TCP Sendbuffer Advertising

Costin Raiciu

University Politehnica of Bucharest

Problem statement

- There is only so much we can find about about a connection by looking at in flight packets (losses, retransmissions, RTT, etc.)
  - sFlow – packet sampling
  - Netflow – aggregate statistics
  - For anything else, it gets expensive

Problem statement

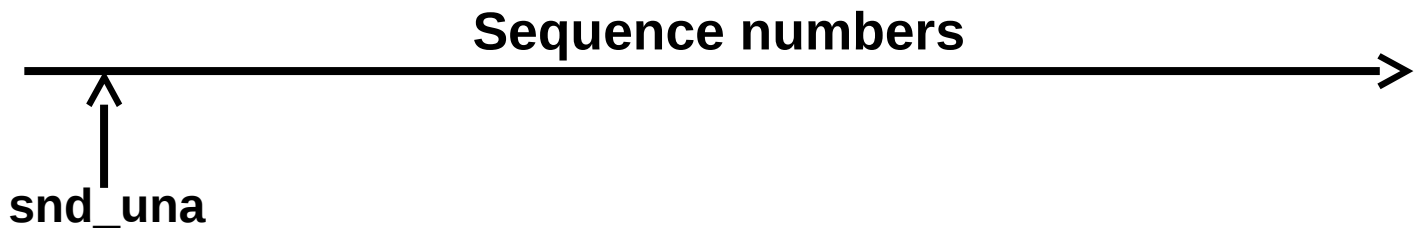
- There is only so much we can find about about a connection by looking at in flight packets (losses, retransmissions, RTT, etc.)
  - sFlow – packet sampling
  - Netflow – aggregate statistics
  - For anything else, it gets expensive

- ***Is the connection limited by the network ?***

What if we advertised send buffer occupancy inside TCP segments ?

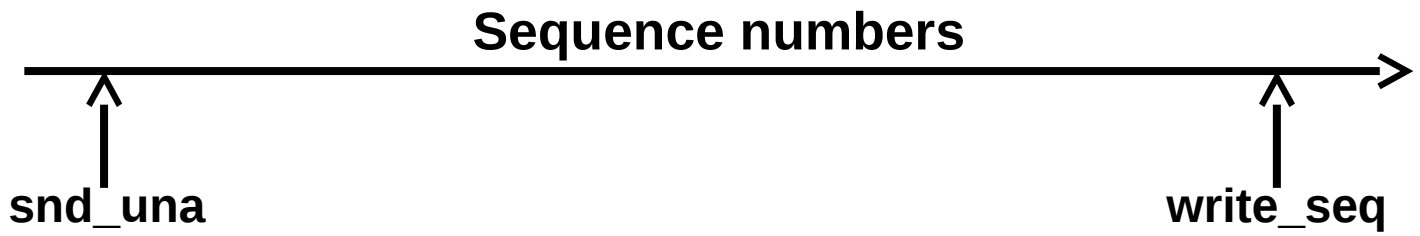
What exactly do we advertise ?

What exactly do we advertise ?



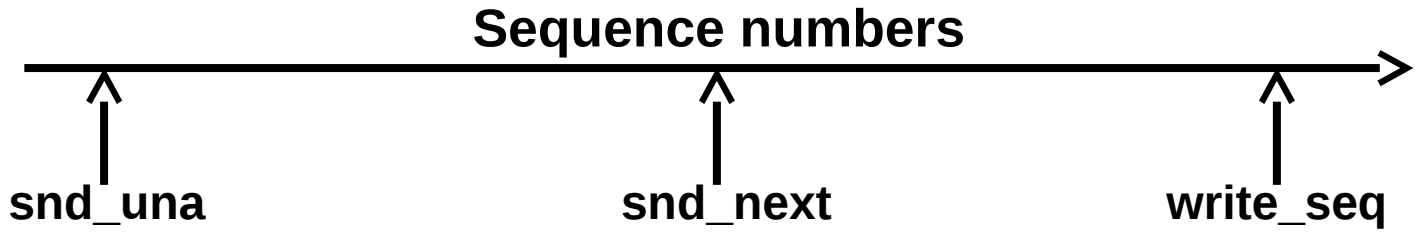
first unacknowledged
sequence number

What exactly do we advertise ?



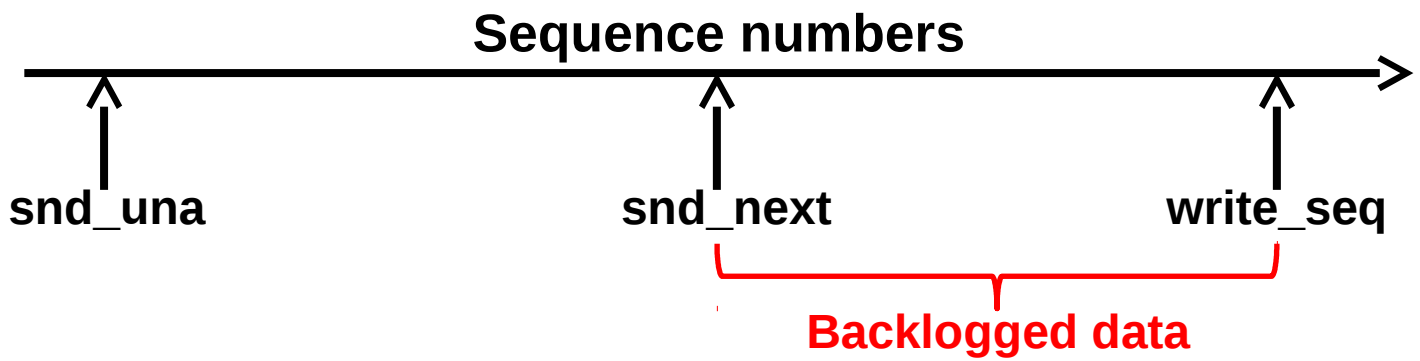
sequence number of
the last written byte

What exactly do we advertise ?



