

文章编号:1007-5321(2019)01-0001-15

DOI:10.13190/j.jbupt.2018-280

# 5G 多接入网络 TCP 研究与展望

王 莹, 李洪林, 费子轩, 赵兹宇, 王 虹

(北京邮电大学 网络与交换技术国家重点实验室, 北京 100876)

**摘要:** 为了对第 5 代移动通信系统(5G)中的传输控制协议(TCP)进行优化,针对 5G 多接入网络中 TCP 协议的优化机制展开综述. 分析了无线接入网络中 TCP 协议的局限性和面临的挑战;将无线 TCP 优化方案分为单路径 TCP 优化和多路径 TCP 优化方案,并对多种优化方案进行了对比分析;最后提出了 TCP 协议优化的发展方向.

**关 键 词:** 多接入网络; 第 5 代移动通信系统; 传输控制协议; 性能优化

**中图分类号:** TN911.22

**文献标志码:** A

## Research and Prospect of TCP Optimization in 5G Multi-Access Networks

WANG Ying, LI Hong-lin, FEI Zi-xuan, ZHAO Hong-yu, WANG Hong

(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

**Abstract:** In order to optimize the transmission control protocol (TCP) in the fifth generation of mobile communications system (5G), a survey on the optimization mechanism of TCP protocols in 5G multi-access networks is given. Firstly, the limitations and challenges of TCP protocols in wireless access networks are analyzed. Then, the TCP optimization schemes in wireless access networks are sorted out, which are mainly classified into single-path TCP optimization and multipath TCP optimization, simultaneously the comparative analysis of various optimization schemes is summarized. Finally, the development directions of TCP protocol optimization are presented.

**Key words:** multi-access networks; the fifth generation of mobile communications system; transmission control protocol; performance optimization

移动互联网在无线通信发展的不断推动下,已经逐渐发展成为人们生活中不可或缺的一部分,从应用服务供给侧数量看,移动互联网应用服务已经成为主角. 随着移动终端设备的快速更新,人们在体验更优质服务的同时也对未来网络有了更多的期待和要求. 移动互联网数据流量的增长持续快速上升,全球移动应用服务市场稳步增长. 据爱立信公司的数据显示,2016 年第一季度全球移动数据流量比 2015 年增幅达 60%,预计 2021 年全球移动数据流量比 2015 年将增加 10 倍. 与此同时,全球应用

服务市场稳步增长. 根据市场研究公司 App Annie 数据显示,2015 年全球移动 App 应用市场的规模约 700 亿美元,到 2020 年将达 1 890 亿美元,5 年内将以 22% 的年复合增长率平稳增长<sup>[1]</sup>.

为应对移动数据流量爆炸式增长的压力,学术界和工业界在无线传输技术方面开展了大量的研究. 传输控制协议(TCP, transmission control protocol)自 20 世纪 80 年代被提出来之后,一直被认为是一种效果较好的 Internet 网络传输控制协议. 统计数据表明,因特网上的数据流有 90% 以上基于

收稿日期: 2018-11-12

基金项目: 国家科技重大专项项目(2017ZX03001009)

作者简介: 王 莹(1976—),女,教授,博士生导师, E-mail: wangying@bupt.edu.cn.

TCP 协议. TCP 作为面向连接的、可靠的、端到端的数据传输控制协议,在传统的有线网络中有很好的性能,然而当传统 TCP 应用于无线网络时,所面临的是与有线网络截然不同的网络环境,原有的算法和策略已不适应新的网络环境,性能显著下降<sup>[2]</sup>.

## 1 TCP 协议的局限性与挑战

TCP 是目前广泛采用的传输控制协议,为各主机之间提供可靠有序的传输服务,在保障网络通信性能方面起着非常重要的作用. TCP 设计主要针对系统稳定性、协议兼容性、业务公平性、资源利用率以及拥塞控制等问题.

TCP 拥塞控制机制是保证网络能够稳定地运行、快速地发展、具备较强鲁棒性的重要因素. 然而传统有线网络中采用的拥塞控制机制性能在具有随机链路差错率高、非对称链路、带宽有限、易受环境影响、移动频繁等特征的有线/无线异构网络环境下急剧下降.

### 1.1 4G 网络中 TCP 协议的局限性

TCP 协议的拥塞控制机制主要是为有线网络而设计的,每当检测到数据丢失都会启动相应的拥塞控制机制,减小发送窗口的大小,降低 TCP 的发送速率. TCP 协议在有线网络中的应用表现出了良好的性能,而在无线网络中性能严重下降. 有线网络中网络拥塞是引起报文丢失的主要原因,而在无线网络中,除了拥塞引起的报文丢失,误码率高、信道信号衰弱等也会引起网络中数据的丢失,此时 TCP 依然会调用拥塞控制机制,此种虚假拥塞判断是 TCP 协议在无线网络中性能下降的主要原因<sup>[3]</sup>.

通信服务市场的广泛需求,对移动通信技术的发展提出了越来越高的要求. 随着第 4 代移动通信系统(4G, the fourth generation of mobile communications system)的商用和广泛部署,移动数据流量急剧增长,4G 移动通信技术需求量也日益增加. 人们在享受越来越高速的数据业务同时,对 4G 移动通信网络提出了更高的要求. 如何为用户提供更加高效的通信服务,实现更好的用户体验,是 4G 移动通信网络所面临的一大挑战.

TCP 协议作为被广泛采用的传输控制协议,在 4G 网络中仍然有举足轻重的地位. 然而目前所采用的 TCP 协议主要是为有线网络而设计的,在 4G 无线网络中其性能急剧下降. 提升 4G 网络中 TCP

协议的传输性能是优化 4G 网络的重要一步<sup>[4]</sup>.

### 1.2 5G 网络中 TCP 协议迎来新的挑战

移动互联网的进一步发展将带来未来移动流量超千倍增长,推动了移动通信技术和产业的新一轮变革. 随着通信技术的快速发展,第 5 代移动通信系统(5G, the fifth generation of mobile communications system)已经成为当前无线通信领域的研究热点,是面向 2020 年及未来移动通信产业需求的新一代无线通信系统,旨在满足 5G 时代大带宽、高容量和低时延的业务要求. 5G 作为未来网络发展的新进程,对网络架构和通信技术提出了重大挑战<sup>[5]</sup>.

未来的网络必将融合多种通信方式,形成多种网络共存的局面. 由于移动通信网络的演进特性,未来的网络将包括 3G、4G 以及 WLAN 网络等多种制式,是无缝、异构、融合的网络. 因此,未来将形成蜂窝与 Wi-Fi 融合组网的新型网络架构,5G 是多种接入技术并存的异构融合网络<sup>[6]</sup>.

传统 TCP 协议在 4G 无线网络中即面临性能危机,带宽利用率很低,当其应用于 5G 无线网络时,性能会进一步恶化,造成带宽资源的极大浪费,难以满足 5G 网络高吞吐量、低时延的要求. 相比于 4G 网络中的 TCP 协议,5G 异构融合的网络架构给 TCP 协议带来了更加严峻的挑战,对 TCP 协议的优化需求迫在眉睫. 基于以上背景,5G 网络中 TCP 协议的优化一直是研究的热点,移动边缘计算(MEC, mobile edge computing)作为 5G 网络架构的演进方向,也为 TCP 协议提供了新的优化方向,图 1 所示为 5G 多接入网络中 TCP 协议的优化方向分类.

## 2 无线接入网络中单路径 TCP 优化

通过对 TCP 协议的分析可知,基于有线网络设计的 TCP 协议在 4G、5G 无线接入网络中存在很大的局限性. 基于此,国内外研究者已经对无线网络环境下的拥塞控制机制进行了研究,以改善 TCP 协议在无线接入网络中的性能. 近年研究的机制主要有端到端机制、分段连接机制和跨层机制<sup>[7]</sup>.

### 2.1 端到端机制

端到端机制适用于无线传输环境,且保留了 TCP 端到端语义,其核心思想是通过改进 TCP 协议提高发送端对网络真实情况的感知能力,从而直接或间接地判断出数据丢失的原因,根据不同的拥塞情况启动不同的拥塞控制策略,并且不需要中间路由节点的支持.

