Latency Reducing TCP modifications for thin-stream interactive applications

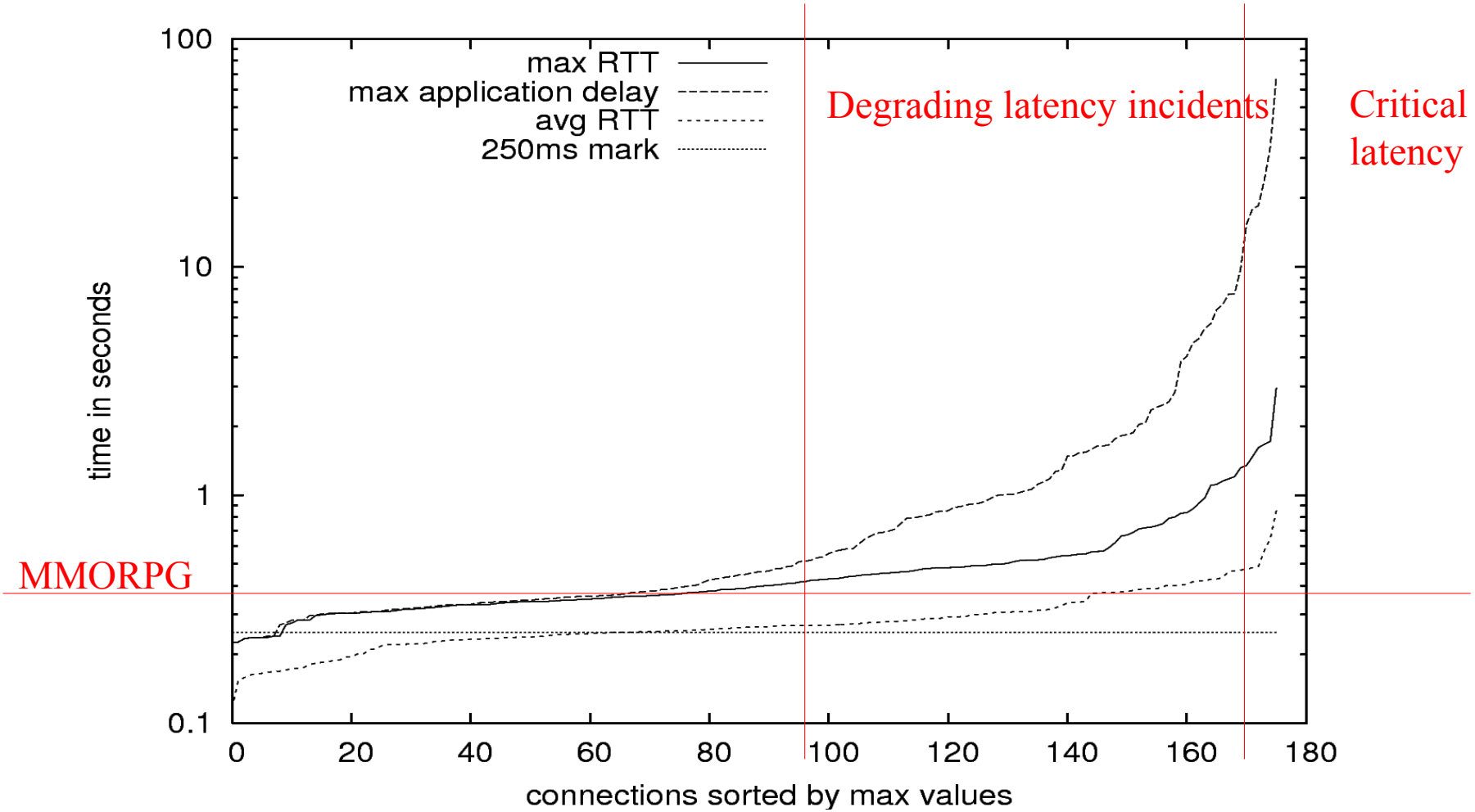
Andreas Petlund / Kristian Evensen
Simula Research Laboratory /
University of Oslo

Linux–Kongress 2008 - Hamburg - Germany

A long time ago in a game-company far, far away...

- Anarchy Online MMORPG server side packet trace from Funcom (171 streams, 1 hour).
- Extreme max. values for latency.
- Most of the streams experienced extreme (game degrading) latencies during the dump period.
- Occasional game ruining latencies (3-4% of clients).

