

数据包队列管理实验报告

张磊 2017K8009922027

一、实验题目

数据包队列管理实验

二、实验内容

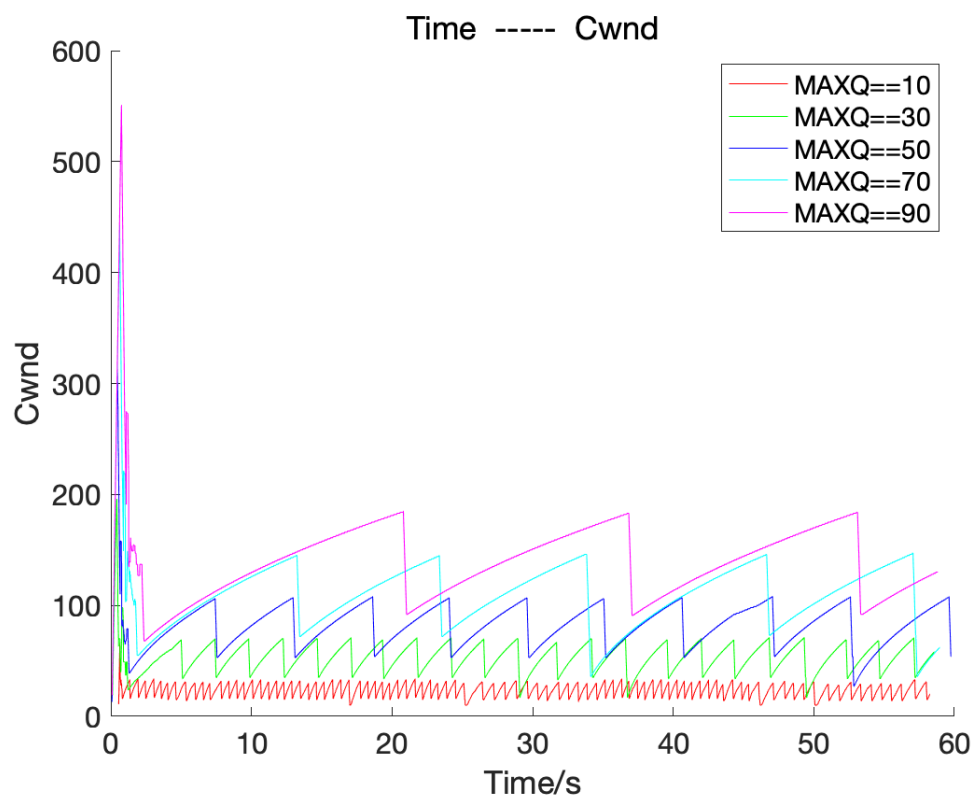
1. 根据附件中提供的脚本，重现 BufferBloat 问题的实验结果；
包含 Cwnd, , Ping, Qlen;
2. 根据附件中提供的脚本，重现解决 BufferBloat 问题的实验结果；
包含 Taildrop, Red, Code1;