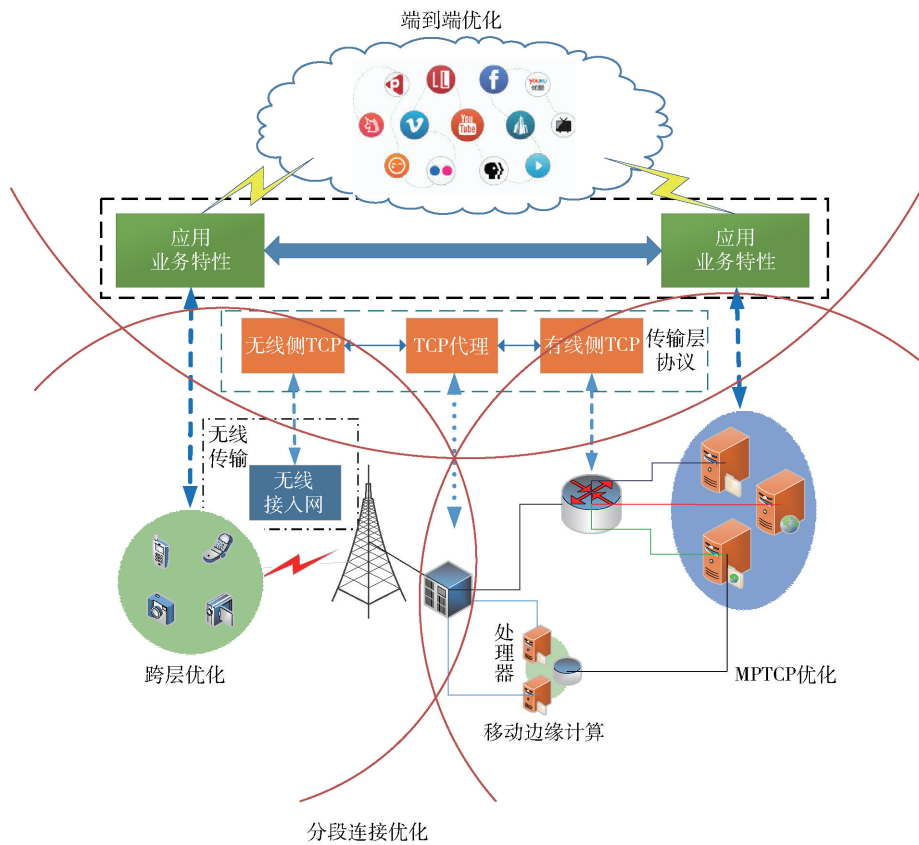


图 1 5G 多接入网络中 TCP 协议的优化方向

1) 基于带宽估计

传统 TCP 机制在有线网络中有较好的性能表现,但在高误码率、低带宽、非对称链路、长时延的有线/无线混合的异构网络中,性能急剧下降. 传统 TCP 拥塞控制机制,例如 Reno 和 Cubic,其拥塞避免阶段的拥塞窗口 (CWnd, congestion window) 随着往返时间 (RTT, round-trip time) 而逐渐增大,窗口的增长速度与当前网络的实际情况并没有建立联系,因此拥塞窗口与可用带宽没有很好地匹配,导致其带宽利用率较低.

基于以上问题,Wu 等<sup>[8]</sup>提出基于可用带宽估计的改进拥塞控制机制 TCP NewBR,以期提高网络带宽利用率以及网络吞吐量. 此机制通过估计当前可用带宽,计算带宽比例因子  $\delta$ ,进而根据网络的真实情况动态调节拥塞避免阶段的 CWnd. 带宽比例因子  $\delta$  的值介于 -1 和 1 之间,当  $\delta$  趋近于 -1 时,表明此时链路处于严重的拥塞状态,应适当减少发送窗口的值;当  $\delta$  趋近于 1 时,表明当前网络处于较好的状态,发送窗口可以继续增加. 在拥塞避免阶段,每收到一个 ACK (acknowledgement) 确认包,则根据式(1)调整 CWnd,从而实现对发送速率的自适应

应调节.

$$w = w + \frac{1}{w} \delta \tag{1}$$

其中  $w$  为拥塞窗口. 此机制根据当前网络实际状况动态调节 CWnd,相比于传统 TCP 拥塞控制机制的窗口增加策略,该机制使得窗口在更长时间内稳定在拥塞避免阶段,CWnd 与可用带宽更加匹配,提升系统的带宽利用率.

通过估测带宽的方法实现 TCP 拥塞控制相比于传统 TCP 过于保守的 AIMD (additive increase multiplicative decrease) 机制更加充分地利用了当前带宽,提高了系统吞吐量. TCP NewBR 的带宽估计调节窗口机制虽然一定程度上提升了系统吞吐量,但其本质上仍为丢包的拥塞控制机制. 事实上,丢包本就不应该作为拥塞的标志,它只是拥塞的一种表现. 同时 CWnd 仅仅规定了当前 TCP 最多可以发送多少数据,却没有明确发送速率,可能会造成数据突发的情况. 基于以上问题,同时为了应对无线网络噪声丢包导致的虚假拥塞判断情况,Cardwell 等<sup>[9]</sup>提出了 BBR (bottleneck bandwidth and round-trip propagation time) 拥塞控制机制.



BBR 不断采集连接时间窗口内的最大带宽和最小 RTT,并依据反馈的当前实际带宽和当前最大 RTT 调节增益系数  $G$ . 基于最大带宽以及当前的增益系数  $G$  计算发送速率和  $CW_{nd}$ ,将这 2 个结果作为拥塞控制算法的输出. 在 TCP 连接的持续过程中,每收到一个 ACK,都会计算即时带宽,然后基于此反馈结果不断调节增益系数  $G$ . BBR 是一个典型的封闭反馈系统,可根据带宽和 RTT 的变化不断调节输出. 由于噪声丢包不会急速拉低带宽的值,所以 BBR 不会因为噪声丢包而迅速削减  $CW_{nd}$ . 可通过区分噪声丢包和拥塞丢包,提高链路带宽利用率,提升系统吞吐量.

## 2) 基于网络编码

通过新技术实现 TCP 优化一直是研究的热点. 随着网络编码的不断发展,人们提出 TCP 网络编码 (TCP/NC, TCP with network coding) 方案,将网络编码与 TCP 相结合,降低丢包对传输层拥塞控制机制的影响,缩短因显式重传通告引发的延时. 这一类机制只需在端处进行修改,无需跨层协作,其改进之处对传输网络而言是透明的.

TCP 网络编码是端到端的传输技术, Zhang 等<sup>[10]</sup>提出了 TCP 友好的 TCP/NC 拥塞控制机制——CEE-TCP/NC. CEE-TCP/NC 依赖于重复确认消息 (Dup-ACK) 与往返时间 RTT 的变化,同时参考拥塞丢包特性,以上述状态信息作为依据进行拥塞暴露,并判断当前的拥塞程度. CEE-TCP/NC 不仅可以避免无线随机丢包对拥塞控制的影响,还可以保证在拥塞的有线链路中,发送端感知到的拥塞程度与拥塞敏感的 TCP 流所感知的程度相同,因此可以公平地竞争资源.

TCP 网络编码可提升无线环境下的数据流吞吐量,在无线链路丢包率相同的情况下,其拥塞窗口变化相比于传统 TCP 而言更加平滑. 得益于 TCP 网络编码的网络传输透明性,其部署不再受限于网络环境,只要收发信息的两端均支持 TCP 网络编码,即可进行数据传输,并具有较强的可实现性和可扩展性.

## 3) 基于业务特性

TCP 改进机制通过获取特定网络状况信息提高 TCP 发送端对当前网络的感知能力,从而实现发送速率的调整,缓解拥塞情况的恶化. 然而考虑到 TCP 业务的多样性,可以根据业务特性的不同采取特定的拥塞控制方案,此种 TCP 改进机制有很强的

针对性,并不能适用于所有的 TCP 数据传输.

对于一些时延容忍度较高的业务,可以通过一段时间内阻止其占用网络资源的方法,来缓解拥塞的恶化. Chang 等<sup>[11]</sup>提出在 LTE-A 网络下,当发生拥塞时,拒绝 IoT (internet of things) 设备的请求,产生一个合适的等待时间,从而缓解拥塞. 此方案主要为上行的拥塞控制方案,其中等待时间的设置尤为重要. 如果等待时间设置过短,设备会马上发出 2 次连接请求,这会加重额外的网络负载;如果等待时间设置过长,可能会导致网络产生闲置时间.

为了防止播放中断影响 QoE (quality of experience),视频业务都有严格的传输下限. 然而,目前的 TCP 传输并没有考虑业务 deadline 特性,对不同 deadline 的业务都采用相同的传输模式,没有实现数据流最小化 deadline-missing 的数量. 基于视频业务的 deadline 特性, Claeys 等<sup>[12]</sup>提出一种基于 TCP Reno 版本的 deadline-aware 的拥塞控制策略. 通过在传输层获取不同视频流的 deadline 信息,可以对视频流业务划定优先级,根据优先级的不同,动态调整拥塞窗口,从而更好地协调不同 deadline 业务之间的传输模式. 此策略需要对发送端协议做出适当改变,不紧急的数据流允许其他数据流优先传输,紧急的数据流相较于公平算法可以获得更高的吞吐量.

## 2.2 分段连接机制

TCP 在无线侧的性能表现很差,一味地改进传统 TCP,对性能的提升幅度有所限制,继而引出分段连接机制 (Split-TCP). TCP 分段连接是将传统端到端的 TCP 链路分裂开,分为有线段和无线段两部分<sup>[13]</sup>,分别独立地执行两套 TCP 拥塞控制算法,从而解决 TCP 端到端链路中有线侧的性能受到无线侧的影响,导致整条链路性能下降的问题. 分段连接方案能够很好地提升系统整体吞吐量. 分段连接机制的场景如图 2 所示,其中, FH (fixed host) 和 SH (split TCP host) 之间为有线段连接, MH (mobile host) 和 SH 之间为无线段连接.

