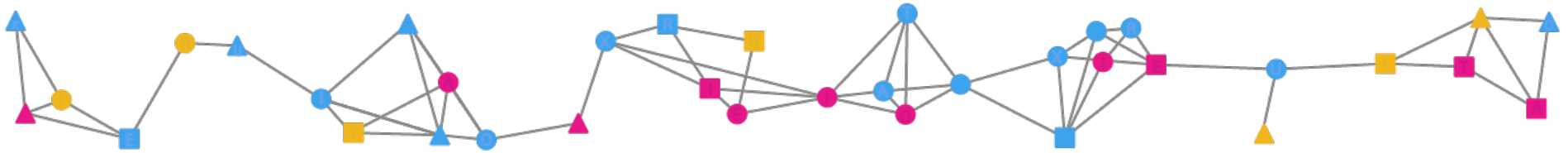


计算机网络

第三章 传输层

谢瑞桃
xie@szu.edu.cn
rtxie.github.io
计算机与软件学院
深圳大学

第三章讲解内容

1. 传输层概述与UDP
   - 需求/服务/协议、多路复用/分解、UDP协议
2. 可靠传输
   - 可靠传输基础知识、TCP可靠传输
3. TCP
   - 报文段结构、超时间隔、流量控制、连接管理
4. TCP拥塞控制
   - 网络拥塞、TCP拥塞控制、吞吐量分析

传输层需求、服务和协议

应用层需求	传输层服务	UDP	TCP
为运行在不同主机上的进程之间提供逻辑通信	进程间交付	✔	✔
检测报文段是否出错	差错检测	✔	✔
解决丢包、差错问题	可靠传输	✘	✔
解决乱序问题	按序交付	✘	✔
解决接收缓存溢出问题	流量控制	✘	✔
应对网络拥塞	拥塞控制	✘	✔

TCP拥塞控制讲解内容

- 网络拥塞对传输性能的影响
- TCP拥塞控制
- 吞吐量分析

网络拥塞

- **太多TCP发送方**向网络**以太快的速率**发送了**太多的数据**，超过了网络的处理能力
- 会导致：
  - 路由器缓存溢出丢包
  - 很长的排队时延

网络拥塞

- 太多TCP发送方向网络以太快的速率发送了太多的数据，超过了网络的处理能力
- **对性能的影响：**
  - **吞吐量下降**
  - **排队时延增加**

