

面向车联网按需驱动的多宿主多链路 TCP 拥塞控制算法

丁男¹, 林滔¹, 宋彩霞^{1,2}, 谭国真¹

(1. 大连理工大学计算机科学与技术学院, 辽宁 大连 116024; 2. 青岛农业大学理学与信息科学学院, 山东 青岛 266109)

摘 要: 针对车联网终端设备中网络应用 QoS 的多样化需求, 保证与安全应用相关的高优先级数据报文发送实时性, 提出了一种新的基于数据优先级与吞吐量评估的按需驱动的 MPTCP 拥塞控制算法 PTLIA。首先, 算法采用数据报文优先级以及吞吐量占比因子, 表征各个数据报文的发送权重; 其次, 利用分批估计理论模型对 MPTCP 中路径状态进行实时评估; 最后, 依据算法模型设计, 实现对网络拥塞时间窗的按需动态调整。实验与仿真验证了 PTLIA 算法在满足大部分网络应用需求的前提下, 提高了高优先级应用的网络传输实时性。

关键词: 车联网; MPTCP; 拥塞控制; 安全应用; 数据优先级; 吞吐量评估

中图分类号: TP302

文献标识码: A

Requirements-driven and multi-homed-based multipath TCP congestion control algorithm for vehicular network

DING Nan¹, LIN Tao¹, SONG Cai-xia^{1,2}, TAN Guo-zhen¹

(1. Institute of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China;

2. College of Science and Information, Qingdao Agricultural University, Qingdao 266109, China)

Abstract: To meet the diversified QoS applications demanded by terminal units of a VANET, especially to ensure the timeliness and reliability when sending the high-priority messages that were related to safety-critical applications, a demand-driven MPTCP congestion control algorithm: PTLIA was proposed, which was based on message priority and throughput estimation. First, message priority and throughput proportion factor were defined to characterize the weight of each message being sent. Second, the patch estimation model was used to make real-time estimation of the state of each MPTCP path. Finally, the algorithm of PTLIA was accordingly designed to adjust, dynamically, the window of the congestion time on demand. The algorithm of PTLIA, as proved in proposed simulations and experiments, has shortened the transmission of the high-priority data, under the premise of MPTCP transmission principles.

Key words: vehicular ad hoc network, MPTCP, congestion control, safety-critical applications, message priority, throughput estimation

1 引言

车联网利用无线通信技术实现车车通信、车路通信、车与服务中心通信, 将人—车—路—环境有机地结合起来, 提高了交通安全性和通行效率^[1]。如何保障车联网进行高效稳定地通信已成为车联网中的关键问题。多宿主(multi-homed)通信技术,

利用在系统中集成多个或多类物理通信介质, 可以提高系统通信的稳定性与可靠性^[2]。随着多宿主通信技术在终端设备上的广泛应用, 如何利用其多物理通信介质并发传输、动态切换, 有效利用网络信道容量, 成为其应用的关键问题。

2013 年 1 月, IETF(Internet Engineering Task Force)工作组制定了 MPTCP (multipath transport

收稿日期: 2016-01-26; 修回日期: 2016-05-10

通信作者: 丁男, dingnan@dlut.edu.cn

基金项目: 国家自然科学基金资助项目 (No.61471084); 中央高校基本科研业务费专项基金资助项目 (No.DUT15QY02)

Foundation Items: The National Natural Science Foundation of China (No.61471084), The Fundamental Research Funds for the Central Universities (No.DUT15QY02)

control protocol)协议框架^[3]。MPTCP 协议可以有效地利用多宿主系统中的网络容量,提高网络传输性能。MPTCP 协议通过对传统 TCP 协议的扩展,利用多 TCP 链路并发传输机制实现了在 TCP 协议传输过程中路径并发建立、动态选优、自动切换等功能,解决了传统 TCP 连接在多通信介质切换过程中的路径建立延时以及拥塞阶段无法动态切换路径的问题。

车联网应用是多样化的,其通信需求也是多元化、海量化并存的。不同类型应用的信息重要性不同,导致其传输优先级需求的差异性较大。如系统广播信息、安全应用信息、交通诱导类应用信息和娱乐应用信息等。然而,现有的大部分 MPTCP 拥塞控制算法的主要考虑因素是传输效率以及对传统单路径 TCP 用户的公平性问题^[4],并未结合网络应用中数据的不同通信需求、不同优先级特性对网络资源进行分配管理。结合车联网的应用需求,设计有效的 MPTCP 拥塞控制算法,是提高 MPTCP 协议在车载终端设备中有效利用多物理通信介质的关键。

针对上述问题,本文提出了一种基于应用优先级与吞吐量评估的 MPTCP 拥塞控制算法。本文不仅考虑了传输效率与公平性因素,还考虑了车联网中高优先级应用的传输需求,按需对不同应用的网络拥塞窗口值动态调整,进一步保证了车联网通信的安全性,并通过吞吐量评估的方式提高了网络传输效率,从而增强了车联网通信的实时性。

2 相关工作

MPTCP 对 TCP 进行了支持多路径的扩展,旨在通过同时使用多条路径提高网络的吞吐量和顽健性。由于 MPTCP 具有的高可靠性、高吞吐量特性,MPTCP 将成为车联网多样化通信的有效方法之一。文献[5]介绍了一种专门用于车联网中多媒体应用的 MPTCP 传输协议,文献[6]使用 MPTCP 协议提高车联网中的文件传输效率。然而由于 MPTCP 多路径的特性,也使 TCP 中的传统拥塞控制算法不适合直接应用于 MPTCP 协议中,故早期 IETF 工作组制定 MPTCP 规范时提出了以下 3 个设计目标^[3]。

1) 提高吞吐量: MPTCP 流的整体吞吐量不应低于仅使用任意一条单路径时所获得的吞吐量。

2) 不产生损害: 当 MPTCP 与 TCP 同时使用某网络共享资源时, MPTCP 用户不能比 TCP 用户

获得更多的网络资源。

3) 拥塞平衡: 若某条 MPTCP 链路发生拥塞,则应尽量将数据转移到其他路径传输,以均衡各子流间的拥塞程度。

针对上诉目标现已有若干拥塞控制算法被提出。在文献[7]中,EWTCP(equal weighted TCP)算法对每个子流设置了路径数的平方分之一作为权重因子,用来平衡 MPTCP 对 TCP 的侵略性。之后 Costin 等^[8]提出 LIA (linked increase)算法, LIA 算法将路径的拥塞窗口值与延迟响应时间的平方的比值作为衡量路径状态的标准,并将当前路径比值与所有的路径比值之和的比例作为拥塞窗口值增量。Khalili 等^[9]通过实验证明了 LIA 算法不能完全实现拥塞平衡,并在 LIA 算法的基础上,进行了部分修正,提出了 OLIA 算法(opportunistic linked increases algorithm)。该算法与 LIA 算法相比,新增了一项与最佳路径相关的机会增长因子用来对 LIA 算法的路径选择进行一定的修正,保证了整体链路的负载均衡,以及 Linux 系统中 MPTCP 默认的拥塞控制算法 Coupled^[10],该算法将多个子流的拥塞窗口联合起来,共同决定某一子流的拥塞窗口值的增长。除以上提及的 MPTCP 拥塞控制算法,还有一些其他的相关文献建议去处理 MPTCP 中的最优化路由^[11]和数据中心网络^[12]等相关拥塞问题。