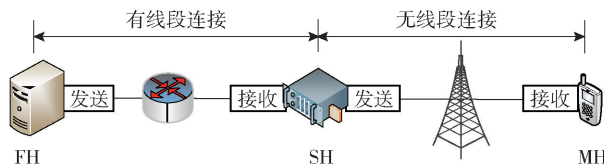


图 2 分段连接机制场景

Bakre 等<sup>[14]</sup>提出的 I-TCP (indirect-TCP) 是最早

建议采用分段连接的。I-TCP 两段均使用标准的 TCP 连接。发往 MH 的数据首先被基站 (BS, base station) 接收, BS 接收到数据后便向 FH 发送 ACK, 然后将数据转发到 MH。I-TCP 有助于有线链路屏蔽无线链路的不确定性, 并且 SH 上的 TCP 协议不需要改变。然而 I-TCP 机制不能够维持 TCP 端到端语义, BS 在收到 MH 的确认之前就向 FH 进行确认, 因此 FH 收到的确认并不能证明 MH 成功接收到数据。同时, BS 和 MH 之间依然使用标准 TCP 连接, 不能有效地处理无线丢包问题, 从而导致整体性能的下降。

为了解决 I-TCP 不能维持端到端语义的问题, Brown 等<sup>[15]</sup>提出了 M-TCP (TCP for mobile cellular networks) 方案。M-TCP 也是一种分段连接的方法, 它的体系结构可以被看作 3 层, 在最低层移动主机 MH 与每个蜂窝 BS 通信; 多个 BS 由一个监视主机控制; 最上层是监视主机和固定主机 FH 通信。

M-TCP 中仍然使用标准的 TCP, 与 I-TCP 中 BS 一旦收到 FH 的数据即发送确认不同, M-TCP 中只有收到来自 MH 的确认时才会发送确认到 FH, 从而维持了 TCP 端到端的特性。另外 M-TCP 还采用了 Freeze-TCP 中的零窗口通告机制, BS 通过检测发往 MH 的数据包和反向确认包的变化来判断 MH 的连接状态。一旦 BS 认定 MH 处于断开状态, 就通过发送声明窗口大小为零的数据包来迫使 FH 的 TCP 进入保持状态。当 MH 和 BS 建立新的链路后, BS 通知 FH 退出保持状态从而进入先前的工作状态, 这样, 发送端 TCP 在切换前后不必降低传输速率, 因此提高了带宽利用率。

M-TCP 虽然解决了端到端语义问题, 然而如果 MH 接收数据发生延迟, 可能会引起 FH 产生不必要的重传。M-TCP 使得基站必须缓冲大量的状态信息, 其任务过于复杂。当业务量很大时, 基站的负荷加重, 如果移动设备频繁切换, 基站之间状态信息的传输会产生较大的时延, 导致丢包。

M-TCP 维持 TCP 端到端语义的方法使得数据得不到及时确认, 为了解决此问题, 更好地保证 TCP 端到端语义, Kim 等<sup>[16]</sup>提出 ES-TCP (enhanced split TCP) 方案。该方案不需要对 MH 和 FH 处的协议进行修改, 将 EH (enhanced split TCP host) 部署于 MH 和 FH 之间。EH 接收端接收来自 FH 的数据, 发送端将数据转发给 MH。在 LTE 网络中, PGW (packet data network gateway) 是部署 EH 的选择之一。

ES-TCP 分别独立地控制有线链路和无线链路, 避免将无线链路的丢包问题引入有线链路, 同时保证端到端语义。当 EH 接收到新数据, 则向 FH 处发送返回 ACK, 但此 ACK 确认的序列号要比实际期望的下一个序列号小 2 bit。这样特殊的 ACK 确认方法是为了保证端到端语义在 MH 处没有收到所有的数据之前不会向 FH 处确认所有已收到的数据 (保留 2 bit 不进行确认)。因此 FH 在收到 ACK 确认时, 知道 MH 是否已经收到所有的数据。由于预留 ACK 的 2 bit 没有确认, 所以 FH 处的 RTO (retransmission time out) 和 EH 处的 RTO 都可能会超时。为了防止超时带来的重传问题, EH 采用 ZWA (zero window advertisement) 机制冻住 FH, 其发送 ACK 包以确认预留 2 bit 之中的 1 bit, 由于最后 1 bit 没有确认, 故端到端语义得以保证。

分段连接机制可以提升 TCP 的性能表现, 然而一味地改进 TCP 拥塞控制机制, 适应无线信道多变的环境, 已经很难使得吞吐量有较大幅度的提升。发生丢包时, 由于存在拥塞控制, 无线侧仍然会留下较大的空闲带宽。与分段连接机制中改进无线侧拥塞控制不同, Le 等<sup>[17]</sup>提出在 Split-TCP 的场景下, 移除 Wi-Fi 无线侧的拥塞控制机制, 允许 TCP 在无线信道状况下尽可能快地发送数据, 最大化利用无线侧带宽。但同时保留一个固定的拥塞窗口进行流量控制, 防止接收端的 buffer 过载溢出。此方案无需对服务器和客户端进行修改, 在中间代理处实现。

## 2.3 跨层机制

基于单一某层的改进使 TCP 协议在异构网络中性能表现不佳, 因此, 以传统 TCP 算法为基础、多个协议层之间相互协作、各层信息相互共享交流、联合决策判定的跨层方案成为异构网络传输控制技术一个新的发展方向。

根据实现方式的不同, 跨层设计主要可分为设计新的接口、合并相邻层、添加分层以及设计层间耦合等方案。跨层设计并不是否定传统的网络分层体系结构, 而是通过模糊层与层之间的界限将网络各层的信息进行共享, 打破各层间的独立性, 从而通过层与层之间的信息交互增强系统的自适应能力。

### 1) ECN 显示拥塞通知

TCP 跨层机制通过中间节点的跨层协作, 协助 TCP 实现相应的拥塞控制。IETF RFC 3168<sup>[18]</sup>提出经典的 ECN (explicit congestion notification) 显示拥塞通知方案, 有效解决了传统 TCP 的时延问题。经

经典的 ECN 采用 RED(random early detection)算法通过对数据包设置 ECN 比特位来标识网络发生拥塞. ECN 的基本思想是:当路由器发生早期拥塞时,不丢弃数据包,而是标记数据包 IP(internet protocol)报头的 CE(congestion encountered)比特位,此处 CE 标记的概率随着路由器队列长度的增加而线性增大. 当接收端收到被标记 CE 的数据包后,通过在其返回的 ACK 包中设置 ECE(ECN-Echo)比特位,将拥塞信号传递给发送端,发送端进而降低发送速率来避免拥塞,此处发送速率降为一半. 发送速率降低以后,发送端通过在发送数据包中标记 CWR(congestion window reduced)来通知接收端在后续的 ACK 包中取消 ECE 比特位的标记.

经典的 ECN 方案消除了恢复丢失数据包的时间,因此降低了 TCP 包的时延. 然而,RED 算法的高、低队列长度参数难以准确设置,且发送速率的削减仅为单一的减半处理,因此吞吐量方面并没有实现很大的提升. 为了同时实现低时延、高吞吐量特性,Alizadeh 等<sup>[19]</sup>对经典的 ECN 加以改进,提出了基于改进 ECN 的 DCTCP(data center TCP)拥塞控制机制. DCTCP 在路由器发生拥塞时,不丢弃数据包,而是标记数据包 IP 报头的 CE 比特位. 与经典 ECN 标记 CE 比特位的方法不同,DCTCP 仅设置一个队列长度阈值的参数  $K$ ,当数据包到达路由器时,如果当前的队列长度超过阈值  $K$ ,则此数据包被标记 CE 比特位;否则不会被标记. 当接收端收到被标记 CE 的数据包后,通过在其返回的 ACK 包中设置 ECE 比特位,将拥塞信号传递给发送端,发送端进而降低发送速率来避免拥塞,此处发送速率的调整与被设置 ECE 比特位的 ACK 包所占的比重有关. DCTCP 相较于经典 ECN 在实现了低时延的同时,对吞吐量的提升也更加明显.

ECN 机制可以实现对网络的拥塞估计,TCP 通过协作此跨层信息,能够更好地调整数据发送速率. 基于此,Jiang 等<sup>[20]</sup>提出了 CLTCP(congestion level based TCP)算法. CLTCP 是基于 Reno 算法的改进,首先使用 ECN 估测拥塞等级,进而根据拥塞等级的不同,在拥塞避免阶段,动态地调节拥塞窗口增长的参数,在丢包发生时,动态地设置慢启动阈值和拥塞窗口的值. 由于路径时延测量容易被反向拥塞和路由路径改变所干扰,CLTCP 的拥塞信号舍弃时延参数的使用,采用拥塞等级划分,因此可以避免因时延测量误差和差错所带来的拥塞虚假判断.

## 2) 底层协作和应用层协作

上述 TCP 跨层机制皆通过 ECN 协助 TCP 实现跨层感知,获取网络状况信息,从而自适应地调节 TCP 发送速率,提高带宽利用率. TCP 跨层机制也可以通过底层预测信息协助 TCP 实现拥塞控制,Zhu 等<sup>[21]</sup>提出了 2 种无线感知的 TCP 优化方案.

松耦合无线感知的 TCP 优化,通过联合考虑基站处提供的协作信息,UE(user equipment)得以对接下来的 ADR(achievable data rate)做出预测,并以此预测速率向服务器请求数据,进而使得 TCP 发送端以该速率发送数据,从根源上避免了拥塞的产生. 而紧耦合无线感知 TCP 优化的核心思想为申请-授权机制. 基站处的 layer2 和 layer1 层根据 IP 层 buffer 的挂起数据,同时考虑无线信道状况与分配给目标 UE 的无线资源产生调度原则. 调度原则产生之后,layer2 和 layer1 层即向上层(IP 层和 TCP 层)请求数据,此时 TCP 层传统的拥塞控制机制将被移除,下层的数据请求全部得到回应,因此可以充分利用空口带宽.