第三章知识点汇总

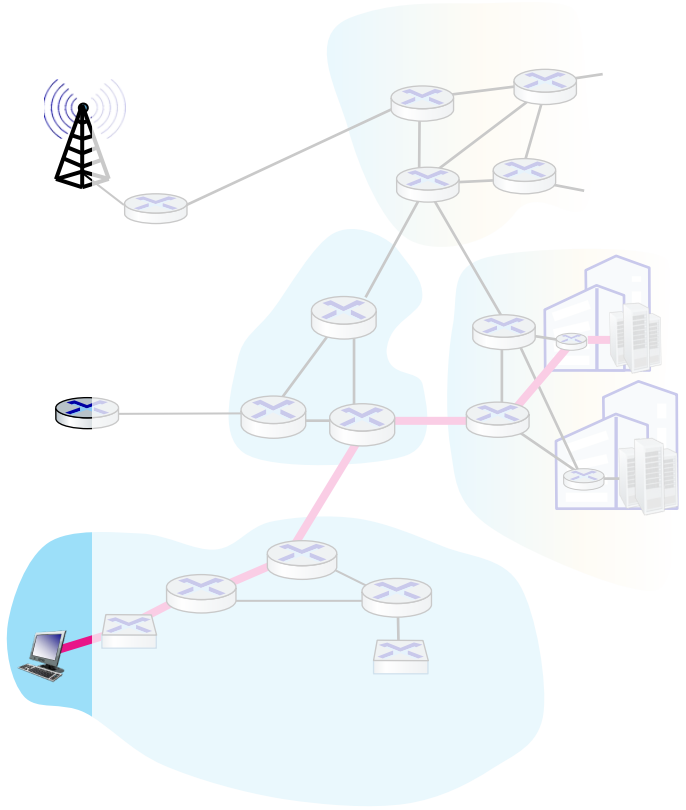
- 了解网络拥塞对传输性能的影响

TCP拥塞控制讲解内容

- 网络拥塞对传输性能的影响
- TCP拥塞控制
- 吞吐量分析

TCP拥塞控制

- TCP发送方的三个问题：
  - 如何限制发送速率？
  - 如何感知网络拥塞？
  - 如何动态调节发送速率？

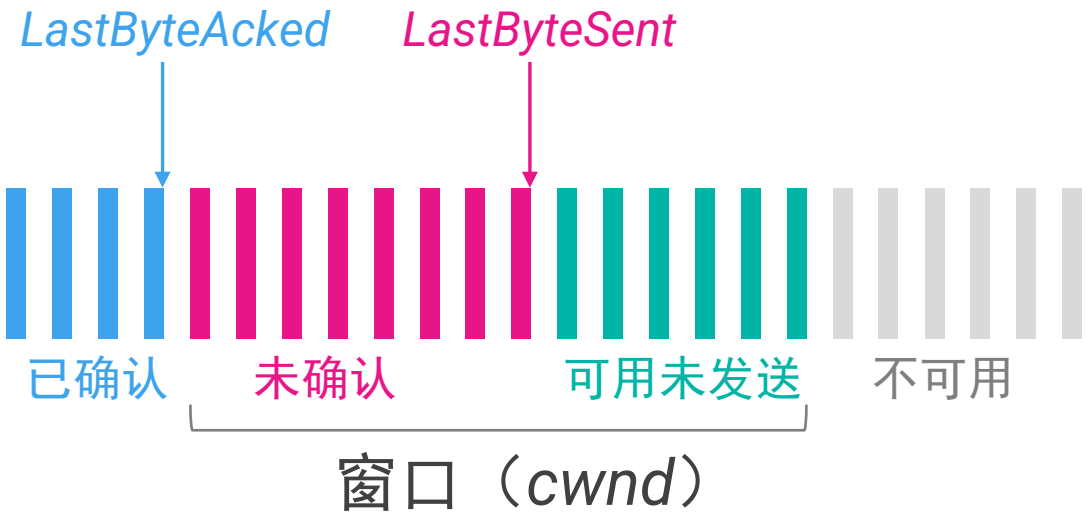


TCP拥塞控制

- 问题一： TCP发送方如何限制发送速率？

TCP拥塞控制

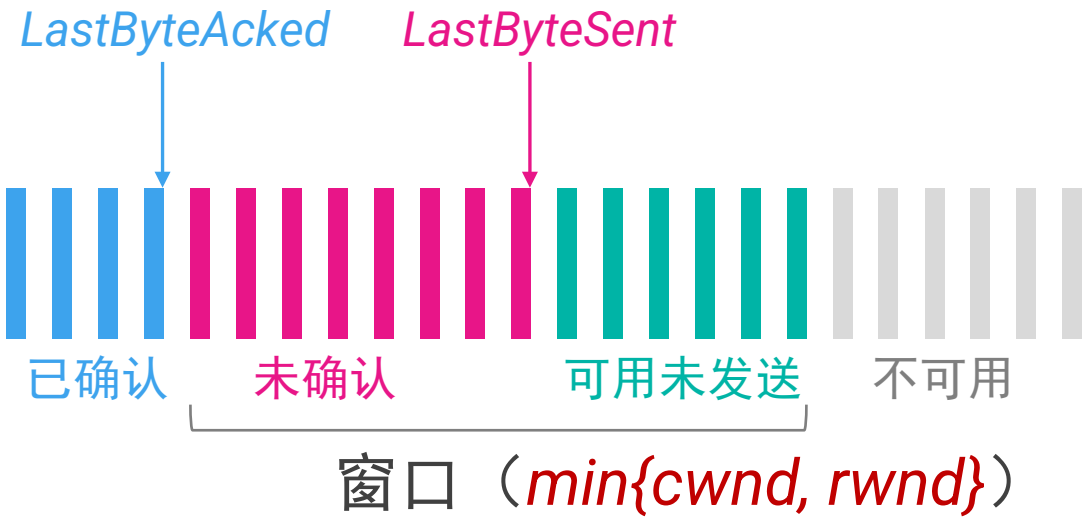
- 问题一： TCP发送方如何限制发送速率？
- 已发送未确认的字节数 <= 拥塞窗口cwnd
- *LastByteSent – LastByteAcked <= cwnd*



图示：发送方的序号空间

TCP拥塞控制+流量控制

- 问题一： TCP发送方如何限制发送速率？
- 已发送未确认的字节数 <= ***min{cwnd, rwnd}***
- *LastByteSent – LastByteAcked* <= ***min{cwnd, rwnd}***