sequence number for the next packet to be sent

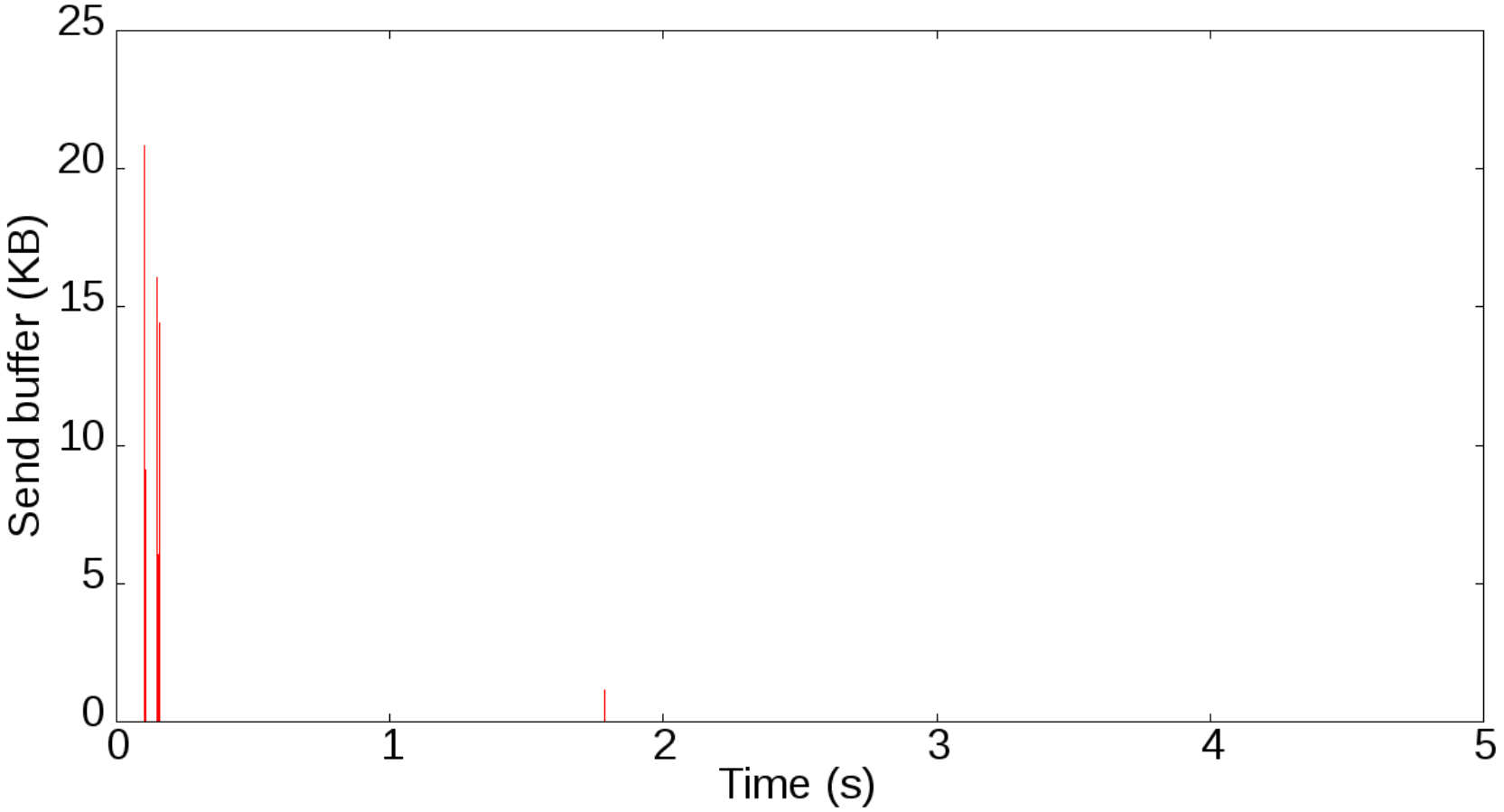
What exactly do we advertise ?



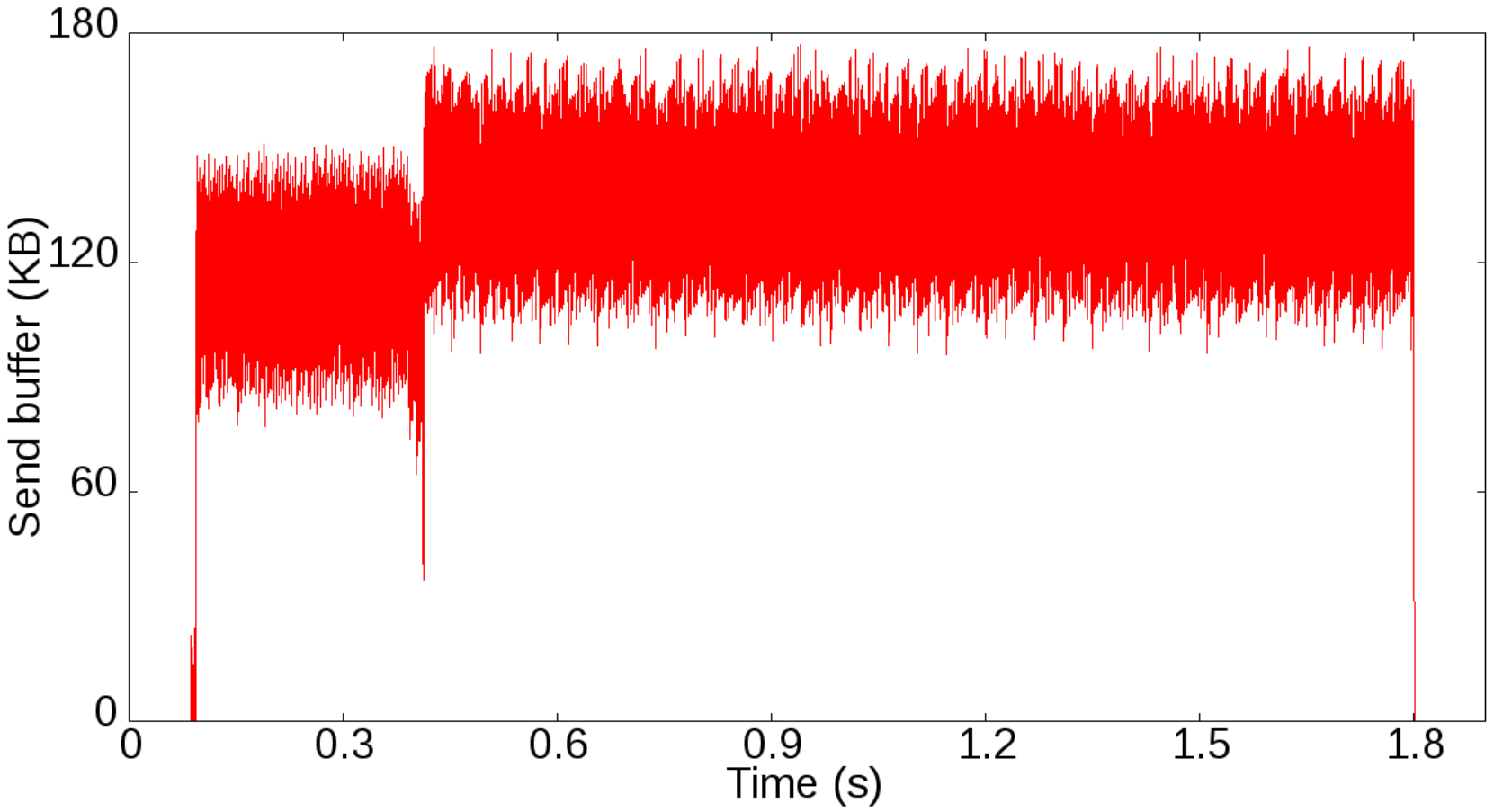
Why do we do it ?

