Effect of IW and Initial RTO changes

Ilpo Järvinen, Aki Nyrhinen, Aaron Yi Ding, Markku Kojo

Department of Computer Science
University of Helsinki

IETF79 / Beijing
Nov 11th 2010

Introduction

- Simulation study to evaluate effects of recently proposed changes:
  - Initial Window change from 3 packets to 10 packets
  - Initial RTO change from 3 seconds to 1 second
- Focus on (typical) slow/moderate bit-rate wireless links like environments
- Initially presented IW10 results in the last ICCRG meeting @ Maastricht

Test setup

- Links (bw/one-way propagation delay)
  - EGDE 160kbps/250ms, BDP = 7 pkts (6.7)
  - HSPA 2Mbps/70ms, BDP = 24 pkts (23.3)
  - LTE 50Mbps/15ms, BDP = 125 pkts
- No wireless errors, nor allocation / error related delays considered
- 11ms propagation delay from sender to wireless link
- Buffer (FIFO) sizes
  - BDP (Bandwidth Delay Product)
  - $2 \cdot$ BDP
  - 50 Packets (EDGE only)
- Workload: A burst of 1, 2, 6 or 18 simultaneous downstream TCP flows (total 180kB) competing against a similar later starting burst (another 180kB), 100 replications
- ns2 TCP SACK in use

