

Разбор задач контрольной работы 2

Смелянский Р. Л.,
Щербинин В. В.

Sliding window

- Интернет-магистраль между Москвой и Новосибирском имеет пропускную способность в 10 Гбит/с, $RTT = 50$ мс. TCP-поток направлен из Москвы в Новосибирск. Размер окна получателя (RWS) равен 100 Мбит. Какова максимальная скорость у отправителя?

Sliding window

- $\text{Max CWND} = R \cdot \text{RTT} = 10 \cdot 10^9 \text{ б/с} \cdot 50 \cdot 10^{-3} \text{ с} = 500 \text{ Мбит}$
- $\text{RWS} = 100 \text{ Мбит}$
- $\text{SWS} = \min(\text{CWND}, \text{RWS}) = 100 \text{ Мбит}$
- Максимальная скорость отправителя:
$$\begin{aligned} \text{SWS}/\text{RTT} &= 100 * 10^6 \text{ бит} / (50 \cdot 10^{-3} \text{ с}) \\ &= 2 \text{ Гбит/с} \end{aligned}$$

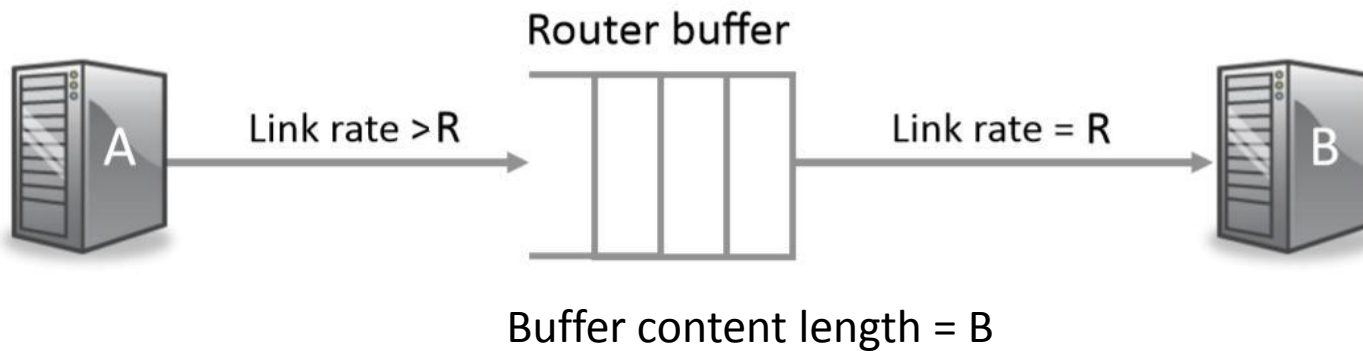
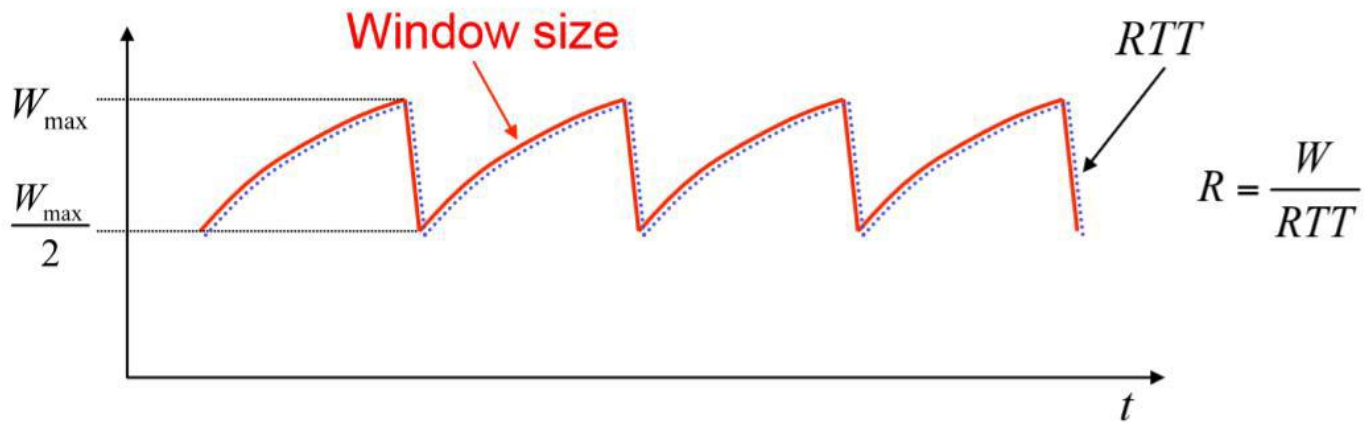
AIMD