TCP 普遍用于流媒体应用中,FEC(forward error correction)编码作为直播流媒体系统中的应用层错误恢复方案,可以弥补 TCP 数据重传机制的缺点,是 TCP 在提高视频传输性能方面有效且互补的解决方法. Wu 等<sup>[22]</sup>基于 FEC 编码,利用 TCP 时延友好性优化实时视频传输. 该方案用于实时视频通信系统中,能够显著地减小端到端时延,并提升视频质量. Wu 等<sup>[23]</sup>基于 FEC 编码,利用优先级感知和 TCP 连接状态最小化视频失真,优化实时视频质量.

## 3) 移动边缘计算

随着 5G 网络的发展,TCP 跨层机制不再局限于传统的跨层协作. MEC 即移动边缘计算,是建设 5G 网络边缘云的普遍模式,是 5G 网络架构的演进方向. MEC 实现了应用网络与无线接入网的融合,为 5G 网络中的 TCP 协议提供了新的优化方向. 如图 3 所示,靠近移动用户终端的无线接入网(RAN, radio access network)基于边缘云节点提供的服务环境具备了计算能力,使得引入 MEC 的 5G 网络具备

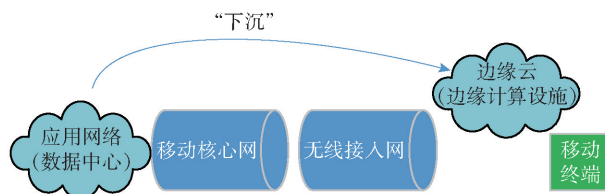


图 3 5G 网络中的 MEC 部署



了本地化、近距离、低时延、位置感知及网络信息开放等特点。

MEC 具有高带宽、低时延以及位置/内容感知等技术特征,基于 MEC 的智能视频加速可以改善 TCP 协议在异构无线网络中遭遇的情况<sup>[24]</sup>。TCP 不再需要主动探测可用的无线网络资源,而是由移

动视频服务器根据 MEC 服务器所提供的无线信息做出更合理的 TCP 拥塞控制决策,包括初始窗口大小的选择、对拥塞避免期间拥塞窗口值的设置、无线链路拥塞情况恶化时对拥塞窗口大小的调整等。

2.4 单路径 TCP 优化机制对比

单路径 TCP 优化机制的对比如表 1~3 所示。

表 1 端到端优化机制对比

文献	年份	指标	方案特点
[8]	2015	Bandwidth, Window size	通过带宽估计判定当前带宽比例因子 $\delta$ ,进而动态调节拥塞避免阶段的 CWnd 增长模式
[9]	2016	Window size, Sending rate	BBR 根据即时计算的带宽以及调整后的增益系数来计算发送速率以及 CWnd 的新值,并不是基于丢包的拥塞控制算法,真实地反映了网络的拥塞状态,能够区分噪声丢包和拥塞丢包
[10]	2014	RTT, Window size	CEE-TCP/NC 不仅依赖于 RTT 的变化,而且也依赖于丢包信息,通过按需重传、拥塞暴露和拥塞控制机制,对当前拥塞程度做出更准确地判断,从而实现更加合理的窗口调整
[11]	2016	Delay, Throughput	此方案为上行的拥塞控制方案,当发生拥塞时,拒绝 IoT 设备的请求,产生一个合适的等待时间,等待时间过去后,再接收其请求,从而缓解拥塞
[12]	2016	Window size	通过在传输层获取不同视频流的 deadline 信息,可以对视频流业务划定优先级,根据优先级的不同,动态地调整拥塞窗口,从而更好地协调不同 deadline 业务之间的传输模式

表 2 分段连接优化机制对比

文献	端到端语义	指标	方案特点
[14]	不能维持	Window size	I-TCP 两段均使用标准的 TCP 连接。发往 MH 的数据首先被基站接收,基站接收到数据后便向 FH 发送 ACK,然后将数据转发到 MH。I-TCP 有助于有线链路屏蔽无线链路的不确定性
[15]	维持	Window size	M-TCP 中仍然使用标准的 TCP。与 I-TCP 中基站一旦收到 FH 的数据即发送确认不同,M-TCP 中只有当收到来自 MH 的确认时才会发送确认到 FH,从而维持了 TCP 端到端的特性,但基站负载较重
[16]	维持	Window size	ES-TCP 分别独立地控制有线段链路和无线段链路,避免将无线链路的丢包问题引入有线链路,同时保证端到端语义,但基站负载较重
[17]	维持	Window size	提出在 Split-TCP 场景中,移除后半段无线侧的拥塞控制机制,实现最大的发送速率,充分利用无线侧带宽,同时保留一个固定的拥塞窗口进行流量控制

表 3 跨层优化机制对比

文献	年份	指标	方案特点
[18]	2001	Window size, Delay	ECN 当路由器发生早期拥塞时,标记数据包的 CE 比特位,此处 CE 标记的概率随着路由器队列长度的增加而线性增大。通过拥塞标记处理,将拥塞信号传递给发送端,发送端进而将发送速率降为一半,避免拥塞。ECN 方案消除了恢复丢失数据包的时间,降低了 TCP 包的时延
[19]	2011	Window size, Delay	DCTCP 是对 ECN 的改进,仅设置一个队列长度阈值 $K$ ,超过阈值 $K$ ,则标记 CE 比特位;否则不会被标记。通过拥塞标记处理,将拥塞信号传递给发送端,发送端进而合理调整拥塞窗口,避免拥塞。DCTCP 相较于经典 ECN 在实现了低时延的同时,对吞吐量的提升也更加明显
[20]	2015	Window size	使用 ECN 估测拥塞等级,根据拥塞等级的不同,在拥塞避免阶段,动态地调节拥塞窗口的增长;当丢包时,动态地设置慢启动阈值和拥塞窗口的值
[21]	2017	Throughput, Window size	松耦合优化方案:UE 得到基站传来的跨层信息,将 UE 预测的吞吐量信息上传给 TCP 发送端,进而适当调整拥塞窗口;而紧耦合方案将传统的 TCP 拥塞控制机制修改为“请求-授权”机制
[22]	2015	Delay, Quality	基于 FEC 编码,利用 TCP 时延友好性优化实时视频传输,显著减小端到端时延并提升视频质量
[23]	2017	Quality, Delay	基于 FEC 编码,利用优先级感知和 TCP 连接状态最小化视频失真,优化实时视频质量
[24]	2015	Throughput, Window size	在 MEC 服务器内部署无线分析应用,为视频服务器提供无线下行接口的实时吞吐量指标,以协助视频服务器做出更合理的 TCP 拥塞控制决策

### 3 无线接入网络中多路径 TCP 优化

多路传输控制协议 (MPTCP, multipath TCP) 是传统 TCP 的一种演进机制, 允许主机同时使用不同的网络接口建立单一 TCP 连接进行发送或接收报文, 更加合理地利用了带宽资源, 提高带宽利用率, 性能表现大大优于传统 TCP 协议. MPTCP 协议与传统的 TCP 协议兼容, 同时使用多条链路进行数据传输, 当其中一条路径因某种故障中断, 其他的路径仍能继续传输, 既减少了数据传输的时间, 又提高了传输可靠性<sup>[25]</sup>.

MPTCP 拥塞控制采用的控制策略就是对 TCP 进行扩展, 为每一个用于传输的子流维持一个单独的拥塞窗口, 每条子流分别根据路径上收到的确认包或者检测的丢包情况来增加或减少拥塞窗口. 多路径拥塞控制将同一个连接上的多个子流进行关联, 并将拥塞程度高的路径上的数据转移到拥塞程度低的路径上继续传输, 从而保证拥塞程度大的子流负载减少, 减小了链路丢包率. 原来拥塞程度小的路径负载增加, 提高了子流的使用效率. 通过设置合理的增加减少策略, 使得拥塞控制算法达到资源共享的理想状态, 最大化利用了网络资源. 同时保证了与当前 TCP 的公平性.

参照 RFC6356, 将多路径 TCP 的拥塞控制目标归结为以下 3 点, 通过这 3 个目标, 增加了网络的鲁棒性, 提升了整体吞吐量<sup>[26]</sup>.

1) 目标 1 (提高吞吐量): MPTCP 必须支持多路径, 并且多条路径上的总吞吐量不应该小于单一的 TCP 连接的吞吐量;

2) 目标 2 (不产生损害): 对于多条路径中的任何一条路径, 多路径 TCP 流不能比传统 TCP 流抢占更多的带宽;

3) 目标 3 (拥塞均衡): 在不违反前两条目标的前提下, 多路径 TCP 流要把最拥塞路径上的流量尽可能多地转移到别的路径上去.

#### 3.1 TCP 友好性

MPTCP 的目标 1 体现了 MPTCP 连接与传统 TCP 相比, 传输性能得到改善. 目标 2 保证了 MPTCP 与 TCP 在瓶颈链路上同时传输时的公平性, 防止 MPTCP 子流对 TCP 数据流的侵占, 导致 TCP 无法传输数据<sup>[27]</sup>.

