

有线/无线混合网络传输控制策略

李红, 沈未名

(武汉大学遥感测绘信息工程国家重点实验室, 武汉 430079)

摘要: 针对有线/无线混合网络存在的数据包易丢失的问题, 提出一种基于探测包的传输控制策略。发送方周期性地向网络发送探测包, 接收方通过观测探测包到达的时间间隔评估网络状态。观测结果能较好地表征可用带宽、网络拥塞以及包丢失原因等信息, 发送方可根据观测结果调整发送速率以适应网络拥塞和丢包特性。仿真实验结果表明, 该策略是有效的。

关键词: 传输控制; 带宽估计; 丢包区分; 拥塞控制

Transmission Control Strategy for Wired/Wireless Mixed Networks

LI Hong, SHEN Wei-ming

(State Key Lab of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Hubei 430079)

【Abstract】 Aiming at the problem that data packet is easily lost in wired/wireless network, a Probe Packet-based Transmission Control(PPTC) strategy is presented. The sender periodically injects probe packets and the receiver estimates the network conditions. The inferences from the observations are utilized to regulate some information such as the transmission rate at the sender, the desirable properties of congestion control and loss packet discrimination. Simulation experimental results show this strategy is effective.

【Key words】 transmission control; bandwidth estimation; loss packet discrimination; congestion control

1 概述

在有线网络中, 由于误码率很低, 因此 TCP 协议假设所有的丢包都是由网络拥塞引起的。然而在有线/无线混合网络中, 这种假设就不成立了, 因为在混合网络中, 有线网络的拥塞和无线网络的误码都会导致包丢失, 这就需要一种传输策略来适应不同类型的丢包, 并同时保持 TCP 的友好性。不正确的丢包区分将导致网络性能的恶化, 如果拥塞丢包被错误地认为是误码丢包, 发送方将不会降低发送速率, 从而导致更严重的拥塞; 如果误码丢包被错误地认为是拥塞丢包, 则发送方将降低发送速率, 从而导致吞吐量的降低。

目前出现了许多区分包丢失的方法, 但这些方法需要改变网络结构或违背了端到端的原则, 牺牲了对 TCP 的友好性。

本文提出一种端到端的基于探测包的传输控制(Peer-to-Peer Packet-based Transmission Control, PPTC)策略, PPTC 不需要改变网络结构, 它在保持 TCP 友好性的同时能够很好地解决有线/无线混合网络中存在的诸如丢包区分等特殊问题。

2 相关工作

目前, 基于码率的传输控制协议得到广泛应用, 如 RAP 和 TFRC, 这是因为与基于 ACK 反馈消息的协议相比, 它们的传输速率更加平滑, 但由于剧烈的带宽变化, 这些协议还不能直接应用于无线网络中。另外一些方法, 如 WTCP^[1], 能够适应无线网络剧烈的带宽变化, 通过测量长时的抖动, WTCP 试图充分利用带宽, 然而由于无线信道的时变性, 长时的抖动并不能精确表征网络状态。在 PPTC 中, 部分地采用 WTCP 方法, 通过测量短时的抖动来监测网络拥塞。

PPTC 通过探测包来估计可用带宽, 这种方法早已被许多离线工具用于测量带宽, 如 bprobe, 但这些离线工具的缺点是所采用的滤波技术缺乏统计鲁棒性。有学者建议采用核

密度估计(Kernel Density Evaluation, KDE)方法对评估带宽的数据进行滤波。由于 KDE 方法计算复杂性太高, 因此 PPTC 采用一种优化的 KDE 算法进行在线实时滤波, 同时保持较高的精度。

有许多改进的 TCP 方法用来区分包丢失。文献[2]通过接收端数据包到达时间间隔来区分丢包。文献[3]通过往返时间与吞吐量的关系来区分丢包, 但这种启发式的方法被用在基于 ACK 反馈的协议中, 这是不合适的, 因为这种方法更适用于基于速率的传输协议。文献[4]采用延迟抖动率来区分丢包, 取得了较好的效果。PPTC 部分地借鉴了上述几种方法。

3 基于探测包的传输控制策略

PPTC 的主要目的是根据可用带宽和网络拥塞调整发送速率。为估计可用带宽, PPTC 发送端周期性地发送探测包, 接收端观测探测包的时间间隔, 从而获得网络的可用带宽。包的时间间隔(也称为抖动)能够表征网络状态, 当网络拥塞时, 抖动会增加, PPTC 由此降低发送速率, 由于速率降低具有前瞻性, 因此可以进行微调, 从而使发送速率更加平滑。由于拥塞丢失和误码丢失在平滑性上具有独特性, 因此可以通过这种特性对它们进行区分, 然后把区分结果反馈到拥塞控制机制, PPTC 能更好地适应网络带宽的变化。