虽然现有的 MPTCP 拥塞控制算法已经较为完善,但并未考虑车联网等系统中安全类应用具有的优先级传输需求。已有的考虑优先级拥塞控制的文献也几乎局限于无线传感网方面。如最早的应用于无线传感网中的基于优先级的拥塞控制算法 PCCP^[13] (priority based congestion control protocol),文献[14~16]均是对 PCCP 算法在某一方面的改进,而文献[17]则介绍了一种在 LTE(long term evolution)移动网络中基于优先级的拥塞控制算法。

3 吞吐量评估优先级调整模型

3.1 MPTCP 链路条件分析

如图 1 所示,在使用 TCP 传输数据时,通常包含 3 个阶段^[18]。

1) 开始阶段(0-knee): 网络上的负载较小时,增加负载,网络上的吞吐量将会近似线性增长。

2) 积聚阶段(knee-cliff): 网络上的负载持续增加,到达网络的容量上限之后,吞吐量的增长速率将逐渐变缓。

3) 过载阶段(cliff-∞): 继续增加负载, 以至于网络严重拥塞, 导致分组丢失的发生。此时, 网络的响应时间显著增加, 网络趋于瘫痪, 吞吐量急剧下降。

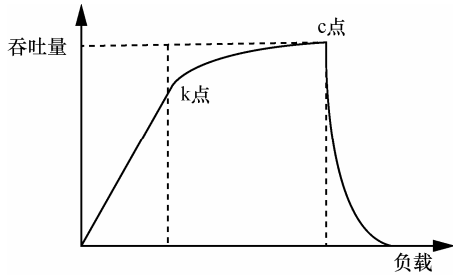


图 1 负载—吞吐量关系

由以上的分析可知, 链路的吞吐量是随链路负载不断变化的, 因此, 本文设想通过分析吞吐量的增长比例去预测 MPTCP 的链路条件。基于文献[19]提出的 TCP 链路吞吐量模型, t 时刻的吞吐量 $x_r(t)$ 可由以下公式获得

$$x_r(t) = \frac{1}{R_r(t)} \sqrt{\frac{2}{p_r(t)}} \quad (1)$$

其中, $R_r(t)$ 表示 t 时刻在路径 r 上的延迟响应时间 RTT(round-trip time), $p_r(t)$ 表示 t 时刻在路径 r 上的分组丢失率。 $r \in M_u$, M_u 是任意应用 u 的所有可用路径集合。 $p_r(t)$ 的计算方式采用了文献[19]中的估计方法 $p_r(t) = \frac{1}{l_r(t)+1}$, 其中, $l_r(t)$ 是应用 u 在上 2 次分组丢失之间成功传递数据报文的数量。根据式(1)对 $x_r(t)$ 求导可得

$$\frac{d(x_r(t))}{dt} = -\frac{\sqrt{2}}{2} \left(\frac{2}{R_r(t)^2 \sqrt{p_r(t)}} \frac{d(R_r(t))}{dt} + \frac{1}{R_r(t) p_r(t) \sqrt{p_r(t)}} \frac{d(p_r(t))}{dt} \right) \quad (2)$$

分析式(2)可知, 任何时刻路径 r 上的吞吐量具有如下 $x_r(t)$ 属性。

1) 如果 t 时刻路径 r 上延迟响应时间 $R_r(t)$ 和分组丢失率 $p_r(t)$ 同时增大, 则 $x_r(t)$ 将减小。

2) 如果 t 时刻路径 r 上延迟响应时间 $R_r(t)$ 和分组丢失率 $p_r(t)$ 同时减小, 则 $x_r(t)$ 将增大。

因此, 为表示 MPTCP 链路吞吐量的变化趋势, 本文定义了集合 C_r 。

$$C_r = \{t | t \in \arg(2p_r(t)\Delta R_r(t) + R_r(t)\Delta p_r(t) > 0)\} \quad (3)$$

C_r 由式(2)推导获得, 其是 t 时刻吞吐量逐渐降低的路径 r 的集合。为方便计算, $\Delta R_r(t)$ 与 $\Delta p_r(t)$ 分别使用 $R_r(t)-R_r(t-1)$ 与 $p_r(t)-p_r(t-1)$ 的简化表示。

3.2 吞吐量预测

若使用基于拥塞窗口调整的拥塞控制机制去尽量避免拥塞和提高 MPTCP 的性能, 则应研究吞吐量的变化趋势。如图 1 所示, 在开始阶段, 拥塞窗口值应该与吞吐量一同急剧增长。而当进入积聚阶段时, 则应该逐渐降低拥塞窗口值的增长速度以减缓吞吐量的增长速度。因此, 为选择合适的拥塞窗口增长策略, 需要去计算和预测链路的吞吐量。虽然吞吐量的长远变化趋势受链路负载的影响, 但仍然可以使用分组丢失率和 RTT 值预测出短期内吞吐量的变化趋势。文献[20]中推荐了几种吞吐量评估策略。

本文使用了分批估计算法^[21]进行吞吐量预测, 其在一维非线性系统中具有良好的顽健性。该算法的工作过程由 2 步组成: 第 1 步为自适应, 分批估计算法通过测量和计算第 t 时刻和 $t-1$ 时刻数据计算出系数 α ; 第 2 步为预测, 该算法使用第 1 步中的数据估计第 $t+1$ 时刻的可能取值。使用分批估计算法获得的估计值表示吞吐量的预测值, 可以预测出未来传输数据流的吞吐量。 $t+1$ 时刻的吞吐量预测值 $\hat{x}(t+1)$ 可由以下公式得出

$$\hat{x}(t+1) = \hat{\alpha}(t)\hat{x}(t) + (1-\hat{\alpha}(t)) \frac{x_r(t) + x_r(t-1)}{2} \quad (4)$$

其中, $\hat{x}(t+1)$ 为 $t+1$ 时刻 x 的估计值, $\hat{\alpha}(t)$ 表示对 t 时刻 α 的估计值, α 的估计值可用式(5)求得

$$\hat{\alpha}(t) = \frac{2\alpha_r(t)^2}{\alpha_r(t)^2 + \alpha_r(t-1)^2} \quad (5)$$

其中,

$$\alpha_r(t) = \sqrt{|x_r(t)^2 - \hat{x}_r(t)^2|}$$

$$\alpha_r(t-1) = \sqrt{|x_r(t-1)^2 - \hat{x}_r(t-1)^2|}$$

因为该算法的递归特性, 故不需要其他额外信息, 仅使用第 t 时刻与第 $t-1$ 时刻吞吐量的测量值与估计值就可以进行实时的预测。该预测方式计算简单, 且该算法的顽健性也能尽可能保证预测值的准确性。

3.3 动态优先级

针对车联网中面向不同应用的数据报文具有不同重要程度的特性, 本文从系统一致性角度, 参

考数据报文所属应用优先级，以及报文所在链路状态定义了数据报文的动态优先级，给出了数据报文的动态优先级以及划分模型，用以反映不同优先级或相同优先级任务产生的报文的实时重要性。

定义 1 数据报文的动态优先级用以表征当前数据报文待发送的实时需求程度。其描述如下

$$TP = p + \left(\frac{R_r}{R_{\max}} \right) \left(\frac{data}{data + \alpha} \right) \quad (6)$$

