向为网络服务质量、新型网络协议、多媒体通信技术，叶澄清 教授，博导，研究领域为高性能体系结构、计算机网络、多媒体应用等，李仁发 教授，研究方向为分布式计算、虚拟技术、智能控制。

cable modem 网络或卫星信道作为下行链路，而用包音频网络或拨号作为上行链路，则由于传输 ACK 的上行链路的低带宽或大延时，将导致 ACK 丢弃或大的队列延时而不能及时到达发送方，发送方在给定间隔由于收不到足够数目的重复 ACK 而导致超时，恶化了端端性能。

2.3 小传输窗口问题

在无线移动环境下，移动主机的可用带宽往往较低，并且目前基于 TCP 的主要应用 Web 传输一般每次传输的数据量较小（典型的 Web 页面），这样将限制 TCP 发送者的传输窗口，而小的传输窗口一旦丢包，发送者就不能接收足够数目的重复 ACK 来触发快速重传，只能通过超时机制恢复，降低了对可用带宽的使用效率。

2.4 标记处理问题

要在无线移动环境下支持最小速率保证，一方面要在发送端或网络边界对注入网络的 TCP 流量进行标记，另一方面要在基站对标记流进行相应的处理，以优先保证标记流的传输，我们将采用令牌算法[3]和在基站设置代理解决这两个问题。

3. 提高端端传输性能

为提高 TCP 在易出错的蜂窝无线网络环境下传输数据的端端性能，业界已进行相当多的研究，主要三种方法：

3.1 链路层协议

链路层协议是解决易出错的无线链路的传统方法，试图对上层协议屏蔽易出错的无线链路，一般采用 ARQ（Acknowledgement-Repeat-reQuest）和 FEC（Forward Error Correction），或二者的组合[6]，但 FEC 要求出错是随机的，并且带宽较大，因此 ARQ 或 ARQ/FEC 更多地用于低速无线链路。

但是采用复杂的可靠链路层协议并不能有效提高 TCP 的效率，因为 TCP 本身也提供端端可靠数据传输，二者独立的定时器[5]和快速重传机[7]制将限制 TCP 性能提高，并且复杂的可靠链路层增大了数据的往返传输延时。

3.2 端端协议

为克服采用复杂的链路层协议屏蔽无线错误存在的缺点，Berkeley 利用交错层协议[7]优化的思想，提出了意识传输层的可靠链路层协议——Snoop 协议[8]。而本文将改进 Snoop 协议来支持最小速率保证。

3.3 分段连接

90年代中期，提出分段连接[12,13]提高 TCP 在无线链路上的传输性能，其基本思想是将一个 TCP 连接在基站分离成两个独立的连接：一个连接从固定主机到基站，另外一个连接基站和移动主机。

文[4]将分段连接方法与其它方案进行了比较分析，虽然分段连接能提高性能，但主要是在基站与移动主机之间采用了无线优化技巧，并不是由于分段的缘故，同样，在 Snoop 中采用无线优化技术，不但不会破坏端端语义，而且能获得更好的性能。

4. Snoop 协议

Snoop 的设计目标包括：采用本地解决方案、消除协议层之间的负面交互作用、支持逐步配置、保持端端语义、使用软状态等。

Snoop 协议只需修改经由无线链路连接的基站和移动主机的协议，对通过有线连接的其它设备的协议无须修改，设置在基站的 Snoop 代理完成丢包检测和丢包恢复功能。

在无线移动环境下，主要是移动主机从固定主机下载数据，本文也主要研究这种情况。基站的 Snoop 代理缓冲未确认的 TCP 数据段，如果是由于无线链路出错导致丢包，则依赖移动主机的 ACK 和自己的定时器完成本地数据重传。这样，Snoop 代理使发送者意识不到无线链路出错，我们也只需修改基站的协议。

如果移动主机上载数据到固定主机，则 Snoop 代理监控从移动主机来的 TCP 数据段，检测由于无线链路出错而损坏的数据段，并将不连续的间隙记录下来。当接收到固定主机的重复 ACK 后，如果代理确认是无线链路出错而导致丢包，则在对应的 TCP 头中设置一个 ELN（Explicit Loss Notification）位，然后转发给移动主机，发送者通过 ELN 位知道丢弃的包不是拥挤原因导致的，它重传该数据段，但不执行任何拥挤控制。