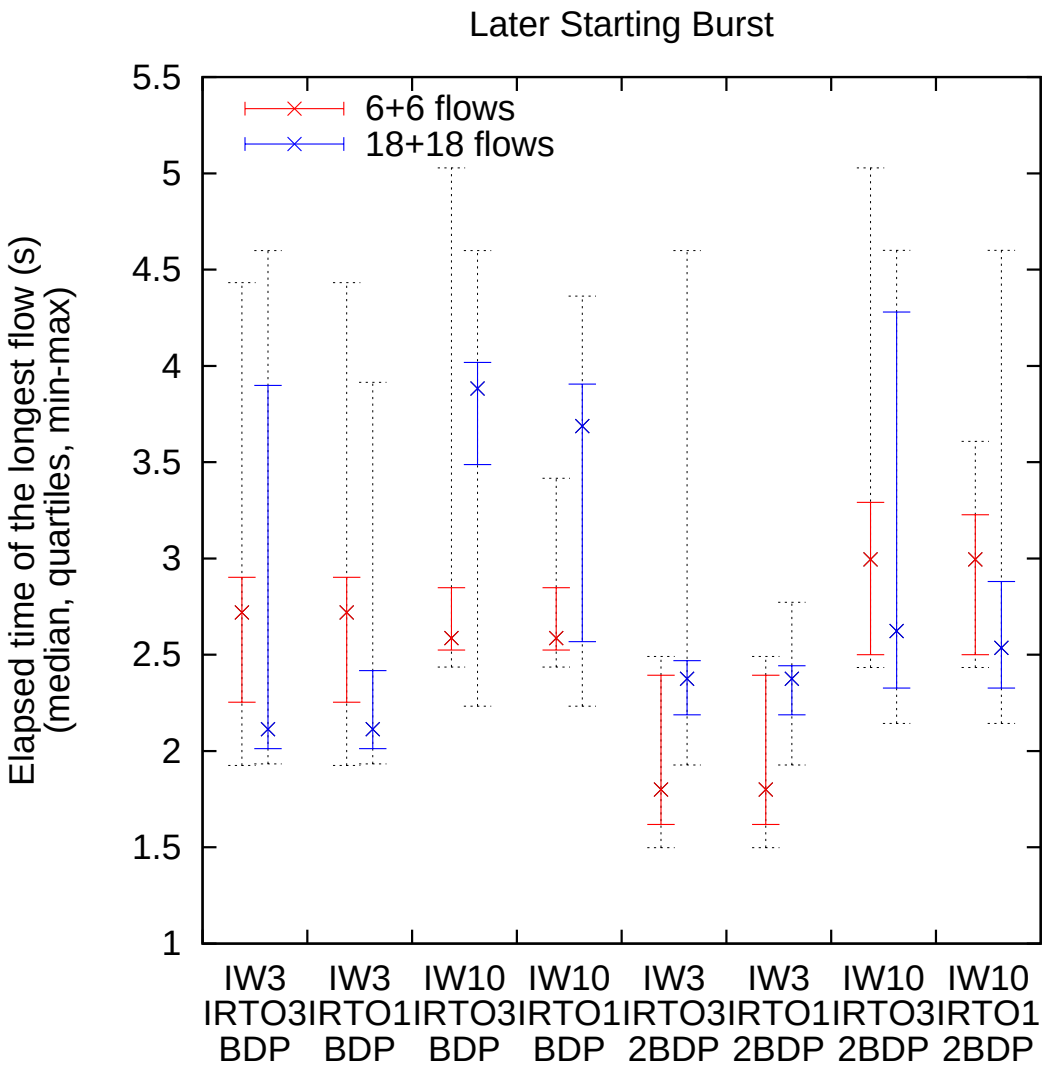
Summary of IW10 Effects

- With small number of TCP flows, IW10 improves performance
- With larger number of flows, IW10 tends to decrease performance - Regardless of IW, too many flows clearly results in suboptimal performance
- Fairness for later starting traffic improves with IW10
- Fairness within both bursts worse with IW10

IRTO: LTE (50Mbps/15ms, BDP=125 Packets)

- No changes
- No spurious RTOs
- RTOs with IW10 when # of flows is 6+6 or 18+18
  - But not in the beginning for the flow that completes last (not for the SYN nor the first packet)
  - $\Rightarrow$ IRTO has no effect

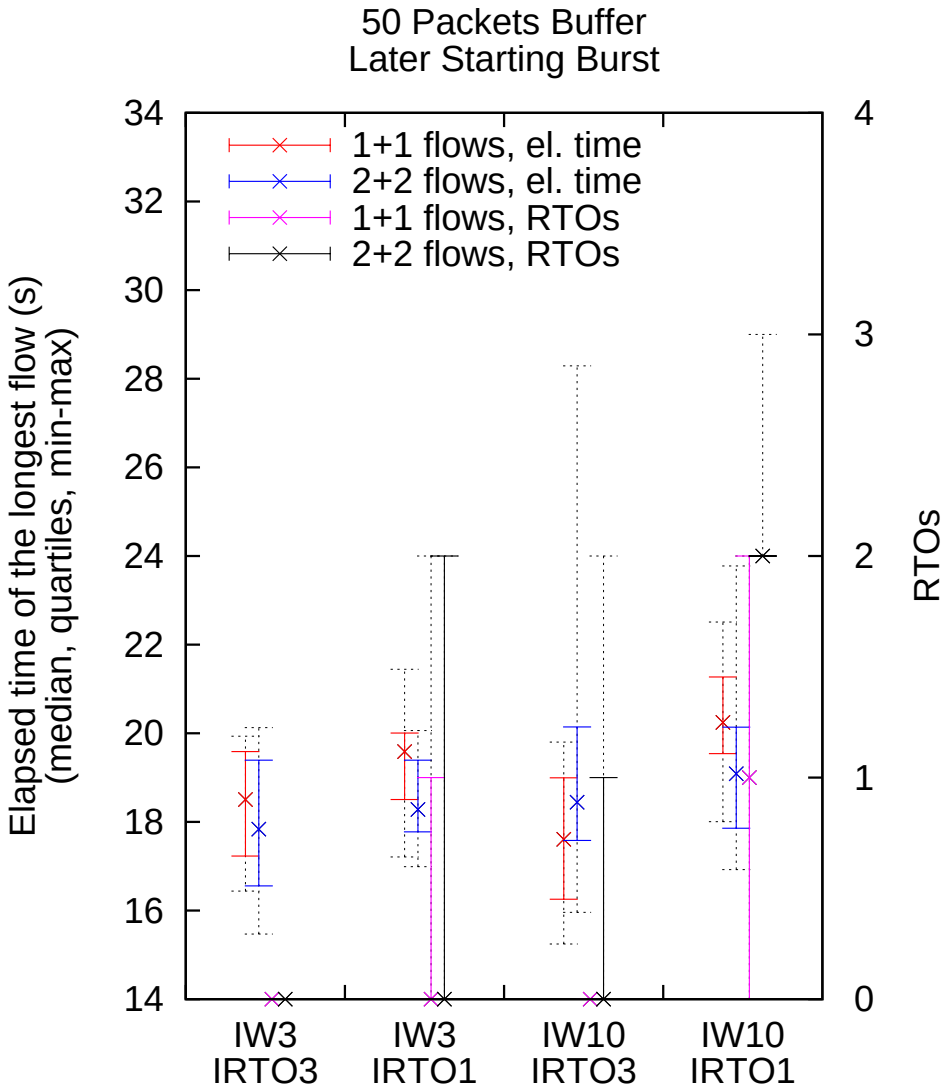
IRTO: HSPA (2Mbps/70ms, BDP≈23 Packets)