其中， p 为所属应用的优先级，根据操作系统任务调度的一般性， p 值越小，其优先级越高。 r 为应用选择的传输路径， R_r 为其上次传输时接收端的延迟响应时间， R_{\max} 是最大延迟响应容忍时间常数，超出这个时间则认为报文丢失，应用当前任务总的数据传输量为 $data$ ， α 为比例控制因子。根据 IEEE TCP 协议^[4]定义，单个数据报文最多可以携带的数据量为 1 500 byte，故本文选择 $\alpha = 1\ 500$ 作为比例控制因子，即认为若某应用传输数据量低于单个数据报文传输的最大数据量，则该应用的传输数据量相对较小。

根据 TP 定义，某数据报文所属的任务优先级与从属的任务优先级、数据报文长度以及 RTT 成正比，即从属任务优先级越高，所在通信链路状态越优，总发送数据报文越小，该数据报文实时发送需求越急迫。动态优先级考虑了 RTT 与传输数据量大小对应用优先级的影响。RTT 较小表示当前路径状态较好，可以适当增大拥塞窗口。一般情况下，某次传输数据的数据量较小，则表示该数据具有更高的即时性与有效性，如一般情况下信息、文档等文件较音乐、视频等多媒体文件有更高的重要性，但理论上应该保持数据优先级对优先级影响力的主导作用，对于本文讨论的情况，即应该保证调整之后的动态优先级不会跨越下一应用优先级。

因此，本文采用了将延迟相关因子与数据量相关因子相乘的方式。由公式简单计算可知 $\left(\frac{R_r}{R_{\max}} \right) \left(\frac{data}{data + 1\ 500} \right)$ 取值范围为 (0,1)，则动态优先级不会发生跨越应用优先级的情况。动态优先级定义既考虑了链路情况与传输数据量对优先级的影响，以区别相同优先级应用的优先程度，同时也保证了应用优先级对动态优先级的支配地位。

针对不同优先级应用的数据传输需求的重要性不同，故首先应该对动态优先级进行划分。若系统中动态优先级的取值范围为从 M 到 N 依次递增

的数，则先取初始中间值 $MID = \frac{M + N}{2}$ ，为尽可能减少高优先级应用，故将初始中间值下取整，即 $MID = \left\lfloor \frac{M + N}{2} \right\rfloor$ 。以 Linux 中 nice 值优先级为例，

应用的优先级范围为 -20 到 20，其中小于 0 的优先级为特权优先级，则在 Linux 系统中动态优先级 $M = -20$ 、 $N = 21$ 、 $MID = 0$ 。故对于 Linux 系统中动态优先级小于 0 的应用，本文认为可以适当加快其数据传输的速度。

3.4 吞吐量占比因子

在车联网系统中高优先级应用自身传输的数据量一般较少，故通常不会引起长时间的链路拥塞。通常只有当低优先级应用大量传输多媒体数据时才会引起链路的长时间拥塞。为保证高优先级应用数据的快速传输，则此时应尽量保证高优先级应用的链路吞吐量占比以避免高优先级应用数据分组的丢失。本文定义了吞吐量占比因子，以避免因低优先级应用大量发送数据造成链路拥塞，从而导致高优先级应用频繁分组丢失。吞吐量占比因子能保证当高优先级应用分组丢失时，若其对当前链路的吞吐量占用比例低于吞吐量占比因子时，则其发生分组丢失将不会使自身拥塞窗口值发生衰减。吞吐量占比因子具体定义如下。

定义 2 吞吐量占比因子用于表示某应用所需占用网络资源的权重。其表达式如下

$$\beta = \left(\frac{TP - MID}{\left\lfloor \frac{N - M}{2} \right\rfloor} \right) \quad (7)$$

若某应用的动态优先级为 TP ，该系统的优先级数值范围为 M 到 N ， $M < N$ ， MID 为该系统的优先级数值的中间值。由定义可知，当传输应用的动态优先级较低，即 $TP \geq MID$ 时，其吞吐量占比因子 $\beta \in [-1, 0]$ ，对低优先级应用不具有保证吞吐量占用比例的作用。而当传输应用的动态优先级较高，即 $TP < MID$ 时，其吞吐量占比因子 $\beta \in [0, 1]$ ，可通过其适当保证高优先级应用在当前链路的最低吞吐量占用比例。

4 PTLIA 算法

本部分将具体介绍基于优先级和吞吐量预测的链接增长拥塞控制算法 (PTLIA)，PTLIA 算法基于拥塞窗口大小调整机制。本节还分析了 PTLIA 算法满足

IETF 设计目标的各项属性，并且证明了其收敛性。

4.1 PTLIA 算法设计

在 PTLIA 算法中，本文使用了吞吐量预测的方式去调整拥塞窗口值的增长速度，以适应不同的传输阶段。具体的 PTLIA 算法如下。

1) 每当路径 r 接收到一个 ACK 时，则下一时刻的拥塞窗口值增量为

$$\Delta w_r(t+1) = \lambda_r(t) \frac{w_r(t)}{R_r(t)^2} + \frac{\alpha_r(t)}{w_r(t)} \quad (8)$$

$$\left(\sum_{p \in M_u} \frac{w_p(t)}{R_p(t)} \right)^2$$

其中， M_u 是用户 u 的所有可用路径集合，当 $r \in C_r$ 时， $\lambda_r(t) = \frac{x_r(t)}{\hat{x}_r(t+1)}$ ， $r \notin C_r$ 时 $\lambda_r(t) = 1$ ，当

$$\sum_{p \in M_u} \Delta x_p(t) \neq 0 \text{ 且 } \Delta x_r(t) > 0 \text{ 时, } \alpha_r(t) = \frac{\Delta x_r(t)}{\sum_{p \in M_u} \Delta x_p(t)},$$

$$\sum_{p \in M_u} \Delta x_p(t) \leq 0 \text{ 或 } \Delta x_r(t) \leq 0 \text{ 时 } \alpha_r(t) = 0. \text{ 若 } r \in C_r,$$

则表示当前路径吞吐量增长比率下降，网络可能发生拥塞，此时因 $\hat{x}_r(t+1) > x_r(t)$ ， λ_r 将会小于 1，可以减缓拥塞窗口值的增长速度，避免过度拥塞，长时间维持链路的高吞吐量。

在式(8)中， $\frac{\alpha_r(t)}{w_r(t)}$ 是用于当吞吐量发生波动和

拥塞时，PTLIA 算法进行拥塞平衡调整。 $\Delta x_r(t)$ 由 $x_r(t) - x_r(t-1)$ 计算得出，通过参数 α_r ，可以使当链路条件较好，即 Δx_r 增长较快时， w_r 的增长也较快，从而使链路条件越好的路径其应用的拥塞窗口值增长越快。相反，当链路条件较差 Δx_r 增长缓慢时， w_r 的增长也逐渐变缓。通过以上分析可知，PTLIA 算法通过参数 λ_r 和 α_r 能在链路变得过度拥塞之前有效地降低拥塞窗口值的增长速度，更有效地维持链路的最大吞吐量。