下面我们分析移动主机从固定主机下载数据时的 Snoop 协议行为：修改基站软件，增加 Snoop 代理监控通过连接的每个数据包，但基站并没有传输层，代理由数据包处理和 ACK 处理两个部分组成。

4.1 数据处理

Snoop 代理根据 TCP 数据段第一个字节的顺序号和长度来识别每个连接的不同包，并且跟踪连接的最后顺序号。代理能从固定主机接收三种类型的数据包，并且分别进行不同的处理：

·按顺序号增加的新包　这是传输的正常情况，代理缓冲之后转发该包，并且记录转发时间以便以后计算无线链路的 RTT。

·已经缓冲的乱序包　三种原因可能导致这种情况。第一，由于有线链路的拥挤，导致单个窗口中多个包丢失、超时，慢速启动重传已被缓冲的数据段。第二，无线链路持续拥挤，没有触发代理的本地重传。第三，

无线链路长时间出错、导致发送者超时重传。

如果到达包的顺序号大于最后 ACK 中确认的顺序号，则移动主机很可能没有收到该数据段，代理转发该包。如顺序号小于最后 ACK 包中确认的顺序号，则移动主机很可能已经收到该包，为简单起见，Snoop 代理直接转发该包，以后一旦收到移动主机的 ACK 则刷新代理的相关状态，因为代理使用的是软状态，即使移动主机已收到该包。

·没有缓冲的乱序包　这种情况可能是有线链路拥挤导致包丢失，或者网络没有按顺序传递，下管是哪种情况，代理缓冲该包后转发给移动主机，并且标记该包已经被发送者重传，以便 ACK 处理算法根据该标记处理重复 ACK。

4.2 ACK 处理

Snoop 代理监控从移动主机来的 ACK，根据其类型采取不同的操作，这些 ACK 可分成四类：

4.2.1　新的、希望的 ACK　这是没有丢包时的正常情况，这类 ACK 清除缓冲在 Snoop 代理中对应的 TCP 数据包，并且更新无线链路的 RTT、然后转发 ACK 给固定主机。

4.2.2　重复 ACK　重复 ACK 表示某个包的后继包在该包之前被移动主机接收到，移动主机为每个没有按顺序到达的包生成一个重复的 ACK（DUPACK）。依赖 DUPACK 的类型和当前 Snoop 代理的状态，可以采取如下行为：

·首先，可能该 DUPACK 对应的包没在代理的缓冲中，或者是在代理的数据处理过程中已被标记为发送者重传的包。对于前者，需要固定主机重传该包，有可能引发发送者进行相应的拥挤控制；而对于后者，当发送者重传一个包时，它基于接收到的重复 ACK 数目来确定所应进入的状态，这两种情况都要求代理将 DUPACK 传回给固定主机，这对发送者正确地快速恢复尤其重要。

·第二种情况是 DUPACK 对应的包由于无线链路的原因而丢失，这并不是代理所希望的 DUPACK，移动主机每收到发送窗口中该包的后继包都生成一个 DUPACK，代理在收到第一个 DUPACK 后就以高优先级重传该包。代理根据丢包后、重传前传输的包的数目可以估计最多的 DUPACK 个数，也可据此估计下一个新的、希望的 ACK。

·第三种情况是"可预见的"DUPACK，这可以根据第二种情况的 DUPACK 的估计来确定。因为收到第一个 DUPACK 后就重传了对应的包，所以对"可预见的"DUPACK 不需再重传该包。

对后两种情况的 DUPACK，代理不传递给发送者，这样固定主机不会因为无线链路出错而导致发送者重传以及相应的拥挤控制。

4.2.3　新的、不是所希望的 ACK　如果一个窗口的数据有多个包丢失，在 DUPACK 的处理过程中，当所有的 DUPACK 到达后，代理跟踪第一个新 ACK，当其顺序号小于估计值时，则该 ACK 不是其所希望的 ACK。这是因为除 DUPACK 对应的包外，还有后继的包丢失，但这个窗口所有的 DUPACK 都是为第一个丢失的包生成的，因此代理清理缓冲，释放所有已确认接收的包，然后重传新 ACK 对应的包。

