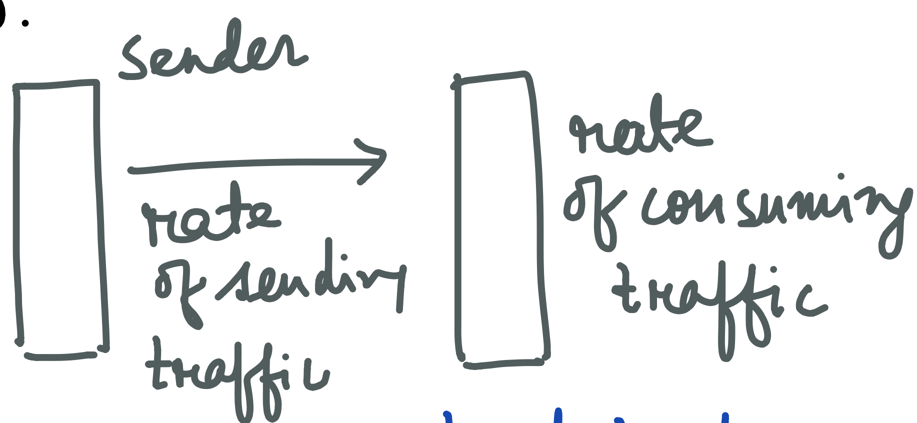
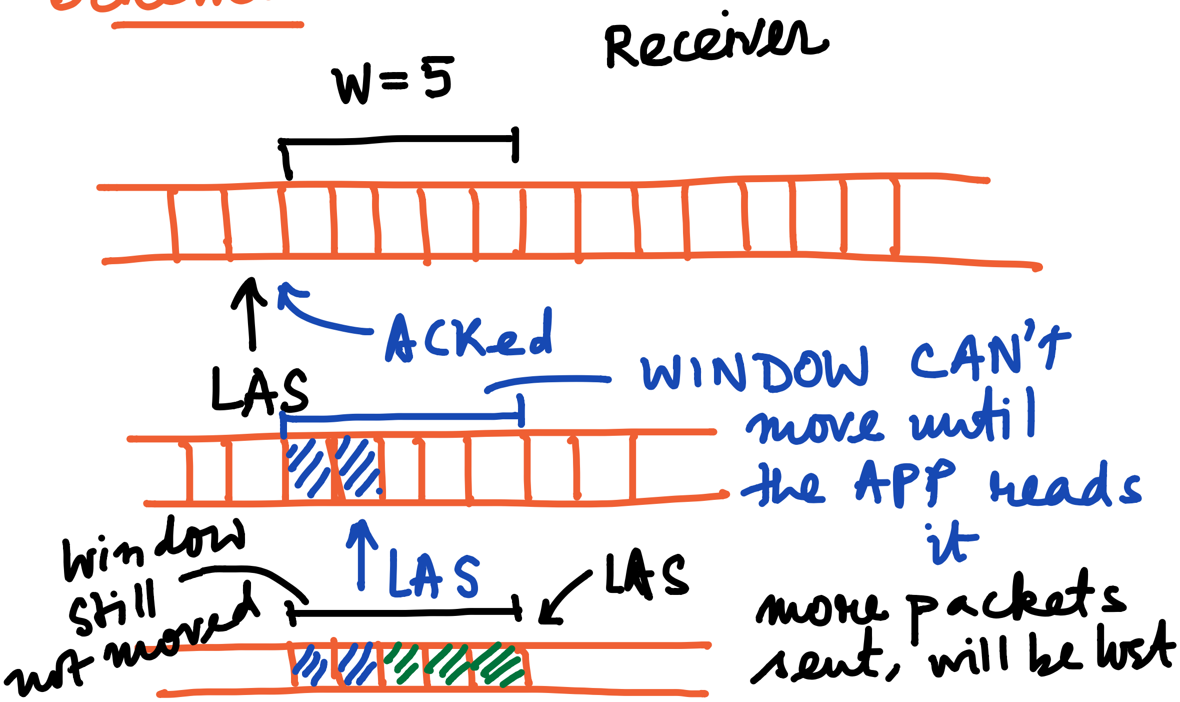
Flow Control mechanism