2) 每当路径 r 发生一次分组丢失，并且

$\beta_u < \frac{x_r}{x_{r\max}}$ 时，下一时刻的拥塞窗口值为

$$w_r(t+1) = w_r(t) - \frac{w_r(t)}{2 + \frac{\beta_u(t)}{2}} \quad (9)$$

其中， β_u 是应用 u 对应的吞吐量占比因子， x_r 为当前吞吐量， $x_{r\max}$ 是应用 u 在路径 r 上曾出过的最大吞吐量数值。

因不同进程的吞吐量占比因子 β_u 不同，故式(9)能使当因链路拥塞而发生分组丢失时，高优先级应用的拥塞窗口值衰减更小，低优先级应用的拥塞窗口值衰减更大，从而在长时间传输过程中逐渐提升高优先级应用在链路中的吞吐量占用比例。同时，通过在分组丢失时进行条件判断，若高优先级应用在吞吐量占用比例过低时发生分组丢失，则不会发生拥塞窗口值衰减，进一步保证了高优先级应用在链路中的吞吐量占用比例。

PTLIA 算法仅在发生分组丢失，即在链路可能出现拥塞时，才对不同优先级应用的拥塞窗口值给予不同的调整策略。因此，PTLIA 算法不会影响不同应用在拥塞窗口值增长上的公平性，同时通过保证吞吐量占比提高了高优先级进程的传输速度。

4.2 PTLIA 算法属性分析

根据 IETF 制定的标准，MPTCP 拥塞控制算法应该尽可能满足本文第 2 节提及的 3 个目标。本文将描述 PTLIA 的相关属性，并且证明其满足以上的 3 个目标。

定理 1 PTLIA 算法在任意应用 u 使用的任意路径 r 上具有以下属性。

- 1) PTLIA 算法满足 MPTCP 设计目标 1：提高吞吐量。
- 2) PTLIA 算法满足 MPTCP 设计目标 2：不产生损害。
- 3) PTLIA 算法满足 MPTCP 设计目标 3：拥塞平衡。
- 4) 对于任意时刻 t 与 $t+1$ ，若 K 与 T 是 2 次连续分组丢失发生的时间，且满足 $K < t < t+1 \leq T$ ，则 $\frac{\Delta w_r(t+1)}{\Delta w_r(t)} < 1$ 。即在任意 2 次连续分组丢失的时间段内，拥塞窗口值的增长数值逐渐变小。

证明 以上属性的具体证明请详见附录 1。

4.3 PTLIA 算法公平性与代价

4.3.1 PTLIA 算法公平性分析

PTLIA 算法的公平性问题即为对 TCP 用户的友好性，具有良好公平性的算法应尽可能减少对 TCP 用户路径的侵占。有关 PTLIA 算法公平性的理论分析，即为 4.2 节中的属性 2)，不产生损害。

4.3.2 PTLIA 算法代价分析

本文通过吞吐量预测的方式，更准确地获取了路径的拥塞情况，从而进一步提升了系统的传输效率，但吞吐量预测也增加了算法运行时的计算量。

不过该计算量与目前计算机的性能相比极其微小，所以，其对传输性能的影响也几乎可以忽略。

另一方面，本文提出的算法主要考虑的是高优先级任务的传输需求，故在降低高优先级任务的传输时间的同时将会增大低优先级任务的传输时间。同时由于算法分组丢失策略的限制，当系统中的传输进程全是极高优先级进程，或全是极低优先级进程时，系统的传输效率也将受到较大影响，但该问题可通过应用设置避免。

5 实验仿真与性能分析

本文实验通过 NS3 构建多宿主通信环境，并针对车联网中大数据量、小数据量等不同的网络传输需求对 PTLIA 算法性能进行验证，并与 LIA、OLIA 算法进行了性能对比分析。

5.1 仿真环境与设置

本部分的优先级对比实验在仿真模型中的 A、B、C 3 个节点上建立相关应用进行数据传输。其中，A、C 为发送节点，B 为接收节点，节点 A 使用 MPTCP，节点 C 使用 TCP。在相同的网络环境下，当所有应用的数据全部传输完成时，实验停止。其仿真实验模型如图 2 所示。

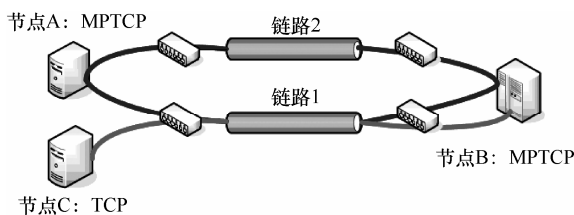


图 2 仿真模型

5.2 优先级对比实验仿真结果分析

针对不同场景的应用需求，本实验环节分为 2 部分：第 1 部分实验应用的传输数据量为 5 MB；第 2 部分实验应用的传输数据量为 50 MB。本节将分别对比传输数据量为 5 MB 和 50 MB 时，PTLIA 算法中不同优先级应用的传输结束时间和拥塞窗口值变化情况。实验参数如表 1 所示，实验结果如图 3 所示。

图 3 为各应用传输数据量为 5 MB 和 50 MB 时的拥塞窗口值变化。由图 3 可以看出不论是传输数据量较低（5 MB），还是传输数据量较高（50 MB）时，应用 1 在其传输期间相比其他应用能维持较高的拥塞窗口值，且分组丢失策略的保护使应用 1 近乎同时使用路径 1 与路径 2 进行传输，从而使高优

优先级应用 1 的数据提前传输完成。

表 1 实验 1 仿真参数

参数	值
链路 1 带宽/(Mbit·s ⁻¹)	15
链路 1 延迟/ms	40
链路 2 带宽/(Mbit·s ⁻¹)	10
链路 2 延迟/ms	60
P1(应用 1 优先级)	-10
P2(应用 2 优先级)	0
P3(应用 3 优先级)	5
P4(应用 4 优先级)	10

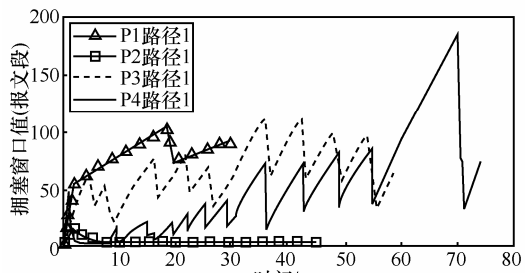
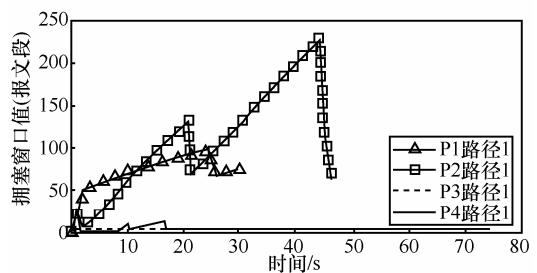
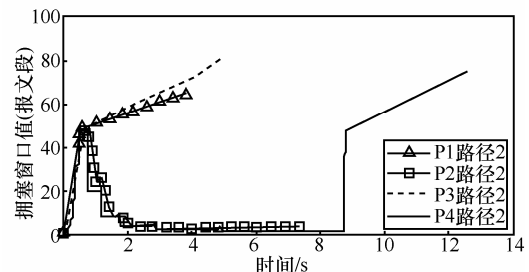
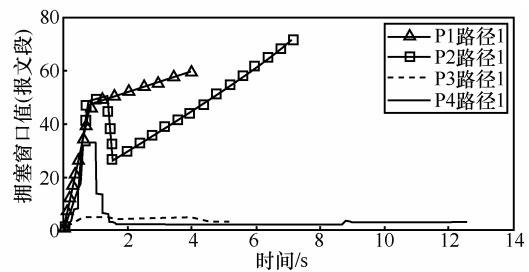