图示：发送方的序号空间

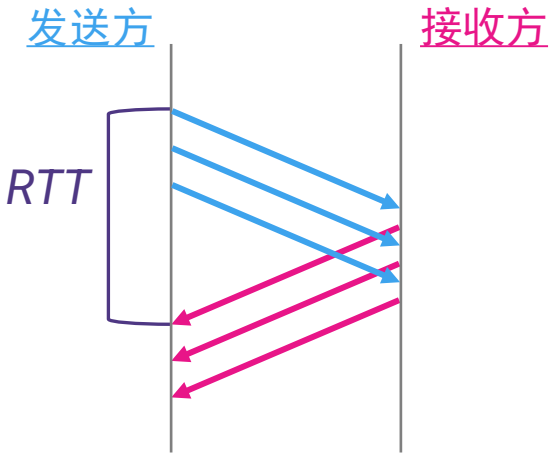
TCP拥塞控制

- 问题一： TCP发送方如何限制发送速率？
- 发送速率是多少呢？

TCP拥塞控制

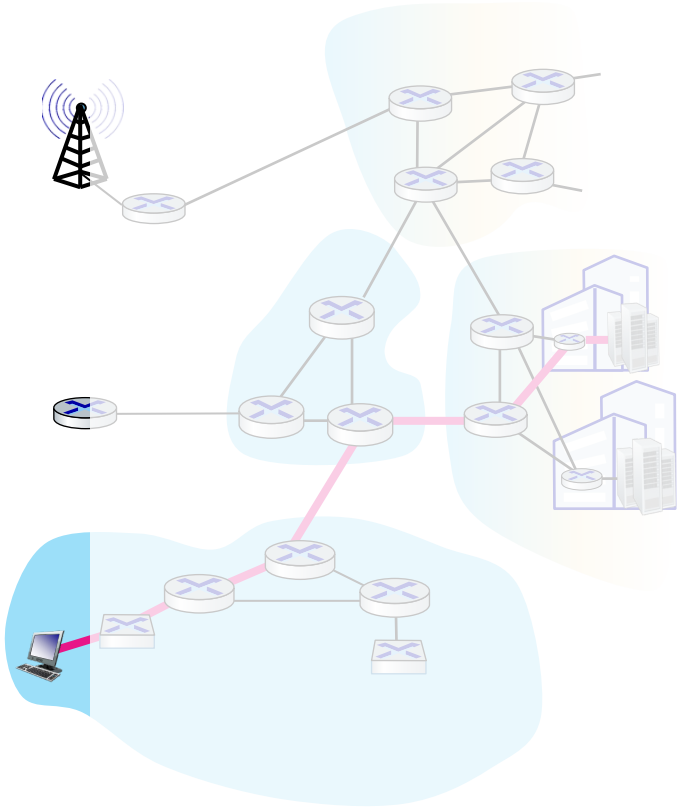
- 问题一： TCP发送方如何限制发送速率？
- 发送速率是多少呢？

$$\frac{cwnd}{RTT}$$



TCP拥塞控制

- 问题二： TCP发送方如何感知网络拥塞？
- 发送端看不到网络内部



TCP拥塞控制

- 问题二： TCP发送方如何感知网络拥塞？
- 丢包事件
  - 定时器超时
  - 收到3个冗余ACK
- 时延增加
- 路由器提供的信号

TCP拥塞控制

- 问题二： TCP发送方如何感知网络拥塞？
- 不同的拥塞控制算法使用不同“拥塞信号”/“网络反馈”

TCP拥塞控制

TCP拥塞控制算法	拥塞信号	参与方
(New) Reno	Loss	Sender
Agile-TCP	Loss	Sender
BBR[12]	Delay	Sender
BIC	Loss	Sender
C2TCP[9][10]	Loss/Delay	Sender
CLAMP	Multi-bit signal	Receiver, Router
Compound TCP	Loss/Delay	Sender
CUBIC	Loss	Sender
ECN	Single-bit signal	Sender, Receiver, Router
Elastic-TCP	Loss/Delay	Sender
FAST	Delay	Sender
H-TCP	Loss	Sender
High Speed	Loss	Sender
Jersey	Loss/Delay	Sender
JetMax	Multi-bit signal	Sender, Receiver, Router
MaxNet	Multi-bit signal	Sender, Receiver, Router
NATCP[11]	Multi-bit signal	Sender
RED	Loss	Router
TFRC	Loss	Sender, Receiver
VCP	2-bit signal	Sender, Receiver, Router
Vegas	Delay	Sender
Westwood	Loss/Delay	Sender
XCP	Multi-bit signal	Sender, Receiver, Router

以前使用最广
本课程所讨论的

目前使用最广

https://en.wikipedia.org/wiki/TCP_congestion_control

习题

- 在Windows10的powershell里执行下列指令，看看你的主机用的TCP拥塞控制算法是哪种算法。
- **netsh interface tcp show supplemental**

TCP拥塞控制

- 问题二： TCP发送方如何感知网络拥塞？
- **丢包事件**
  - **定时器超时**
  - **收到3个冗余ACK**

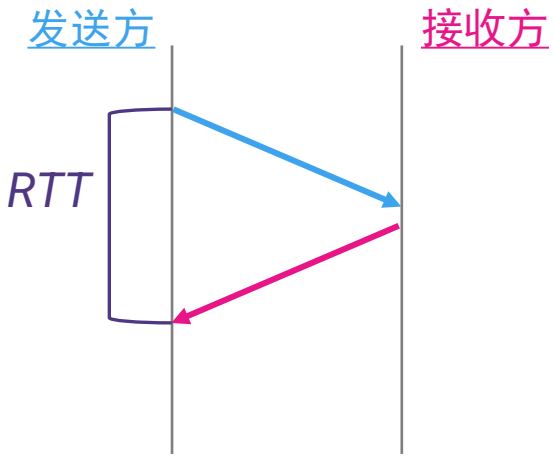
  } **TCP Reno**
- 时延增加
- 路由器提供的信号

TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 发送方如何确定它的发送速率？

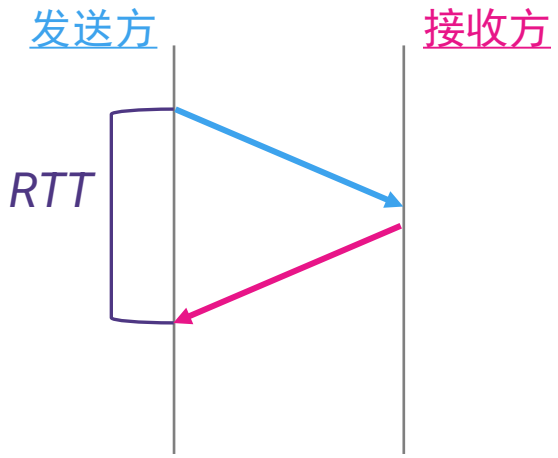
TCP拥塞控制

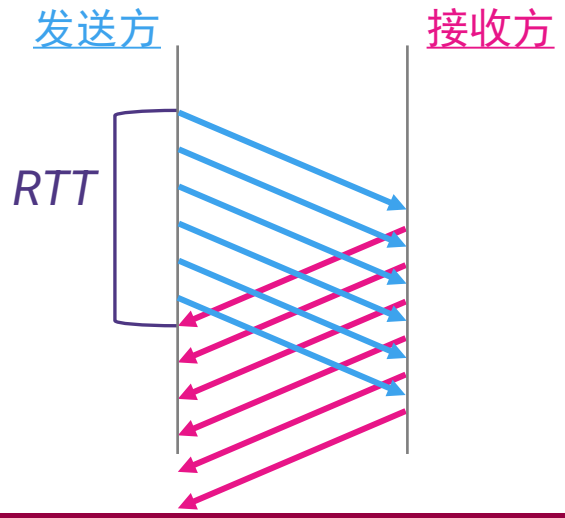
- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 发送方如何确定它的发送速率？
- 每个RTT发送一个分组（停等协议），信道利用率太低，通常不会使网络拥塞。



TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)？
- 发送方如何确定它的发送速率？
- 每个RTT只要信道空闲就发送分组，信道利用率最高，通常会使网络拥塞。