- Некто качает видео на скорости 1 Мбит/с с сервера. Все пакеты имеют длину 250 Байт, ring до сервера в отсутствие передачи данных 5 мс. Во время передачи максимальный и минимальный размеры окна в AIMD не изменяются.
 - Оценить минимальный достаточный размер буфера маршрутизатора в узком месте.
 - Оценить наибольший размер окна отправителя и RTT при его достижении.

AIMD

- Раунд – интервал времени от отправки пакета до получения подтверждения этого пакета. RTT – время раунда.
- Правило AIMD:
 - Если все нормально, каждый **раунд** $W := W + 1$
 - Если пакет потерян, то $W := W / 2$

AIMD



AIMD

- Минимальный размер окна для поддержания скорости R :

$$W_0 = R \cdot RTT$$

- Если размер окна фиксирован и равен W_0 , то на узкой линии поддерживается скорость R и буфер не заполняется.

AIMD

- Раунд 0: послали W_0 бит за RTT_0 , буфер пуст

AIMD

- Раунд 1: послали $W_1 = W_0 + 1 \cdot MSS$ бит за RTT_0
 - буфер не успел опустошиться, в нем осталось $1 \cdot MSS$ бит

$$B = 1 \cdot MSS$$

- RTT увеличилось на $1 \cdot \frac{MSS}{R}$ из-за задержки буферизации:

$$RTT_1 = RTT_0 + 1 \cdot \frac{MSS}{R}$$

- $(RTT_1 - RTT)$ для последнего пакета, посланного за 1-й раунд)

AIMD

- Раунд 2: послали $W_2 = W_0 + 2 \cdot MSS$ бит за RTT_1
 - буфер не успел опустошиться, в нем осталось $2 \cdot MSS$ бит

$$B = 2 \cdot MSS$$

- RTT увеличилось еще на $1 \cdot \frac{MSS}{R}$ из-за задержки буферизации:

$$RTT_2 = RTT_0 + 2 \cdot \frac{MSS}{R}$$

AIMD

- Раунд k : послали $W_k = W_0 + k \cdot MSS$ бит за RTT_{k-1}
 - $W_k = W_0 + k \cdot MSS$
 - $B = k \cdot MSS$
 - $RTT_k = RTT_0 + k \cdot \frac{MSS}{R}$

AIMD

- Если $B > B_{max}$, то пакет теряется и в следующем раунде окно уполовинивается.
- Для сохранения скорости R B_{max} должен быть таким, чтобы после уполовинивания $W \geq W_0$