图 3 各应用拥塞窗口值

图 4 为传输数据量分别为 10 MB、20 MB、30 MB、40 MB 和 50 MB 时，重复进行 20 次传输实验后统计获得的各应用的平均结束时间。由图中可以看出，当传输数据量较低时，各应用的传输时间差距较小，当传输数据量逐渐增大时，应用 1 的传输时间明显低于其他应用的传输时间。由实验结果可知，PTLIA 算法通过在发生分组丢失时保证高优先级应用的吞吐量和对应用的拥塞窗口丢失值进行调整的方式，保证高优先级应用优先传输的目的。

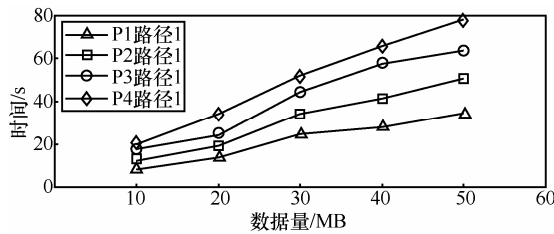


图 4 各应用平均结束时间

5.3 各算法性能对比与分析

5.3.1 性能对比实验仿真参数

本节针对 LIA、OLIA 和 PTLIA 这 3 种算法分别在 NS3 中进行仿真。实验 2 在图 2 仿真模型中的 A、B 节点上建立了应用 5，应用 5 分别使用不同的拥塞控制算法传输数据，传输数据量均为 10 MB。为排除 PTLIA 算法中优先级控制对 PTLIA 算法性能的影响，实验 2 将应用 5 的优先级设置为 0，则分组丢失处理时 PTLIA 算法将与其他算法相同。表 2 为实验 2 的各项仿真参数。

表 2 实验 2 仿真参数

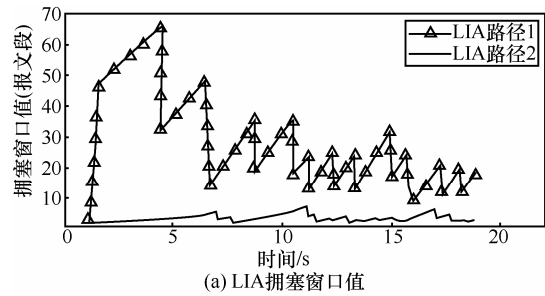
参数	值
链路 1 带宽/(Mbit·s ⁻¹)	5
链路 1 延迟/ms	40
链路 2 带宽/(Mbit·s ⁻¹)	3
链路 2 延迟/ms	60
P5(应用 5 优先级)	0

5.3.2 拥塞窗口表现分析与比较

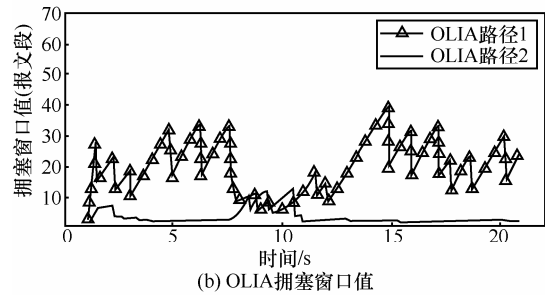
图 4 中给出了 3 种算法在路径 1 和路径 2 上的拥塞窗口值变化情况。

由图 5 可以看出，因 LIA 算法不具有拥塞平衡的作用，故 LIA 算法在整个传输过程中一直在同时使用链路条件更差的路径 2，而 OLIA 算法与 PTLIA 算法仅仅在路径过度拥塞时使用路径 2 传输。对比 OLIA 与 PTLIA 算法还可以看出，OLIA 算法在拥

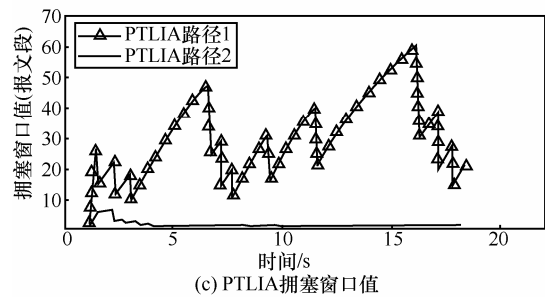
塞避免阶段的拥塞窗口值近乎为线性增长，而 PTLIA 算法的拥塞窗口值则因吞吐量预测而呈现出曲线趋势，能持续地维持拥塞窗口值平稳增长和降低分组丢失率，故 PTLIA 算法将能更有效地避免如 OLIA 算法在 8~12 s 时间段出现的过度拥塞的情况。表 3 给出了各个算法的分组丢失次数。



(a) LIA 拥塞窗口值



(b) OLIA 拥塞窗口值



(c) PTLIA 拥塞窗口值

图 5 各拥塞控制算法的拥塞窗口值

表 3 分组丢失次数

算法	分组丢失次数
LIA	21
OLIA	23
PTLIA	13

实验在算法传输过程中统计了各个算法所有路径上发生的分组丢失次数总和，若某次分组丢失导致该路径拥塞窗口值降低，则统计为一次有效分组丢失次数。PTLIA 算法通过吞吐量预测的方式更准确地对路径状态进行评估，保持路径进行长时间的稳定传输，从而降低了分组丢失次数。

5.3.3 吞吐量表现分析与比较

图 6 中给出了 3 种算法在路径 1 和路径 2 上的

吞吐量变化情况。

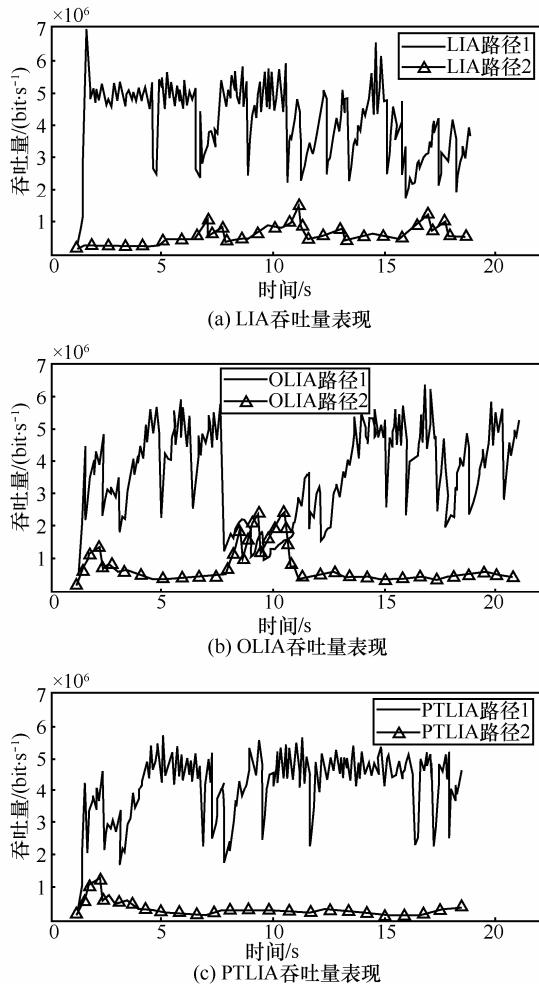


图 6 各拥塞控制算法的吞吐量表现

对比图 6 中的各个算法可以看出，PTLIA 算法的吞吐量在整个数据传输过程中表现更加平稳，其最大吞吐量值几乎维持在同一水平线。而 LIA 算法与 OLIA 算法的吞吐量均在传输过程中具有很大波动，OLIA 算法甚至出现了因链路过度拥塞而导致吞吐量急剧下降的情况。对比分析可知，PTLIA 算法通过吞吐量预测的方式相比 LIA 与 OLIA 算法，能更好地控制链路的拥塞情况，维持链路的最大吞吐量。表 4 为各个算法的传输结束时间，证明了 PTLIA 算法通过吞吐量预测进一步地节约了传输时间。