TCP拥塞控制

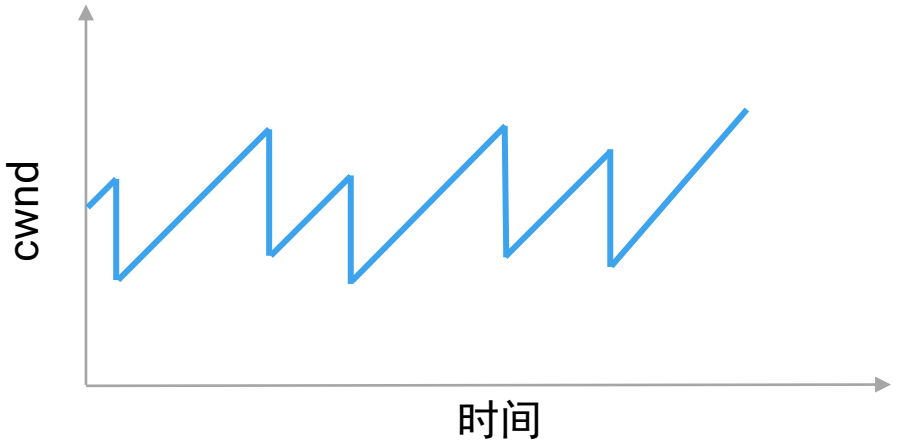
- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- **信道利用率和网络拥塞是一对矛盾**
- 发送方如何确定它的发送速率，使得网络不会拥塞，同时又能充分利用所有可用的带宽？

TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 信道利用率和网络拥塞是一对矛盾
- 发送方如何确定它的发送速率，使得网络不会拥塞，同时又能充分利用所有可用的带宽？
- 网络里的流量是动态变化的，如何**动态地**调节发送速率？

TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
  - 发送方**由低到高**的增加拥塞窗口，探测有多少可用网络带宽
  - 如果没有发生意外（丢包），**增加速率**
  - 如果发生了（丢包），立刻**降低速率**
  - 重复以上过程



TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 子问题1：一个TCP连接开始的时候，cwnd初始值应该设多少？

TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 子问题1：一个TCP连接开始的时候，cwnd初始值应该设多少？
- 因为对网络流量和带宽一无所知，应该设置为一个很小的值。一般设为1、4或10个MSS 。
- 在Windows10的powershell里执行下列指令，看看你的主机用的TCP初始cwnd是多少？
- **Get-NetTCPSetting**

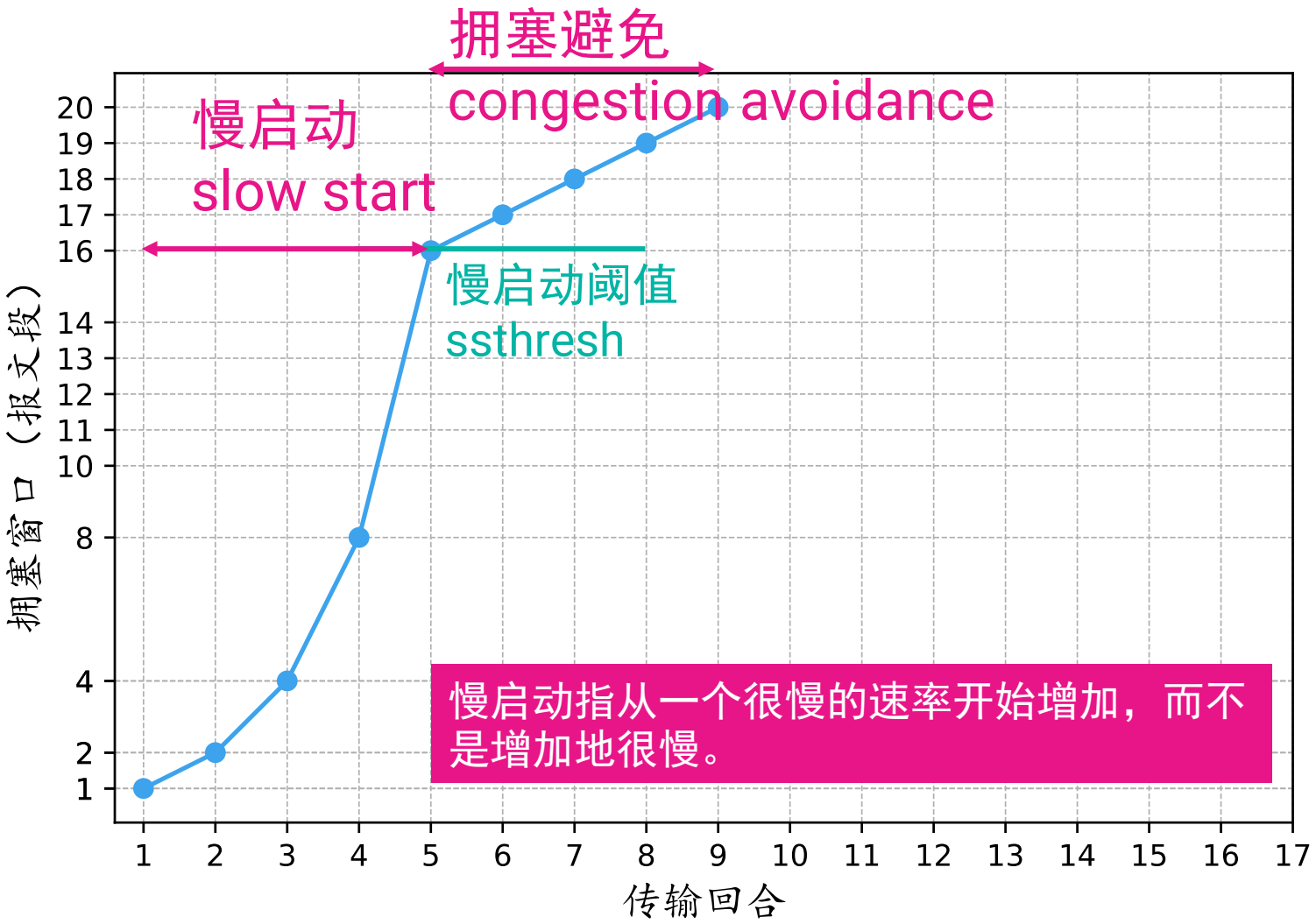
TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 子问题2：如何增加cwnd?
  - 线性增加：
    - 每个RTT增加一个MSS
  - 指数增加:
    - 每个RTT增加一倍MSS
- 用哪种方法好呢？

TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)？
- 子问题2：如何增加cwnd？
  - 线性增加：
    - 每个RTT增加一个MSS
    - **增长速度慢，但一旦发生拥塞损失小**
  - 指数增加：
    - 每个RTT增加一倍MSS
    - **增长速度快，但一旦发生拥塞损失大**
- 用哪种方法好呢？
- **先指数增加，超过一定阈值以后线性增加**

TCP拥塞控制


拥塞避免
congestion avoidance
慢启动
slow start
慢启动阈值
ssthresh
慢启动指从一个很慢的速率开始增加，而不是增加地很慢。
拥塞窗口（报文段）
传输回合
1
2
4
8
10
11
12
13
14
15
16
17
18
19
20
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 子问题3：丢包时，如何降低cwnd?
- 定时器超时
- 收到3个冗余ACK

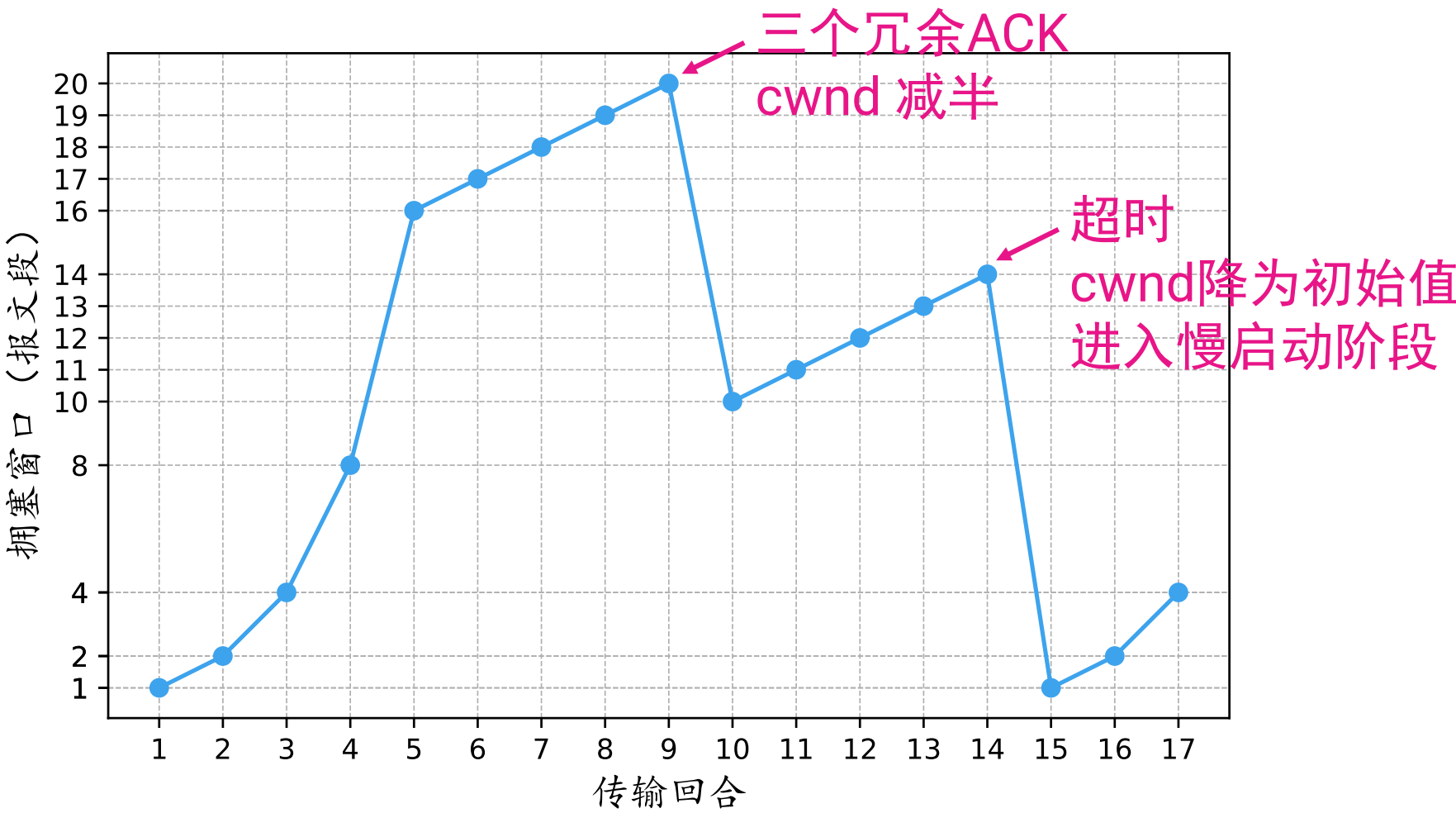
TCP拥塞控制

- 问题三：如何动态调节发送速率（cwnd）？
- 子问题3：丢包时，如何降低cwnd？
- 定时器超时
  - 说明拥塞
- 收到3个冗余ACK
  - 虽然某个分组丢失了，但有些分组被收到了，说明拥塞不是很严重