Observations

- When overloaded, small improvement for the longest cases among later starting traffic
- Opposite effect for the first starting burst (the shortest cases delayed)
- No changes due to IRTO1 with 1+1 or 2+2 flows

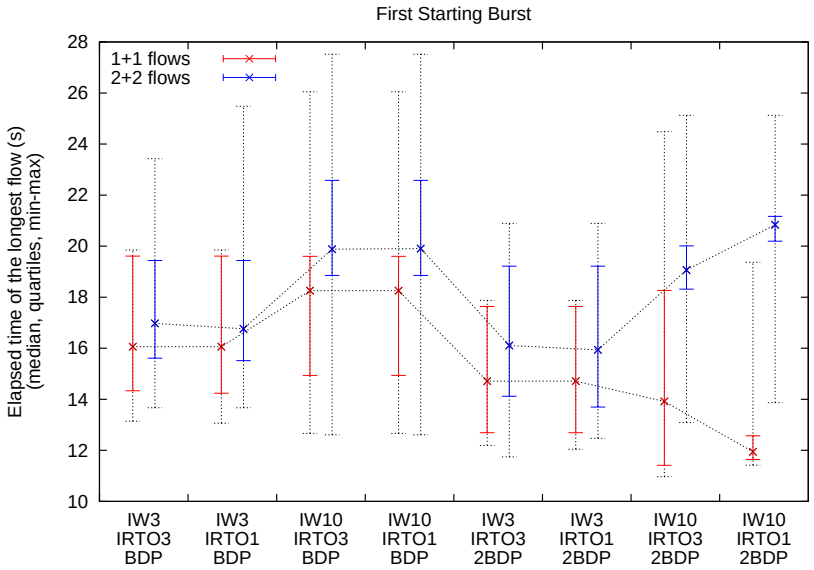
IRTO: EDGE (160kbps/250ms, BDP≈7 Packets)

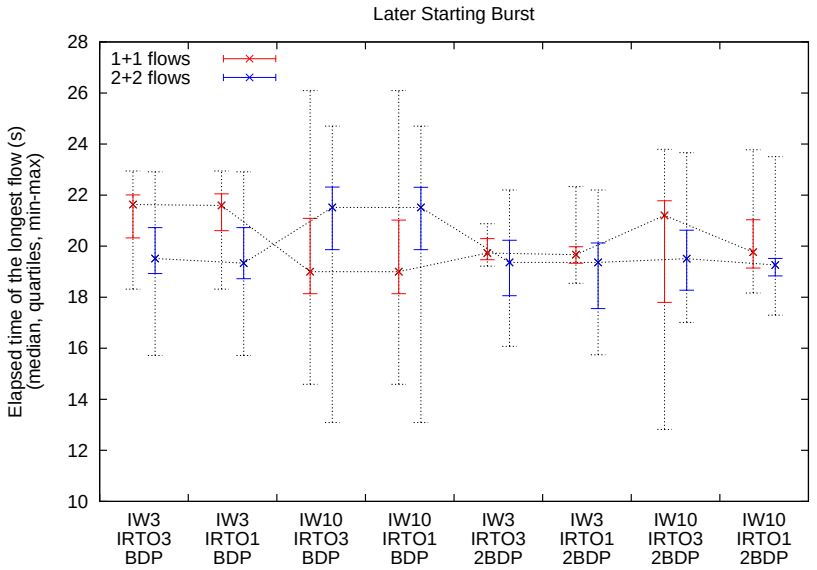


Observations

- With large buffer, number of RTOs increase
  - Mostly spurious RTOs
- ⇒ Completion of the longest flow is delayed
- The same trend with larger number of flows
- When IW10 in use, the first starting burst is able to take advantage and completes unfairly early

IRTO: EDGE (160kbps/250ms, BDP≈7 Packets)