在考虑 MPTCP 公平性方面, 多路径拥塞控制可以分为两类: Decoupled 和 Coupled. Decoupled 多路

径拥塞控制是对单路径 TCP 的简单扩展, 子流之间的拥塞窗口比较独立; 而 Coupled 多路径拥塞控制是将所有子流的拥塞窗口关联起来, 控制总拥塞窗口的大小.

MPTCP 早期的版本在每个子流上独立的运行 AIMD 机制. Decoupled MPTCP<sup>[28]</sup> 采用的策略是对每条子流进行单独控制. 在每条子流上分别独立的运行 TCP 拥塞控制, 然后检测每条子流的性能, 对子流  $r$  的拥塞窗口  $w_r$  作相应的增加或者减少.

1) 子流  $r$  上每收到一个 ACK 确认包, 拥塞窗口  $w_r \leftarrow w_r + 1/w_r$ ;

2) 子流  $r$  上检测到丢包时, 拥塞窗口  $w_r \leftarrow w_r/2$ .

在多路径传输的每个子流上都独立运行该算法, 若一条子流发生阻塞, 并不影响其他子流的传输性能. 但该算法也存在严重的问题, 当一个 MPTCP 连接的  $N$  条子流与单路径 TCP 连接共享同一个瓶颈链路时, 会严重抢占单路径 TCP 的带宽, 造成  $N$  倍的竞争, 所以, 这种方法并不能保证传输过程中的链路公平性问题.

为了解决上述的 TCP 友好性问题, 提出了多种 Coupled MPTCP 机制. Honda 等<sup>[29]</sup> 提出 EWTCP 机制, 通过对各子流设置侵略权值, 限制每条子流, 从而保证 MPTCP 流与 TCP 流公平共享瓶颈链路, 不会过多地抢占带宽资源.

1) 子流  $r$  上每收到一个 ACK 确认包, 拥塞窗口  $w_r \leftarrow w_r + 1/(\sqrt{n}w_r)$ ;

2) 子流  $r$  上检测到丢包时, 拥塞窗口  $w_r \leftarrow w_r/2$ .

相比于 Decoupled MPTCP, EWTCP 初步解决了瓶颈链路中的公平性问题. 但是该算法没有考虑每条子流的特性, 统一为每条子流分配相同的权值, 因此数据流量的分配不均衡, 不能将网络流量转移到拥塞程度低的链路上.

与 EWTCP 窗口的增长模式不同, Wischik 等<sup>[30]</sup> 提出 Semi-coupled MPTCP, 基本思想是基于链路耦合的拥塞控制策略, 通过多个子流之间进行协商, 并对属于每条子流的拥塞窗口做出相应的改变.

1) 子流  $r$  上每收到一个 ACK 确认包, 拥塞窗口  $w_r \leftarrow w_r + a/w_{\text{total}}$ ;

2) 子流  $r$  上检测到丢包时, 拥塞窗口  $w_r \leftarrow w_r/2$ .

$w_{\text{total}}$  表示所有子流的总窗口大小,  $a$  为侵略因子. Semi-coupled 算法基于各条子流的高耦合性, 通过总的拥塞窗口和侵略因子来调整每个子流的窗口增长模式. 在解决数据流之间侵占带宽的问题上,



相比于 EWTCP, Semi-coupled 算法总窗口参数的采用更加高效, 性能更优。

MPTCP 的目标 2 和目标 3 要保证在多路径传输过程中对现有 TCP 连接的公平性, 还要实现拥塞均衡, 确保算法的有效性。为了这个目标, RFC 6356<sup>[26]</sup>中提出, 一种改进的 Semi-coupled 机制 LIA (linked increases algorithm)。

1) 子流  $r$  上每收到一个 ACK 确认包, 拥塞窗口  $w_r \leftarrow w_r + \min \left( \frac{a}{w_{\text{total}}}, \frac{1}{w_r} \right)$ ;

2) 子流  $r$  上检测到丢包时, 拥塞窗口  $w_r \leftarrow w_r / 2$ 。

相比于 Semi-coupled 机制, LIA 进一步修正了拥塞窗口的增长模式, 确保多路径 TCP 流拥塞窗口的增长更加合理, 平衡与单路径 TCP 之间的带宽竞争能力, 从而更好地保证瓶颈链路的公平性。

### 3.2 拥塞均衡

MPTCP 的目标 3 主要是为了实现链路的拥塞均衡, 确保算法的有效性。拥塞均衡是将拥塞程度高的链路负载转移到拥塞程度低的子流上去, 降低了丢包率, 提升了网络的利用率, 以达到网络资源的最大化利用<sup>[31]</sup>。

为了满足 MPTCP 的设计目标, 实现多路径拥塞均衡, 已经提出多个 MPTCP 方案。Key 等<sup>[32]</sup>联合多路径路由和多路径拥塞控制, 在多路径数据传输时, 分别考虑协同拥塞控制和不协同拥塞控制的方法。在实现拥塞均衡的目标上, 协同多路径拥塞控制有着更好的性能表现。Ding 等<sup>[33]</sup>提出一种基于滑动模型的拥塞控制和流量调度的联合优化方案, 以保证无线网络中的服务质量 (QoS, quality of service)。该方案考虑每个链路队列, 解决网络效用最大化问题, 可以实现多路径的拥塞均衡, 具有很好的鲁棒性。

Bredel 等<sup>[34]</sup>使用 OpenFlow 来实现 MPTCP 中数据流的拥塞均衡。OpenFlow 用于根据当前服务器负载重新分配客户端请求, 通过 OpenFlow 控制器协调多路径传输, 实现网络内部每个子流的拥塞均衡, 从而提升带宽利用率。Alizadeh 等<sup>[35]</sup>提出一种数据中心处分布式拥塞感知的拥塞均衡机制。仿真结果表明, 该机制在实现 MPTCP 拥塞均衡方面有很好的性能。

MPTCP 的拥塞均衡目标, 不仅仅是将尽可能多的数据流量转移到拥塞程度较轻的路径, 而且也要给其他路径留有足够多的数据流量。为此, Zhao

等<sup>[36]</sup>提出基于 RTT 侵略因子来控制子流的拥塞窗口增长, 从而实现多路径的拥塞均衡。侵略因子的采用不仅仅使得拥塞程度较轻的路径得到充分利用, 而且也给其他路径分配了足够多的数据流量, 因此很好地实现了多路径之间的拥塞均衡。

### 3.3 其他目标

MPTCP 的 3 个设计目标保证了多路径传输的有效性, 增加了网络鲁棒性。在满足这 3 个目标的基础上, 为了应对不同场景的特定需求, MPTCP 也应该考虑特定的性能指标。

1) 进一步提升 MPTCP 性能

为了实现 MPTCP 数据流的公平性和拥塞均衡, RFC 6356 提出 LIA 算法, 然而该算法在子流之间转移了过多的负载, 不能完全实现拥塞均衡。基于此问题, Khalili 等<sup>[37]</sup>通过对 LIA 算法进行部分修正改进, 提出 OLIA (opportunistic linked-increases algorithm) 算法, 该算法具有比 LIA 算法更好的拥塞平衡能力, 同时能够很好地保证公平性。OLIA 是一种基于窗口的拥塞控制机制, 通过参考丢包率和 RTT, 选择出网络条件较理想的路径, 将负载从拥塞路径上转移, 更多地分配到不拥塞的、理想路径之上, 从而达到了拥塞均衡的目的。OLIA 克服了 LIA 算法无法完全实现拥塞均衡的缺陷, 流体模型证明了该算法是帕累托最优的。

OLIA 算法在子流中拥塞变化时, 反应能力较差。为了进一步提高算法的公平性和网络变化时的反应能力, Peng 等<sup>[38]</sup>提出 Balia 算法。Balial 算法探索一个新的设计框架, 在公平性和反应能力之间进行权衡, Balia 算法已在 Linux 内核中实现。

MPTCP 相较于单路径 TCP 而言, 在占用同样带宽的前提下, 会取得更大的吞吐量增益。然而受传输层协议限制, 当 RTT 和路径损耗速率相对固定时, MPTCP 的吞吐量并不会随着带宽的增加而线性增加, 其增益幅度很有限。这意味着增加的带宽并没有得到有效利用, 因此导致带宽利用率很低。基于此, Wang 等<sup>[39]</sup>提出一种带宽调度方案, 为每个链路分配合适的频谱资源。首先, 基于 Mathis Model 估测所需的带宽 VBW (validity bandwidth); 接着, 采用 Kalman Filter 对受多变网络环境影响的 VBW 进行平滑处理; 最后根据估测的带宽为每个链路分配带宽资源。此方案权衡吞吐量和频谱效率之间的关系, 不能一味地牺牲大带宽而很小幅度的提升吞吐量, 实现二者之间的动态

平衡,追求效益最大化。

MPTCP 虽然提高了无线接入网络中移动视频传输的效率,但现有的机制对于传输满足用户需求的视频服务仍然面临着很大的挑战。为了实现实时视频流的高质量传输,Wu 等<sup>[40]</sup>提出一种质量驱动的 MPTCP 机制。该机制基于速率分配和前向纠错编码,在端到端时延和丢失性能之间取得平衡,从而最大限度的减少视频失真,提高移动视频的传输质量。