Sender and Receiver should stay in sync so that the buffers don't overflow.



analogous to overhead tank mechanism

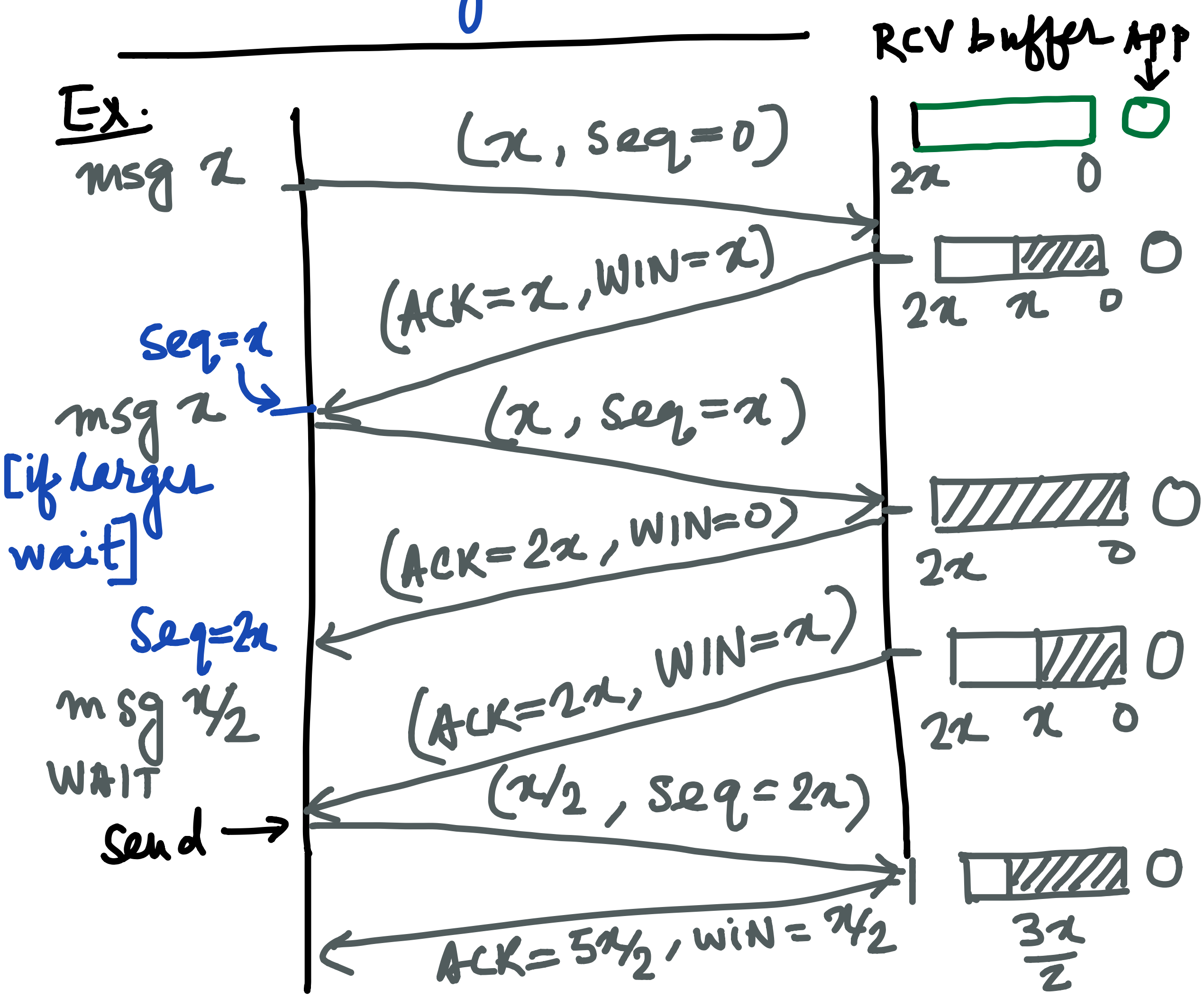
Schematic



The flow control ensures that the remaining buffer size is also communicated to the sender

- sender pretends that is the window size and will send if the next message is $\leq$ window.