3.1 带宽估计

PPTC 通过估计网络可用带宽来调整发送速率, 但由于网络带宽是个动态值, 它会随着网络流量模式的变化而变化, 因此测量可用带宽是困难的。通过探测包测量瓶颈带宽并结合短时的抖动历史数据, PPTC 可以决定网络拥塞趋势, 然

作者简介: 李红(1972-), 男, 博士研究生, 主研方向: 多媒体通信, 嵌入式图像处理; 沈未名, 教授、博士生导师

收稿日期: 2009-01-30 **E-mail:** lihong2047@163.com

后反过来进行传输速率的调整, 如果不仔细对此过程进行控制, 将不可避免地引入估计错误。本文对此进行详细描述, 并采用统计方法来解决导致不精确评估的潜在原因。

设 B_{bn} 为瓶颈带宽, S 为包大小, t_0^s 和 t_1^s 为 2 个探测包在发送端出发的时间, t_0^r 和 t_1^r 为 2 个探测包在接收端被接收到的时间, 则接收端的时间可用式(1)表示。

$$t_1^r - t_0^r = \max(S/B_{bn}, t_1^s - t_0^s) \quad (1)$$

从式(1)可计算瓶颈带宽。

$$B_{bn} = S / (t_1^r - t_0^r) \quad (2)$$

式(2)成立的条件是 2 个探测包只在瓶颈链路中进行排队, 但这种假设是很难满足的, 因为探测包会经过多个路由器, 每个路由器又承载了不同的数据流。由于网络的动态特性, 其他流可能会插入到 2 个探测包之间, 这就增加了探测包到达的时间间隔, 从而低估网络可用带宽。探测包也可能在后续的路由器中进行排队, 导致探测包到达的时间间隔减少, 从而会高估可用网络带宽。为采用探测包正确估计网络可用带宽, 需要用统计鲁棒的滤波算法来消除这些影响, 而 PPTC 就采用了优化的 KDE 算法。

KDE 算法的时间复杂性为 $O(n^2)$, 其中, n 为连续的抖动值采样点的数量。PPTC 需要在线实时执行 KDE 算法, 这就需要对 KDE 算法进行优化。PPTC 采用标准的线性差分法对 KDE 算法进行线性化, 虽然此线性化的 KDE 可能会引入估计误差, 但由于此误差不会累加, 且其幅度远远小于估计的带宽值, 因此可以忽略。

3.2 速率和拥塞控制

PPTC 每隔 4 个数据包发送一对探测包以估计网络可用带宽, 每次接收方估计一个新的带宽值, 就采用权重平均的方法来更新速率。PPTC 采用标准的指数加权滑动平均(Exponent Weight Moving Average, EWMA)算法来更新速率, 其算法为 $newrate = oldrate\alpha + newrate_est(1-\alpha)$ 。

采用探测包的方法估计带宽常导致估计的带宽高于实际带宽, 从而引起带宽的不公平共享并导致进一步拥塞。为此, PPTC 采用一种拥塞避免机制。由于抖动能很好地表征网络的拥塞趋势, 因此它通过跟踪抖动来合理地调整发送速率, 以维持网络带宽的公平共享。

数据包在网络中的传输时间 t_n 包括传播时间 pt_n 和路由器排队时间 q_n , q_n 又包括数据包等待处理的时间和处理时间。当网络负载增加时, 传输中的数据包在路由器中要经历更长的等待时间, 从而导致端到端更长的传输时间。当网络负载减小时, 端到端的传输时间将减小。

抖动可以通过 2 个连续包的传输时间之差进行测量。假设所有包的传播时间 pt_n 相同, 由于 $t_n = pt_n + q_n$, 则可以得到式(3)。

$$jitter_n = (t_n - t_{n-1}) \approx (q_n - q_{n-1}) \quad (3)$$

从式(3)可以看出, 抖动可以只用排队时间来衡量, 从而可以把抖动作为网络拥塞的指示。

PPTC 采用时间戳来决定发送时间。设 2 个连续的数据包 $n-1$ 和 n 的发送时间和接收时间分别为 t_{n-1}^s , t_n^s 以及 t_{n-1}^r , t_n^r , 则第 n 个包的抖动为 $jitter = (t_n^r - t_{n-1}^r) - (t_n^s - t_{n-1}^s)$ 。

抖动可以是负值、正值或零。由于正的抖动是网络拥塞的早期指示, 因此正的抖动将触发拥塞避免并促使 PPTC 在

源端降低发送速率, PPTC 根据前面 3 个正的抖动值的平均来降低发送速率。另外, PPTC 检测到拥塞丢包时会将其发送速率减半。

3.3 丢包区分

由于拥塞控制的性能依赖于数据包丢包区分的准确度, 因此 PPTC 接收端发现丢包后, 需要查看抖动测量的历史记录来区分丢包原因。正的抖动指示网络拥塞的增加, 如果拥塞缓冲区溢出, 包就会被丢弃, 从而使正在处理的和后续的包在缓冲区中靠得更近。因此, PPTC 接收端在拥塞丢包后会关注负的抖动, 负的抖动可能指示网络负载的减少(当 PPTC 发送端降低发送速率时)。