网络编码技术的出现为多路径 TCP 的优化开拓了新方向,将网络编码与 MPTCP 相结合,进一步优化 MPTCP。作为一类新机制,TCP 网络编码要提高 TCP 流在无线传输中的吞吐量,同时需要保证引入网络编码后,原始机制的有利设计原则没有被破坏,保证公平性。多路径 TCP 中,编码对子流的保护程度不同将造成负载的不均衡分配。基于此问题,Xue 等<sup>[41]</sup>提出适用于多路径 TCP 网络编码的拥塞控制技术 Couple +。该机制可使编码子流及时发现拥塞,进行合理的负载迁移。此外,该机制还保证编码子流对同一瓶颈中的 TCP 数据流保持友好性。Couple + 可以使得网络编码更合理地与多路径 TCP 相结合,进一步实现 MPTCP 的 3 个目标。

## 2) 数据包重排序问题

MPTCP 多路径传输过程中,由于路径间时延、带宽和损失率不同,会产生乱序数据包,引发数据包重排序现象。数据包重排序会引起不必要的重传,极大地降低网络资源利用率,影响多路径数据传输的性能。减少数据包重排序的发生可以有效提升多路径传输的吞吐量,对于优化 MPTCP 起着至关重要的作用。

为了解决数据包重排序问题,Tsai 等<sup>[42]</sup>针对实时流传输,提出一种联合带宽聚合和分组调度的多路径传输机制。带宽聚合机制可区分拥塞丢包和无线丢包,以此获得路径间最优的传输速率。分组调度机制根据反馈信息调整传输序列,减少乱序数据包的产生,有效缓解乱序重排的影响,提高多路径传输的系统吞吐量。Kimura 等<sup>[43]</sup>分析了多路径传输中不同调度策略的影响,提出一种可选择调度策略的多路径传输机制。该机制在最大发送速率、最大窗口和最小时间/空间 3 种调度策略之间选择,旨在降低路径时延和损失,减少乱序现象的发生,提升多路径传输的吞吐量。

负载均衡旨在优化网络资源的使用,实现多路

径的流量分配策略。路径间有效的流量分配可以减少数据包重排序的发生,提升数据传输的性能。基于此,Kandula 等<sup>[44]</sup>提出一种动态的负载均衡机制,通过对数据包传输间距的限制避免数据包重排序的发生,精准地实现多路径传输的流量分配。Delgado 等<sup>[45]</sup>提出一种时延感知的负载均衡机制,通过多路径间合适的流量分配策略,降低平均丢包率和端到端时延,进而减少乱序重排的发生,提升多路径算法的传输性能。

## 3) 时延优化目标

MPTCP 相比于单路径 TCP 而言显著提升了带宽利用率,提高了吞吐量。然而对于一些交互式应用和服务而言,低时延特性同样非常重要。Park 等<sup>[46]</sup>权衡时延性能和成本效率之间的关系,采用贪婪启发式算法实现多路径子流之间的流量分配,提高了时延性能,同时降低了用户成本。Frommgen 等<sup>[47]</sup>提出一种减小时延和抖动的方案,该方案通过在多条路径上冗余的发送数据,避免了丢包引起的重传现象,改善了队头堵塞的问题,从而降低了链路的整体时延,保证了数据传输的可靠性。此方案通过牺牲一部分带宽来换取较低的时延,仿真结果表明,在实际的移动场景中该方案可以极大地降低往返时延。

## 4) 能耗优化目标

由于移动设备的能量有限,所以在保证视频服务质量的同时,如何实现较小的能量消耗成为 MPTCP 的突破点。在异构无线网络多路径视频传输中,Wu 等<sup>[48]</sup>提出 EDAM (energy-distortion aware MPTCP) 机制。该机制基于效用最大化理论,按照视频流量分配原则,在保证视频质量的同时,最小化移动设备的能量消耗。EDAM 机制对发送和接收两端的协议都有所更改,其核心模块为流量分配。在发送端,决策模块包括参数控制单元、流量分配和重传控制器;在接收端,信息反馈单元负责返回信道状态信息和 ACK 信息。参数控制单元根据接收端的反馈信息估计输入参数,流量分配和重传控制器用于数据包发送和重传策略的制定。通过流量分配模块实现异构无线网络中多路径视频数据的合理分配,从而权衡能量和视频质量之间的关系。

为了使移动设备在相同的能耗下能够接收更多的数据,Wang 等<sup>[49]</sup>提出一种基于节能的多路径拥塞控制算法。首先通过遗传算法输出速率分配向量以实现节能的目标,进而联合考虑节能和路径特征

调整多路径子流的拥塞窗口. 该方案基于 LIA 算法实现最佳的数据分配策略,通过子流间的流量转移最大限度地提高移动设备能效,同时保证吞吐量和

网络的负载均衡.

**3.4 多路径 TCP 优化机制对比**

多路径 TCP 优化机制的对比如表 4~6 所示.

表 4 TCP 友好性优化机制对比

文献	年份	指标	方案特点
[28]	2006	RTT, Bandwidth, Throughput	在多个子流上独立运行 AIMD 机制,与单路径 TCP 共存时,存在严重的公平性问题
[29]	2009	Throughput	通过对各子流设置侵略权值,限制每条子流,从而保证 MPTCP 流的友好性,但该算法没有考虑每条子流的特性,统一分配相同的权值,因此存在拥塞均衡的问题
[30]	2011	Throughput, RTT, Window size	Semi-coupled MPTCP 是基于链路耦合的拥塞控制策略,通过多个子流之间进行协商,设置侵略权值,从而确保公平性和高效性
[26]	2011	Throughput, RTT, Window size	LIA 是 Semi-coupled 的改进机制,其对拥塞窗口的增长模式进一步加以修正,确保多路径 TCP 流拥塞窗口的增长更加合理,实现更好的公平性保障

表 5 拥塞均衡优化机制对比

文献	年份	指标	方案特点
[32]	2011	Sending rate, Bandwidth	多路径数据传输时,分别考虑协同拥塞控制和不协同拥塞控制的方法. 在实现拥塞均衡的目标上,协同多路径拥塞控制有着更好的性能表现
[33]	2013	Sending rate, Queue length	基于滑动模型联合优化拥塞控制和流量调度,可保证无线网络中的 QoS,实现多路径的拥塞均衡目标,具有很好的鲁棒性
[34]	2014	Mean transfer time	通过 OpenFlow 控制器实现 MPTCP 中数据流的拥塞均衡,从而提升网络带宽利用率
[35]	2014	Delay, Throughput	提出一种数据中心处分布式拥塞感知的拥塞均衡机制,实现 MPTCP 数据流负载均衡
[36]	2015	Throughput, Window size	通过基于 RTT 的侵略因子来控制子流拥塞窗口的增长,从而实现多路径的拥塞均衡

4 发展方向与优化目标

在高速发展的移动互联网和不断增长的物联网业务需求的共同推动下,5G 网络要具备高速率、低时延、低能耗、安全可靠等特性. 3GPP 提出了 5G 未来移动通信的 3 个应用场景:增强移动宽带 (eMBB, enhanced mobile broadband); 海量机器类通信 (mMTC, massive machine type of communication); 超可靠低时延通信 (uRLLC, ultra reliable and low latency communication). 其中,eMBB 场景作为目前移动互联网的增强版主要侧重速率提升,未来 5G 标准要求单个 5G 基站至少能够支持 20 Gbit/s 的下行速率以及 10 Gbit/s 的上行速率. mMTC 场景是海量大连接,对应物联网等连接密度要求较高的场景. uRLLC 聚焦对时延极其敏感的业务,要求 5G 时延低于 1 ms,对应于无人驾驶、智能工厂等低时延应用.

随着 5G 网络的进一步发展,未来网络将形成新型架构,TCP 协议优化在新的网络架构下迎来了

新的挑战,同时也有了新的发展方向和优化目标.

**4.1 时延优化**

eMBB 聚焦是对带宽有极高需求的业务,如视频等大流量移动宽带业务,以满足人们对于数字化生活的需求. 以超高清视频、4 K/8 K 视频、VR (virtual reality)/AR (augmented reality)、全景视频等为代表的新一代视频应用正在蓬勃发展,大视频时代已经来临. 满足 eMBB 场景大带宽、高速率要求的同时,VR/AR 以及视频直播等业务对时延也有着较高的要求,实现高速率和低时延的双重目标是 MPTCP 协议优化的一大挑战.