Ex.



Retransmission timeouts

- Start timer when a message is sent
- Receive ACK - stop timer
- Timer expires - Retransmit

Setting the right "timeout"

- long : too much idling
- short : many retransmissions

ideally it should be just above the RTT

measuring RTT

easy in LAN

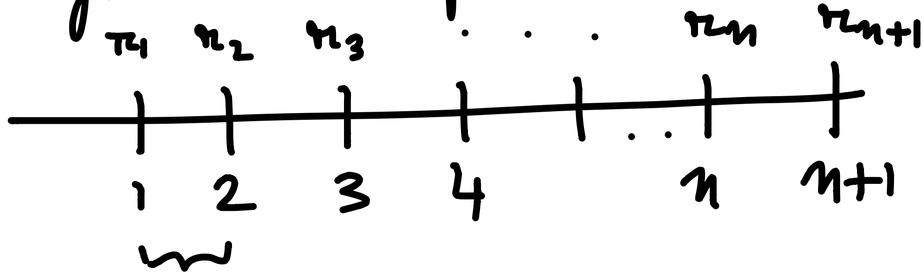
difficult on Internet.

- on Internet, the RTT varies
- doesn't make sense to design either based on the best or worst cases

Adaptive timeout

based on moving average and variance.

Imagine time epochs 1, 2, 3, . . . ,



measure

smoothened RTTs within every epoch.

$$sr_{n+1} = 0.9\, sr_n + 0.1\, r_n \,, \quad sr_0 = r_0$$

$$sv_{n+1} = 0.9\, sv_n + 0.1 \times \left| r_{n+1} - sr_{n+1} \right| ,$$

$$sv_0 = 0,\ sv_1 = \left| r_1 - sr_0 \right|$$

- intuition: making the changes slow and smooth.

Timeout$_n$ (conservative estimate) [Adaptive]

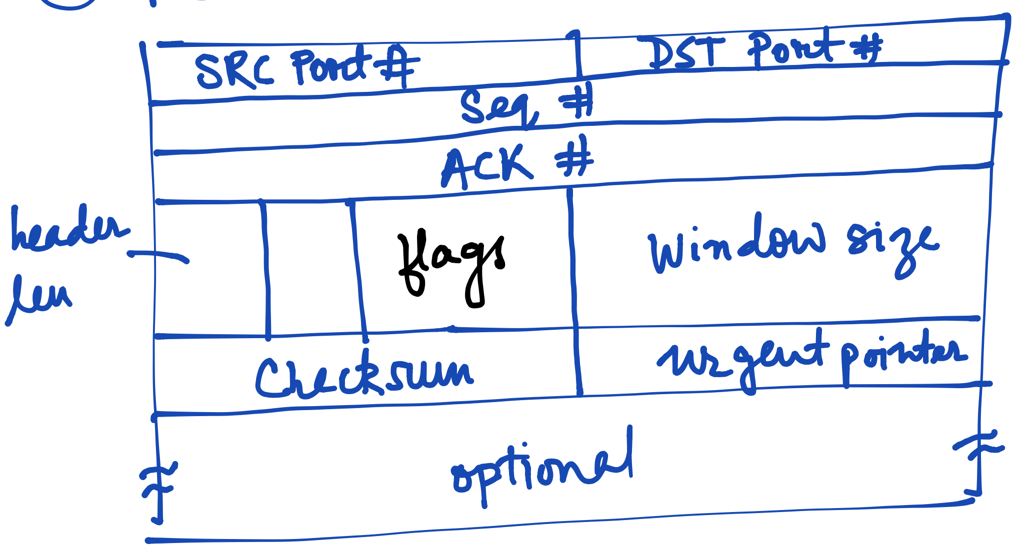
$$= sr_n + \underline{4} \times sv_n$$

numbers are fit wrt the data.

Transmission Control Protocol

How and where the ideas are used.