Max delay values for Anarchy Online



Griwodz et al.: "The fun of using TCP for an MMORPG" (2006)

TCP and thin streams

- High interarrival time.
- Small packets.
- Optional kernel mechanisms that affect thin streams:
  - Nagle: Wait for small packets to assimilate.
  - Delayed ACKs: Save ACKs by waiting for more segments to arrive.
- Both of these increase latency for thin streams.

Examples of thin-stream applications

application (platform)	payload size (Bytes)			packet interarrival time (ms)						avg. bandwidth requirement	
								percentiles			
	average	min	max	average	median	min	max	1%	99%	(pps)	(bps)
World of Warcraft	26	6	1228	314	133	0	14855	0	3785	3.185	2046
Anarchy Online	98	8	1333	632	449	7	17032	83	4195	1.582	2168
Age of Conan	80	5	1460	86	57	0	1375	24	386	11.628	12375
BZFlag†	30	4	1448	24	0	0	540	0	151	41.667	31370
Casa (sensor network)	175	93	572	7287	307	305	29898	305	29898	0.137	269
Windows remote desktop	111	8	1417	318	159	1	12254	2	3892	3.145	4497
Skype (2 users)†	236	14	1267	34	40	0	1671	4	80	29.412	69296
SSH text session	48	16	752	323	159	0	76610	32	3616	3.096	2825

† Application using TCP fallback due to UDP being blocked by a firewall.

Analysis of TCP for thin streams

- Linux TCP flavours (2.6.16) analysed:

Griwodz et al.: “The fun of using TCP for an MMORPG” (2006)

- New Reno -SACK -DSACK -FACK
- DSACK+FACK -Westwood -BIC -Vegas

- Poor overall performance for interactive thin streams with all tested flavours.
  - New Reno best “on average” for thin-stream latency.

It's all about timeouts

- Methods of triggering retransmissions:
  - Timeout -Fast retransmit
- 3 dupACKs needed to trigger a fast retransmission.
- Thin streams mostly stay below 1 packet per RTT.
- In effect for thin streams: “Only” retransmissions by timeout.