- Backlogged applications are usually network-limited (unless receive window limited or facing very rare issues)
- Advertising the backlog size is more informative than checking a binary threshold

Disk bound transfer



Network bound transfer



Use cases

Detecting network hotspots

- *High loss rate = congestion ?*

Detecting network hotspots

- *High loss rate = congestion ?*

Not really! Example: **incast**

Detecting network hotspots

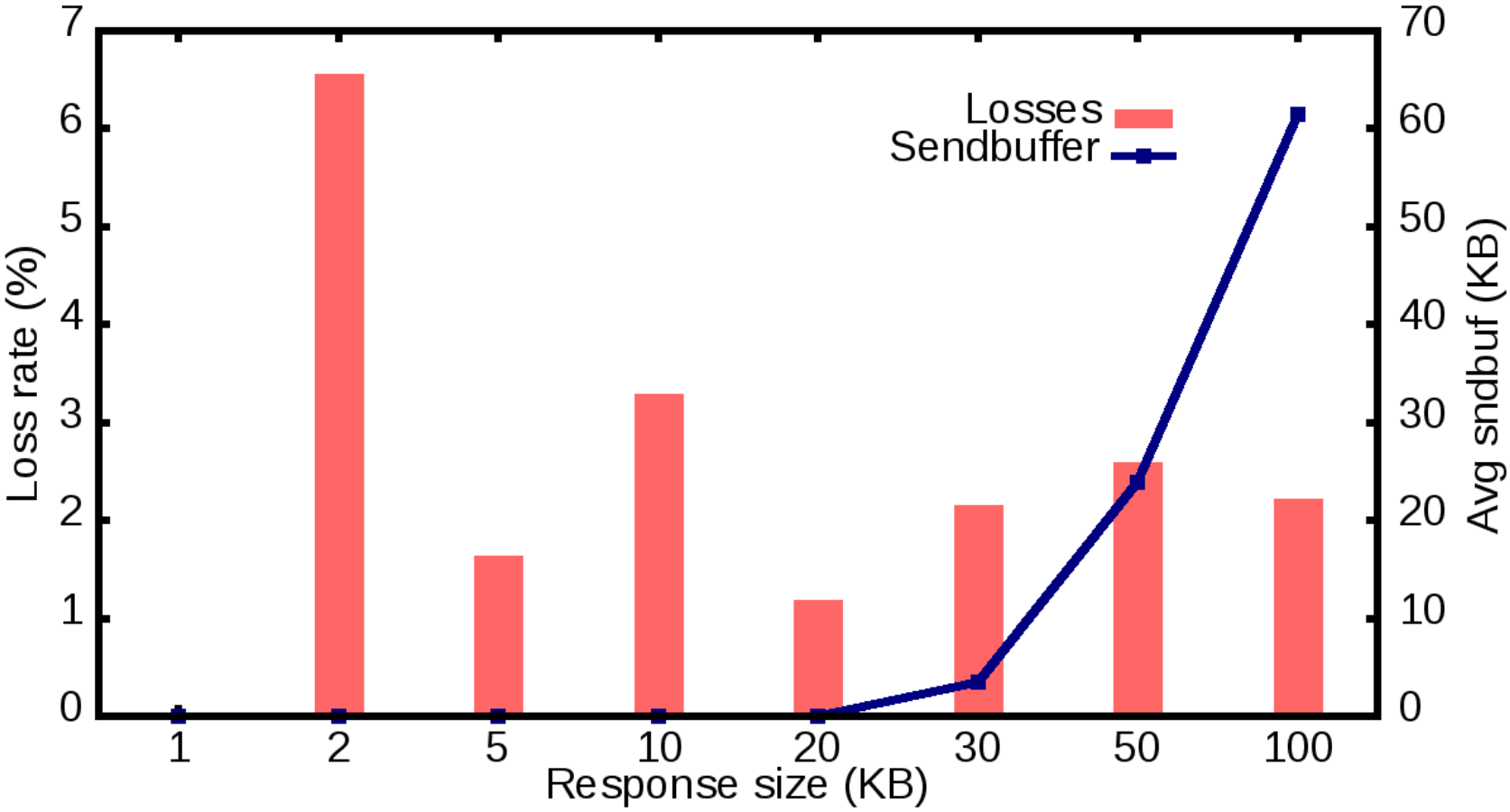
- *High loss rate = congestion ?*

Not really! Example: **incast**

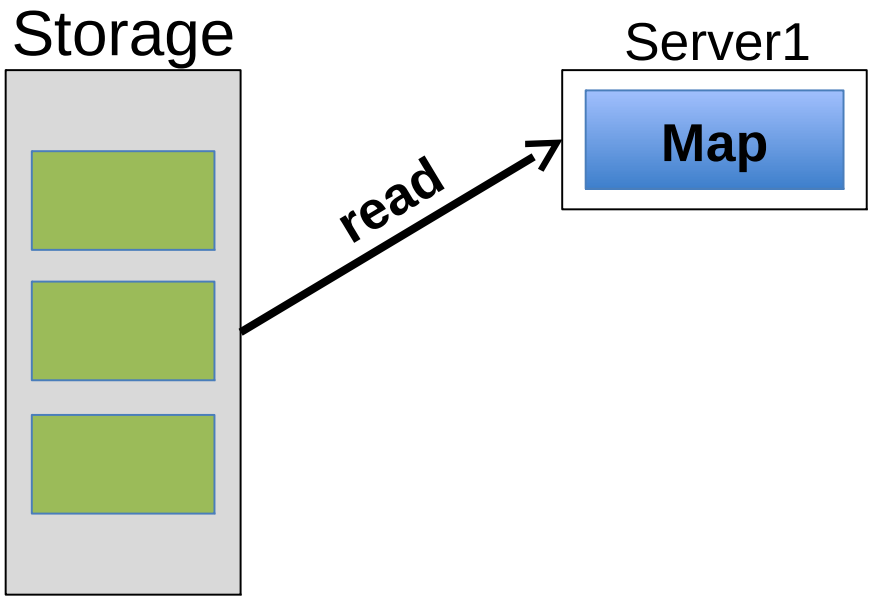
- EC2 incast scenario:

- 99 synchronized senders and a single receiver
- variable transfer size per round
- average loss rate ~2.5%

