

TCP Versions: Tahoe

Kameswari Chebrolu

All the figures used as part of the slides are either self created or from the public domain with either 'creative commons' or 'public domain dedication' licensing. The public sites from which some of the figures have been picked include:

<http://commons.wikimedia.org> (Wikipedia, Wikimedia and workbooks); <http://www.sxc.hu> and <http://www.pixabay.com>

Recap

- TCP relies on a variety of techniques to achieve congestion control
 - Slow start → filling the pipe
 - RTT estimation → conservation
 - Congestion avoidance (AIMD) → adapting
- Lets put it all together

TCP Tahoe

- Basic go-back-N with slow start, congestion avoidance
- Uses two variables cwnd (congestion window) and ssthresh (slow-start threshold)
- At the beginning: slow-start phase
 - cwnd=1;
 - When new data is acked, $cwnd += 1$
- Bound to overestimate leading to loss of data

$$w = 25$$

1, 2, 4, 8, 16, 32
↑

Estimate = 16

→ Slow Start

TCP Tahoe

- On time out: Congestion avoidance

- set ssthresh = $\text{cwnd}/2$; cwnd = 1

32
↓
→ cwnd = 16
↑

- When new data is acked,

- If ($\text{cwnd} < \text{ssthresh}$) $\text{cwnd} += 1$;

ack

- Else $\text{cwnd} += 1/\text{cwnd}$;