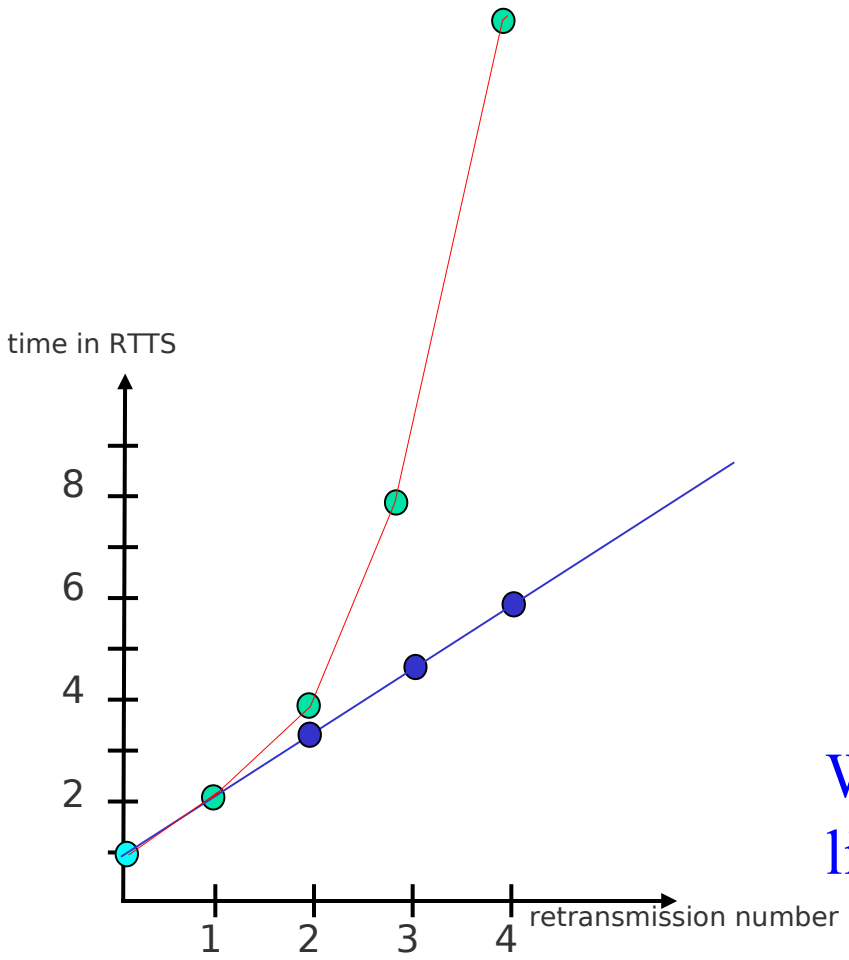
Thin-stream modifications

- We have developed ways to improve latency for the thin-stream scenario without affecting other streams.
- Detection:
  - Packets in flight (PIF) <= 4
  - size_unacked(p1) + size(p2) < MSS
- Modifications triggered only when these conditions are met.

IOCTL enabling of mechanisms

- Activate mechanism on a per-stream basis using socket options.
- Options:
  - TCP_THIN_RM_EXPB
  - TCP_THIN_DUPACK
  - TCP_THIN_RDB
- The dynamic triggering of the thin-stream mechanism will then be active.

Exponential backoff

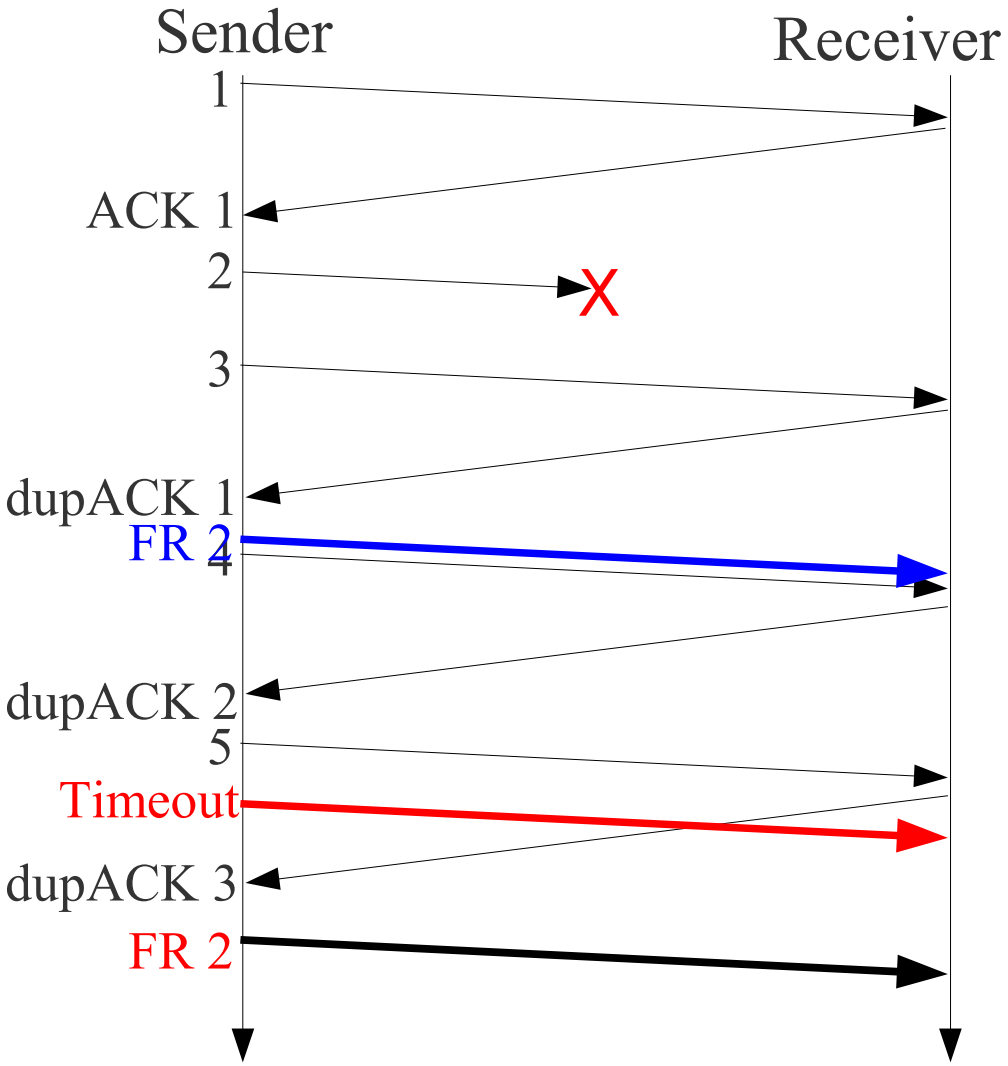


Lost packet

1. retransmission
2. retransmission
3. retransmission
4. retransmission

When thin streams are detected, linear timeouts are used.

