UiO : University of Oslo

Update on the TCP Evaluation Suite

David Hayes <davihay@ifi.uio.no>
David Ros <David.Ros@telecom-bretagne.eu>
Lachlan Andrew <landrew@swin.edu.au>
Sally Floyd <floyd@acm.org>

ICCRG — IETF 87, Berlin

Outline

- Background to the test suite
- Overview of the test suite
- Changes from previous tmrg draft
- Future extensions

Background to the Test Suite

Why?

- Broad tests for new mechanisms
- Help facilitate comparisons
- Avoid CC wars

Background to the Test Suite

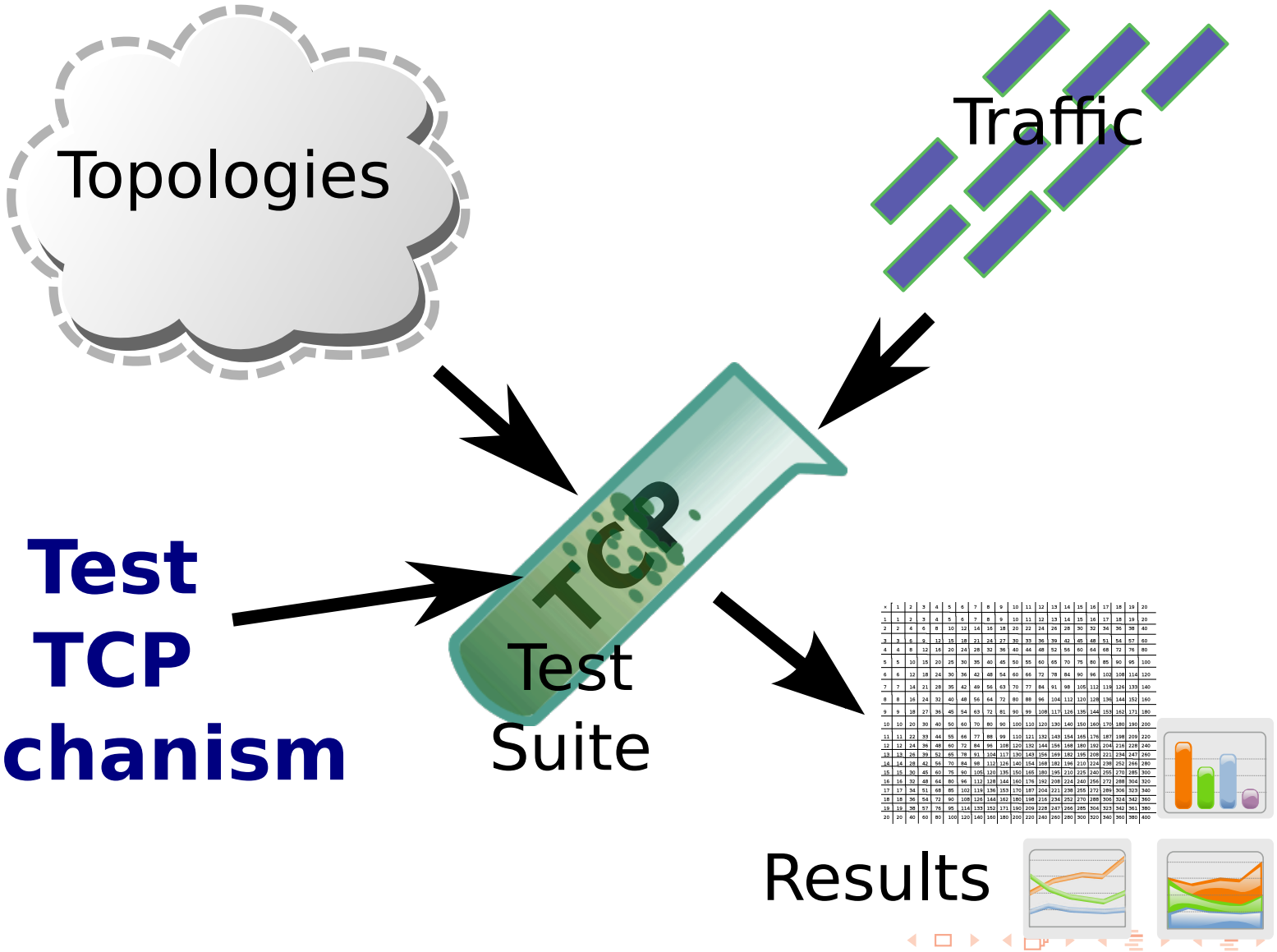
Why?

- Broad tests for new mechanisms
- Help facilitate comparisons
- Avoid CC wars

Objectives

- Base realistic test for new TCP mechanisms
- Standardised
- Publicly available implementations
- Small set of summary results