表 4 传输完成时间

算法	完成时间/s
LIA	18.825 5
OLIA	20.853 7
PTLIA	18.359 6

5.3.4 拥塞窗口增量分析与比较

图 7 给出了 3 种算法在路径 1 上的拥塞窗口增量变化情况。

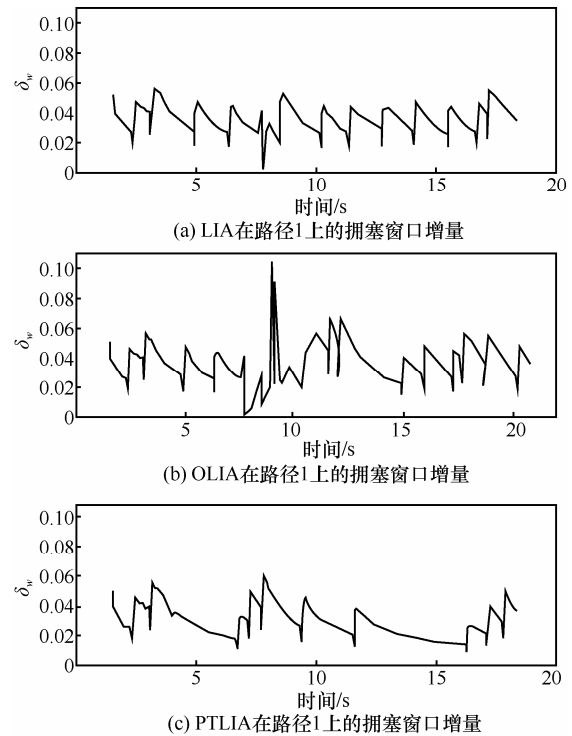


图 7 各拥塞控制算法在路径 1 上的拥塞窗口增量

因路径 2 条件较差，在传输过程中拥塞窗口值几乎没有变化，图 7 仅为 LIA、OLIA 和 PTLIA 算法在路径 1 上的拥塞窗口增量值的变化。为更加直观地表现拥塞避免阶段算法对拥塞窗口值的调整作用，本部分移除了慢开始阶段的拥塞窗口增量值。图中竖线表示传输过程中发生一次分组丢失，因此，从图 7 中可以清晰地看出 PTLIA 算法在 2 次分组丢失之间的拥塞窗口增量值逐渐减小，而 LIA 与 OLIA 算法并不完全符合该特征，即图 7 验证了 PTLIA 算法满足理论 1 中的属性 4。

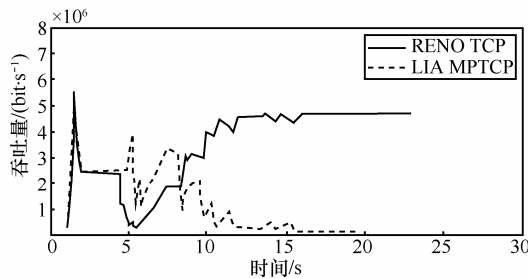
5.4 公平性分析与比较

为测试 PTLIA 算法对传统单路径 TCP 用户的公平性，本部分进行了 LIA、OLIA 和 PTLIA 算法的公平性对比实验。实验 3 依然使用图 2 中的仿真模型，传输数据量均为 10 MB，具体仿真参数如表 5 所示。传输时应用 6 使用 MPTCP，并同时使用链路 1 与链路 2，应用 7 使用 TCP，仅使用链路 2。为避免优先级的影响，应用 6 与应用 7 的优先级均设置为 0。

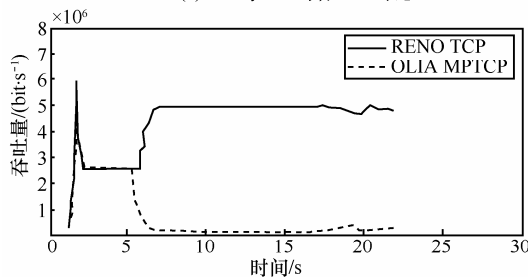
表 5 实验 3 仿真参数

参数	值
链路 1 带宽/(Mbit·s ⁻¹)	5
链路 1 延迟/ms	40
链路 2 带宽/(Mbit·s ⁻¹)	5
链路 2 延迟/ms	40
P6(应用 6 优先级)	0
P7(应用 7 优先级)	0

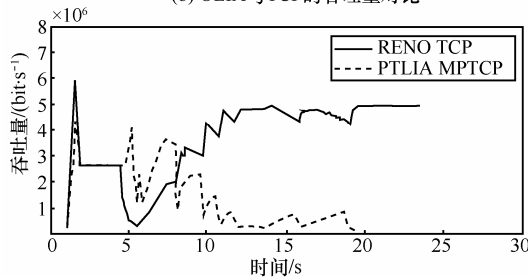
图 8 为使用 MPTCP 时 LIA、OLIA 以及 PTLIA 算法与使用 TCP 时的 RENO 算法的吞吐量对比。按照 MPTCP 拥塞控制算法公平性的理论设计, 当 MPTCP 与 TCP 同时使用瓶颈链路链路 1 时, 应尽可能使 TCP 与 MPTCP 用户获得相同的链路带宽资源。针对本实验环境, 因链路 1 与链路 2 带宽相同, 则 MPTCP 应用应尽可能少地侵占链路 1。使 MPTCP 应用与 TCP 应用具有相同的带宽资源。



(a) LIA与TCP的吞吐量对比



(b) OLIA与TCP的吞吐量对比



(c) PTLIA与TCP的吞吐量对比

图 8 MPTCP 与 TCP 在瓶颈链路 1 上的吞吐量对比

由图 8 可以看出, 相比 PTLIA 与 LIA 算法, OLIA 算法的公平性最好, 公平性调整最迅速, 且

稳定地保持着对链路 1 带宽资源的低占有率。LIA 算法的公平性其次, PTLIA 算法公平性最低。但由图 8 中依然可以看出 PTLIA 算法具有公平性调整的作用, 有效降低了对 TCP 使用的瓶颈链路的带宽占有率, 验证了 PTLIA 算法满足理论 1 中的属性 2。

6 结束语

本文在参考已有的 MPTCP 拥塞控制算法的基础上, 根据未来车联网等系统中即将出现的数据传输需求, 提出了基于优先级和吞吐量预测的 MPTCP 拥塞控制算法 PTLIA。本文建议的拥塞控制算法是为了解决具有高优先级需求的实时数据的传输问题以及进一步提高 MPTCP 的数据传输效率。仿真结果表明, 该算法较其他的 MPTCP 拥塞控制算法, 明显提升了高优先级应用的数据传输速度, 并提升了整体数据的传输效率, 将进一步提高车联网系统的安全性。