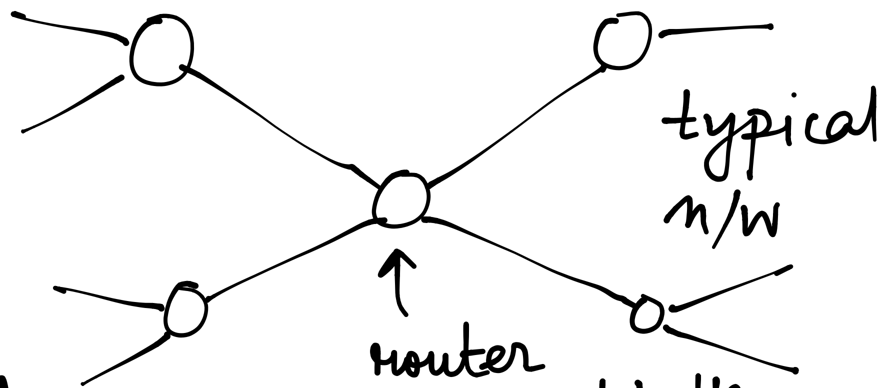
① TCP header

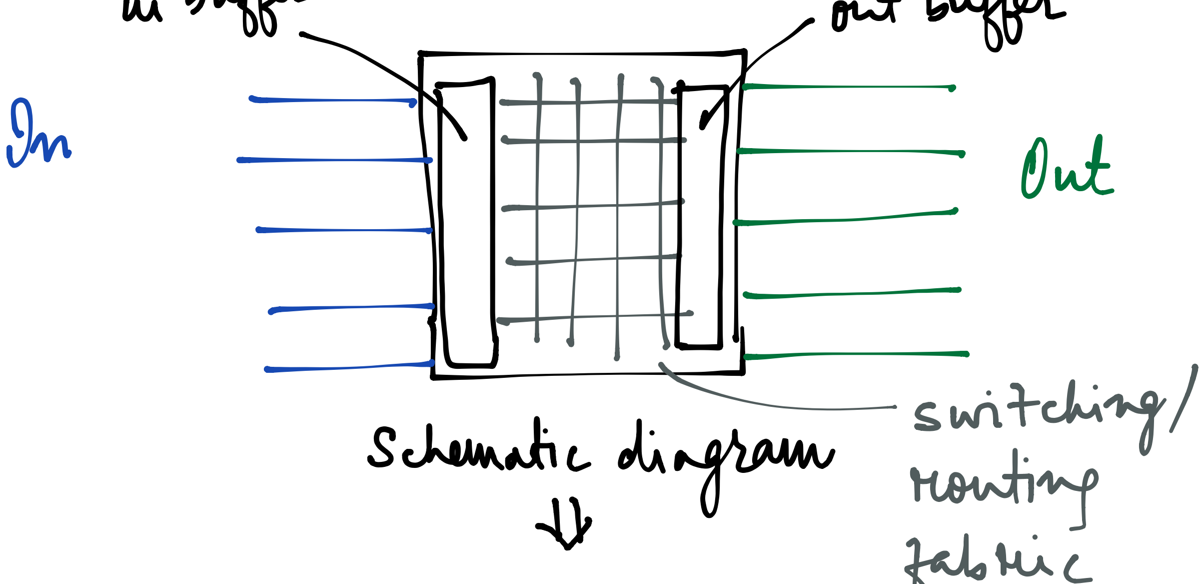


- Ports : 16 bit – identified the Apps
- Selective Repeat sliding window
- Window size
- more details and analysis are topics of advanced networks course.
- Pending topic : congestion control.

