



面向时延需求的主动队列管理方法

赵玉宇^{1,2}, 程光^{1,2*}, 李昊冬³, 张慰慈⁴

1. 东南大学网络空间安全学院, 南京 211189
2. 计算机网络与信息集成教育部重点实验室 (东南大学), 南京 211189
3. 东南大学苏州研究院, 苏州 215123
4. 东南大学计算机科学与工程学院, 南京 211189

* 通信作者. E-mail: gcheng@njnet.edu.cn

收稿日期: 2019-01-16; 修回日期: 2019-04-29; 接受日期: 2019-06-18; 网络出版日期: 2019-10-17

国家重点研发计划 (批准号: 2017YFB081703)、国家自然科学基金 (批准号: 61602114)、赛尔网络下一代互联网技术创新项目 (批准号: NGII20150108, NGII20170406) 和江苏省自然科学基金 (批准号: BK20151416) 资助项目

摘要 网络智能终端的普及和发展对网络总体的拥塞控制和以时延、抖动为主的服务质量提出了更高的要求和挑战. 现有的网络中间设备, 主要依靠队列管理控制时延, 解决拥塞. 针对现有主动队列管理算法易导致的高时延和网络资源分配不公下的大量丢包问题, 本文提出了面向时延需求的主动队列管理 (time demand-active queue management, TD-AQM) 方法. 该方法将队列管理的基础从端系统和中间设备得到的网络测量反馈回归到写入数据报文 IP 层 option 字段的时延需求, 即报文在路由器最长存在的时间上. 为了避免洪峰效应, TD-AQM 利用限制向下查找范围和遏制格子虚占用命中率方法, 将数据包按照时延需求插入队列. 实验结果表明, TD-AQM 能够有效地维持队列的稳定性, 保证数据报文按照时延需求出队并相较现有队列管理算法提高了吞吐率等各项性能.

关键词 主动队列管理, 时延需求, 洪峰效应, 队列稳定性, 低时延

1 引言

中国互联网络信息中心 (CNNIC) 在 2018 年第 42 次“中国互联网发展状况统计报告”中指出, 以网络直播为主的各类应用年增长率最高, 体现了现在国际国内智能终端的普遍发展. 这种现象对网络总体的拥塞控制施加了很大的压力, 给予了更多的挑战. 视频和将来必定会普及的 VR (virtual reality, 虚拟现实) 技术除了对带宽的要求之外, 还对以延迟、抖动为主的服务质量有更高的要求. 所以, 仅依靠端系统源节点降低发送窗口等拥塞避免方式已经于事无补. 由于全局网络概貌描绘的困难性高, 从网络中间设备上进行针对性的队列管理, 尤其是在路由器上谋求高缓存和高吞吐, 有效降低网络的整体时延是现在的主流研究趋势.

引用格式: 赵玉宇, 程光, 李昊冬, 等. 面向时延需求的主动队列管理方法. 中国科学: 信息科学, 2019, 49: 1321–1332, doi: 10.1360/N112019-00010
Zhao Y Y, Cheng G, Li H D, et al. Active queue management algorithm for time delay demand (in Chinese). Sci Sin Inform, 2019, 49: 1321–1332, doi: 10.1360/N112019-00010

但是这种方法如果只关注路由器状态, 则降低了数据报本身的重视程度. 现有队列管理方法的丢弃策略重点依赖的网络测度大多以队列的状态为主, 虽然宏观上解决了突发流量带来的高丢包率以及网络拥塞状况, 但是对报文不加以区分的丢包对部分响应类应用的 QoS (quality of service) 造成了严重的影响. 其次, 作为分组交换网的核心, 路由器必须携带一个 Buffer 用来平滑泊松分布的包到达和固定速率的包发送之间的关系, 这对如何利用路由器的 Buffer 也提出了很高的要求. 所以, 现有方法 Buffer-bloat 问题严重^[1]. Buffer-bloat 带来的高延迟使得 RTT (round-trip time) 增加, 必定使得网络吞吐率下降, 网络状态变差, 无法满足现有网络应用如互动直播和 VoIP (voice over Internet protocol) 等对稳定性和高速性的要求. 最后, 路由器需要平衡 UDP (user datagram protocol) 和 TCP (transmission control protocol) 两个协议之间的链路瓶颈的公平共享性^[2]. 由于两个协议设计的目的相似, 只是依赖队列状态进行队列管理使得基于 UDP 的众多实时应用的数据流会影响基于 TCP 协议的响应流, 导致 TCP 流的性能受到干扰. 特别的, 在瓶颈链路资源中, 这种干扰还会直接导致实施保守策略数据流的用户无法完成任务, 但是实施激进策略的用户迅速地挤占链路资源, 完成任务后带宽闲置, 造成浪费.

缓解上述问题的根本思想是重点利用报本身的信息, 辅以队列状态进行主动队列管理, 控制链路的总体时延. 本文将队列管理的焦点聚集在数据报本身, 提出了一种面向时延需求的主动队列管理方法 (time demand AQM, TD-AQM). 该方法在网络层的 option 上给每个数据报打入时延需求, 即是报文存在于单一路由器中的最长时间, 并将其作为路由器内主动队列管理的依据. 由于这种方法的构建颠覆了之前的队列管理架构, 可能会引发时延需求与 Buffer 资源不协调, 重要数据分组在资源占用程度上不匹配的“洪峰效应”. TD-AQM 方法利用多种入队策略结合时延需求紧迫程度遏制了洪峰效应, 通过时延需求规避了不同策略的流所带来的资源浪费, 保障了数据分组的低时延性.