MPTCP 多路径传输过程中,多路径子流之间存在一定的时延差异,因此会产生乱序数据包导致实际吞吐量低于理论吞吐量. 基于以上问题,为了满足 eMBB 场景的高速率需求,同时保证低时延特性,降低多路径子流之间的时延差异,提出异构网络下基于时延的 MPTCP 拥塞控制算法 (DBCCA, delay-based congestion control algorithm for MPTCP). DBCCA 算法联合优化吞吐量和时延目标,降低多路径子



表 6 其他目标优化机制对比

文献	年份	指标	方案特点
[37]	2013	Throughput	OLIA 是一种基于窗口的拥塞控制机制,通过参考丢包率和 RTT,将负载更多地分配到不拥塞的、理想路径之上,从而实现拥塞均衡. 通过流体模型证明 OLIA 算法是帕累托最优的
[38]	2014	Convergence time, Throughput	Balia 算法探索新的设计框架,在公平性和反应能力之间进行权衡
[39]	2017	Throughput, Spectral efficiency	此方案权衡吞吐量和频谱效率的关系,为每个链路分配合适的频谱资源,实现吞吐量和频谱效率之间的动态平衡,追求效益最大化
[40]	2016	Quality, Delay	一种质量驱动的 MPTCP 机制,其基于速率分配和前向纠错编码,在端到端时延和丢失性能之间取得平衡,最大限度的减少视频失真
[41]	2017	Throughput, RTT, Window size	Couple + 使得网络编码与多路径 TCP 的结合更加合理,使得编码子流可以及时发现拥塞,进行合理的负载迁移,同时保证友好性
[42]	2010	Throughput, Bandwidth	一种联合带宽聚合和分组调度的多路径传输机制,减少乱序数据包的产生,缓解乱序重排的影响,提高多路径传输的系统吞吐量
[43]	2017	Throughput, Loss	在最大发送速率、最大窗口和最小时间/空间三种调度策略之间选择,降低路径时延和损失,减少乱序现象的发生
[44]	2007	Throughput, Delay	一种动态的负载均衡机制,通过对数据包传输间距的限制避免发生数据包重排序,精准的实现多路径传输的流量分配
[45]	2017	Throughput, Delay	通过多路径间合适的流量分配策略,降低平均丢包率和端到端时延,减少乱序重排的发生,提升多路径算法的传输性能
[46]	2014	Delay, Cost	权衡时延性能和成本效率之间的关系,采用贪婪启发式算法实现子流之间的流量分配,提高了时延性能,同时降低了用户成本
[47]	2016	Delay	通过牺牲一部分带宽来换取较低的时延,其在多条路径上冗余的发送数据,避免了丢包引起的重传现象,从而降低了链路的整体时延,保证了数据传输的可靠性
[48]	2017	Energy, Throughput	EDAM 是一种能量权衡的 MPTCP 机制,其基于效用最大化理论,通过视频流量分配原则,在保证视频质量的同时,最小化移动设备的能量消耗
[49]	2018	Energy, Throughput	实现最佳的数据分配策略,通过子流间的流量转移最大限度地提高移动设备能效,同时保证吞吐量和网络的负载均衡

流之间的时延差异,提升用户感知速率,同时满足时延敏感业务的时延要求.

DBCCA 基于最小化子流间时延差异的优化问题,通过遗传算法找到多路径子流近优的速率分配原则,设定速率分配因子. 进而根据流体模型将速率分配原则作为 MPTCP 多路径拥塞控制的窗口调节准则,联合考虑吞吐量和时延目标,在拥塞避免阶段每收到一个 ACK 确认即可调节多路径子流拥塞窗口的大小. 最小化子流间时延差异的优化问题如

$$\text{minimize} \sum_{r \in \mathcal{R}} (|D_r - \bar{D}|) \tag{2}$$

其中: $\mathcal{R}$  为多路径子流; $D_r$  为子流  $r$  对应的时延,满足一定时延限制条件; $\bar{D}$  为子流时延均值. DBCCA 算法通过时延感知提升系统的整体吞吐量性能,满足 eMBB 场景的高速率、低时延要求.

4.2 能耗优化

随着移动互联网的快速发展,移动终端设备的数量也在逐年增长. 由于移动终端能量有限的特性,未来 TCP 的优化方向应该满足节能要求. 在未

来 5G 物联网环境下,同时会有海量传感器和智能设备连网,这要求物联网应具有低成本、低能耗、安全可靠等特点. mMTC 作为 5G 物联网深度应用的场景之一,通过改善连接密度和网络节能效率为物联网提供服务,支持海量设备以较低能耗接入网络. 节能需求与提升吞吐量目标本身是一对难以调和的矛盾,权衡能量与吞吐量之间的关系,实现移动设备的效用最大化是未来 TCP 优化的一大挑战.

相对单路径而言,TCP 的多路径传输由于同时需要维持多个传输接口,因此会有更多的能量消耗,移动设备终端受限的能量资源决定着多路径 TCP 连接的时长和可靠性. 能量资源的合理有效利用,是未来 TCP 优化急需解决的问题.

目前实现节能的方法主要是流量整形和流量转移. 流量整形是一种主动调整流量输出速率的措施,周期性发送流量,避免接收端一直处于活跃模式,从而实现减少能量消耗的目标. 流量转移是将流量从高能量消耗接口转移到低能量消耗接口,从

而使得数据的传输更加节能. 未来的 TCP 优化应不断协调能量与吞吐量之间的动态平衡, 当能量相对充足时, 可以适当消耗较多的能量以满足吞吐量目标. 而当能量极其有限时, 则需要降低吞吐量以保证能量资源的合理利用. 根据预定目标的不同, 自适应调节能量消耗与吞吐量之间的关系, 实现设备效用最大化, 这是 TCP 协议的优化目标.

### 4.3 基于网络架构优化

为提高系统的灵活性和效率, 并降低成本, 未来 5G 网络将引入新的技术, 出现新型网络架构. 5G 网络架构中对 MEC、SDN (software defined network) 和 NFV (network function virtualization) 的支持, 为 TCP 协议带来了新的挑战, 同时也为 TCP 优化指明了新的发展方向.

MEC 是一种基于移动通信网络的全新分布式计算方式, 通过使一定的网络服务和网络功能脱离核心网络, 实现节省成本, 降低时延等目标. MEC 通过实时监控空口传输质量, 通知 TCP 服务器实现跨层优化, TCP 服务器根据空口状态信息及时调整传输层发送机制. TCP 服务器也可以告知 MEC 无线侧应用的相关信息, 如应用数据优先级或应用特性信息等, MEC 根据应用数据信息特征, 调整无线协议策略. 基于 MEC 的 TCP 优化可以满足视频、游戏等业务对带宽和时延的要求, 保证终端获得更加极致的用户体验. 然而, MEC 的安全性以及移动业务的连续性保证等也是基于 MEC 优化 TCP 协议的技术挑战.

5G 网络通过引入 SDN 技术提高了数据传输效率, 实现更好的资源分配, 通过 NFV 技术的部署, 使得部分功能网元以虚拟功能网元的形式部署在云化的基础设施上, 网络功能由软件实现, 不再依赖于专有通信硬件平台. 基于 SDN 和 NFV 的 TCP 能够获得网络资源的全局信息, 并根据业务需求进行全局调配和优化, 实现应用与网络的无缝集成. SDN 和 NFV 的灵活性要求 TCP 协议应具备更加完善的动态自适应机制, 网络的自动化部署和运行维护对 TCP 协议提出了更高的标准.

## 5 结束语

传统 TCP 协议在 4G、5G 无线接入网络中的性能急剧恶化. 随着通信技术的快速发展, 未来网络中 TCP 协议的优化已经成为亟需解决的问题. 基于此, 国内外学者已经提出多种无线侧的 TCP 优化方

案. 笔者对近年来的解决方案进行了分类和比较, 总结了 TCP 优化的发展进程. 笔者认为, 很难建立一个满足所有要求的无线 TCP 机制. TCP 优化主要分为单路径和多路径 TCP 优化, 旨在实现高吞吐量和低时延的目标. 新的网络架构下, 结合 5G 网络的新技术、新场景, TCP 协议实现特定场景下特定目标的优化是未来的发展方向.

### 参考文献:

- [1] Cisco Incorporated. Cisco visual networking index: global mobile data traffic forecast update, 2015—2020 White Paper[EB/OL]. The United States: Cisco Inc, 2016.
- [2] 王志明, 曾孝平, 刘学, 等. 一种异构网络 TCP 拥塞控制算法[J]. 电子与信息学报, 2016, 38(4): 780-786.
- [3] Wang Z M, Zeng X P, Liu X, et al. TCP congestion control algorithm for heterogeneous networks[J]. Journal of Electronics and Information Technology, 2016, 38(4): 780-786.
- [4] Singh J P, Li Y, Bambos N, et al. TCP performance dynamics and link-layer adaptation based optimization methods for wireless networks[J]. IEEE Transactions on Wireless Communications, 2007, 6(5): 1864-1879.
- [5] 袁传博, 董育宁. 4G 网络中基于无线丢包感知的跨层拥塞控制方法[J]. 南京邮电大学学报(自然科学版), 2016, 36(2): 65-73.
- [6] Yuan C B, Dong Y N. Wireless-loss aware cross-layer congestion control algorithm in 4G networks[J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition), 2016, 36(2): 65-73.
- [7] IMT-2020. 5G 概念白皮书[EB/OL]. <http://www.imt-2020.org.cn/zh/documents/1>.
- [8] 张平, 陈昊. 面向 5G 的定位技术研究综述[J]. 北京邮电大学学报, 2018, 41(5): 1-12.
- [9] Zhang P, Chen H. A survey of positioning technology for 5G[J]. Journal of Beijing University of Posts and Telecommunications, 2018, 41(5): 1-12.
- [10] Li P, Fang Y. On the throughput capacity of heterogeneous wireless networks[J]. IEEE Transactions on Mobile Computing, 2012, 11(12): 2073-2086.
- [11] Wu R Q, Jie H, Ding N. An improved TCP congestion control algorithm of based on bandwidth estimation in heterogeneous networks[J]. Journal of Communications, 2014, 9(10): 792-797.
- [12] Cardwell N, Cheng Y, Gunn C S, et al. BBR: congestion-based congestion control[J]. Communications of the ACM, 2017, 60(2): 58-66.