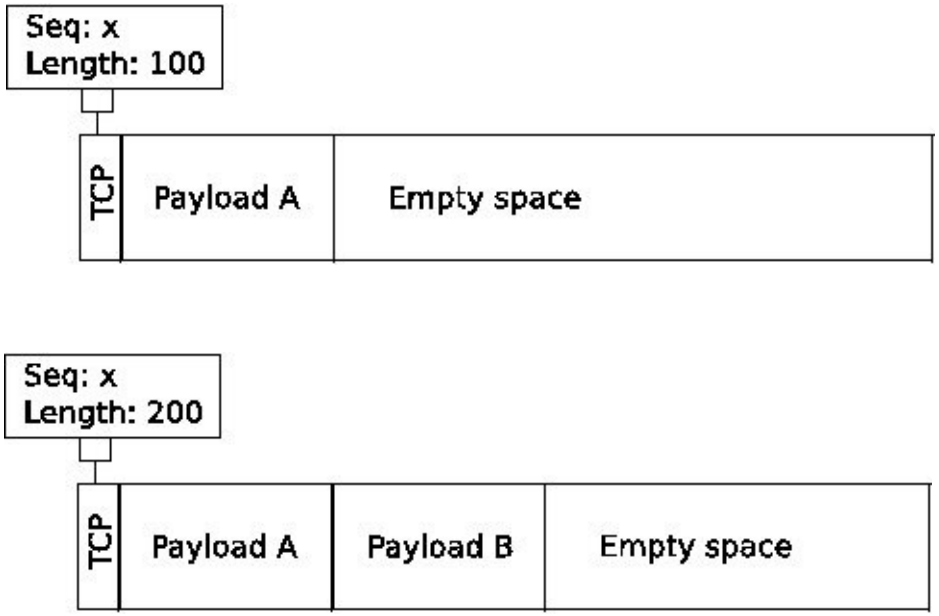
Fast retransmit with thin-streams



- Thin streams often have < 1 packet per RTT.
- Before 3 dupACKs has arrived, a timeout will already have triggered a retransmission.
- When thin streams are detected, we trigger a FR after one dupACK.

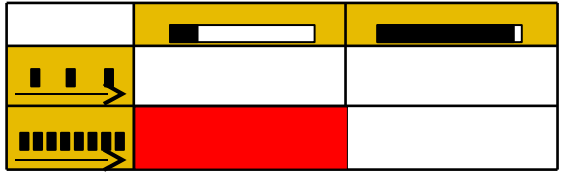
Redundant data bundling

- Preempting the experience of loss.
- Introduces inherent redundancy.
- Will not increase number of sent packets.