2 相关研究

队列管理方法在数据传输上有着重要的作用, 它能够有效地提高数据传输性能并且尽可能保证各个流之间的公平性和可靠性. 为了剔除传统的队列管理机制中的满队列、缓存死锁和全局同步问题, 早期的主动队列管理方法便由被动队列管理方法进化而来. 以队列长度为指标的 RED^[3], REM^[4], AVQ^[5] 等方法有效地解决了被动队列管理中的满队列问题. 这些方法增加了两种新的机制, 首先是预测性的随机丢弃部分新到达的分组, 有效地避免了发生拥塞的可能. 其次是使用平均队列长度作为依据, 计算出丢弃概率的值并调整分组丢弃的概率, 避免了短暂的突发数据流量造成的影响. 近年来提出的 CARED, HRED, CHOKe^[6] 也都是像上述方法一样, 希望其队列长度收敛在阈值以内.

以音、视频为首的多种应用造成了网络中的数据流量的增加, 网络的时延成为了一项重要的测度, 上述策略的弊端逐渐显现. 为了使端到端的延迟下降, 并同步提高网络的吞吐量, BLUE 算法被提出, 它是一种以链路空闲和缓冲区溢出事件作为拥塞度量的队列管理方法^[7]. 为了提高算法的稳定性, 克服网络拥塞反应迟缓等缺点, 避免突发数据流量到来时造成的网络性能下降, Alsaaidah 等^[8] 提出了 Gentle-BLUE 算法, 其使用两个参数, 将队列长度作为依据, 在拥塞发生前动态地调整丢弃概率.

为了更好地满足用户对实时业务流的需求, 面向时延测度继续优化, Nichols 等^[9] 提出了 CoDel 算法. CoDel 算法无需设置参数, 对于算法中参数的设置不敏感, 只是聚焦了队列的发送时延, 在突发数据流量到来时, 能保持较低的时延, 满足用户的需求. 随着用户数量的增大, 对于运行稳定的网络有了更高的需求^[10]. Zhu 等^[11] 提出了一种基于主动队列管理中平均队列长度差距的 TCP 拥塞控制方法, 方法定义了队列长度差距的不同状态, 利用其离散熵, 将队列管理转化为 Wiener 过程. Bisoy

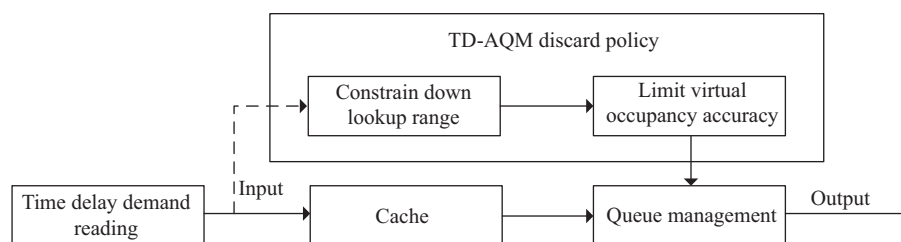


图 1 TD-AQM 的整体框架

Figure 1 Frame of TD-AQM

等^[12]设计出了名叫 SAQM (stable AQM) 的稳定的主动队列管理方法, SAQM 基于传输速率和往返 RTT 控制 TCP 连接中的目标队列长度, 提高了 Internet 路由器队列长度的稳定性. Iskandar 认为利用主动队列管理方法可以降低网络的拥塞, 而网络拥塞发生时都出现了分组到达比输出链路容量大的情况, 所以作者改进 CoDel 方法中控制时延方式, 利用 buffer-brutt 进行实时管理^[13].

无论是基于对丢弃概率参数的改变还是基于 RTT 的改变, 现有方法都在努力保证网络的通畅. 问题在于, 上述各种方法中, 主要针对流的各项测量属性如长度、阈值、速率等作为判断队列是否丢包的依据, 忽略了数据报文本身的重要程度. 在多种网络业务并存的情况下, 严重地影响了用户的使用体验. 本文以时延需求作为切入点, 将调整拥塞控制的依赖参数回归到数据包本身, 利用数据报文的时延需求作为队列插入和丢弃概率的依据. 方法在降低算法复杂度的同时, 更加易于在普通网络或工业控制、医用手术远程操作等具体情境的网络体系结构下的实际部署.

3 TD-AQM 的设计与实现

利用数据包本身的属性进行队列管理是本文提出方法的宗旨. 由于整体网络环境复杂, 拓扑结构未知, 控制数据报文在链路中的总体时延只能通过控制其通过每个路由器的时间或者调整其优先级来确定^[14]. 然而, 按照报文优先级的方式进行调整只能提高数据报文之间相对的发送时间, 并不能在宏观上进行时延的控制. 方法使用的时延需求 (time demand) 是指报文在一个路由器中最长存在的时间. 通过控制报文在单个路由器中的时间来达到整体的时延控制是本文的核心思想.

3.1 TD-AQM 的主要框架

TD-AQM 的主要架构包括预处理、丢弃、数据分组缓存入队和队列管理等模块. 数据包按照时延需求查找或计算出预估入队位置, 然后通过丢弃策略判断是否丢弃或缓存入队. TD-AQM 的丢弃策略设计思想是满足时延需求的同时抑制洪峰效应, 其主要包括限制向下查找范围和遏制格子虚占用命中率两个机制. 整体框架如图 1 所示.