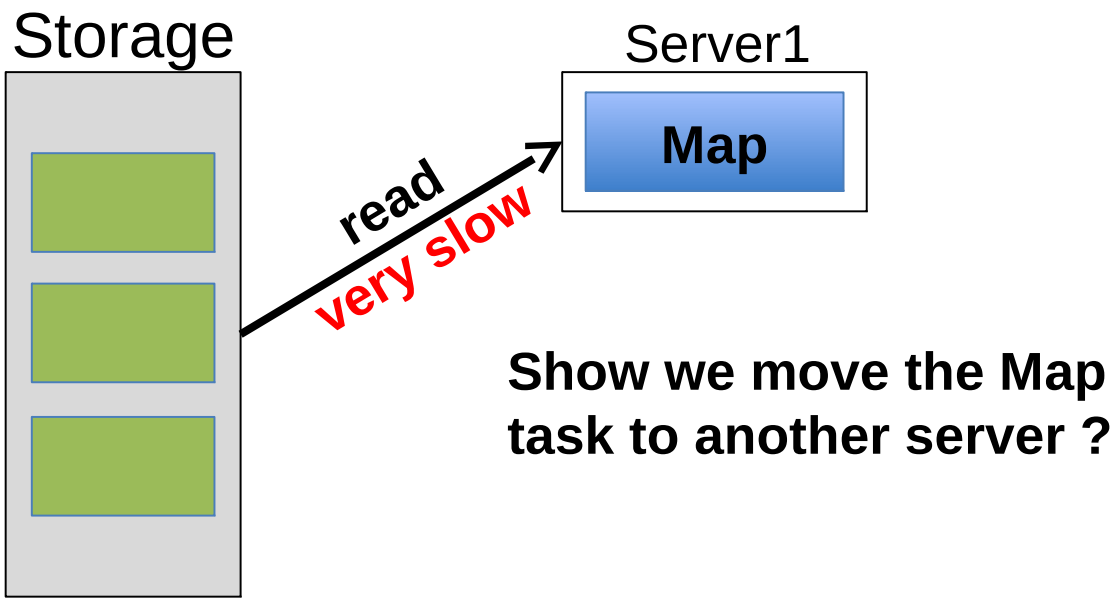
Incast results



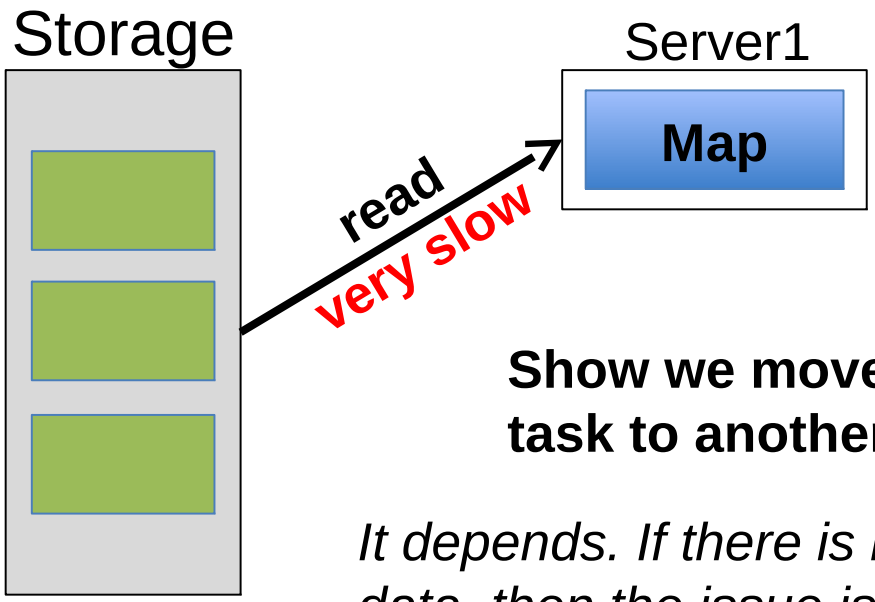
Helping datacenter applications



Helping applications



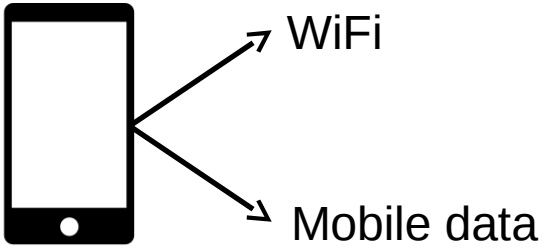
Helping applications



Show we move the Map task to another server ?

It depends. If there is no backlogged data, then the issue is at the storage node.