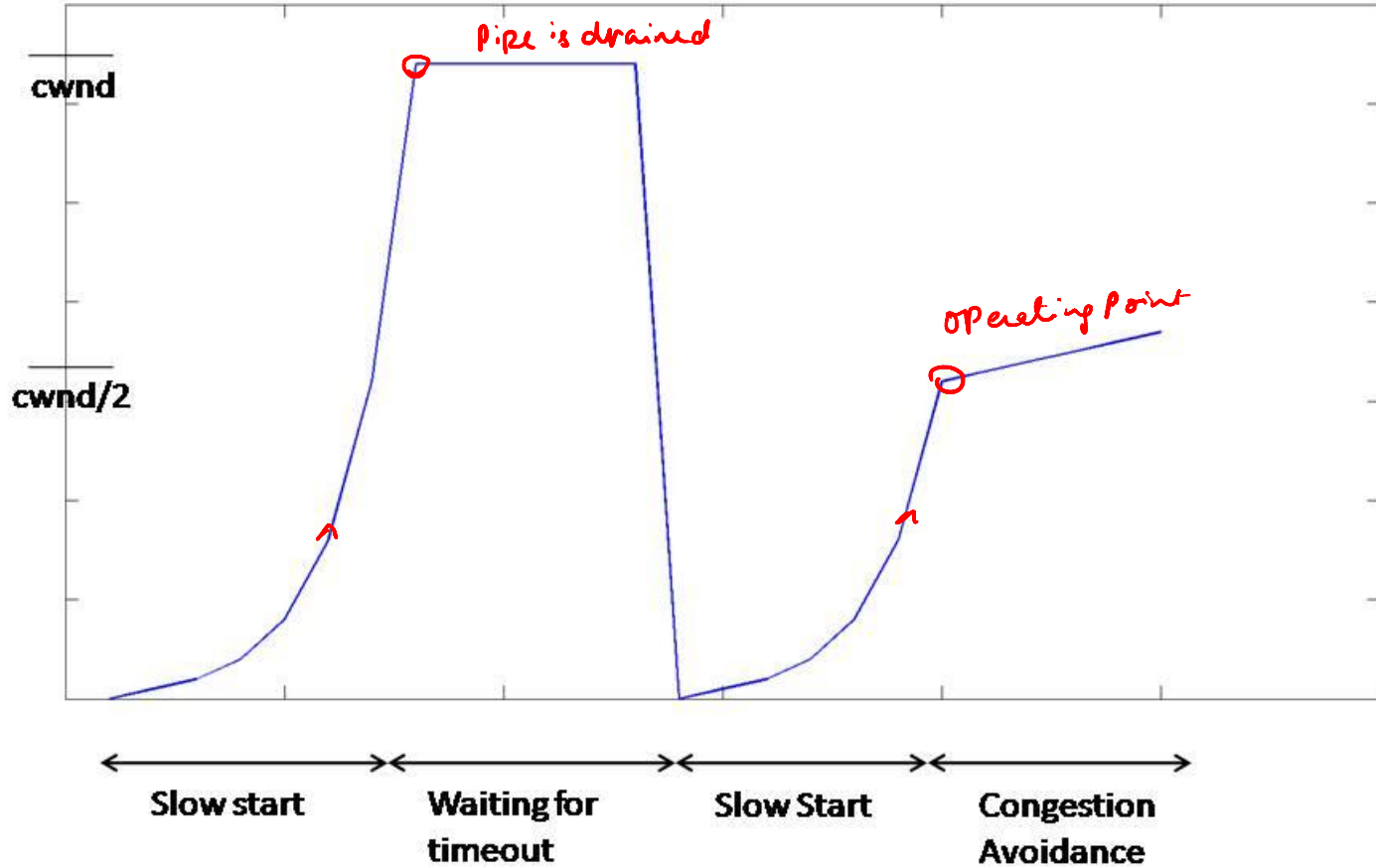
ack

ignore → duplicate

16
1, 2, 4, 8, 16
↑

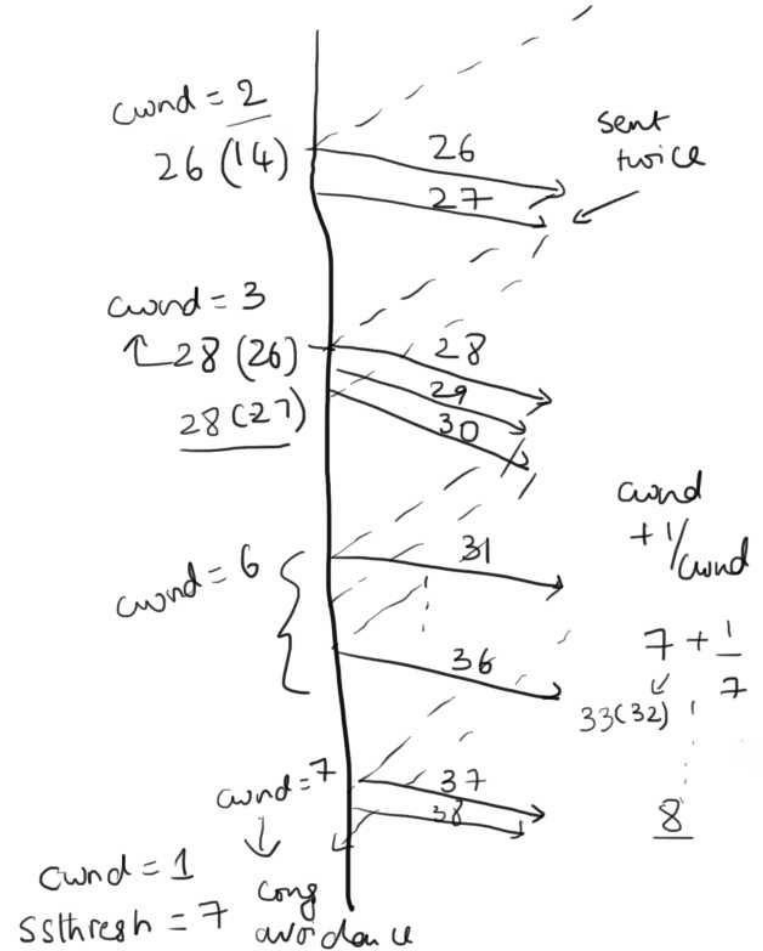
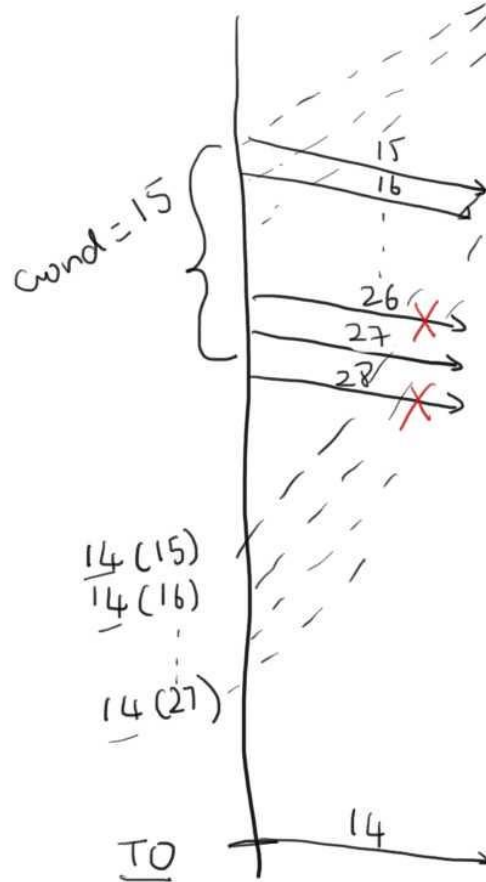
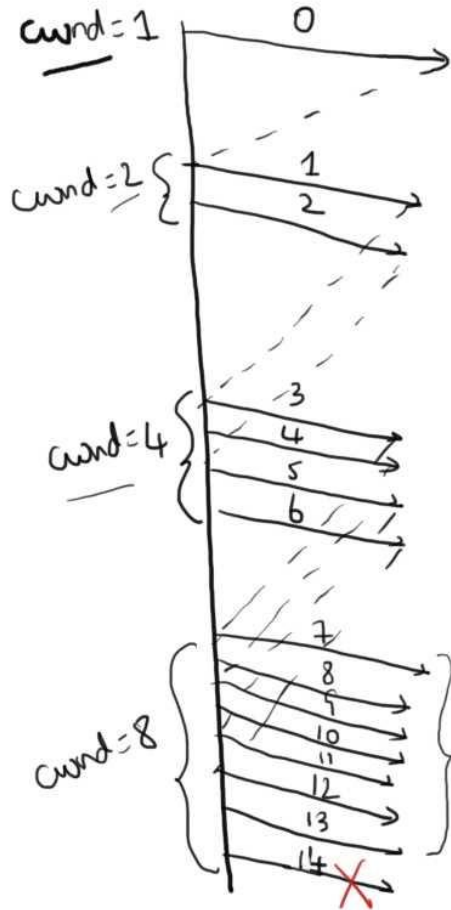
additive increase

16, 17, 18, ... 25



x(Y) → who generated
 ↳ asking for

TCP Tahoe (Not drawn to scale)



TCP Tahoe Drawbacks

- Slow recovery of losses
- Timeouts drain the pipe → Forces one to do slow start which takes time to fill the pipe

Break