Congestion Control

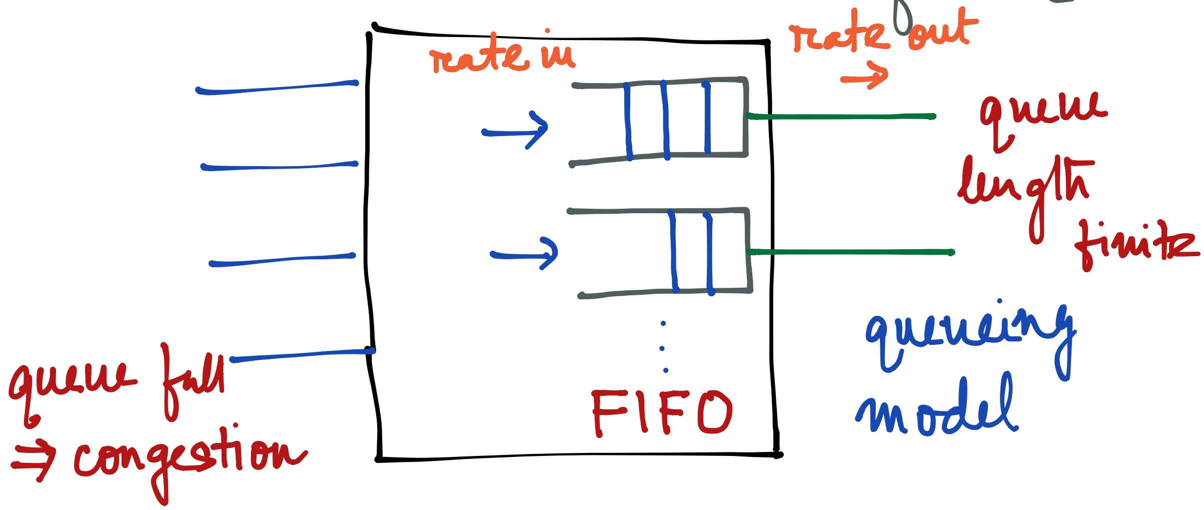
Ex.





Schematic diagram

⇓



rate in < rate out [Good operating point]

rate in ≥ rate out [congestion and overflow]

equality is fragile and avoided (more interesting from analysis viewpoint but limited use in practice).

Congestion depends on

- traffic patterns
- link capacities
- router buffering capacities

Effects

- delay and loss of packets
- throughput reduces - retransmission

How to handle?

Goal: allocate network resources to the users in a "good" way

Efficient: most of the network capacity is used

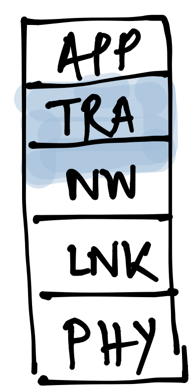
Fair: every sender gets a 'reasonable' share of the capacity

Often conflicting in nature – will see in an example later.

Obs: NW and TRA layers need to work together for congestion control

NW witnesses the congestion

TRA causes the congestion



Network Resource Allocation

Challenges:

1. The network entities, their load are constantly changing
2. network is distributed, no single entity has a complete picture
   - need a decentralized solution

Solution approaches need

1. decentralization – each entity decides based on its local info
2. adaptive – changes based on current state of network
3. ensure efficiency and fairness.

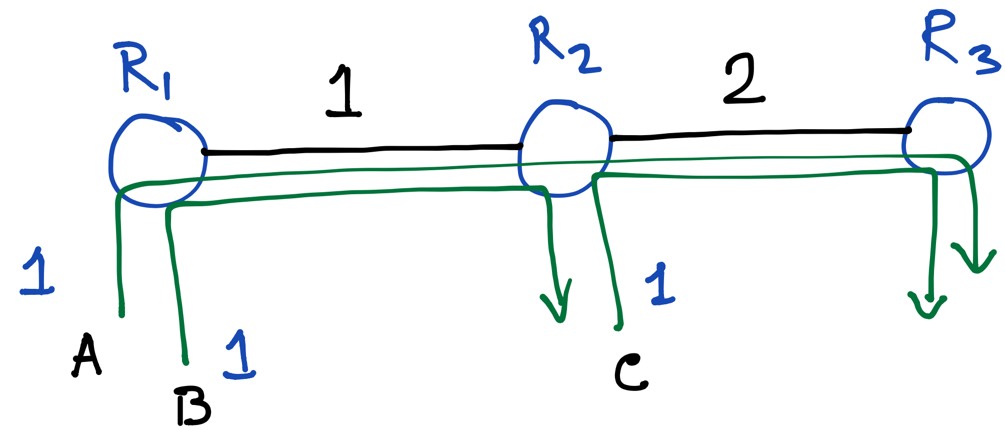
Fairness in Network Resource Allocation

Also applies to several other resource allocation settings.