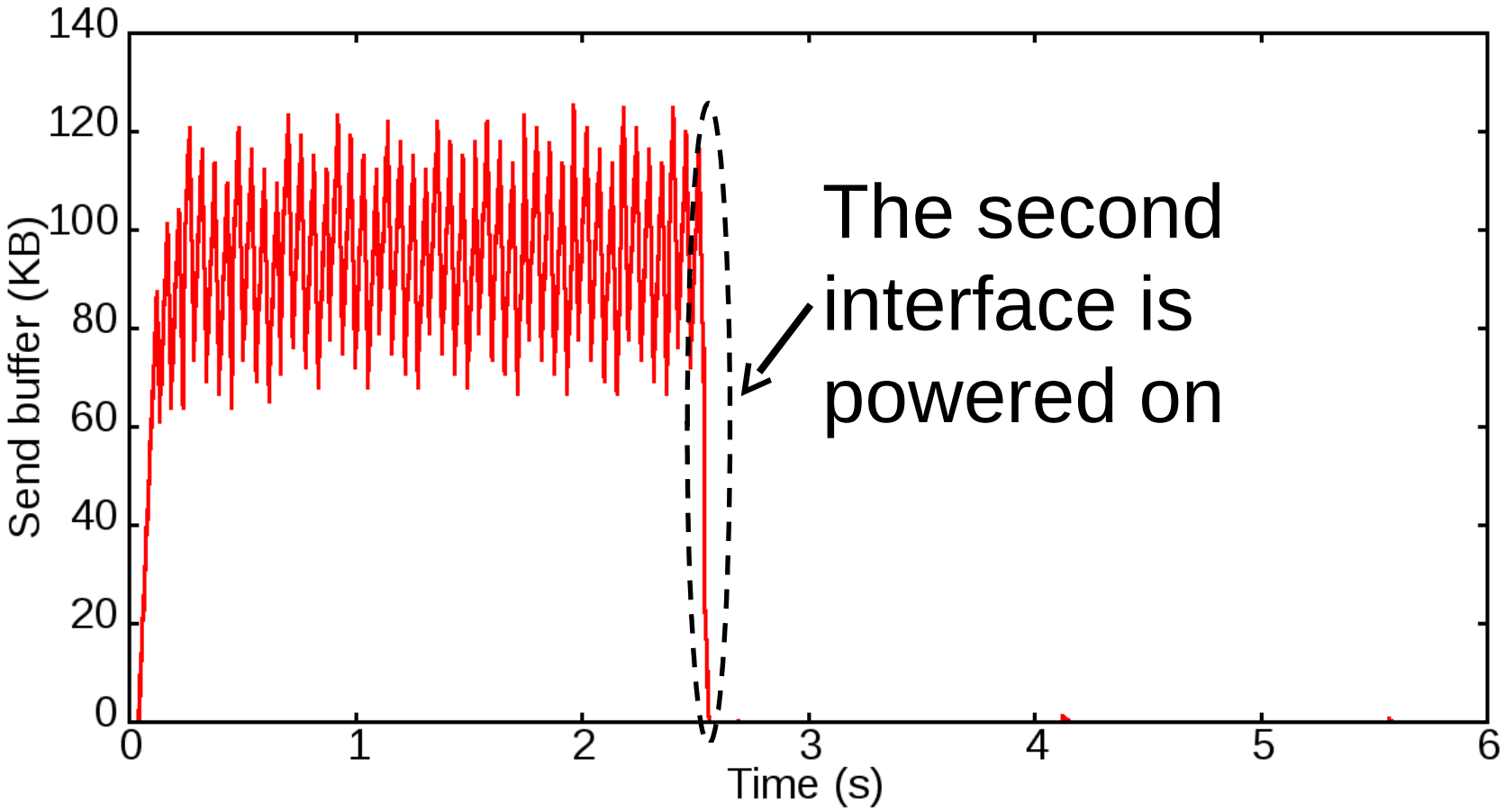
Improving mobile performance



Mobile data is generally not used if a WiFi network is available.

Some applications (video streaming for example) may benefit from also using the other interface, especially in poor network conditions.