3.2 时延需求

时延需求是数据报文能够在路由器中存在的最长时间. 这个需求是从应用层人为的或者自适应的分配给不同的流. 每条流在确定了自身数据包的时延需求之后, 利用用户态和内核态交互, 将需求打在每个报文的固定位置, 即是 IP 层的 option 选项字段中提供给路由器内核进行识别、提取以及计算等操作.

时延需求是接下来进行队列操作的基础. 其描述了报文的紧迫程度, 提供给 TD-AQM 安排先后顺序, 逐步插入队列. 时延需求正确写入后, 对路由器来说每个报文都拥有了其自身期望的入队位置.

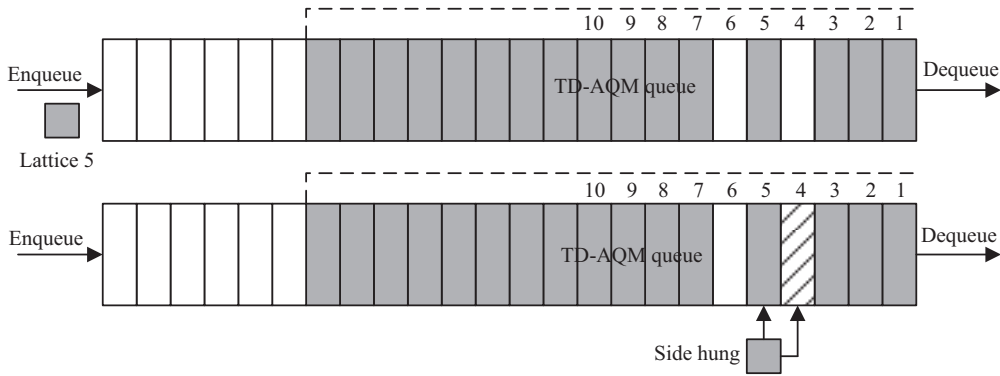


图 2 队列构造
Figure 2 Queue structure

数据包也便可以依据路由器当前状态被判断丢弃或者入队。

3.3 队列构造

众多队列构造的基本思想是将一维的队列二维化^[15]。路由器内队列可以理解为一个缓冲区, TD-AQM 将缓冲区逻辑地分为带有标志位的格子。TD-AQM 队列构造的主要思想是在逻辑队列划分的格子内部, 将时延需求相同的数据包进行再次排队, 保证公平性。为了更好地利用队列, 尽快将时延需求紧迫的数据包转发进入网卡, 需要对队列的每个格子分配不同的标志位。对于队列的某一个格子, 如果没有数据包和任何的标志位, 则格子状态为空; 如果里面已经有了一个数据包, 则这个格子的状态是占用, 而其他数据包经过计算后 TD-AQM 认为也应当放入这个格子中, 则将这个数据包插入至第一个数据包下面, 这种行为称之为侧挂。由于队列申请的存储空间是一定的, 侧挂的条件是向队列下游查找有空格子, 对这个空格子进行虚占用操作, 如图 2 所示。

图 2 中, 灰色的格子是被占用的格子, 代表了当前时间片内此格子有数据包; 白色的格子为空格子, 代表了当前时间片内此格子没有数据包。以同样需要占用 5 号格子的数据包入队为例, 由于 5 号格子被占用, 方法向下搜索到 4 号空格子, 并虚占用 4 号格子 (黑色斜线方块)。虚占用和占用这两个状态会随着格子内数据包的发送而改变, 这两种状态的存在也逻辑地将路由器队列二维化, 提高发送效率。

3.4 TD-AQM 的实现

3.4.1 时延需求的解析

数据包入队的依据是时延需求这一参数。在 TD-AQM 提取出数据包的时延需求后, 利用搭载方法的设备 (一般是路由器) 发送数据包的平均耗时和时延需求计算出其应该插入的队列位置。首先, 根据端口出口速率发送计算单个包的平均耗时 t , 则第 1 个格子的时延为 t , 第 n 个格子的时延为 $Delay(n) = n \times t$ 。然后, 当算法解析到某数据包的时延需求为 $TimeDemand$ 时, 计算其预估入队位置 $pre_position$ 为

$$pre_position = \frac{TimeDemand}{t} \tag{1}$$

预估入队位置 $pre_position$ 并不是数据包最终插入的队列位置。由于其希望插入的队列位置在当前时间片可能已经被占用或者被虚占用, 所以 TD-AQM 继续根据设置的丢弃策略判断是否将此数据

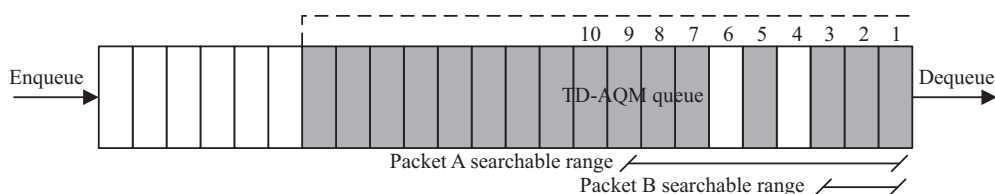


图 3 洪峰效应示例

Figure 3 Example of flood peak effect

包丢弃或者准许其入队。

3.4.2 TD-AQM 的丢弃策略

TD-AQM 的设计初衷是将目光聚焦到数据包, 根据时延需求, 在其入队时就希望按照紧迫程度安排数据包的发送顺序, 从而尽可能地满足各种包的时延需求. 然而实践中如果有数据包的时延需求紧迫度较小, 其预估入队位置靠后, 插入队列时可以向下游查找空格子进行虚占用. 而某一数据包紧迫程度较高, 其预估入队位置靠前, 由于队列申请缓存空间一定, 该数据包却无法向上游查找. 再加上现行网络中的各项拥塞控制机制, 这些需求紧迫的包反而容易受限制, 最终被饿死, 导致队列中没有位置可以供其插入从而被丢弃. 这种现象称为洪峰效应^[16].