The tension between Efficiency and Fairness

- Efficiency ≡ Minimal wastage (Maximal used BW)

- Fairness (for now) ≡ Equal per flow



Efficient : A=1, B=0, C=1
used BW = 3

Fair : $A = \frac{1}{2}$, $B = \frac{1}{2}$, $C = \frac{1}{2}$ = Equal per flow

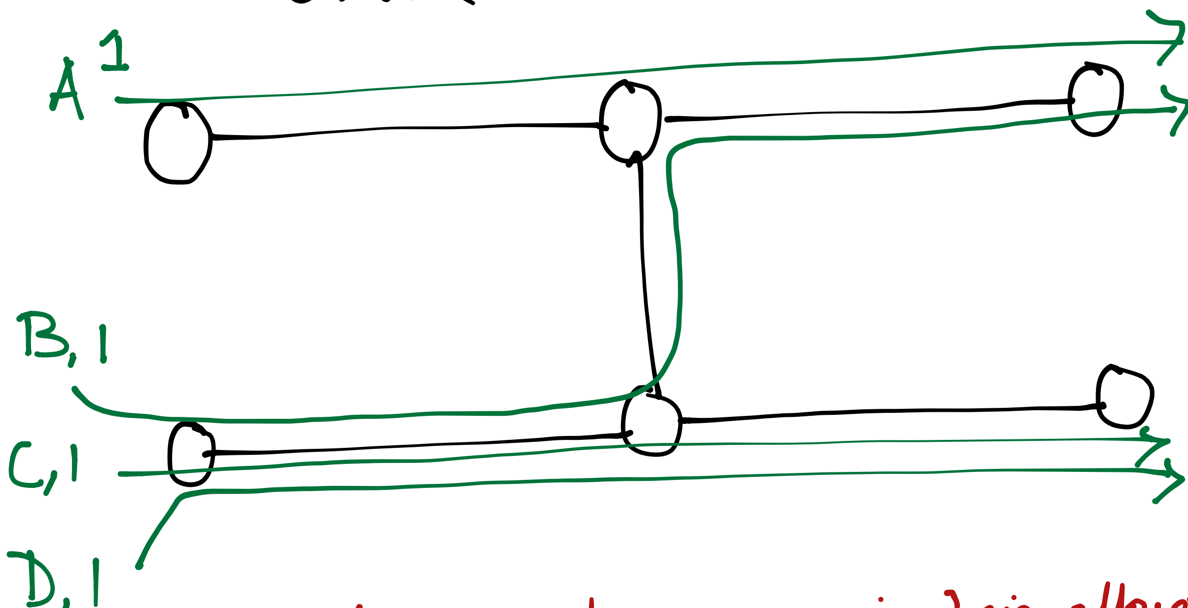
BW usage ratio $= \frac{2}{3} = \frac{\text{Fair}}{\text{Eff.}}$

Extend this example to show it $= \frac{1}{2}$

Different notions of fairness

① Equal per flow

② Max-Min flow –
maximize the minimum flow in the network.



How to compute max-min fair allocation?

– Waterfilling solution

① – start all flows with zero

② – increase unfrozen flow at same rate
- freeze the rate when capacity is reached

③ – go back to ②

Max-min solution

$A = \frac{2}{3}$, $B = C = D = \frac{1}{3}$

BW used

$\frac{2}{3} + 1 + \frac{1}{3} + 1 + \frac{2}{3}$

Efficient solution

$A = 1$, $C = 1$, $B = D = 0$ or similar

$1 + 1 + 0 + 1 + 1$

These (flow based) notions are global notions and difficult to satisfy, as network devices operate locally.

In practice, a local level solution is preferred, e.g., at a router since it can provide decentralization

— fairness per incoming link
efficiency at a router.