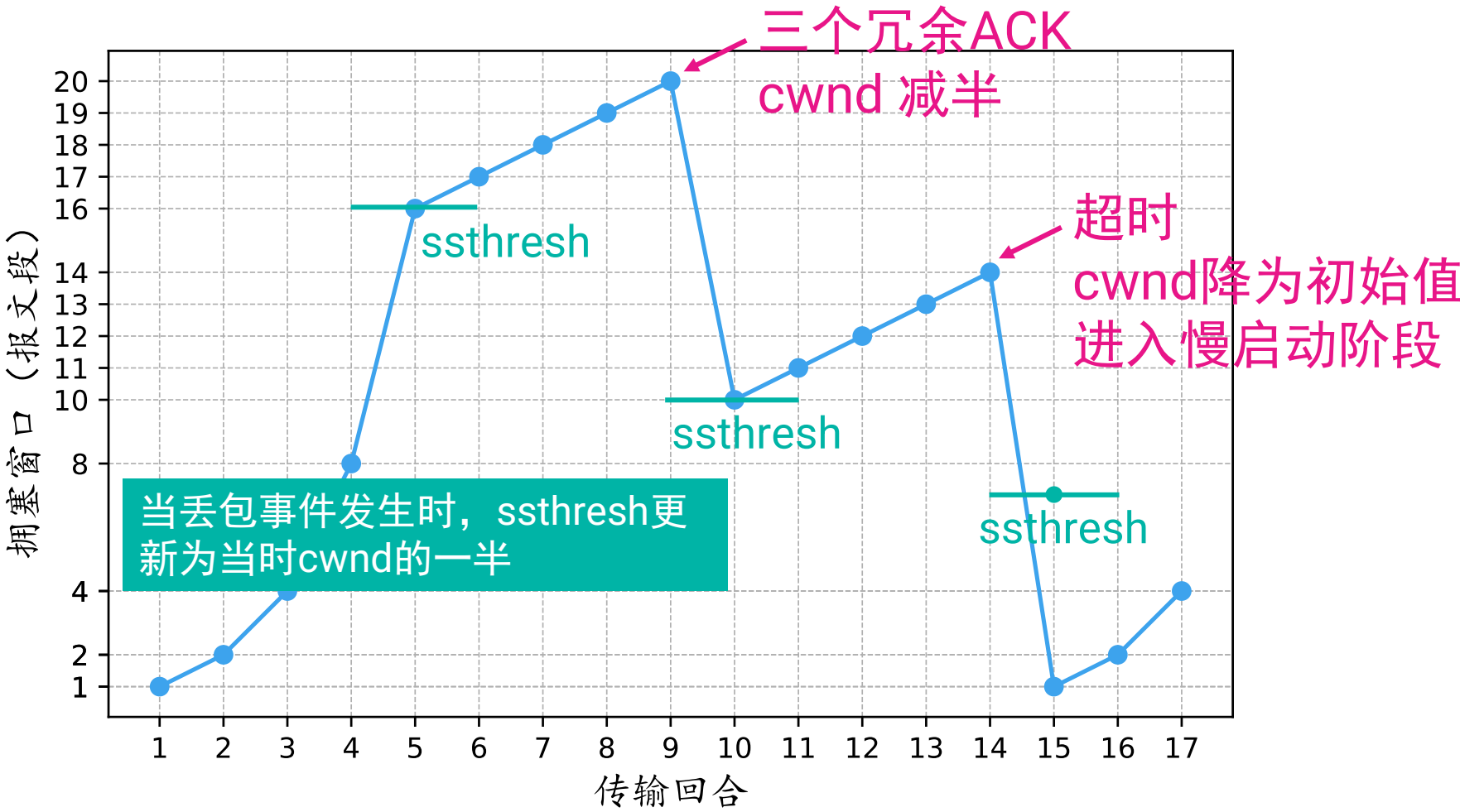
TCP拥塞控制

- 问题三： TCP发送方如何动态调节发送速率(cwnd)?
- 子问题3：丢包时，如何降低cwnd?
- 定时器超时
  - 说明拥塞
  - **将cwnd降为初始值，进入慢启动（指数增长）阶段**
- 收到3个冗余ACK
  - 虽然某个分组丢失了，但有些分组被收到了，说明拥塞不是很严重
  - **将cwnd减半，进入拥塞避免（线性增长）阶段**

TCP拥塞控制

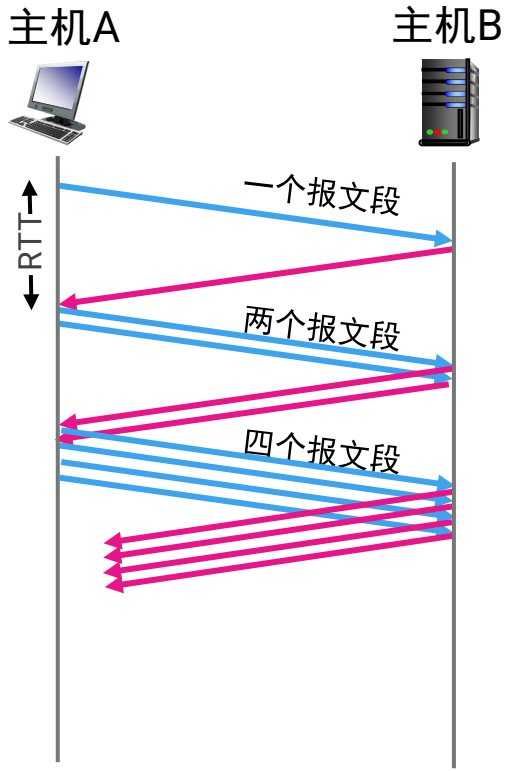

三个冗余ACK
cwnd 减半
超时
cwnd降为初始值
进入慢启动阶段
拥塞窗口（报文段）
传输回合
1
2
4
8
10
11
12
13
14
16
17
18
19
20
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