如图 3 所示, 若 A 包计算得到其预估入队位置为 10 号格子, B 包计算得到其预估入队位置为 4 号格子. 若当其预估入队位置当前时间片被占用或被虚占用, 其能够向下查找的范围如图 3 所示, A 包可查找的范围显然大于 B 包. 为了尽可能规避这种现象的发生, TD-AQM 使用了限制向下查找范围和遏制格子虚占用命中率两种方法对数据包的虚占用操作进行控制, 并通过这一步对数据包进行选择性的丢弃.

为解决洪峰效应, TD-AQM 首先对数据包允许向下查找的范围进行限制. 时延需求紧迫的数据包允许向下查找的范围大, 时延需求不紧迫, 预估入队位置靠后的数据包允许向下查找的范围小.

TD-AQM 在得到数据包的 TimeDemand 和路由器的单包发送平均时延 t 后, 计算得到此时延需求的数据包的预估入队位置 $pre_position$ 和向下查找的极值位置 $last_position$, 并将两个参数传回. 为了满足时延需求紧迫程度与向下查找范围的正相关性, 本算法依据时延需求计算得到的预估入队位置对不同的数据包采用了分治策略. 通过对不使能抑制洪峰效应算法的实验分析, 发现在正常带宽情况下, 序号为 100 及其以下的格子标志位较为密集, 即基本上处于占用和虚占用状态, 认定这一范围的时延需求比较紧迫, 需要给予足够的资源. 所以当预估入队位置小于 100 时, 可直接查找至最下游; 当预估入队位置大于 100 时, 借鉴 BLUE 等经典方法的思想, 设置每个数据包都被允许向下查找的基量 α 和与时延需求紧迫程度正相关的增量 β 来定义数据包能够向下查找的最终范围. 其中基量 α 设置为 50. 增量 β 则利用反比函数得出, 其计算公式如下:

$$\beta = \frac{1}{1 + (pre_position - 100)}. \quad (2)$$

从式 (2) 中可以看出, 当预估入队位置过大后数据包能够向下查找的范围收敛到只有 50 个格子. 限制向下查找范围的算法如算法 1 所示.

遏制格子虚占用命中率算法主要是为了控制数据包向下游查找并虚占用空格子的成功率. 为了能够有效地控制洪峰效应, 每次数据包向下查找到可占用格子并占用此格子的成功率会被遏制. 设计这个概率遏制方法的指导思想是依据路由器的状态和时延需求两方面的综合评价, 调高或者降低数据包

Algorithm 1 Constrain down lookup range**Input:** Time delay demand TimeDemand, Average delay of single packet t ;**Output:** Extreme searchable downward position last_position;

```

1: last_position  $\leftarrow$  1;
2: if TimeDemand = 0 then
3:   pre_position  $\leftarrow$  1;
4:   last_position  $\leftarrow$  1;
5: else
6:   pre_position =  $\frac{\text{TimeDemand}}{t}$ ;
7: end if
8: if pre_position  $\leq$  100 then
9:   last_position  $\leftarrow$  1;
10: end if
11: if pre_position > 100 then
12:   last_position  $\leftarrow$  50 + 50  $\times$   $\frac{1}{1+(\text{pre\_position}-100)}$ ;
13: end if

```

虚占用某个格子的成功率.

一个数据包能够占用的格子范围为 pre_position 号格子到 last_position 号格子. 在这个范围区间内的第 n 号格子被此数据包虚占用的概率为

$$P_{(n)}^{\text{occupy}} = \frac{1}{\text{pre_position} - n} \times \log_{q_i} q_{\text{avg}}, \quad (3)$$

其中 q_{avg} 是路由器的平均队列长度, q_i 为瞬时队列长度. 参数 $\log_{q_i} q_{\text{avg}}$ 随着队列长度 q_i 的变化而变化. 如果瞬时队列长度大于平均队列长度时, 代表排队情况严重, 必须使得丢包的概率变大, 即虚占用概率变小; 如果瞬时队列长度小于平均队列长度, 证明空格子较多, 队列资源相对富余, 则调高格子的虚占用命中率. 具体如算法 2 所示.

Algorithm 2 Limit virtual occupancy accuracy**Input:** Estimate enqueue position pre_position, extreme searchable downward position last_position, lattice n , instantaneous queue length q_i , average queue length q_{avg} ;**Output:** Lattice n hit rate P ;

```

1:  $P \leftarrow$  1;
2: if  $n >$  pre_position &&  $n <$  last_position then
3:    $P \leftarrow$  0;
4: end if
5: if pre_position > 100 then
6:    $P = \frac{1}{\text{pre\_position}-n} \times \log_{q_i} q_{\text{avg}}$ ;
7: end if

```

通过上述算法 2, TD-AQM 控制了每一个数据包在可查找范围内虚占用格子的命中率, 使得时延需求不紧迫的数据包不占用, 或少占用时延需求紧迫的数据包的格子资源, 遏制了洪峰效应, 保证了方法的可靠性.