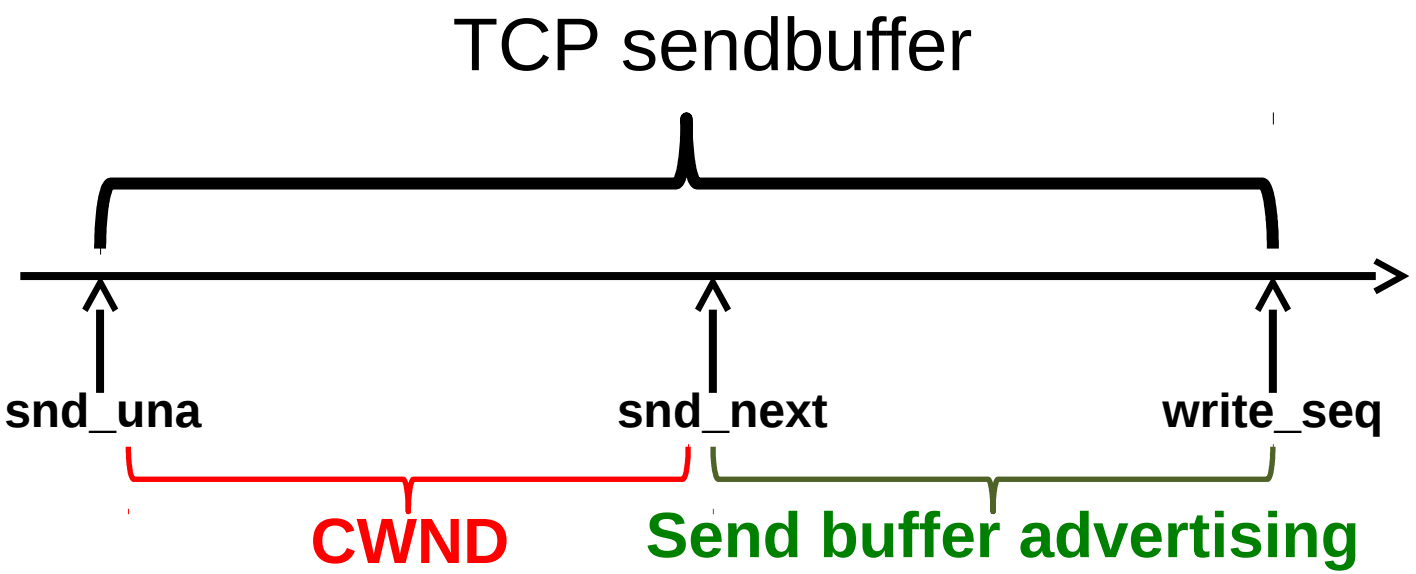
Improving mobile performance



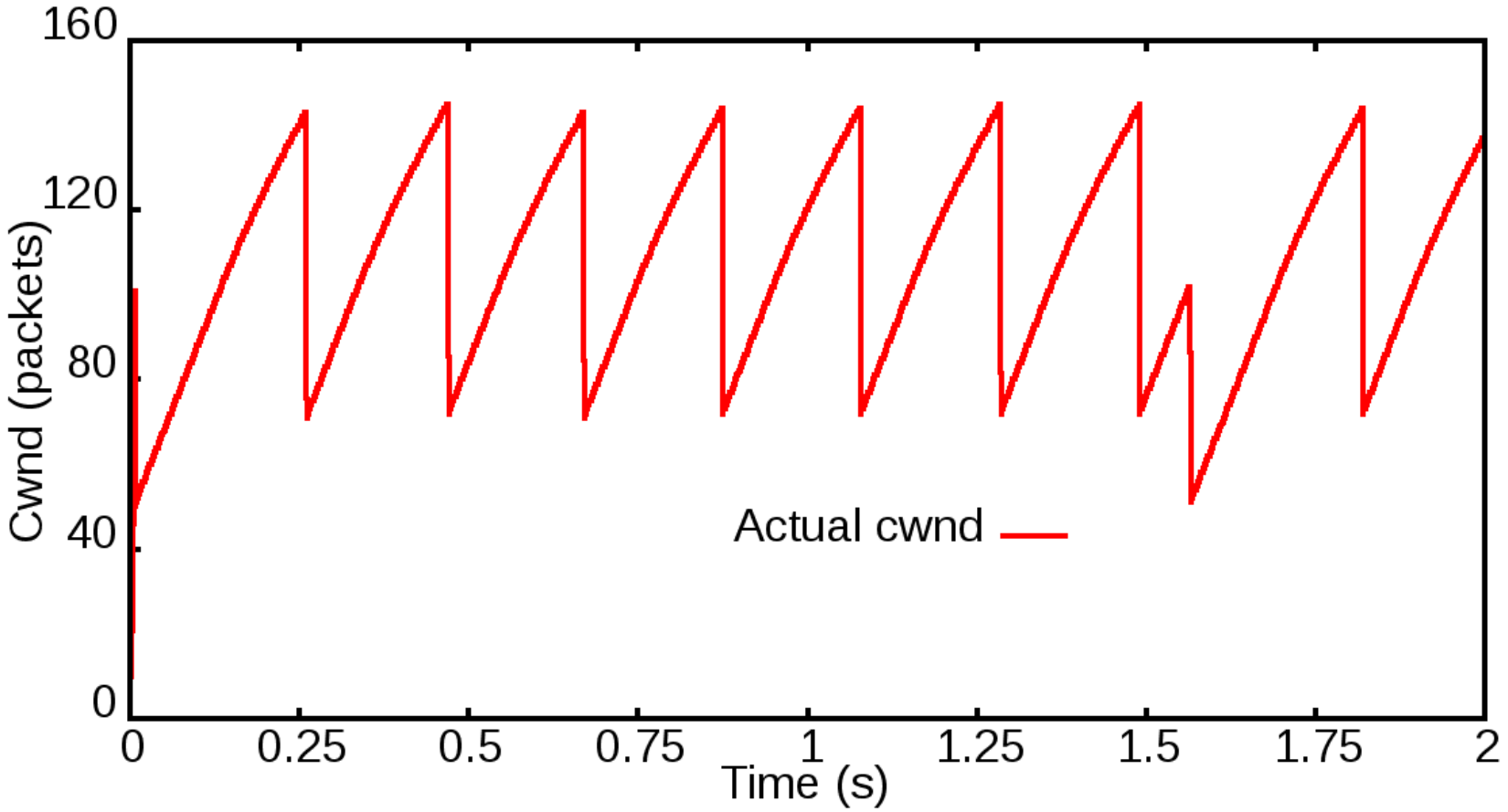
Troubleshooting flow performance

- Using of sendbuffer information to infer other flow characteristics
- For example, we try to estimate the presence of congestion events by analysing the evolution of the sendbuffer

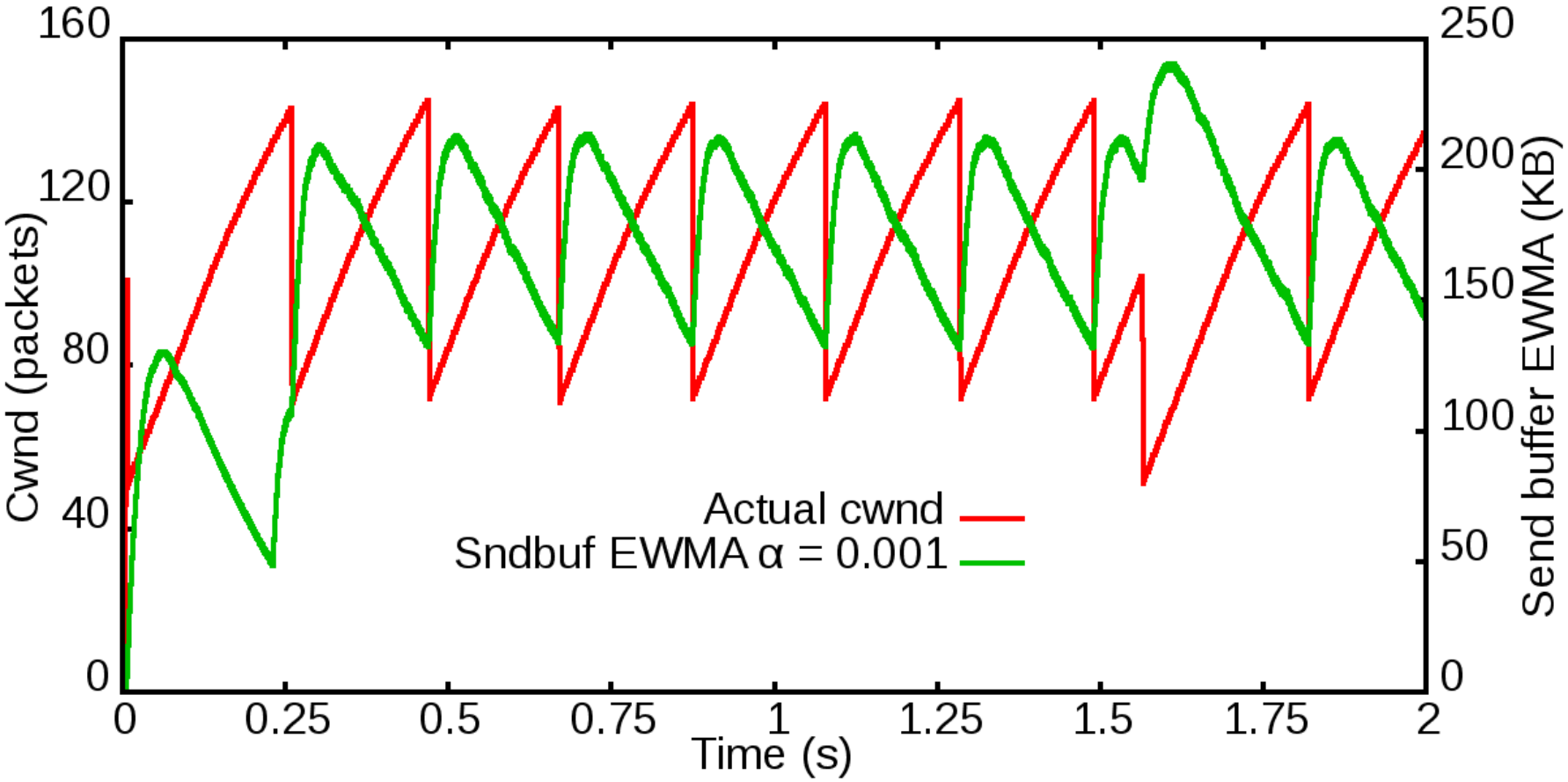
Sendbuffer information is a proxy for the congestion window