4.2.4　其它　我们前面提到以高优先级重传丢失的包，这样使得丢失的包比在基站队列中待发送的其它包更快地到达移动主机，减少了重复 ACK 的数目，并且减少了协议对精确重复 ACK 计数的敏感性。一般情况下，在高优先级重传之后，第一个新 ACK 会小于估计值，这样可以减少突发性，并且在多个丢包的情况下，后面的丢包也可能生成 DUPACK。

Snoop 代理除采用基于 ACK 的类型和数目重传（基于数据的重传）外，还可以基于定时器的超时重传。

5. 流量整形、标记

TCP 协议基于窗口的拥挤控制本质上有较大的突发性，一般可利用漏斗算法[5]和令牌桶[3]在网络边界对注入网络的 TCP 流量进行整形和标记。但漏斗算法不管突发通信量的大小，强迫输出模式保持一个固定的平均速率，然而，许多应用希望当大的突发通信量到来时，输出能相应加速，因此需要一个更有弹性的、更合适的决不会丢失数据的解决方案，即令牌桶算法。漏斗算法不允许空闲主机保留发送权，以备以后有大的突发通信量时使用，而令牌桶却允许保留发送权到桶的深度 b。两者的另一个重要区别是令牌桶算法在桶满时丢失令牌，但绝不丢失分组，而漏斗算法在桶满时丢失分组。

本文描述的机制可以不需要 RSVP 信令，只是在网络入口点（源主机或 Intranet 和 Internet 的接口处）根据预先协商的流量说明 Tspec 对连接进行整形和标记处理，然后在路由器根据包是否标记进行区分处理。Tspec 包括以下参数：r_m 说明该连接的平均速率，r_p 说明其峰值速率，b 是应用生成的最大突发块大小，此外还包括包的最大和最小长度。

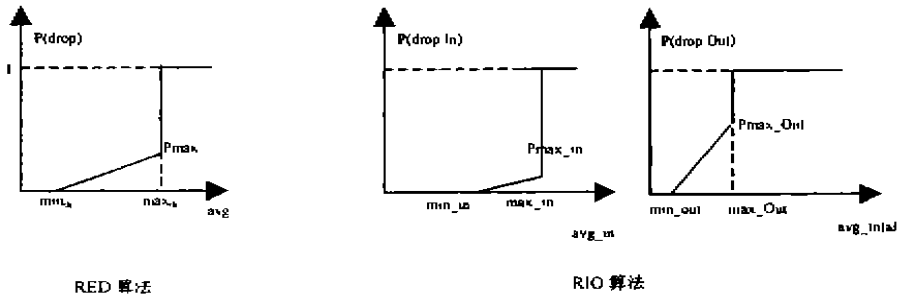
使用令牌桶在网络入口点整形流量，令牌生成进程根据源的 Tspec 生成令牌，平均生成速率为 t_m，短期峰值速率为 t_p，令牌桶的深度为 b，每当有包被注入网络时如果有相应数目的令牌则消耗这些令牌，这些包称之为顺从流（conformant traffic），并对这些包进行标记，否则归入非顺从流（nonconformant traffic），不标记这些包。

目前，为简单处理，我们只进行简单的二进制标记，即区分包为顺从的(In 包)和非顺从(Out 包)的，如果要将包划分成更多的级别，可采用 DIFFSERV[12]对 DSCP 的定义。