在物理因素方面, 本文所提出算法的有效性将主要依赖吞吐量测量值与吞吐量预测模型的准确性, 在链路条件极度不稳定的 Wi-Fi 网络中可能将会有较差的表现。未来的研究方向或将考虑针对系统当前传输任务的优先级对算法的分组丢失策略进行适当调整, 以及调整吞吐量预测模型以适用于各种不同的网络环境。

附录 1 对 PTLIA 算法属性的证明

证明 属性 1)和 2): 提高吞吐量和不产生损害。

因 PTLIA 算法的拥塞窗口值增长因子的结构与 OLIA^[9] 算法相同, 故参考 OLIA 算法, PTLIA 算法中的拥塞窗口值增长率如下

$$\frac{dx_r(t)}{dt} = x_r(t)^2 \left(\frac{\frac{\lambda_r}{R_r^2}}{\left(\sum_{p \in M_u} x_p(t) \right)^2} - \frac{p_r}{2} \right) + \frac{\alpha_r}{R_r^2} \quad (10)$$

当 $\frac{dx_r(t)}{dt} = 0$ 时, 即当前链路的吞吐量已经达到最大值,

由式(8)的定义可知, 此时吞吐量增量 $\Delta x_r(t) = 0$, 故 $\lambda_r = 1, \alpha_r = 0$, 代入式(10)中得到

$$x_r(t)^2 \left(\frac{1}{R_r^2} - \frac{p_r}{2} \right) = 0 \quad (11)$$

因 $x_r(t)^2 > 0$, 故化简整理后得到

$$\sum_{q \in M_u} x_q(t) = \frac{1}{R_r} \sqrt{\frac{2}{p_r}} \quad (12)$$

若 r_{\max} 表示 MPTCP 中的最好路径，文献[9]则可将 MPTCP 设计目标一转化为吞吐量关系

$$\sum_{q \in M_u} x_q(t) \geq \frac{1}{R_{r_{\max}}} \sqrt{\frac{2}{p_{r_{\max}}}} \quad (13)$$

将 MPTCP 设计目标二转化为吞吐量关系

$$\sum_{q \in M_u} x_q(t) \leq \frac{1}{R_{r_{\max}}} \sqrt{\frac{2}{p_{r_{\max}}}} \quad (14)$$

即若要同时满足目标一与目标二，则只有满足

$$\sum_{q \in M_u} x_q(t) = \frac{1}{R_{r_{\max}}} \sqrt{\frac{2}{p_{r_{\max}}}} \quad (15)$$

式(12)表明 MPTCP 任意路径 q 上的吞吐量总和 $\sum_{q \in M_u} x_q(t)$ 与其作为单一路径 TCP 使用时所产生的吞吐量 $\frac{1}{R_r} \sqrt{\frac{2}{p_r}}$ 相等，且因 MPTCP 的路径选择策略为每次均选择

当前的最好路径，故当常规 TCP 使用 MPTCP 中的最好路径 r_{\max} 传输时，则能够获得与使用 MPTCP 时相同的吞吐量，即满足式(15)，故算法 PTLIA 满足属性 1)和 2)。

证明 属性 3): 拥塞平衡

拥塞平衡的作用是为了将数据从拥塞链路转移到非拥塞链路，故对于非拥塞链路其拥塞窗口值增长应该更加快速。若 r_1 、 r_2 分别表示 2 条不同的路径，且当前路径 r_1 较为拥塞，路径 r_2 条件较好， $\Delta w_{r_1}(t)$ 为一用户在路径 r_1 上 t 时刻的拥塞窗口值增量， $\Delta w_{r_2}(t)$ 为同一用户在路径 r_2 上 t 时刻的拥塞窗口值增量，即应该证明

$$\Delta w_{r_1}(t) < \Delta w_{r_2}(t) \quad (16)$$

由式(8)可知，

$$\Delta w_{r_1} = \lambda_{r_1} \frac{\frac{w_{r_1}}{R_{r_1}^2}}{\left(\sum_{p \in M_u} \frac{w_p}{R_p}\right)^2} + \frac{\alpha_{r_1}}{w_{r_1}}$$

$$\Delta w_{r_2} = \lambda_{r_2} \frac{\frac{w_{r_2}}{R_{r_2}^2}}{\left(\sum_{p \in M_u} \frac{w_p}{R_p}\right)^2} + \frac{\alpha_{r_2}}{w_{r_2}}$$

为方便讨论，本文假设在极端情况下路径 r_1 已经接近完全拥塞，故此时路径 r_1 上的吞吐量将几乎不再增长，此时 $\lambda_{r_1} \leq 1, \alpha_{r_1} \approx 0$ 。链路 r_2 状态非常良好，即 $r_2 \notin Cr$ ，故此时 $\lambda_{r_2} = 1$ 。由网络传输相关知识可知，当链路拥塞时，其 RTT 值将比条件良好链路的 RTT 值大，即有 $R_{r_1} > R_{r_2}$ 。

情况 1 当 $w_{r_1} \leq w_{r_2}$ 时，代入以上相关参数，对比后明

显可知此时

$$\lambda_{r_1} \frac{\frac{w_{r_1}}{R_{r_1}^2}}{\left(\sum_{p \in M_u} \frac{w_p}{R_p}\right)^2} < \frac{\frac{w_{r_2}}{R_{r_2}^2}}{\left(\sum_{p \in M_u} \frac{w_p}{R_p}\right)^2} + \frac{\alpha_{r_2}}{w_{r_2}}, \text{ 即式(16)成立。}$$

情况 2 当 $w_{r_1} > w_{r_2}$ 时，代入以上参数得到

$$\Delta w_{r_2} = \frac{\frac{w_{r_2}}{R_{r_2}^2}}{\left(\sum_{p \in M_u} \frac{w_p}{R_p}\right)^2} + \frac{\alpha_{r_2}}{w_{r_2}} > \frac{\sqrt{2\alpha_{r_2}}}{R_{r_2}} \frac{1}{\left(\sum_{p \in M_u} \frac{w_p}{R_p}\right)}$$

此时若有

$$\sqrt{2\alpha_{r_2}} \left(\sum_{p \in M_u} \frac{w_p}{R_p}\right) > \frac{w_{r_1}}{R_{r_1}} \quad (17)$$

则 $\Delta w_{r_1}(t) < \Delta w_{r_2}(t)$ ，即当式(17)成立时，此时 PTLIA 算法将开始对链路进行拥塞平衡调整。综上所述，当链路拥塞时，PTLIA 具有拥塞平衡功能，满足属性 3)。

证明 属性 4)。

因 K 与 T 是 2 个连续发生分组丢失的时刻，假定 t 时刻与 $t+1$ 时刻 RTT 值相同，且 $\lambda_r(t+1) = \lambda_r(t)$ ，则 $\Delta w_r(t+1)$ 与 $\Delta w_r(t)$ 的关系可以如下

$$\frac{\Delta w_r(t+1)}{\Delta w_r(t)} = \frac{\lambda_r \frac{\frac{w_r(t+1)}{R_r^2}}{\left(\sum_{p \in M_u} \frac{w_p(t+1)}{R_p}\right)^2} + \frac{\alpha_r(t+1)}{w_r(t+1)}}{\lambda_r \frac{\frac{w_r(t)}{R_r^2}}{\left(\sum_{p \in M_u} \frac{w_p(t)}{R_p}\right)^2} + \frac{\alpha_r(t)}{w_r(t)}}$$