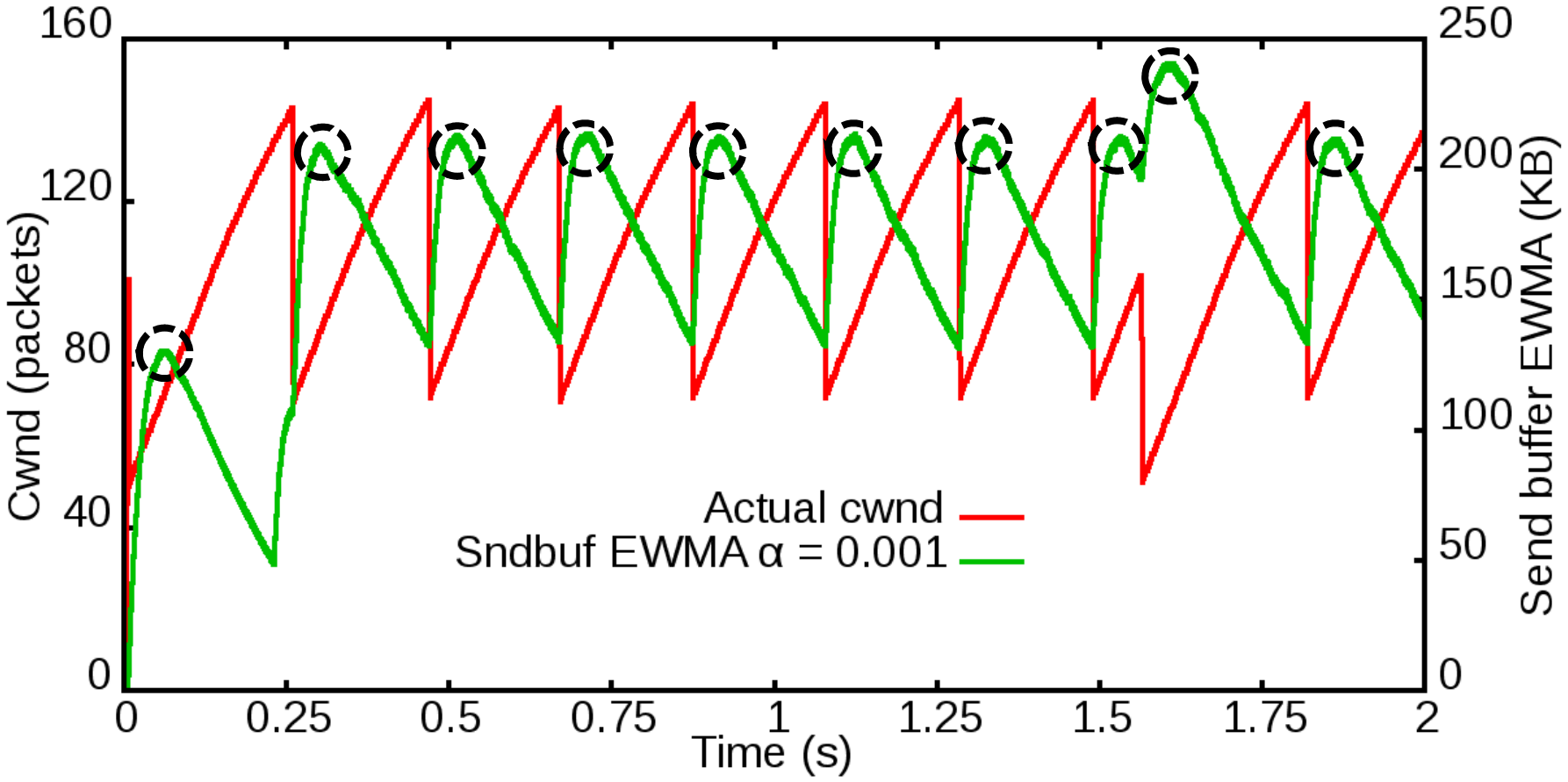
Inferring congestion events



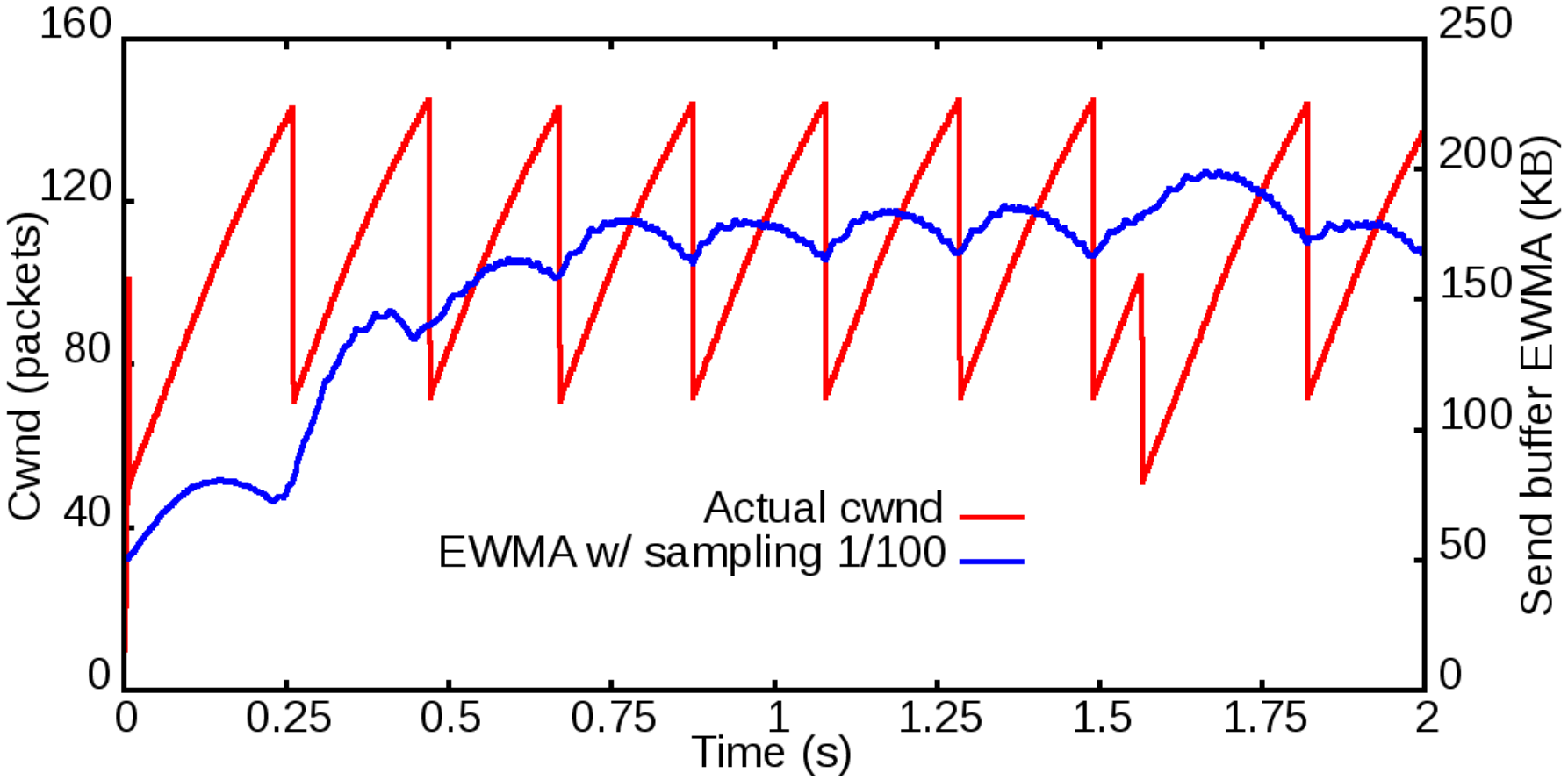
Inferring congestion events



Inferring congestion events



Inferring congestion events



Encoding sendbuffer information

Negotiating sendbuffer advertising

- New TCP option in the SYN handshake?