TD-AQM 为了提高算法效率, 将常用的时延需求统计出来, 分别计算其预估入队位置, 向下查找极值位置并存建表存储. 这样进行丢弃判断时可直接查表, 降低了算法的时间复杂度.

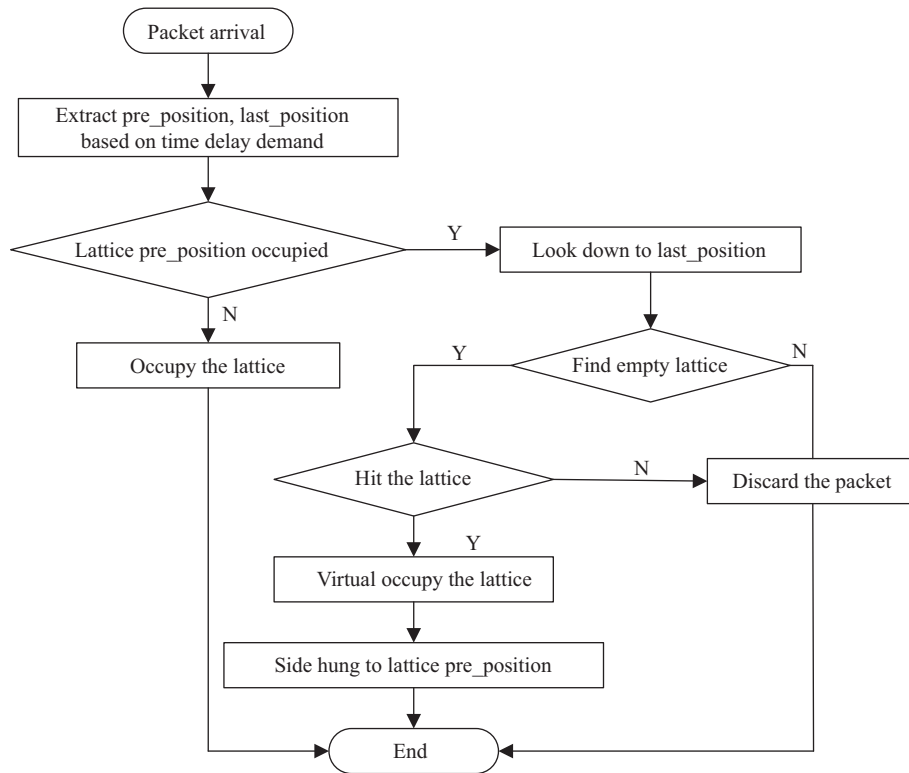


图 4 TD-AQM 的队列管理策略

Figure 4 Queue management policy of TD-AQM

3.4.3 TD-AQM 入队流程

在经过了时延需求提取、预估入队位置解析、限制向下查找范围和遏制格子虚占用命中率这几个步骤之后, 数据包能否入队已经确定. 经过上述分析, 能够入队的数据包大多数处于侧挂状态. 侧挂是在格子内部, 将时延需求一致的数据包按照先入先出顺序进行排队, 并按照这个队列发送至网卡.

TD-AQM 有效地防范了洪峰效应, 并利用了“占用”, “虚占用”两个标志位以及侧挂机制逻辑上将数据报文在缓存队列中按照时延需求排队, 其入队流程如图 4 所示.

图 4 中, 在向下查找到可以虚占用的队列格子时, TD-AQM 首先在此格子上打入虚占用标志位, 然后将数据包侧挂到预估入队位置格子里的数据包下. 在时间片抵达此格子时, 按照格子内部的数据包位置进行发送, 完成出队. 出队过程中, 已出队格子的虚占用、占用标志位随时间片的结束而撤销.

4 实验与分析

为了验证本文提出方法的有效性和可靠性, 本文搭建了真实实验环境对 TD-AQM 进行验证. 从队列稳定性、吞吐率、时延满足度和公平性 4 个角度在普遍被认可的 Dumbbell 拓扑中进行实验, 将 TD-AQM 与 RED, CoDel 和 CHOCe 算法进行了一系列的对比.