三、实验流程

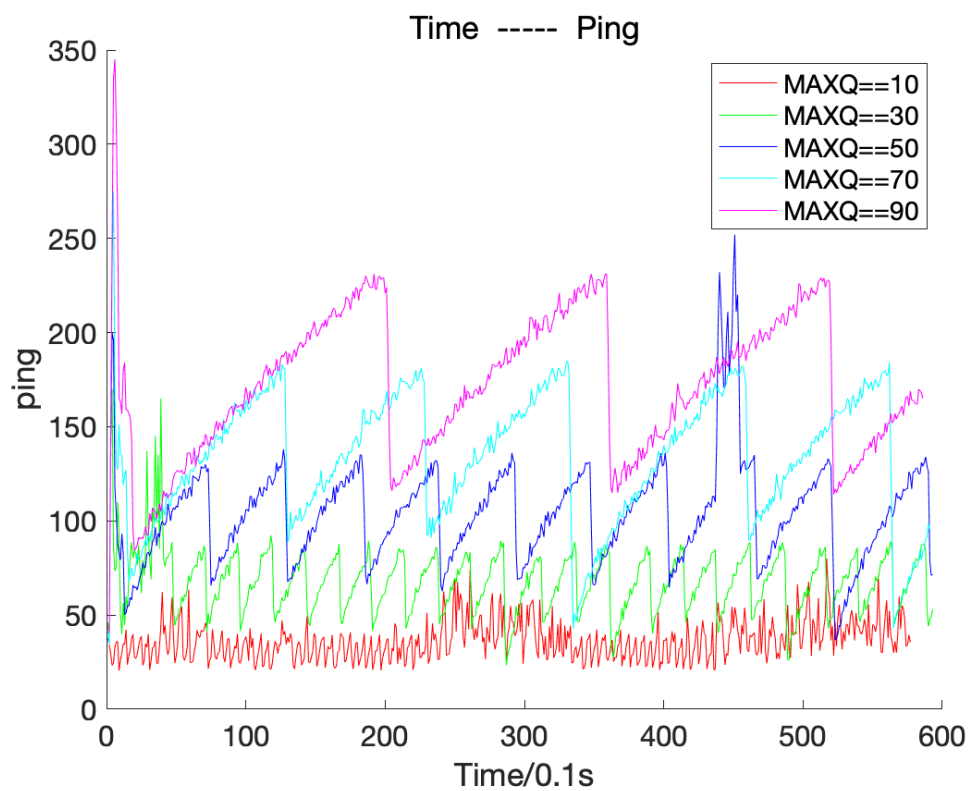
1. 阅读附件中的脚本，理解实验流程；
2. 终端输入 `sudo python reproduce_bufferbloat.py --maxq MAXQ`
运行脚本，MAXQ 为实验中采用的最长队列个数，本次实验，我采集了 MAXQ 分别为 10, 20, 30, 40, 50, 60, 70, 80, 90 的数据；
3. 终端输入 `sudo python mitigate_bufferbloat.py --algo ALGO` 运行脚本，ALGO 为解决 BufferBloat 问题的算法，本次实验，采用 taildrop, red, code1 三种方法；
4. 提取中出数据中的有效数据，使用 MATLAB 绘图；
5. 由于 Cwnd 的数据短时间内波动较大，所以我选取了数据点中的所有极大值点进行绘图；
6. 实验每次采集数据耗时 60s, 60s 内带宽每隔 10s 分别变化为 100, 10, 1, 50, 1, 100, 单位 Mbps；

四、实验结果

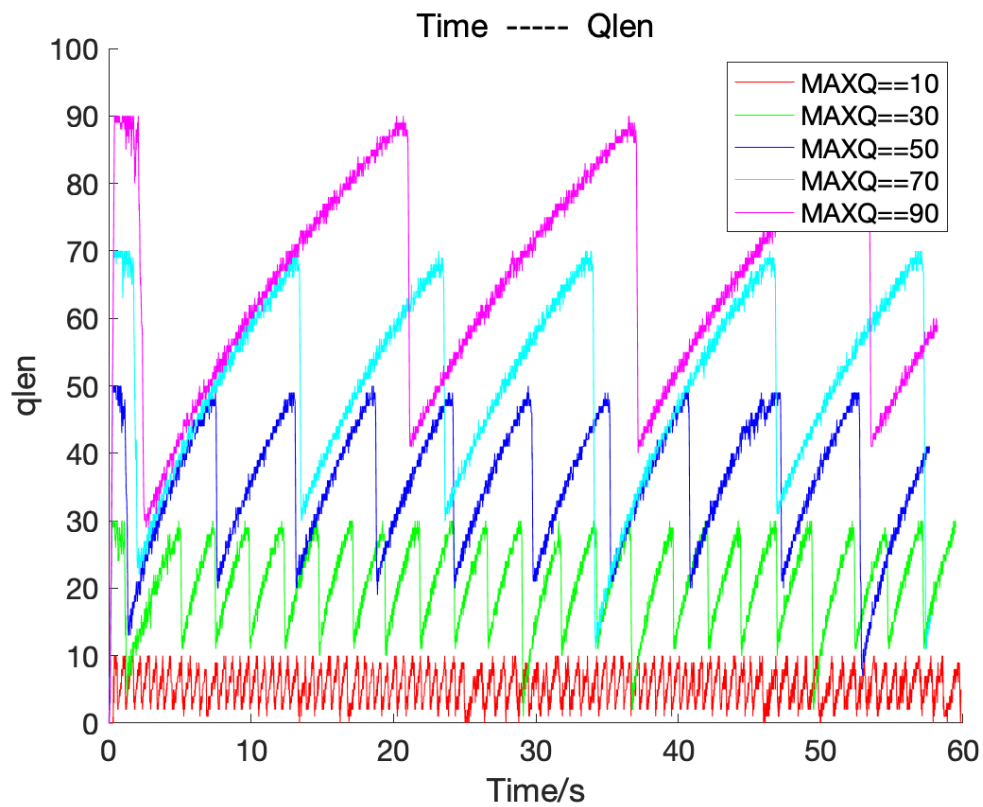
1. Time--Cwnd--Maxq:



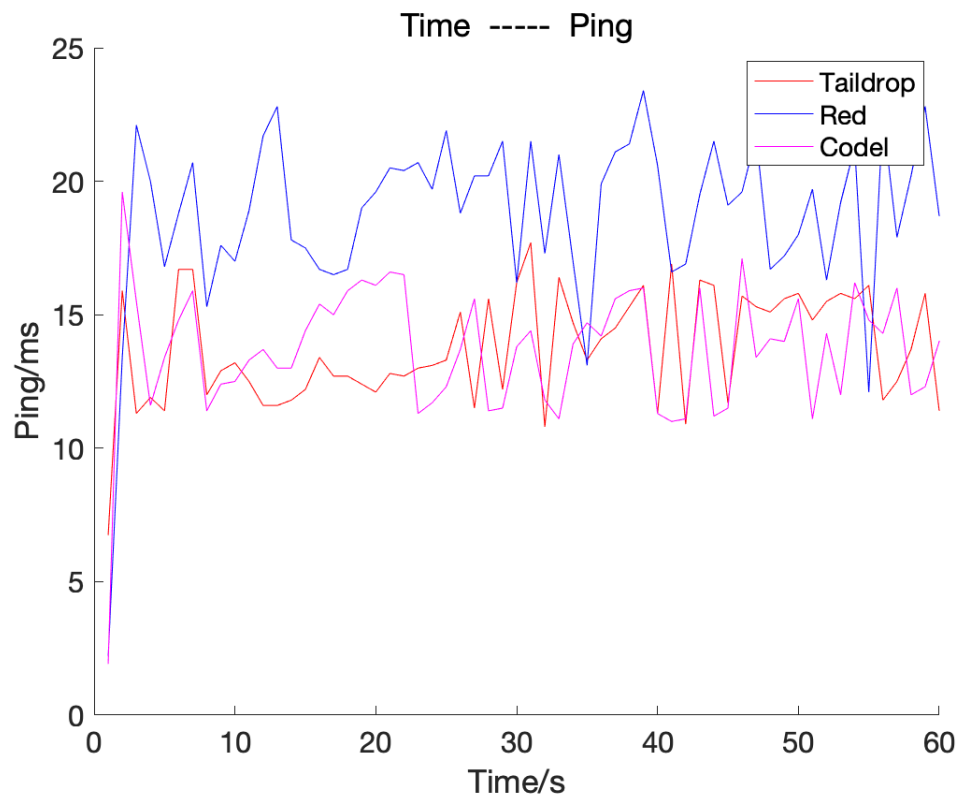
2. Time--Ping--Maxq:



3. Time--Qlen--Maxq:



4. Time--Ping--BufferBloatSolutions:



五、 实验分析

1. 由实验结果可知：

Cwnd, Ping 和 Qlen 都与 MAXQ 成正相关，随着带宽的变化，Cwnd, Ping 和 Qlen 都会发生剧烈变化，在带宽稳定不变的情况下，3 者都随时间增长而增长；

2. 3 种解决 BufferBloat 问题的算法，Red 效果最差，Taildrop 和 Code1 效果相差不大，但是 Code1 要比 Taildrop 效果稍好一些；

六、 反思总结

1. 本次实验又学习到了影响网络传输质量的一个新的因素，认识到了由于数据包队列过短和过长而产生的原因和影响，以及对于数据包队列过长引起的 BufferBloat 问题的 3 种解决方案，其中 Taildrop 最为简单，Red 最复杂，Code1 相对 Red 简单且效果更明显；

七、 参考文献

i

ii

ⁱ 中国科学院大学 2020 春计算机网络研讨课 02 数据包队列管理实验课件

ⁱⁱ 中国科学院大学 2020 春计算机网络研讨课 02 数据包队列管理实验脚本