6. 支持优先级处理的 Snoop 代理

在所有的中间路由器以及 Snoop 代理，均识别两种不同类型的包，网络的瓶颈往往在基站，因此我们重点讨论 Snoop 代理顺从包和非顺从包的处理。

在所有的中间节点以及 Snoop 代理我们都采用 RIO[13]进行区分处理，RIO 的基本思想类似 RED，但对顺从包和非顺从包分别设置不同的丢包率和最大、最小队列长度限制，如图1所示。在 RED 算法中，用三个参数配置：min_{th}、max_{th}、p_{max}，其工作方式如图1所示，x 轴表示平均队列长度，y 轴表示丢包概率，min_{th}、max_{th}将 x 轴分成三个区间[0，min_{th})、[min_{th}，max_{th})、[max_{th}，∞)，分别对应正常操作、拥挤避免、拥挤控制三个阶段，在正常操作阶段，平均队列长度小于 min_{th}，丢包概率为0；在拥挤避免阶段，平均队列长度介于两个阈值之间，其丢包概率是 p_{max}的函数，队列管理通过丢包或采用明确的拥挤通知(ECN)来告诉发送方减慢发送速率，从而减少平均队列长度；当平均队列长度超过 max_{th}时，丢包概率或 ECN 标记概率为1。



RED 算法　　RIO 算法

图1　RED 算法与 RIO 算法比较

RIO 算法的机制类似 RED，但采用两套参数(min_in，max_in，p_{max_in})，(min_out，max_out，p_{max_out})，分别对应顺从的(In 包)和非顺从(Out 包)，其工作原理见图1。在 RIO 算法中，从三个方面保证 In 包的转发。首先，设置 min_out 小于 min_in，则 out 包比 in 包更早丢失；第二，设置 p_{max_out}大于 p_{max_in}，Out 包比 In 包更早地进入拥挤控制阶段。

Snoop 协议结合 RIO 算法，在保持 Snoop 协议原有特性的基础上，实现了不同优先级包的区分处理，从而可以支持带宽的预留，满足关键应用的需求，提高带宽的使用效率。

结束语　本文提出了改进的 Snoop 协议，保证在易出错、丢包的无线链路上能有效提高 TCP 传输性能的同时，结合令牌桶标记和 RIO 队列管理机制，能有效地支持最少带宽保证，从而可提高有最少带宽需求应用的高效运行。我们在 ns[14]进行了初步的仿真实验，验证了本文方法的可行性。

参 考 文 献

1 Shenker S, Partridge C, Guerin R. Specification of Guaranteed Quality of Service. RFC2212, Sept. 1997

2 Wroclawski J. Specification of the Controlled-load Network Element Service. RFC 2211, Sept. 1997

3 Sahu S, Nain P, Towsley D, Diot C, Firoiu V. On Achievable Service Differentiation with Bucket Marking for TCP: [Computer Science Tech-report-99-72]. Univ. of Massachusetts, 1999

4 Balakrishnan H. Challenges to Reliable Data Transport over Heterogeneous Wireless Networks: [PhD thesis]. University of California, Berkeley, 1998. Availble at: http://www.cs.berkeley.edu/hari

5 Siu K, Tzeng H. Intelligent congestion control for ABR service in ATM networks. Computer Communication Review, 1994, 24(5): 81～106

6 Bakre A, Badrinath B. Handoff and System Support for Indirect TCP/IP. In: Proc. Second Usenix Symp. On Mobile and Location-Independent Computing, April 1995

7 Balakrishnan H, Padmanabhan V N, Seshand S, Katz R H. A Comparison of Mechanisms for Improving TCP Performance over Wireless Links. IEEE/ACM Trans. On Networking, 1997, 5(6): 756～769.

8 Balakrishnan H, Seshan S, Katz R H. Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks. In: Proc. ACM Mobile Computing and Network Conf., ACM, 1995. 2～11

9 DeSimone A, Chuah M C, Yuye O C. Throught Performance of Transport-Layer Protocols over Wireless LANS. In: Proc. Globecom'93. Dec. 1993

10 Bakre A, Badrinath B R. Implementation and Performance Evaluation of Indirect-TCP. IEEE Trans. On Computer, 1997, 46(3)

11 Wireless Application Protocol. Available at: http://www.wapforum.com, 1998

12 Balke S, Black D, Carlson M, Davies E, Wang Z, Weiss W. An Architecture for Differentiated Services, RFC 2475, Dec. 1998

13 Clark D D, Fang W. Explicit allocation of best effort packet delivery service. IEEE/ACM Transaction on Networking, 1998, 6(4): 362～373

14 McCanne S, Floyd S UCB/LBNL/VINT Network Simulator, 1999. Available at: http://www-mash.cs.berkeley.edu/ns