4.1 测量指标

为了设计可靠高效的面向时延需求的主动队列管理方法, TD-AQM 综合考虑了吞吐率、公平性、

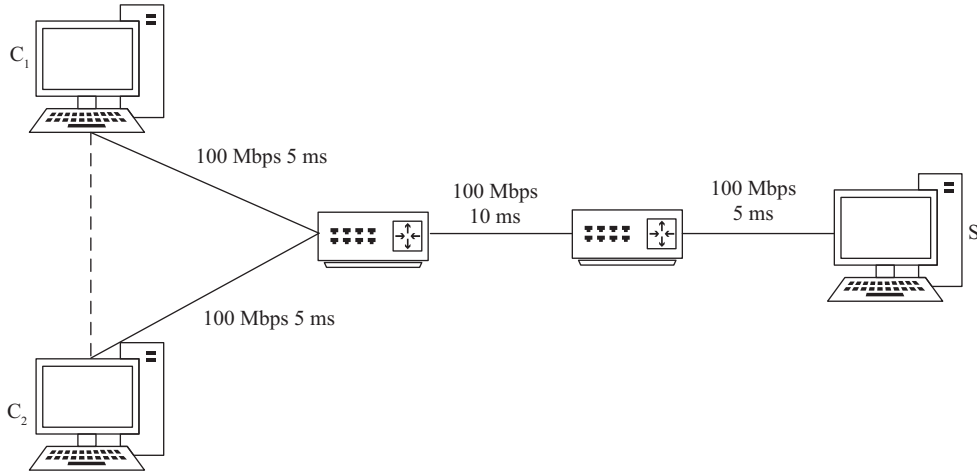


图 5 实验拓扑

Figure 5 Experimental topology

队列稳定性和时延满足度 (time demand satisfaction, TDS) 4 个评价指标. 其中, 吞吐量采用端系统测量的接收速率.

队列稳定性是测量队列的长度是否稳定在一定范围内的一个参数. 一般的, 稳定性由队列长度的标准差 q_{std} 表示^[17], 其定义如下:

$$q_{std} = \sqrt{\frac{1}{N} \sum_{i=1}^N (q_i - q_{avg})^2}, \quad (4)$$

其中 q_{avg} 是路由器的平均队列长度, q_i 为瞬时队列长度. 标准差所代表的队列稳定性也从一定程度上衡量了算法的效率, 标准差越低, 稳定性越高, 效率越高.

为了易于实际部署, 方法也考虑了公平性这一指标. 公平性可以采用 JFI (Jain's fair index)^[18] 进行度量, 其定义如下:

$$JFI = \frac{(\sum_{i=1}^N x_i)^2}{N \sum_{i=1}^N x_i^2}, \quad (5)$$

其中 x_i 代表流 i 的吞吐量, N 表示流的数量. JFI 的取值范围为 $[\frac{1}{N}, 1]$, JFI 值越大, 算法公平性越优良.

由于 TD-AQM 的入队性质, 只有满足时延需求的数据包可以插入队列, 所以其 TDS 表示如下:

$$TDS = \frac{\text{packet_sum} - \text{packet_loss}}{\text{packet_sum}}, \quad (6)$$

其中 packet_sum 和 packet_loss 分别代表被测试流的发送包数量和丢包数量.

4.2 实验环境

为了验证 TD-AQM 在瓶颈链路拓扑中的性能, 选取 Dumbbell 拓扑完成 TD-AQM 和其他算法进行对比实验, 实验拓扑如图 5 所示.

图 5 中设置了两条路径, 利用二打一的方式在路由器内部形成拥塞点. 将拓扑中所有的网卡调至 100 Mbps 的全双工模式. 除加载 TD-AQM 模块的路由器和作为交换机的路由器之间的链路外, 其

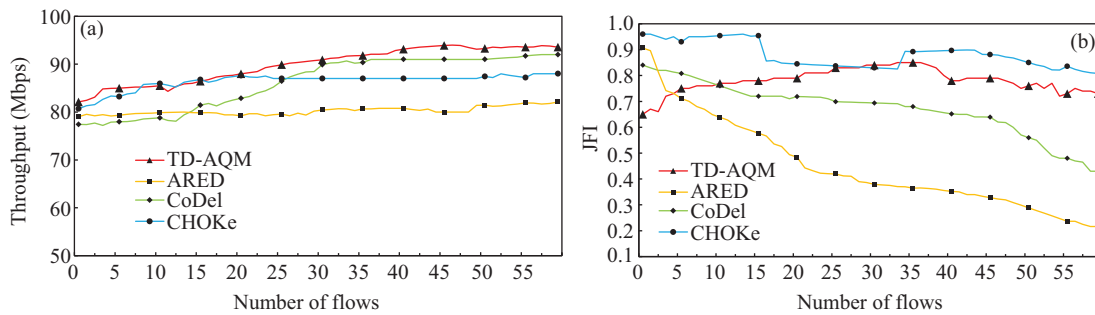


图 6 (网络版彩图) 性能对比. (a) 吞吐率; (b) 公平性

Figure 6 (Color online) Performance comparison. (a) Throughput rate; (b) fairness

余链路利用 TC (traffic control) 设置时延为 5 ms, 路由器之间链路时延为 10 ms. 由于设置的 BDP (bandwidth-delay product) 可以满足单个发送端到路由器, 即 $C_1 \rightarrow S$ 和 $C_2 \rightarrow S$ 能够分别加载至 50 条流, 其数量符合现有流行队列管理方法实验验证时的流数量要求.

实验环境中发送端设置 socket 流多线程与接收端进行交互并完成流量测试. 实验利用可以多路复用的接收模块配合内核探针测量丢包率, 利用接收端加载吞吐率测量算法, 完成对设置测度的精确测量, 结果输出给用户态.

为了能够有效地模拟出真实使用时的时延需求, 实验与其他队列管理算法对比时, 时延需求满足标准正态分布. 对比时延满足度这一特殊测度时, 实验采用标准正态分布, 均匀分布两种概率模型分配多流情况下的时延需求, 并引入无时延需求的流进行对比实验对照.

4.3 实验结果分析

实验首先将流的数量从 0 增加至 60 条, 以便观察 TD-AQM 算法的各项性能. 图 6 是吞吐率和公平性两种测度的实验对比结果.