Observations

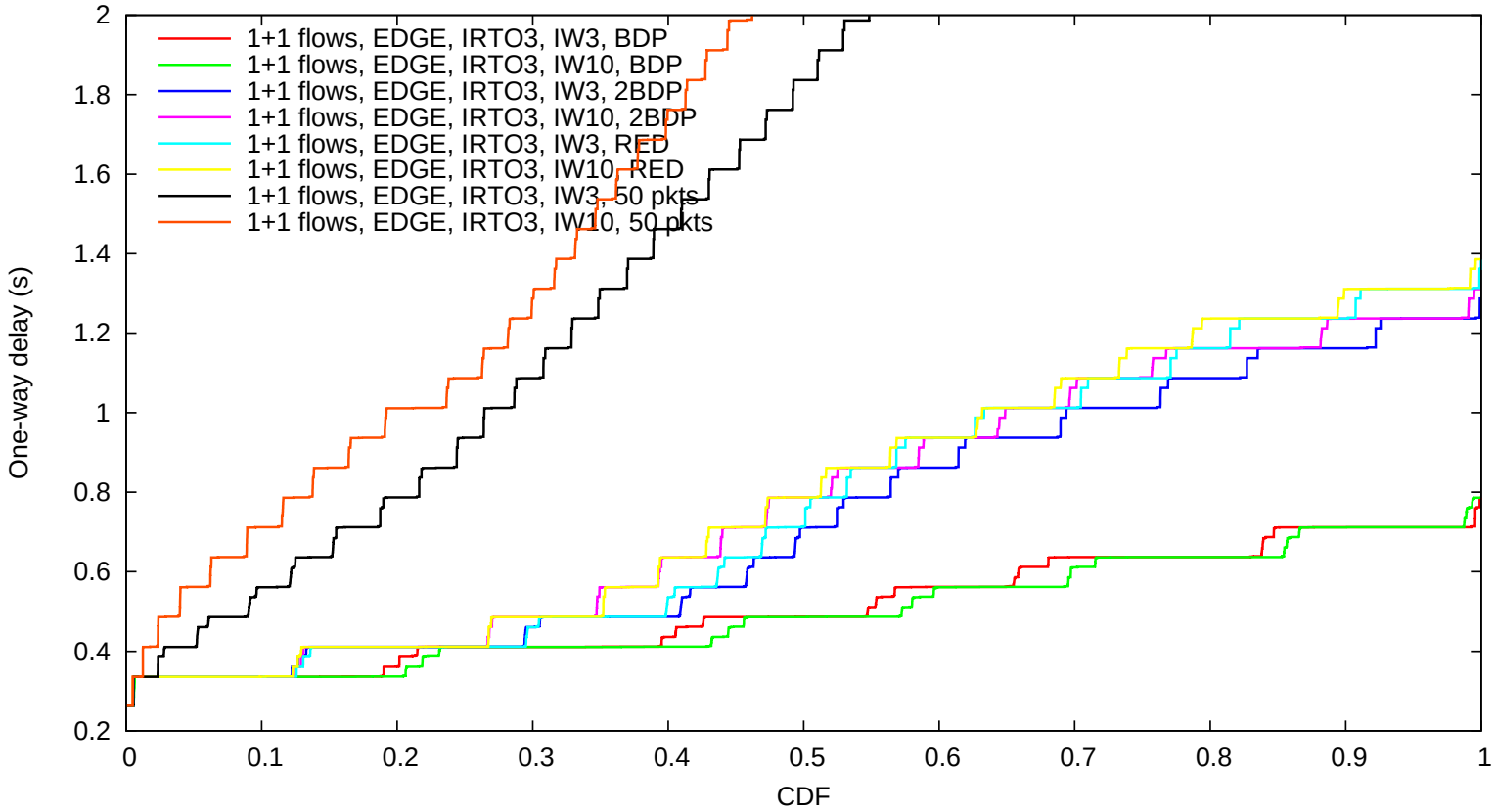
- Mostly the same regardless of IRTO
- IW10+IRTO1 becomes more fair
  - RTO occurred sooner for the later starting burst (a spurious one)

RED Configuration

Cfg	RED		REDok		
Link	EDGE	HSPA/LTE	EDGE	HSPA	LTE
w_q	0.002	0.002	0.2	0.02	0.001
max_p	0.1	0.1	0.65	0.65	0.1
th_{min}	3	5	3	3	5
th_{max}	9	20	40	50	125
buffer size	$2 \cdot BDP$	$2 \cdot BDP$	50	100	400