TCP拥塞控制


三个冗余ACK
cwnd 减半
超时
cwnd降为初始值
进入慢启动阶段
ssthresh
ssthresh
ssthresh
当丢包事件发生时，ssthresh更新为当时cwnd的一半
拥塞窗口（报文段）
传输回合
1
2
4
8
10
11
12
13
14
16
17
18
19
20
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

TCP拥塞控制

- 如何算法实现cwnd增加？
- 慢启动中的指数增加：每个RTT增加一倍MSS
- **每收到一个ACK，cwnd增加一个报文段**



TCP拥塞控制

- 如何算法实现cwnd增加？
- 拥塞避免中的线性增加：每个RTT增加一个MSS
- **每收到一个ACK，cwnd增加多少字节？**

TCP拥塞控制

- 如何算法实现cwnd增加？
- 拥塞避免中的线性增加：每个RTT增加一个MSS
- **每收到一个ACK，cwnd增加多少字节？**
- **一个RTT的发送数据是cwnd字节，等同于cwnd/MSS个报文段**
- **由 x * (cwnd / MSS) = MSS**
- **得 x = MSS * (MSS / cwnd)**

TCP流量控制+拥塞控制

- 两种控制联合分析方法
  - 假设没有丢包
  - 假设接收缓存只有数据写入没有数据读取

单位: MSS ⌊	第i个RTT	第i+1个RTT
起始时拥塞窗口cwnd	cwnd(i)	慢启动阶段：cwnd(i+1) = cwnd(i) + swnd(i)
		拥塞避免阶段：cwnd(i+1) = cwnd(i) + 1
起始时接收窗口rwnd	rwnd(i)	rwnd(i+1) = rwnd(i) - swnd(i)
起始时发送窗口swnd =min{cwnd, rwnd}	swnd(i)	swnd(i+1) = min{cwnd(i+1), rwnd(i+1)}

- 注意：当有丢包或接收缓存有数据读取时，需要具体问题具体分析，但方法类似

第三章知识点汇总

- 理解拥塞控制算法的三个基本问题
  - 如何限制发送速率？
  - 如何感知网络拥塞？
  - 如何动态调节发送速率？
- 理解TCP(Reno)拥塞控制算法的原理
- 理解TCP(Reno)拥塞控制算法的设计逻辑
- 理解实现cwnd增加的算法
- 掌握流量控制和拥塞控制的联合分析方法

习题

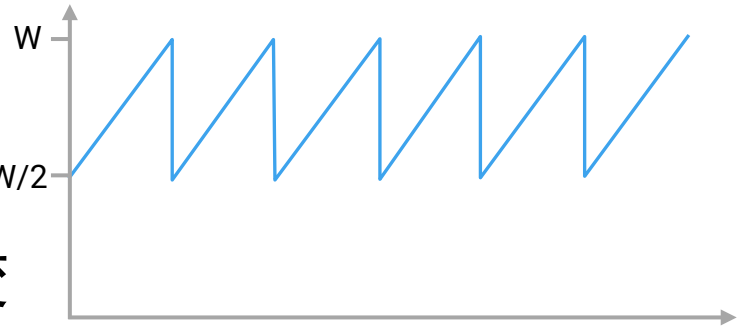
- 【2017年考研39题】若甲向乙发起一个TCP连接，最大段长MSS=1 KB， RTT=5 ms，乙开辟的接收缓存为64 KB，则甲从连接建立成功至发送窗口达到32 KB，需经过的时间至少是
- A. 25 ms
- B. 30 ms
- C. 160 ms
- D. 165 ms

TCP拥塞控制讲解内容

- 网络拥塞对传输性能的影响
- TCP拥塞控制
- 吞吐量分析

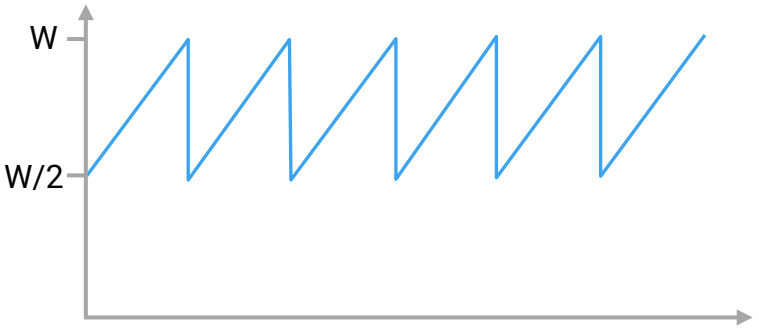
TCP平均吞吐量分析

- 分析一个高度简化的模型
  - 忽略慢启动阶段
  - 设丢包发生时的cwnd为W字节
  - 设一段持续时间内W与RTT不变
  - 设丢包事件均为三次冗余ACK