图 6 中较粗标红曲线为 TD-AQM 在测试中的实验结果曲线. 从图中可以明显看出, 当流的数量从 0 增加到 60 的时候, TD-AQM 在吞吐率和公平性表现中能够优于现有主流的主动队列管理方法. TD-AQM 在流数较低的时候, 由于其时延需求分布不均, 导致格子占用较为单一, 所以吞吐率较低, 丢包率略高. 随着流的数量增加, 时延需求分布更加合理, 队列管理效率提升, 最终能够保持在 94.5 Mbps 左右, 其吞吐率均高于 CHOCe, RED 以及 CoDel 算法. 由于 TD-AQM 算法在设计之初针对公平性的依据是时延需求, 所以能够看出 TD-AQM 的公平性表现比较平稳. 不同于 RED 和 CoDel 算法, 流的数量对其影响不大, JFI 一直稳定在 0.8 左右, 其公平性与 CHOCe 算法相当.

为了更完整地展现算法的队列稳定性, 实验在测试 TD-AQM 时将流的数量从 0 增加至 80 条, 其他方法依然从 0 增加至 60 条, 观察队列稳定性测度的实验结果. 结果如图 7 所示. 在队列稳定性方面, TD-AQM 与其他算法不同, 有一个明显的收敛过程. 原因是 TD-AQM 作为面向时延需求这一特定 QoS 的主动队列管理方法, 在时延需求过于单一, 即流数量较少的时候反而会使得其丢弃策略效果差. 所以, 在流数小于 30 的时候 TD-AQM 的队列稳定性较差. 随着流数量的增加, 时延需求的分布开始离散化, 丢弃策略被充分地利用, 算法稳定性开始提高. 在流数量增加到 30 左右时开始与其他算法队列稳定性相当, 流数量在 40~65 条的阶段稳定性维持在 5 左右, 之后随着流数量的继续增加其稳定性逐步降低.

为了体现时延需求被满足的程度, 本文设置了多流并行实验, 且多流的时延需求为标准正态分布

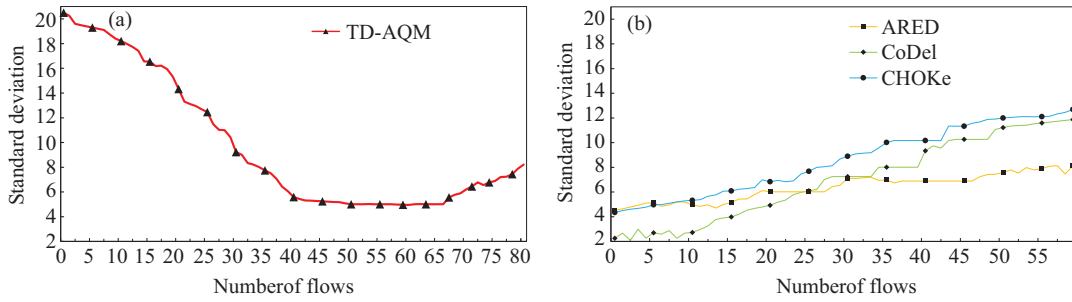


图 7 (网络版彩图) 队列稳定性对比. (a) TD-AQM; (b) 其他方法
 Figure 7 (Color online) Queue stability comparison. (a) TD-AQM; (b) other algorithms

表 1 时延满足度对照

Table 1 Comparison of time demand satisfaction

Number of flows	Standard normal distribution	Uniform distribution	No time delay demand
20	0.81	0.82	0.91
40	0.84	0.81	0.85
60	0.90	0.78	0.82
100	0.86	0.77	0.81

和均匀分布, 以便测量方法的 TDS. 对比实验流删除了时延需求这一字段, 其余参数保持不变, 对其丢包率进行测量. 本文计算其丢包率与 1 的差值的绝对值作为其 TDS. 实验结果如表 1 所示.

没有时延需求的流, 其丢包率随着流数的增加而增加. 在时延需求服从均匀分布的情况下, 随着流数的增加, 时延满足度略微下降, 丢包增加. 在时延需求服从标准正态分布的情况下, TD-AQM 表现最优, 能够将时延满足度稳定在 0.9 左右. 较高的时延满足度代表了 TD-AQM 的低丢包率, 也证明 TD-AQM 能够规避洪峰效应的影响, 保障数据包按照时延需求出队.

5 结论

本文针对现有主动队列管理方法中的队列构造以及报文丢弃策略等做出了改进. 现有的大多数主动队列管理方法只从队列长度、RTT 等一般网络测度决定是否将数据包不加区分地按照一定的概率丢弃. 本文提出的 TD-AQM 方法把数据包的时延需求, 即其在路由器内存在的最长时间作为丢弃数据包依据. 通过构建路由器的多标志位队列和限制向下查找, 遏制格子虚占用命中率两种方法, TD-AQM 能够在控制洪峰效应的前提下保证大多数的数据包按照其时延需求出队并转发至网卡. 本文与现有主流的主动队列管理方法进行对比, 通过真实网络拓扑实验, 验证了提出方法的正确性和有效性. 下一步的工作主要是在网络中心等更复杂的场景中进行部署和测试, 调整部分参数, 优化 TD-AQM 的性能.