$$\begin{cases} B_{max} = K \cdot MSS \\ \frac{W_K}{2} \geq W_0 \end{cases}$$

$$W_0 + K \cdot MSS \geq 2 \cdot W_0$$

$$\min K = \frac{W_0}{MSS}$$

$$\min B_{max} = W_0$$

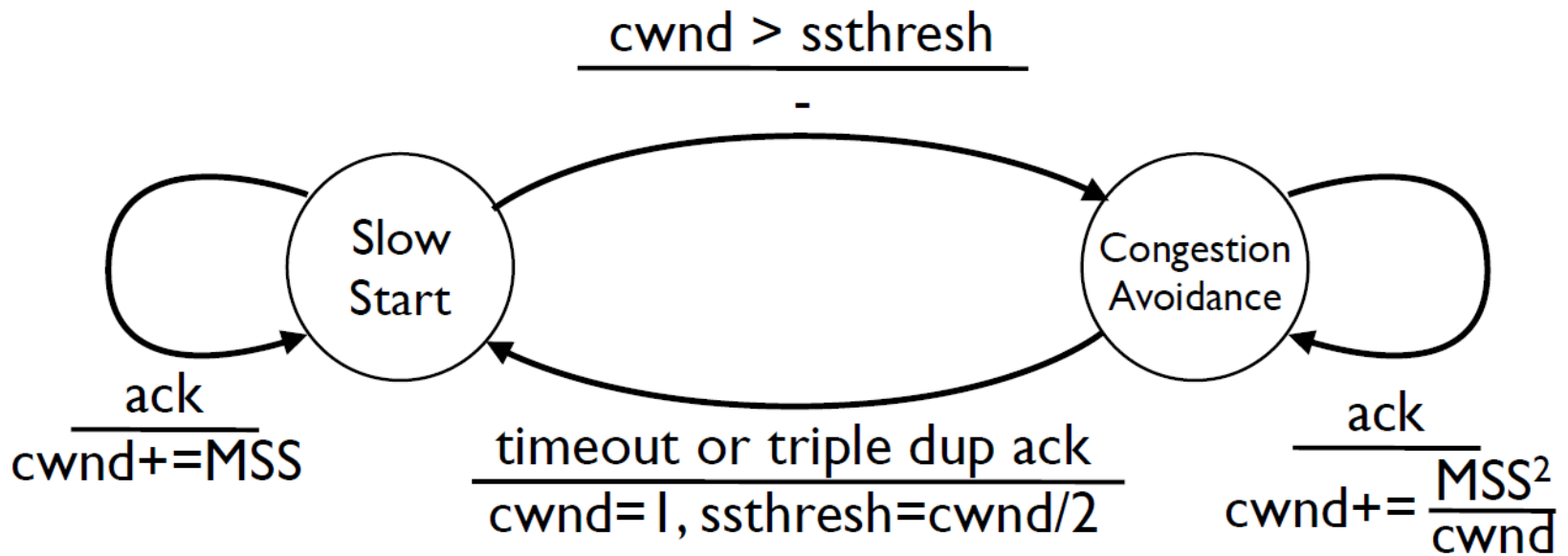
AIMD

- По условию задачи
 - $RTT_0 = 5 \text{ мс}$
 - $R = 1 \text{ Мбит/с}$
 - $MSS = 250 \text{ Байт} = 2000 \text{ бит}$
- Т. о.
 - $W_0 = R \cdot RTT_0 = B_{max} = 5000 \text{ бит}$
 - $W_{max} = 2 \cdot W_0 = 10000 \text{ бит}$
 - При $K = 6$ буфер переполняется
 - Время достижения макс. размера окна
$$\sum_{i=0}^6 RTT_i = \sum_{i=0}^6 \left(5 \cdot 10^{-3} + i \cdot \frac{1000}{10^6} \right) = 56 \text{ мс}$$

TCP Tahoe

- Рассматривается TCP Tahoe. Пусть сеть выдерживает у отправителя CWND на 16 пакетов, после чего пакеты начинают теряться из-за переполнения очереди на «узкой» линии. Пусть начальный размер CWND = 1, а SSTHRESH = 8. Выписать через запятую значения размеров окна, получающиеся во время передачи данных.

TCP Tahoe



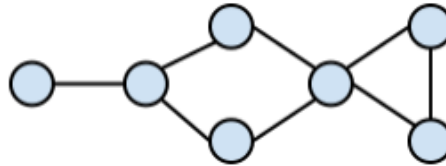
TCP Tahoe

- Размеры окна:

1, 2, 4, 8, 9, 10, ..., 15, 16, 17, 1, 2, ...

Маршрутизация по обратному пути

- Рассмотрим сеть, показанную ниже, в которой $N=7$ маршрутизаторов и $L=8$ линий.



- Предположим, что к каждому маршрутизатору подключен один хост и один из хостов хочет передавать один пакет всем другим хостам в этой сети, используя алгоритм вещания по обратному пути. Считаем, что пакеты не теряются.
 - Сколько всего копий пакетов будет послано по всем линиям маршрутизатор – маршрутизатор?
 - Сколько всего пакетов будет сброшено?

Маршрутизация по обратному пути

- Каждый маршрутизатор R кроме первого шлет $d(R) - 1$ пакетов, где d – степень вершины, своим соседям.
- Первый маршрутизатор шлет $d(R)$ пакетов.
- Послано по линиям:
$$\sum_R (d(R) - 1) + 1 = 2L - N + 1 = 10 \text{ пакетов}$$
- В соединяющем дереве $N - 1$ дуга. Каждый пакет, не прошедший по одной из этих дуг, будет сброшен. Всего сброшено:
- $10 - (N - 1) = 4$ пакета