TCP平均吞吐量分析

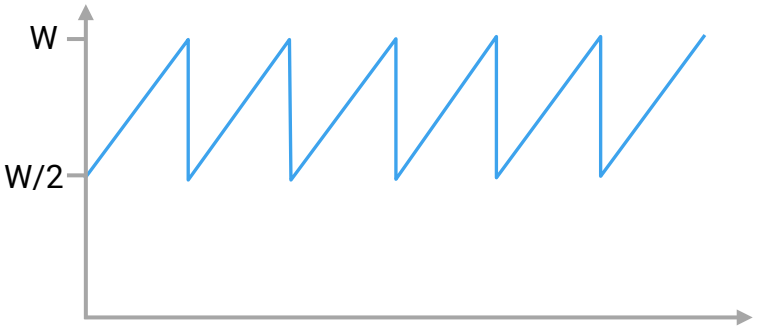
- 分析一个高度简化的模型



- 在这个简化模型里
  - 窗口在W/2到W之间变化
  - 传输速率在W/2RTT和W/RTT之间变化
  - 平均传输速率是多少？

TCP平均吞吐量分析

- 分析一个高度简化的模型



- 在这个简化模型里
  - 窗口在W/2到W之间变化
  - TCP传输速率在W/(2RTT)和W/RTT之间变化
  - 平均传输速率(吞吐量)是多少？

$$\text{平均吞吐量} = \frac{3}{4}\frac{\text{W}}{\text{RTT}}$$

第三章知识点汇总

- 理解简化模型下TCP平均吞吐量的分析方法

习题

- 假定一个客户端从一个HTTP服务器上获取一个对象，传输层使用TCP Reno协议。已知客户端已经完成了域名解析，忽略客户端接收窗口的影响。
- 如果该对象需要10个MSS才能够传输完，那么从该客户端发起连接到它收到该文件需要几个RTT，说明理由？假定初始拥塞窗口为1MSS，并且整个传输中没有丢包。

思考题

- 子问题1：一个TCP连接开始的时候，cwnd初始值应该设多少？
- 因为对网络流量和带宽一无所知，应该设置为一个很小的值。一般设为1、4或10个MSS 。
- 上述这种方法好吗？有没有更好的方法呢？

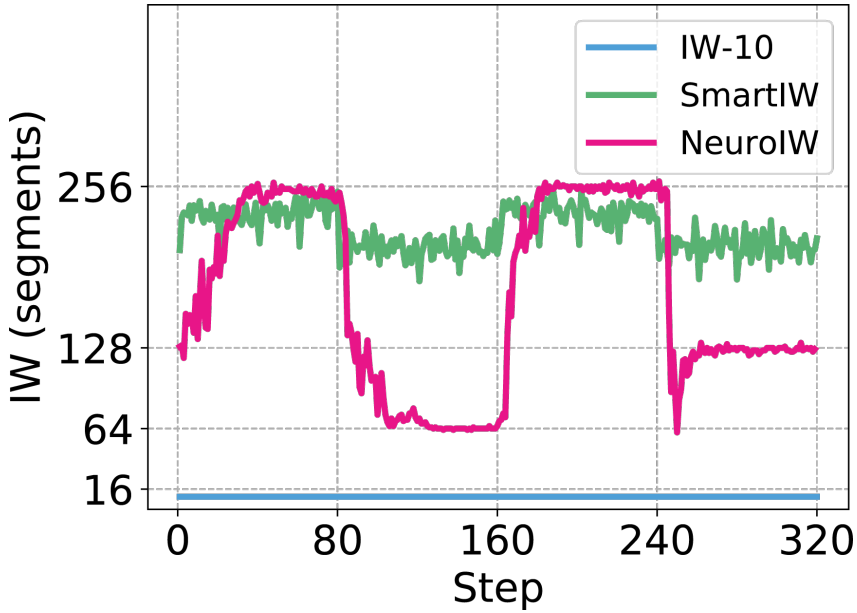
思考题

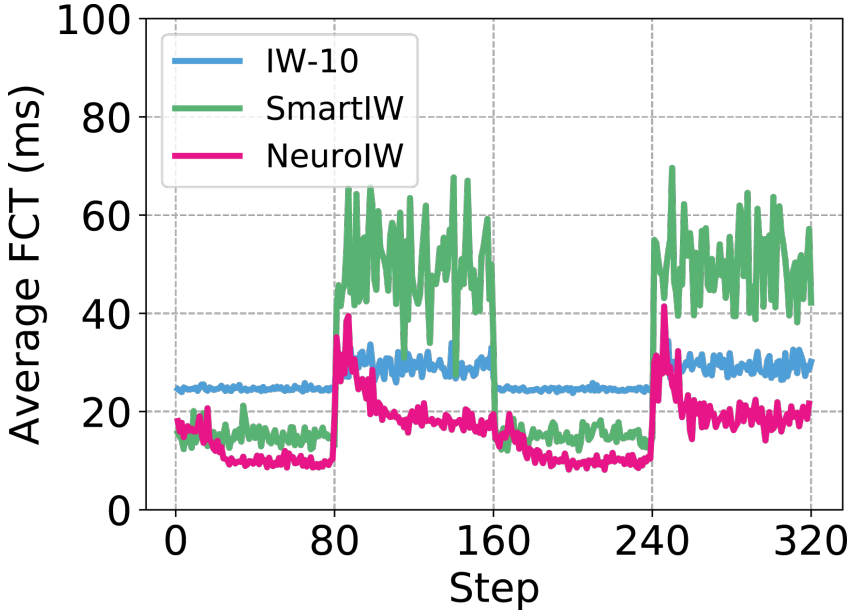
IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 38, NO. 2, FEBRUARY 2020

Adaptive Online Decision Method for Initial Congestion Window in 5G Mobile Edge Computing Using Deep Reinforcement Learning

Ruitao Xie, Xiaohua Jia, *Fellow, IEEE*, and Kaishun Wu

https://rtxie.github.io/rtxie.github.io/wp-content/uploads/2020/03/NeuroIW-s.pdf





Sit still, my heart, do not raise your dust.
Let the world find its way to you.

静静地坐着吧，我的心，不要扬起你的尘土。
让世界自己来寻你。

——*Tagore*