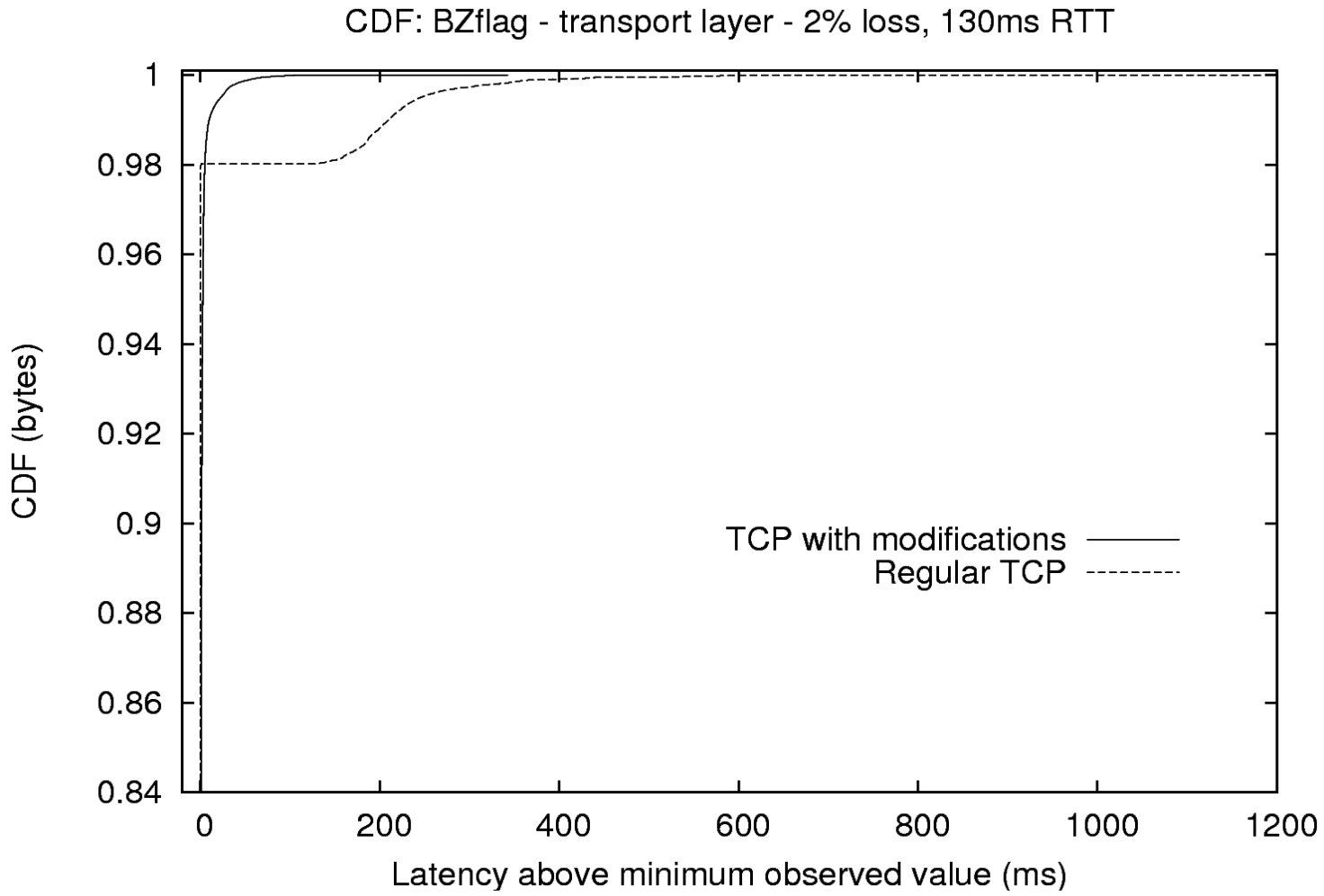


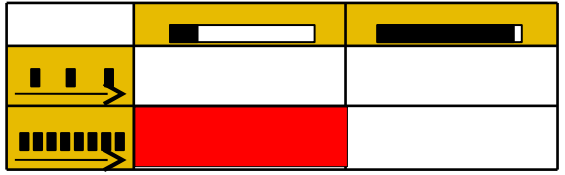
Applicability of modifications

	Small Packets 	Large Packets 
High IA 	*Typical thin stream* **RDB + Retransmission modifications**	*Rare occurrences* **Retransmission modifications**
Low IA 	**RDB**	*Thick* **No modifications**