- Sender-only change
- Receiver or network can use information only if they know the standard
- Just need to ensure legacy boxes are not affected

Negotiating sendbuffer advertising

- New TCP option in the SYN handshake?

Send buffer advertising is not negotiated at all
Sender system-wide configuration decides if it
s used or not

- Just need to ensure legacy boxes are not affected

Information encoding in data segments

Use a TCP option

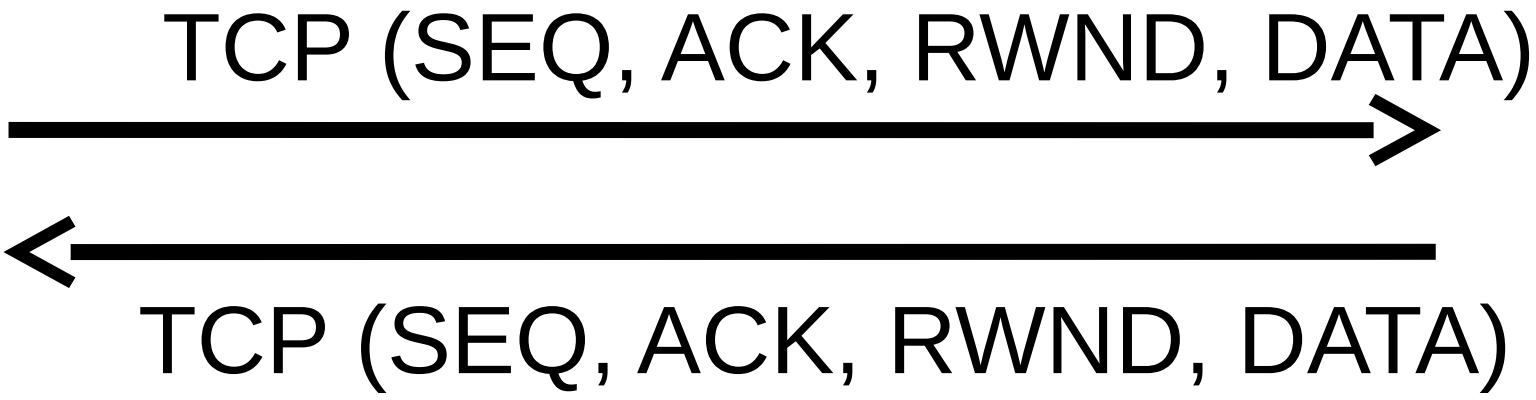
Adds overhead and we don’t have much space in the options field

Should work just fine in the Internet

- Middleboxes either allow unknown options or just scrub them
- Receiver stacks should just ignore unknown options (need to check, though)

Information encoding in data segments

data traffic is (mostly) uni-directional



Information encoding in data segments

data traffic is (mostly) uni-directional

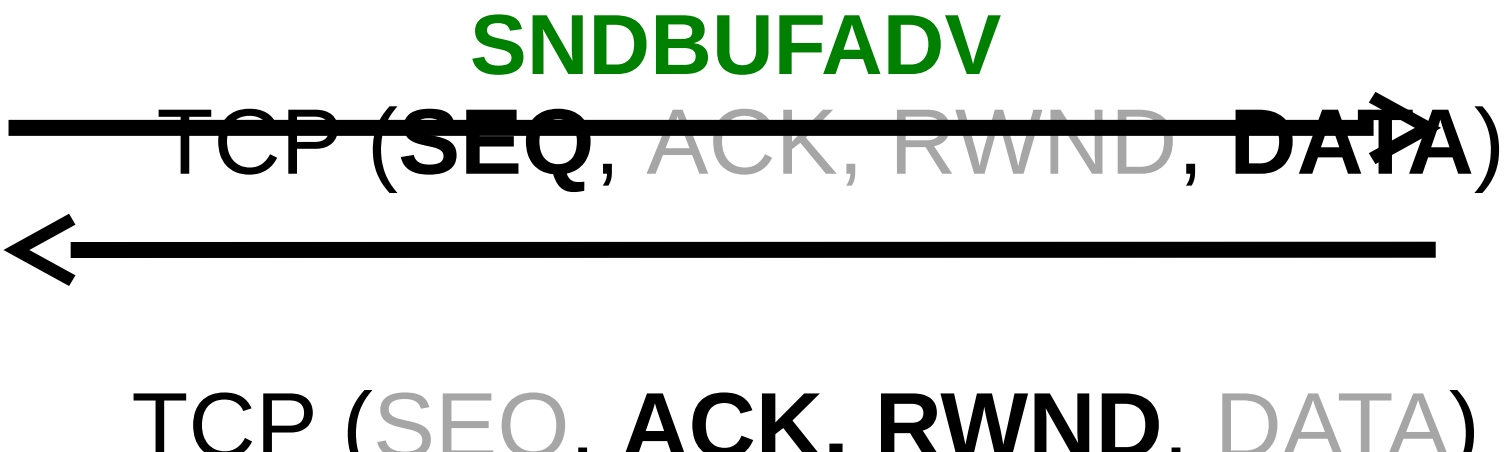
TCP (**SEQ**, ACK, RWND, **DATA**)

TCP (SEQ, **ACK, RWND**, DATA)

Information encoding in data segments

Reuse ACK and RWND fields to encode

sendbuffer information



Information encoding in data segments

Case 1: encoding for datacenters

- Redefine one reserved flag as sendbuffer advertising flag
- To encode sendbuffer advertising:
  - Disable the ACK flag
  - Set the SNDBUF flag
  - Encode value in the ACK field

Information encoding in data segments

Case 2: Internet

- To encode sendbuffer advertising:
  - Disable the ACK flag
  - Encode **value** in the ACK field
  - Encode **HMAC of value** in the RWND field

Conclusions

- Having sendbuffer information in TCP segments can prove useful in many situations
- It can be encoded in every segment without any overhead in terms of space

Documents

- HotCloud 2015 paper
- *draft-ietf-tcpm-agache-sndbufadv-00*