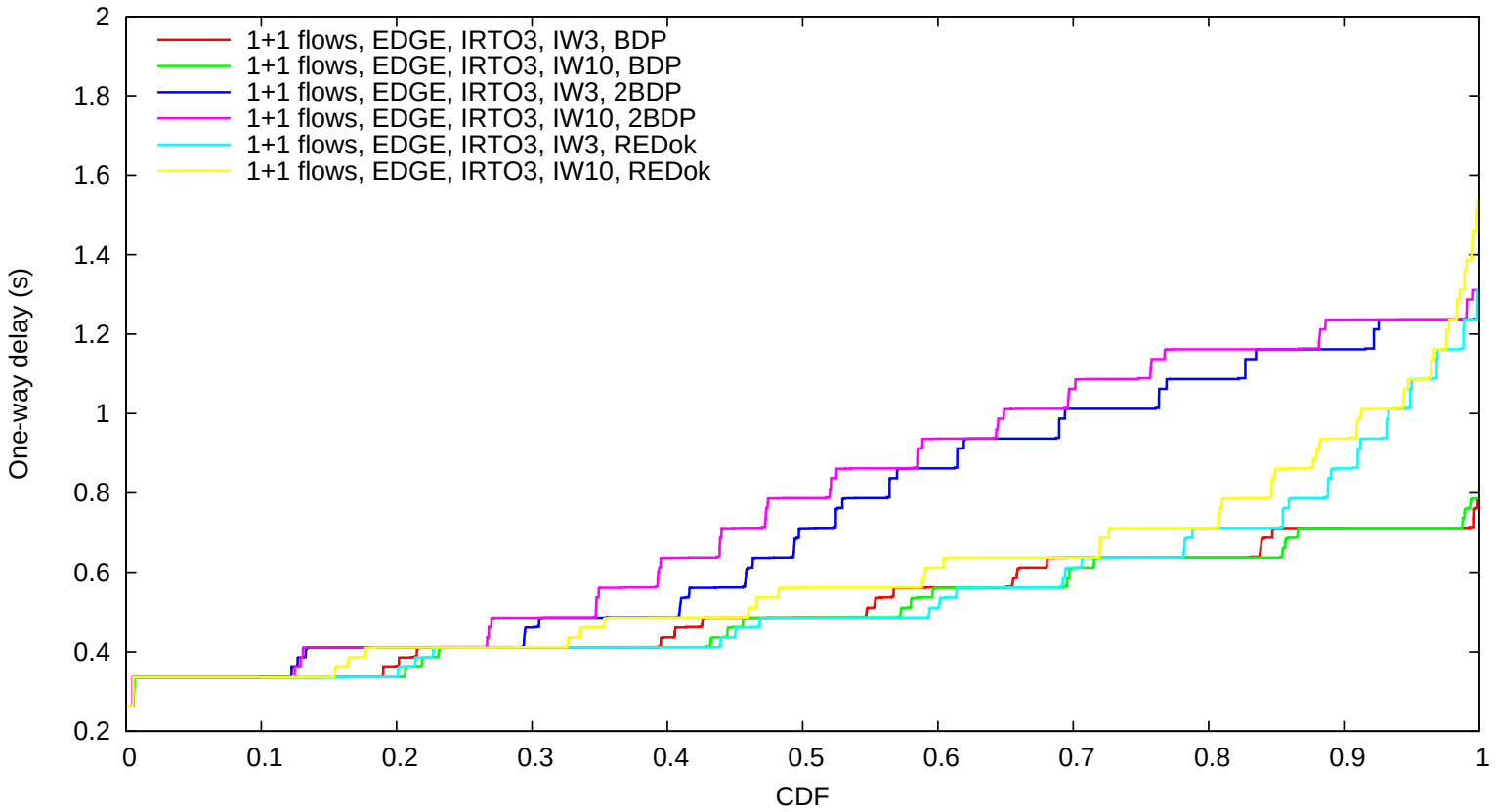
- Large buffers with *RED* configuration were not tested
  - Not useful because of $avg > th_{max}$ dropper
- *REDok* config aimed to highly varying load
  - Thus vastly different from "default configuration"
  - Aggressive enough to respond to slow start
  - Parameters are link characteristics dependent

Single Flow One-way Delay (FIFO, RED and IW3, IW10)