参考文献

- 1 Grazia C A, Patriciello N, Klapez M, et al. Mitigating congestion and bufferbloat on satellite networks through a rate-based AQM. In: Proceedings of the IEEE International Conference on Communications, Paris, 2017. 1-6
- 2 Casoni M, Grazia C A, Klapez M, et al. How to avoid TCP congestion without dropping packets: an effective AQM called PINK. Comput Commun, 2017, 103: 49-60

- 3 Patel S. Performance analysis on RED for stabilized queue. In: Proceedings of the IEEE 7th International Conference on Contemporary Computing, Noida, 2014. 306–311
- 4 Patel S. Performance analysis and modeling of congestion control algorithms based on active queue management. In: Proceedings of the IEEE International Conference on Signal Processing and Communication, Vancouver, 2013. 449–454
- 5 Jain T, Basava A, Tahiliani M, et al. Performance evaluation of CoDel for active queue management in wired-cum-wireless networks. In: Proceedings of the IEEE 4th International Conference on Advanced Computing Communication Technologies, Rohtak, 2014. 381–385
- 6 Tahiliani M P, Shet K C, Basavaraju T G. CARED: cautious adaptive RED gateways for TCP/IP networks. *J Netw Comput Appl*, 2012, 35: 857–864
- 7 Liu W Y, Liu B, Zou X L. Congestion control algorithm based on dynamic threshold. *Appl Res Comput*, 2013, 30: 3459–3461
- 8 Alsaaidah A, Zalisham M, Fadzli M, et al. Gentle-BLUE: a new method for active queue management. In: Proceedings of the International Conference on Advanced Computer Science Applications and Technologies, Amman, 2015. 67–72
- 9 Nichols K, Jacobson V. Controlling queue delay. *Commun ACM*, 2012, 55: 42–50
- 10 Tian Z H, Yu X Z, Zhang H L, et al. A real-time network intrusion forensics method based on evidence reasoning network. *Chin J Comput*, 2014, 37: 1184–1194 [田志宏, 余翔湛, 张宏莉, 等. 基于证据推理网络的实时网络入侵取证方法. *计算机学报*, 2014, 37: 1184–1194]
- 11 Zhu J, Luo T, Yang L, et al. An average queue-length-difference-based congestion detection algorithm in TCP/AQM network. *Int J Adapt Control Signal Process*, 2018, 32: 742–752
- 12 Bisoy S K, Pattnaik P K, Pati B, et al. Design and analysis of a stable AQM controller for network congestion control. *Int J Commun Netw Distrib Syst*, 2018, 20: 143–167
- 13 Iskandar M N. Active queue management (AQM) performance analysis based on controlled delay (CoDel) against bufferbloat on real-time application. *Procedia Comput Sci*, 2017, 2: 119
- 14 Sheikhan M, Shahnazi R, Hemmati E. Adaptive active queue management controller for TCP communication networks using PSO-RBF models. *Neural Comput Applic*, 2013, 22: 933–945
- 15 Okokpujie K, Chukwu E C, Noma-Osaghae E, et al. Novel active queue management scheme for routers in wireless networks. *Int J Commun Antenna Propag*, 2018, 8: 52
- 16 Hamidian H, Beheshti M T H. A robust fractional-order PID controller design based on active queue management for TCP network. *Int J Syst Sci*, 2018, 49: 211–216
- 17 Tahiliani M P, Shet K C. Analysis of cautious adaptive RED (CARED). In: Proceedings of the International Conference on Advances in Computing, Communications and Informatics, Mysore, 2013. 1029–1034
- 18 Mühlenthaler M, Wanka R. Fairness in academic course timetabling. *Ann Oper Res*, 2016, 239: 171–188

Active queue management algorithm for time delay demand

Yuyu ZHAO^{1,2}, Guang CHENG^{1,2*}, Haodong LI³ & Weici ZHANG⁴

1. School of Cyber Science and Engineering, Southeast University, Nanjing 211189, China;

2. China Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, Nanjing 211189, China;

3. Suzhou Research Institute (Southeast University), Suzhou 215123, China;

4. School of Computer Science and Engineering, Southeast University, Nanjing 211189, China

* Corresponding author. E-mail: gcheng@njnet.edu.cn

Abstract With the popularization and development of network intelligent terminals, additional requirements and challenges have arisen concerning network congestion control and the quality of service based on delay and jitter. Existing network intermediate devices primarily rely on active queue management to control the delay and solve congestion. This paper proposes an active queue management algorithm focusing on time delay demands to solve the large number of packet loss problems in existing active queue management algorithms. Packet loss problems are caused by long delays and unequal network resource allocation. In the TD-AQM algorithm, the network measurement feedback is obtained from the end system and intermediate equipment is returned to the data packet as the basis for queue management. The delay requirement is written in the option field of the IP layer of the data packet, i.e., the longest time of the message in the router. To avoid the flood peak effect, TD-AQM constrains the down lookup range and limits the occupancy accuracy of the lattice. The queue location of packets is determined according to the delay requirements. The experimental results show that TD-AQM can effectively maintain the stability of the queue, ensure that data packets are queued according to the delay requirements, and guarantee an overall low latency compared to other algorithms.

Keywords active queue management, time delay demand, peak effect, queue stability, low delay



Guang CHENG was born in 1973. He received the Ph.D. degree from Southeast University in 2003. He is a professor and Ph.D. supervisor at Southeast University. His research interests include cyberspace security monitoring and protection and network traffic big data analysis.



Yuyu ZHAO was born in 1994. He is currently pursuing his Ph.D. degree in the School of Cyber Science and Engineering at Southeast University, China. His main research interests include cyber security and TCP/IP.