为区分拥塞丢包引起的负抖动和网络负载减少引起的负抖动, PPTC 分析丢包前后的抖动记录。如果丢包之后的负抖动紧跟在丢包之前的正抖动之后, 则表示发生了拥塞丢包。当然这种方法也会出现误判, 因为误码丢失也会引起 2 个包在传输过程中靠得更近。在这种情况下, PPTC 会把误码丢包认为是拥塞丢包, 从而降低发送速率, 然而由于诸如 IEEE802.11 这种 MAC 层协议对误码丢包实施了重传, 因此多次重传的时间消耗常常能够减少甚至消除负抖动, 从而减少了误判频率。

虽然上述方法有一定效果, 但并不能提供足够高的区分精度, 这是因为由拥塞丢包导致的可再使用的缓冲区可能被其他数据流占据, 从而使误码丢包看起来像拥塞丢包。这种误判的结果是严重的, 这会使得 PPTC 在拥塞丢包发生时也不会降低发送速率。

为此, 本文采用了保守的方法, 当分析到丢包是由误码引起时, PPTC 采用下面的方法对此做进一步判断: PPTC 按式(4)计算比率(其中, $ROTT$ 为单程时延), 然后通过设置一个比率阈值来决定网络状态, 网络的拥塞直接与此比率相关, PPTC 采用此比率进一步确定丢包原因。

$$(ROTT_{curr} - ROTT_{min}) / (ROTT_{max} - ROTT_{min}) \quad (4)$$

4 性能评估

通过 NS-2 网络仿真器对 PPTC 性能进行仿真评估。图 1 为仿真网络拓扑结构。

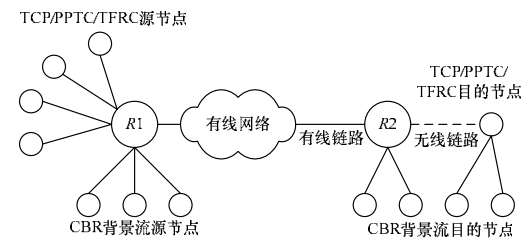


图 1 仿真网络拓扑结构

无线链路与路由器 R2 连接, 带宽为 2 Mb/s, 其 MAC 层协议采用 IEEE802.11。有线链路和其他连接到路由器 R1 的源端都具有 10 Mb/s 的带宽。无线网络的包丢失率在 0%~5% 之间变化。背景流为 CBR 或 TCP 流, 它们在仿真过程中随机加入和退出。每个仿真执行 200 s 并重复 100 次, 把 100 次仿真结果的平均作为最后的仿真结果。

4.1 KDE 算法的带宽估计效果

把 EWMA 算法中的 α 设为 0.8, KDE 算法中的 n 设为 8, 图 2 为仿真结果。从仿真结果可以看出, 与 EWMA 算法相比, KDE 算法能很好地适应带宽变化, 说明 KDE 带宽估计效果更好。

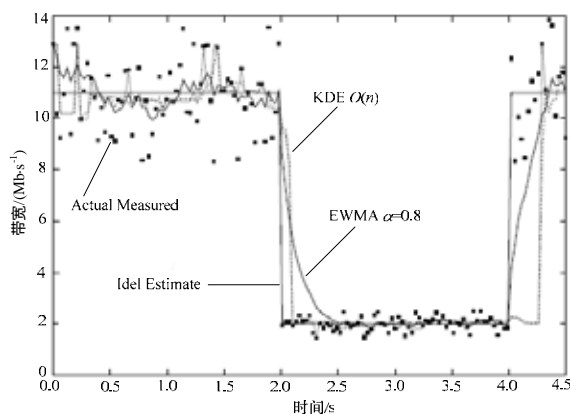


图2 各种算法的带宽估计效果比较

4.2 PPTC 的包丢失区分效果

把 PPTC 和 PPTC-ND(不采用包丢失区分)、PPTC-PD(理想的包丢失区分)、TCP Newreno、TFRC 和 TCP-Westwood 进行对比, 仿真结果如图 3 所示。