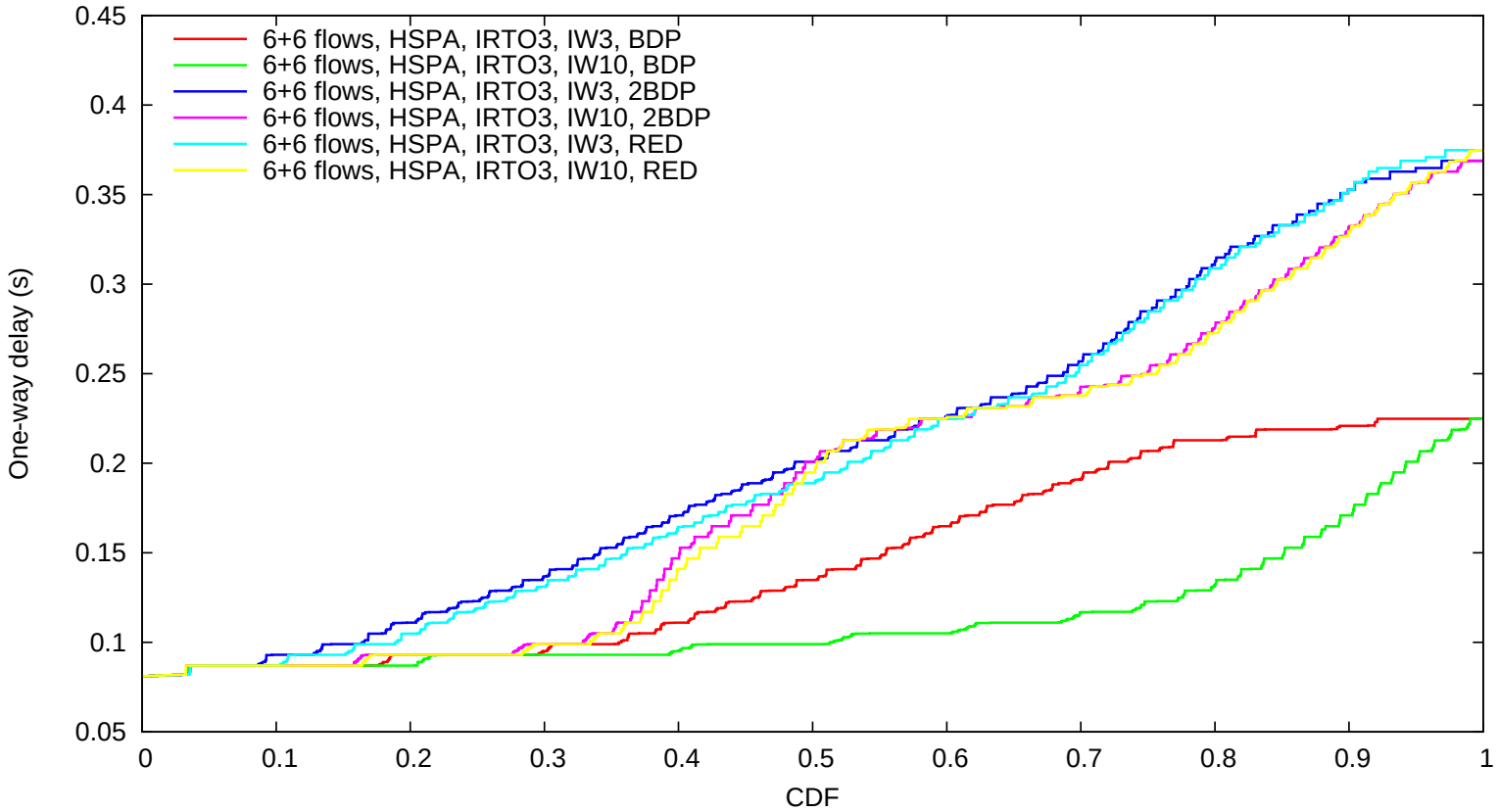
- IW10 slightly more aggressive
- RED similar to FIFO behavior (too slow to react)
- With BDP IW10 hurts itself due to self-congestion
  - Slightly smaller delays except for the highest end

Single Flow One-way Delay (FIFO, REDok and IW3, IW10)