因 t 与 $t+1$ 处于 2 次连续分组丢失的时间段内，故 w_r 和 x_r 在此时间段内将持续增长， $w_r(t+1) > w_r(t)$ ， $x_r(t+1) > x_r(t)$ ，且 $\alpha_r(t+1) \geq \alpha_r(t)$ ，即 $\frac{\alpha_r(t+1)}{w_r(t+1)} > \frac{\alpha_r(t)}{w_r(t)}$ 。故若要证明

$\frac{\Delta w_r(t+1)}{\Delta w_r(t)} < 1$ ，则只需证明

$$\frac{\frac{w_r(t+1)}{R_r^2}}{\left(\sum_{p \in M_u} \frac{w_p(t+1)}{R_p}\right)^2} < \frac{\frac{w_r(t)}{R_r^2}}{\left(\sum_{p \in M_u} \frac{w_p(t)}{R_p}\right)^2} \quad (18)$$

若在 t 与 $t+1$ 时刻其他路径均未发生分组丢失，则当 $j \in R_u, j \neq r$ 时有 $w_j(t+1) \geq w_j(t)$ 。

$$\text{故 } \sum_{p \in M_u} \frac{w_p(t+1)}{R_p} > \sum_{p \in M_u} \frac{w_p(t)}{R_p}, \text{ 因式 } \frac{\frac{w_r(t+1)}{R_r^2}}{\left(\sum_{p \in M_u} \frac{w_p(t+1)}{R_p}\right)^2}$$

中所有项均为正数,且分母为平方增量,故分母增量远大于分子增量,由相关数学公式可知式(18)成立, $\frac{\Delta w_r(t+1)}{\Delta w_r(t)} < 1$

得证。PTLIA 算法满足属性 4)。

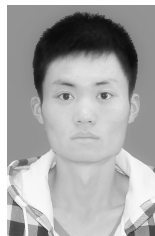
参考文献:

- [1] LEE E, GERLA M, et al. Vehicular cloud networking: architecture and design principles[J]. Communications Magazine IEEE, 2014, 52(2): 148-155.
- [2] DENG S, NETRAVALI R, SIVARAMAN A, et al. WiFi, LTE, or both? measuring multi-homed wireless internet performance[C]//2014 Conference on Internet Measurement. ACM, c2014: 181-194.
- [3] FORD A, RAICIU C, HANDLEY M, et al. Architectural guidelines for multipath TCP development[EB/OL]. <https://tools.ietf.org/html/rfc6182>.
- [4] WISCHIK D, RAICIU C, GREENHALGH A, et al. Design, implementation and evaluation of congestion control for multipath TCP[C]// Usenix Conference on Networked Systems Design and Implementation. USENIX Association, c2011:99-112.
- [5] CLOUD J, PIN C F D, ZENG W, et al. Multi-path TCP with network coding for mobile devices in heterogeneous networks[C]//IEEE Vehicular Technology Conference Institute of Electrical and Electronics Engineers (IEEE). c2013:1-5.
- [6] KUMAR G S, KHARA S. Technique to improve the file transfer outcomes between road side unit and vehicles in vehicular ad hoc networks[C]//2015 International Conference on Computing, Communication & Automation (ICCCA). IEEE, c2015:360-366.
- [7] HONDA M, NISHIDA Y, EGGERT L, et al. Multipath congestion control for shared bottleneck[C]//Proceedings of Protocols for Future, Large-Scale & Diverse Network Transports Work-shop, c2009.
- [8] RAICIU C, WISCHIK D, HANDLEY M. Practical congestion control for multipath transport protocols[EB/OL]. <http://tools.ietf.org/pdf/rfc6356.pdf>.
- [9] KHALILI R, GAST N, POPOVIC M, et al. MPTCP is not pareto-optimal: performance issues and a possible solution[J]. ACM/IEEE Transactions on Networking, 2013, (5):1651-1665.
- [10] RAICIU C, HANDLEY M, WISCHIK D. Coupled congestion control for multipath transport protocols[EB/OL]. <http://tools.ietf.org/pdf/rfc6356.pdf>. 2011.
- [11] WISCHIK D, HANDLEY M, RAICIU C. Control of multipath TCP and optimization of multipath routing in the internet network control and optimization[M]. Springer Berlin Heidelberg, 2009.
- [12] RAICIU C, PLUNTKE C, BARRE S, et al. Data center networking with multipath TCP[C]//ACM Workshop on Hot Topics in Networks. Monterey, Ca, USA, c2010:1-6.
- [13] YAGHMAEE M H, ADJEROH D. A new priority based congestion control protocol for wireless multimedia sensor networks[C]//2013 IEEE 14th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM). IEEE, c2008:1-8.
- [14] PATIL D, DHAGE S N. Priority-based congestion control protocol (PCCP) for controlling upstream congestion in wireless sensor network[C]//International Conference on Communication, Information & Computing Technology. IEEE, c2012:1-6.
- [15] JAN M A, NANDA P, HE X, et al. PASCCC: Priority-based application specific congestion control clustering protocol[J]. Computer Networks, 2014, 74(1):92-102.
- [16] LI Z, LIU P X. Priority-based congestion control in multi-path and multi-hop wireless sensor networks[C]// IEEE International Conference on Robotics and Biomimetics. c2007: 658-663.
- [17] TUNG L C, LU Y, GERLA M. Priority-based congestion control algorithm for cross-traffic assistance on LTE networks[C]// 2013 IEEE 78th Vehicular Technology Conference (VTC Fall). c2013:1-5.
- [18] CHIU D M, JAIN R. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks[J]. Computer Networks & Isdn Systems, 1989, 17(1):1-14.
- [19] MATHIS M, SEMKE J, MAHDAVI J, et al. The macroscopic behavior of the TCP congestion avoidance algorithm[J]. ACM Sigcomm Computer Communication Review, 2001, 27(3):67-82.
- [20] SHAILENDRA S, BHATTACHARJEE R, BOSE S K. Improving congestion control for concurrent multipath transfer through bandwidth estimation based resource pooling[C]// 2011 8th International Conference on Information, Communications and Signal Processing (ICICS). c2011:1-5.
- [21] DING N, TAN G Z, ZHANG W, et al. Character-aware traffic flow data quality analysis based on cusp catastrophe theory and wireless sensor network[J]. Ad hoc & Sensor Wireless Networks, 2013,18(1):277-292.

作者简介:



丁男 (1978-), 男, 辽宁沈阳人, 博士, 大连理工大学副教授、硕士生导师, 主要研究方向为信息物理系统、嵌入式技术、移动网络通信技术、物联网与车联网及其应用等。



林滔 (1992-), 男, 四川南充人, 大连理工大学硕士生, 主要研究方向为车联网、多链路 TCP 等。

宋彩霞 (1977-), 女, 山东即墨人, 青岛农业大学讲师, 主要研究方向为车联网安全通信协议、信道资源分配、拥塞控制。

谭国真 (1960-), 男, 辽宁本溪人, 博士, 大连理工大学教授、博士生导师, 主要研究方向为车联网与无人驾驶技术、人工智能、智能交通系统、网络算法与复杂性理论等。