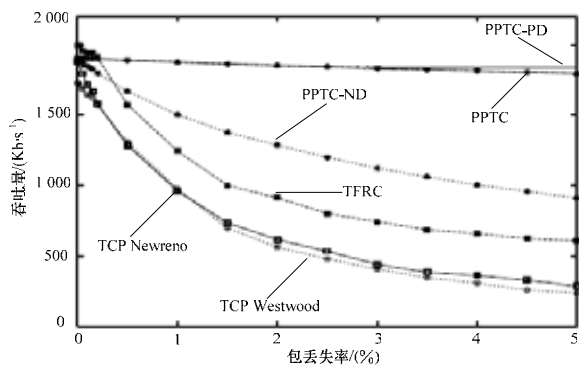


图3 各种传输控制策略包丢失区分效果比较

从仿真结果可以看出, 随着网络包丢失率的增加, PPTC-ND、TCP Newreno、TFRC 和 TCP-Westwood 的吞吐量迅速下降, 这是由于它们把所有的丢包都认为是拥塞丢包。PPTC 和 PPTC-PD 的曲线几乎重合, 性能非常接近, 说明 PPTC 有很好的包丢失区分能力。

4.3 PPTC 的带宽共享性能

图 4 为单个 PPTC 流与单个 TCP Newreno 流共享带宽时的吞吐量仿真比较, 其中, CBR 背景流占用了无线链路 1 Mb/s

带宽。从仿真结果可以看出, PPTC 的带宽利用率得到了很大提升并维持了 TCP 的友好性, 这是因为它能正确地对包丢失进行区分。

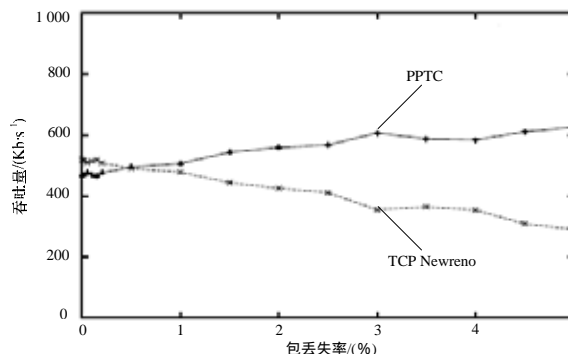


图4 PPTC 与其他流共享带宽的性能

5 结束语

本文针对有线/无线混合网络提出一种基于探测包的传输控制策略, 通过探测包到达接收端的时间间隔来区分丢包, 采用观测的时间间隔序列和有效的统计滤波来估计带宽, 根据估计的带宽和丢包区分结果来进行拥塞控制。仿真结果表明, 该策略具有较强的包丢失区分能力, 能够适应网络带宽的动态改变并维持 TCP 的友好性。下一步工作是在 Linux 操作系统下实现该策略, 以便了解其在真实有线/无线网络环境中的性能。

参考文献

- [1] Sinha P. WTCP: A Reliable Transport Protocol for Wireless Widearea Networks[J]. Wireless Networks, 2002, 8(2): 301-316.
- [2] Biaz S. Discriminating Congestion Losses from Wireless Losses Using Inter-arrival Times at the Receiver[C]//Proc. of IEEE Symp. on Application-specific Systems and Software Engineering and Technology. Richardson, USA: [s. n.], 1999.
- [3] Samaraweera N. Non-congestion Packet Loss Detection for TCP Error Recovery Using Wireless Links[C]//Proc. of the IEEE Int'l Conf. on Communications. Vancouver, Canada: [s. n.], 1999.
- [4] 李 淇, 陈 涤. 基于无线-有线混合网的 TCP 友好速率控制算法[J]. 计算机工程, 2007, 33(3): 85-87.

编辑 陈 文

(上接第 95 页)

索方法。目前该模型系统正处于积极的开发阶段, 取得了初步成果。

6 结束语

本文参考国外已经成熟的领域本体技术, 遵从 RAS 这一规约进行研究, 对资产进行组织、存储和文档化。结合领域本体的特点, 对资产的描述模型进行研究改进, 构建了资产本体库, 以支持语义上的检索, 设计实现了语义检索框架及检索算法, 以便企业完成后续资产的共享和复用。

本文提出的方法具有直观性、简明性以及实用性。实验证明该方法为企业软件资产管理提供了一个便捷、高效的管理平台, 具有一定的现实意义。

进一步的工作主要集中在研究本体结构较复杂的情况下, 采用剪枝技术及记录本体关联日志等方法, 对语义检索

算法做性能上的改进。

参考文献

- [1] OMG. Reusable Asset Specification Version 2.2[Z]. 2005.
- [2] William S, Austin T. Ontologies[J]. IEEE Intelligent Systems, 1999, 14(1): 18-19.
- [3] Smith M K, Welty C, Deborah L. OWL Web Ontology Language Guide[EB/OL]. (2004-02-10). <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>.
- [4] Stanford Center. The Protégé Ontology Editor and Knowledge Acquisition System[EB/OL]. (2007-08-15). <http://protege.stanford.edu/>.
- [5] 朱德利. SQL Server 2005 数据库管理与应用高手修炼指南[M]. 北京: 电子工业出版社, 2007: 371-398.

编辑 顾逸斐