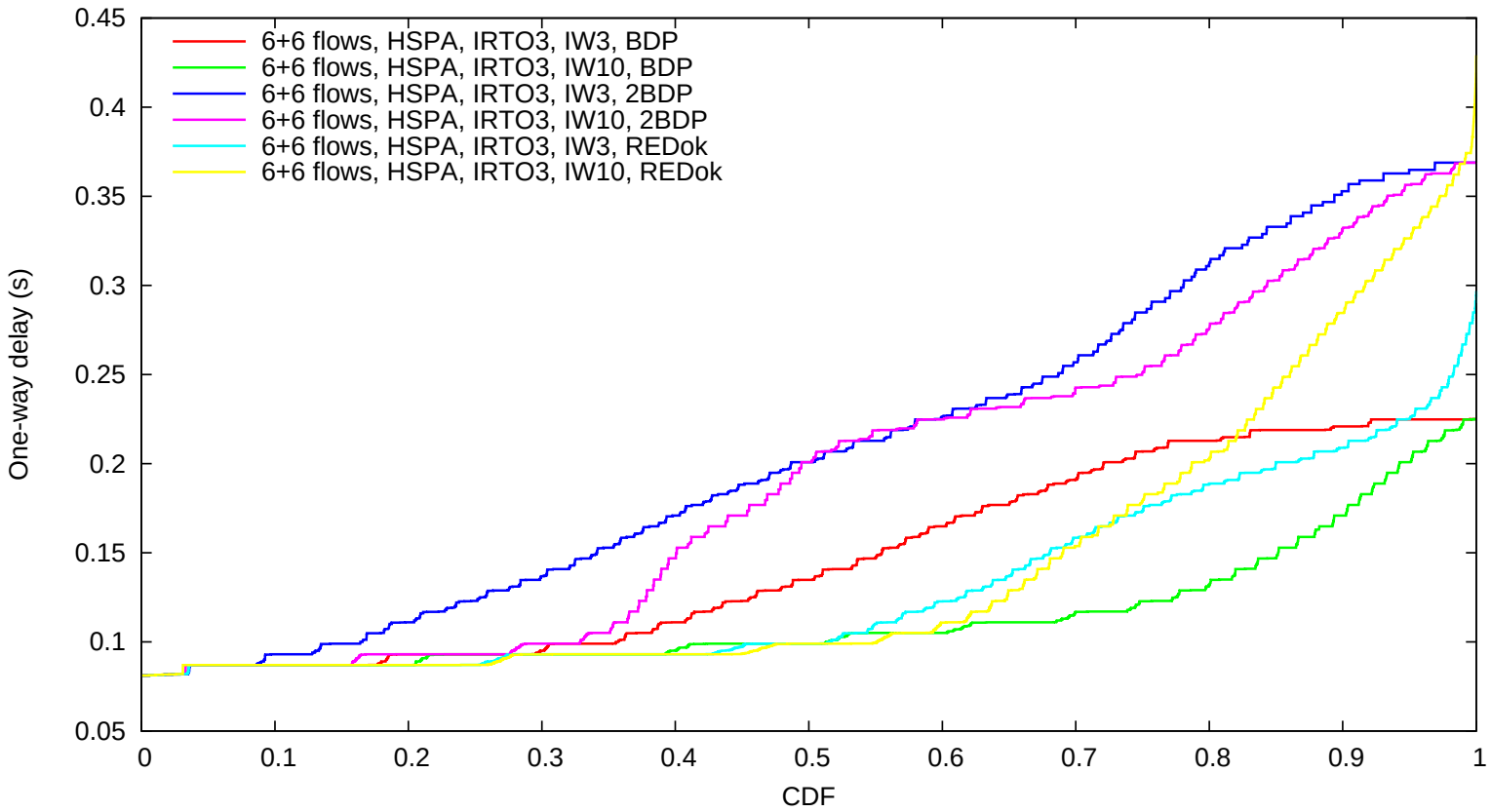
- Also *REDok* fails to control the delay increase IW10 imposes
- Maximum values with *REDok*:
  - IW10: 2.80s
  - IW3: 2.06s

6 Flows One-way Delay (FIFO, RED and IW3, IW10)



- Again, RED reacts too slowly
- IW10 less aggressive due to self-congestion ⇒ more bursty

6 Flows One-way Delay (FIFO, REDok and IW3, IW10)



- With REDok, traffic regulation works without heavy tail-drop
  $\Rightarrow$ IW10 shows to be significantly more aggressive
- Maximum values with *REDok*:
  - IW10: 0.429s
  - IW3: 0.296s

One-Way Delay in Rest of Cases

- Similar behavior observed:
  - Self-congestion $\Rightarrow$ IW10 is less aggressive
    - Except for the very highest end (in some of the cases)
  - With low enough load, IW10 is slightly more aggressive
- IRTO1 only slightly "shifts" curves
  - Only happening when IRTO1 has some effect in the first place
  - Quite insignificant in numbers
- Actual shape of the delay curves vary per queue size and type, however, those differences are out of scope here

Conclusions

- Smaller initial RTO performs better when effective e2e RTT smaller than 1 second
- More controversial when e2e RTT is larger
- IW10, while improving elapsed times, imposes higher queuing delay than IW3
  - However, if self-congesting, IW3 is more aggressive in terms of queuing delay
  - AQM (RED) failed to control the increase in the queuing delay