- [10] Zhang H, Xue K, Hong P, et al. Congestion exposure enabled TCP with network coding for hybrid wired-wireless network [C] // 2014 23<sup>rd</sup> International Conference on Computer Communication and Networks (ICCCN). Shanghai: IEEE, 2014: 1-8.
- [11] Chang C W, Lin Y H, Ren Y, et al. Congestion control for machine-type communications in LTE-A networks [C] // 2016 IEEE Global Communications Conference (GLOBECOM). Washington: IEEE, 2016: 1-6.
- [12] Claeys M, Bouten N, De Vleeschauwer D, et al. Deadline-aware TCP congestion control for video streaming services [C] // 2016 12<sup>th</sup> International Conference on Network and Service Management (CNSM). Montreal: IEEE, 2016: 100-108.
- [13] Kim B H, Calin D. On the Split-TCP performance over real 4G LTE and 3G wireless networks [J]. IEEE Communications Magazine, 2017, 55(4): 124-131.
- [14] Bakre A, Badrinath B R. I-TCP: Indirect TCP for mobile hosts [C] // Proceedings of the 15<sup>th</sup> International Conference on Distributed Computing Systems. Vancouver: IEEE, 1995: 136-143.
- [15] Brown K, Singh S. M-TCP: TCP for mobile cellular networks [J]. ACM SIGCOMM Computer Communication Review, 1997, 27(5): 19-43.
- [16] Kim B H, Calin D, Lee I. Enhanced split TCP with end-to-end protocol semantics over wireless networks [C] // 2017 IEEE Wireless Communications and Networking Conference (WCNC). San Francisco: IEEE, 2017: 1-6.
- [17] Le F, Wong S H Y, Raghavendra R, et al. Removing TCP congestion control on the last hop in split TCP environments [C] // 2016 14<sup>th</sup> International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt). Tempe: IEEE, 2016: 1-8.
- [18] Ramakrishnan K, Floyd S, Black D. The addition of explicit congestion notification (ECN) to IP [R/OL]. (2001-09-15) [2018-11-01]. <https://www.rfc-editor.org/info/rfc3168>.
- [19] Alizadeh M, Greenberg A, Maltz D A, et al. Data center TCP (DCTCP) [J]. ACM SIGCOMM Computer Communication Review, 2011, 41(4): 63-74.
- [20] Jiang X, Jin G. CLTCP: An adaptive TCP congestion control algorithm based on congestion level [J]. IEEE Communications Letters, 2015, 19(8): 1307-1310.
- [21] Zhu X, Zheng R, Yang D, et al. Radio-aware TCP optimization in mobile network [C] // 2017 IEEE Wireless Communications and Networking Conference (WCNC). San Francisco: IEEE, 2017: 1-5.
- [22] Wu J, Yuen C, Chen J. Leveraging the delay-friendliness of TCP with FEC coding in real-time video communication [J]. IEEE Transactions on Communications, 2015, 63(10): 3584-3599.
- [23] Wu J, Cheng B, Wang M, et al. Priority-aware FEC coding for high-definition mobile video delivery using TCP [J]. IEEE Transactions on Mobile Computing, 2017, 16(4): 1090-1106.
- [24] European Telecommunications Standards Institute. Mobile-Edge Computing (MEC) Service Scenarios [EB/OL]. [2018-11-03]. [http://www.etsi.org/deliver/etsi\\_gs/MEC-IEG/001\\_099/004/01.01.01\\_60/gs\\_MEC-IEG004v010101p.pdf](http://www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/004/01.01.01_60/gs_MEC-IEG004v010101p.pdf).
- [25] Wu J, Yuen C, Cheng B, et al. Bandwidth-efficient multipath transport protocol for quality-guaranteed real-time video over heterogeneous wireless networks [J]. IEEE Transactions on Communications, 2016, 64(6): 2477-2493.
- [26] Raiciu C, Handley M, Wischik D. Coupled congestion control for multipath transport protocols [R/OL]. (2011-10-10) [2018-10-15]. <https://www.rfc-editor.org/info/rfc6356>.
- [27] Zhou D, Song W, Cheng Y. A study of fair bandwidth sharing with AIMD-based multipath congestion control [J]. IEEE Wireless Communications Letters, 2013, 2(3): 299-302.
- [28] Jacobson V. Congestion avoidance and control [J]. ACM SIGCOMM Computer Communication Review, 1988, 18(4): 314-329.
- [29] Honda M, Nishida Y, Eggert L, et al. Multipath congestion control for shared bottleneck [C] // Proc PFLD-NeT workshop. Tokyo: [s. n.], 2009: 19-24.
- [30] Wischik D, Raiciu C, Greenhalgh A, et al. Design, implementation and evaluation of congestion control for multipath TCP [C] // Proc Usenix NSDI, Boston: [s. n.], 2011: 99-112.
- [31] Wu J, Yuen C, Cheng B, et al. Goodput-aware load distribution for real-time traffic over multipath networks [J]. IEEE Transactions on Parallel and Distributed Systems, 2015, 26(8): 2286-2299.
- [32] Key P, Massoulié L, Towsley D. Path selection and multipath congestion control [J]. Communications of the ACM, 2011, 54(1): 109-116.
- [33] Ding Z, Wu D. Sliding-mode-based congestion control and scheduling for multiclass traffic over per-link queue-



- ing wireless networks[J]. IEEE Transactions on Vehicular Technology, 2013, 62(3): 1276-1288.
- [34] Bredel M, Bozakov Z, Barczyk A, et al. Flow-based load balancing in multipath layer-2 networks using OpenFlow and multipath-TCP[C] // Proceedings of the third workshop on Hot topics in software defined networking. Chicago: ACM, 2014: 213-214.
- [35] Alizadeh M, Edsall T, Dharmapurikar S, et al. CONGA: Distributed congestion-aware load balancing for datacenters[C] // ACM SIGCOMM Computer Communication Review. Chicago: ACM, 2014, 44(4): 503-514.
- [36] Zhao J, Xu C, Guan J, et al. A fluid model of multipath TCP algorithm: Fairness design with congestion balancing[C] // 2015 IEEE International Conference on Communications (ICC). London: IEEE, 2015: 6965-6970.
- [37] Khalili R, Gast N, Popovic M, et al. MPTCP is not pareto-optimal: performance issues and a possible solution[J]. IEEE/ACM Transactions on Networking, 2013, 21(5): 1651-1665.
- [38] Peng Q, Walid A, Hwang J, et al. Multipath TCP: analysis, design, and implementation[J]. IEEE/ACM Transactions on Networking, 2016, 24(1): 596-609.
- [39] Wang W, Wang X, Wang D. Bandwidth scheduling for multipath TCP based concurrent multipath transfer[C] // 2017 IEEE Wireless Communications and Networking Conference (WCNC). San Francisco: IEEE, 2017: 1-6.
- [40] Wu J, Yuen C, Cheng B, et al. Streaming high-quality mobile video with multipath TCP in heterogeneous wireless networks[J]. IEEE Transactions on Mobile Computing, 2016, 15(9): 2345-2361.
- [41] Xue K, Han J, Zhang H, et al. Migrating unfairness among subflows in MPTCP with network coding for wired-wireless networks[J]. IEEE Transactions on Vehicular Technology, 2017, 66(1): 798-809.
- [42] Tsai M F, Chilamkurti N, Park J H, et al. Multi-path transmission control scheme combining bandwidth aggregation and packet scheduling for real-time streaming in multi-path environment[J]. IET communications, 2010, 4(8): 937-945.
- [43] Kimura B Y L, Lima D C S F, Loureiro A A F. Alternative scheduling decisions for multipath TCP[J]. IEEE Communications Letters, 2017, 21(11): 2412-2415.
- [44] Kandula S, Katabi D, Sinha S, et al. Dynamic load balancing without packet reordering[J]. ACM SIGCOMM Computer Communication Review, 2007, 37(2): 51-62.
- [45] Delgado O, Labeau F. Delay-aware load balancing over multipath wireless networks[J]. IEEE Transactions on Vehicular Technology, 2017, 66(8): 7485-7494.
- [46] Park S Y, Joo C, Park Y, et al. Impact of traffic splitting on the delay performance of MPTCP[C] // 2014 IEEE International Conference on Communications (ICC). Sydney: IEEE, 2014: 1204-1209.
- [47] Frommgen A, Erbschäuer T, Buchmann A, et al. ReMP TCP: low latency multipath TCP[C] // 2016 IEEE International Conference on Communications (ICC). Kuala Lumpur: IEEE, 2016: 1-7.
- [48] Wu J, Cheng B, Wang M, et al. Quality-aware energy optimization in wireless video communication with multipath TCP[J]. IEEE/ACM Transactions on Networking, 2017, 25(5): 2701-2718.
- [49] Wang W, Wang X, Wang D. Energy efficient congestion control for multipath TCP in heterogeneous networks[J]. IEEE Access, 2018(6): 2889-2898.