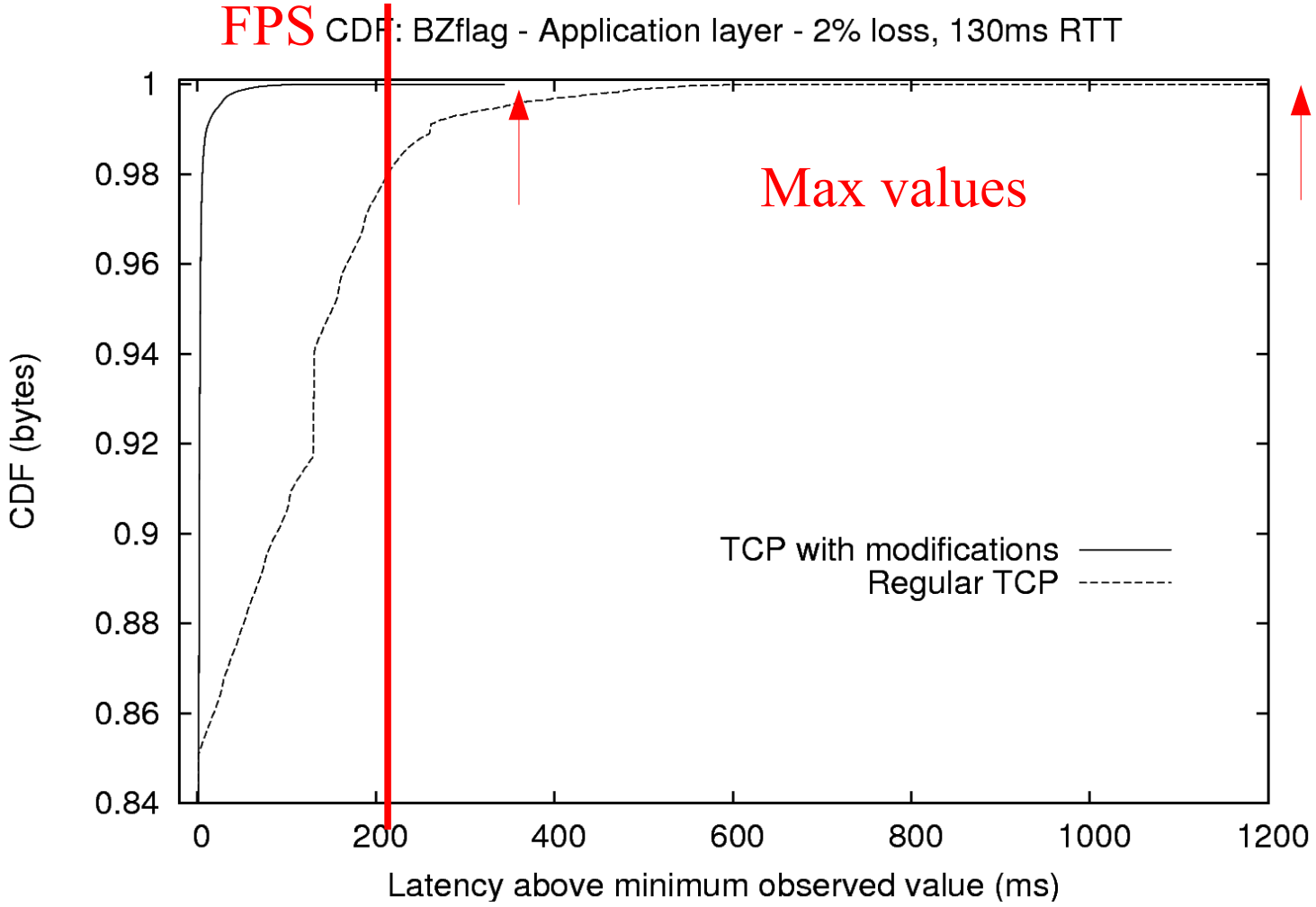


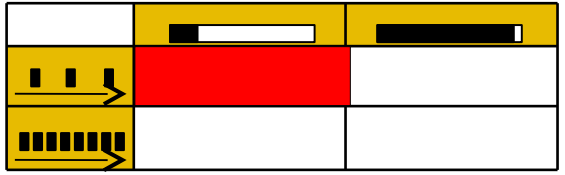
Test results: BZFlag – Transport layer



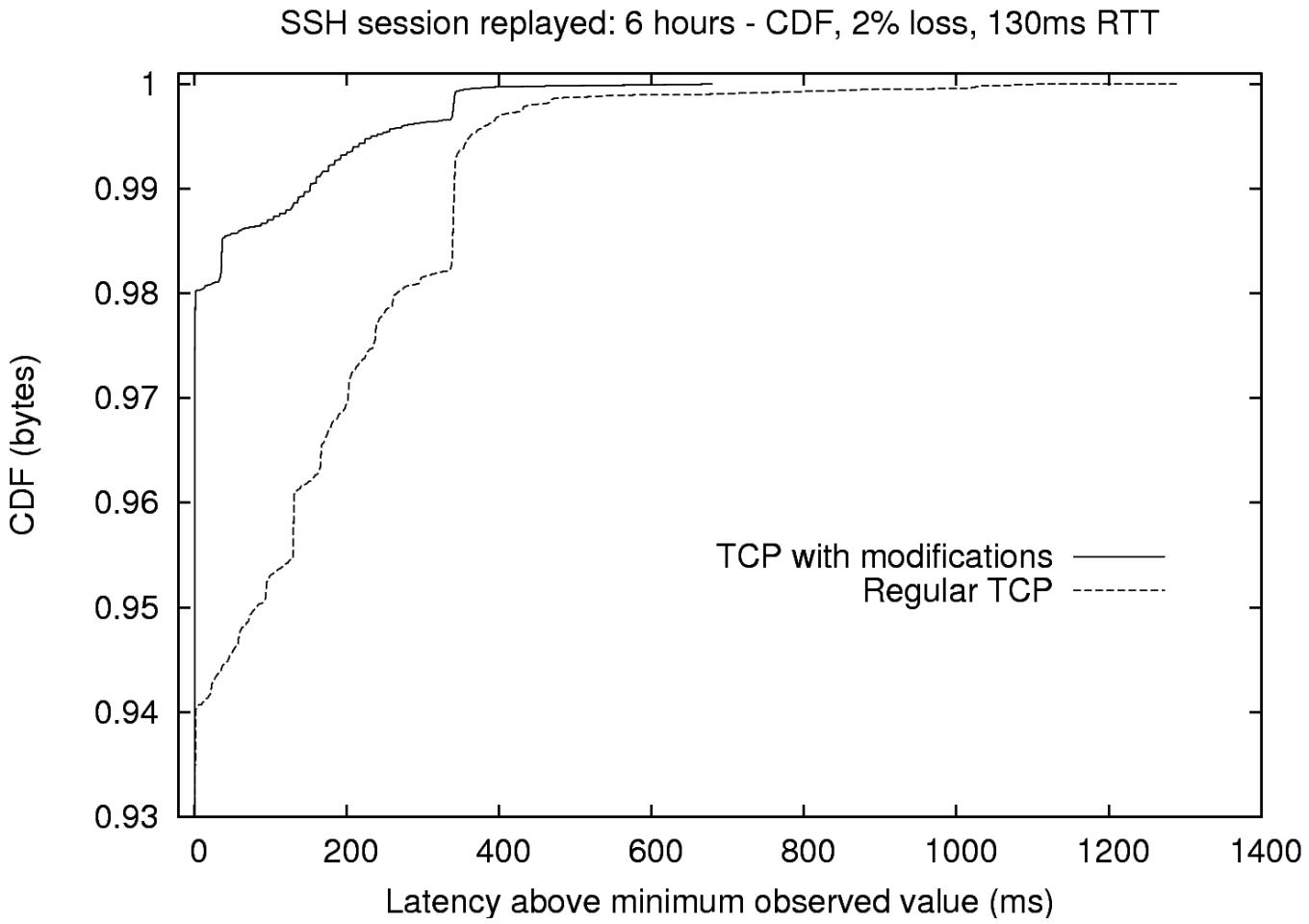


Test results: BZFlag – Application layer





Test results: SSH text session



Considerations

- Fairness:
  - RDB will get a fairness advantage due to absence of timeouts.
  - For high RTTs and (relatively) low IATs, the MSS will fill up.
    - Regulate bundling by introducing a byte limit:
      - Sysctl: net.ipv4.tcp_rdb_max_bundle_bytes
- Implementation issues:
  - Retransmission modifications do only small changes to existing code.
  - RDB more complex.

Linux support for interactive thin streams

- The typical thin stream is an interactive application.
- Games for Linux is the new frontier.
  - Game servers: Benefit from stand-alone patching.
  - Game clients: Two-way benefit if generally included.
- Financial applications with real-time demands.
- Peer to peer interactive applications on the rise.

..fuel to the fire..

- There may be other ways to “help” interactive thin streams.
- If PIF <=4, auto-disable Nagle's algorithm
- :(Takes the responsibility off the shoulders of ignorant developers.
- :) Will benefit interactive applications which very often show these properties
- Such changes always spawn heated discussions on netdev :)

Conclusions

- Thin streams are very often created by time-critical (or interactive) applications.
- Small changes can be made to improve latency for thin streams without affecting other streams.
- Using thin-stream modifications could mean the difference between a well-running application and a ruined experience.

Questions?

?

Thin vs Thick