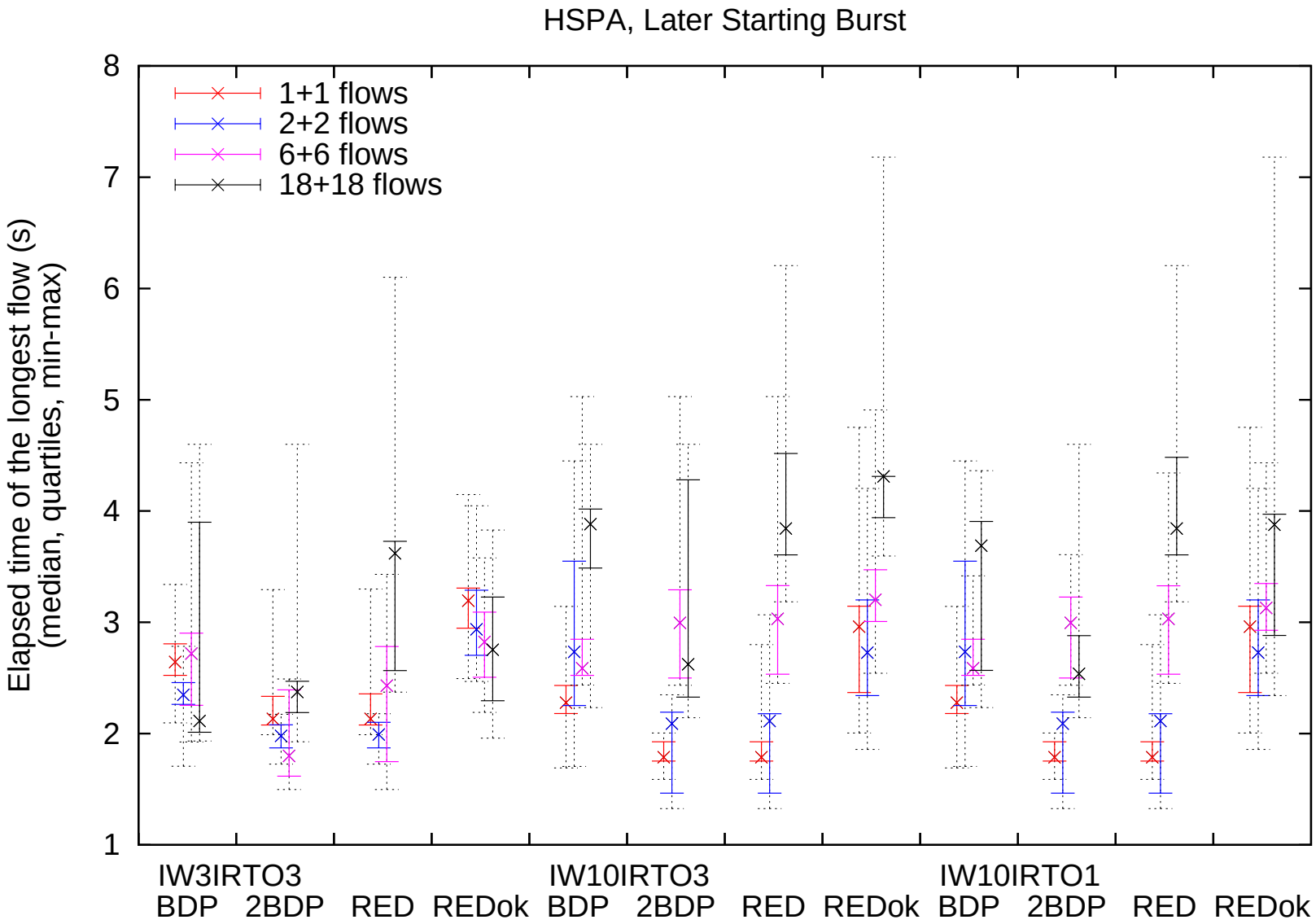
Questions?

Backup slides

RED config (detailed ns2)

```
Queue/RED set bytes_ true
Queue/RED set queue_in_bytes_ true
Queue/RED set gentle_ false
Queue/RED set setbit_ false
Queue/RED set use_mark_p_ false
Queue/RED set mean_pktsize_ 1500
Queue/RED set idle_pktsize_ 1500
Queue/RED set q_weight_ $wq
Queue/RED set thresh_ $minth
Queue/RED set maxthresh_ $maxth
Queue/RED set linterm_ [expr 1.0/$maxp]
Queue/RED set wait_ false
```

6 Flows Elapsed Times (FIFO, RED, REDok and IW)



6 Flows Fairness (FIFO, RED, REDok and IW)