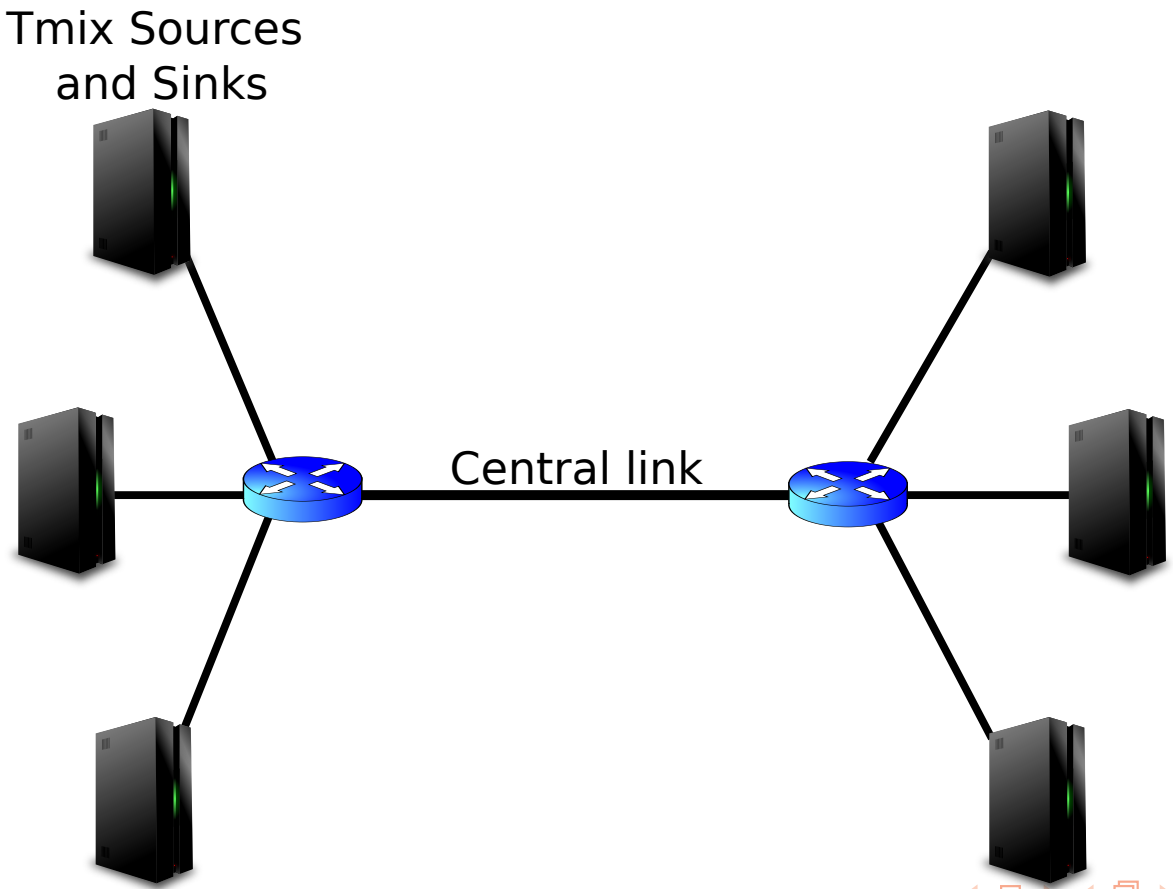
Test Suite Design



Tests

Basic

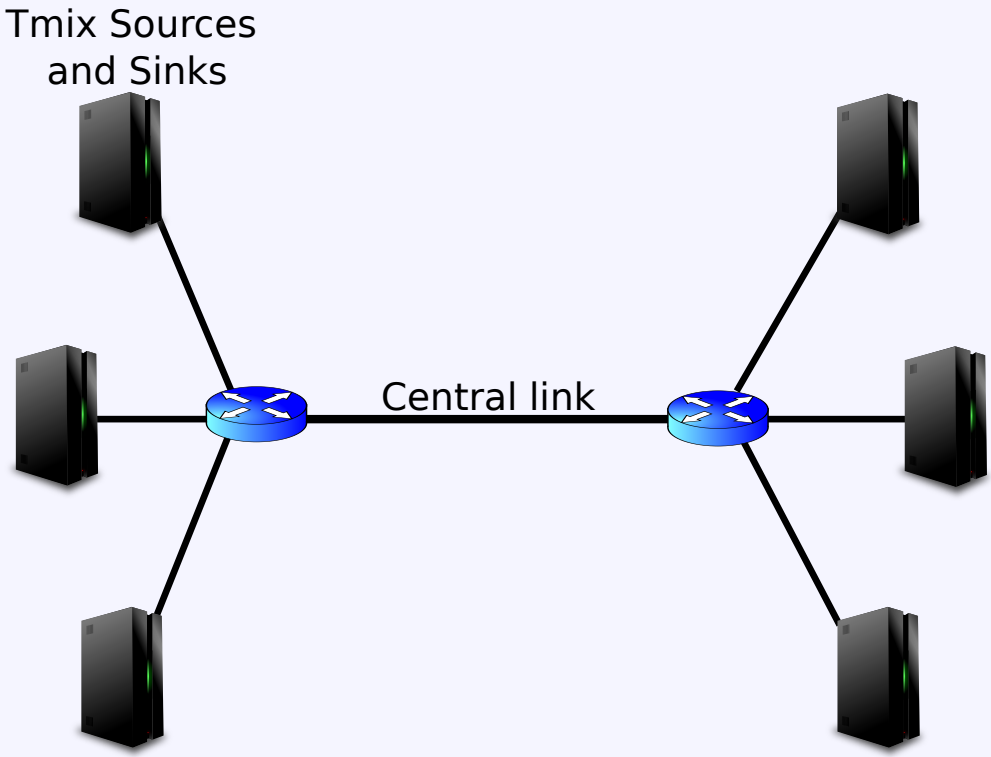
- access link, data center, trans-oceanic, satellite, wifi and dial up



Tests – cont.

Latency specific

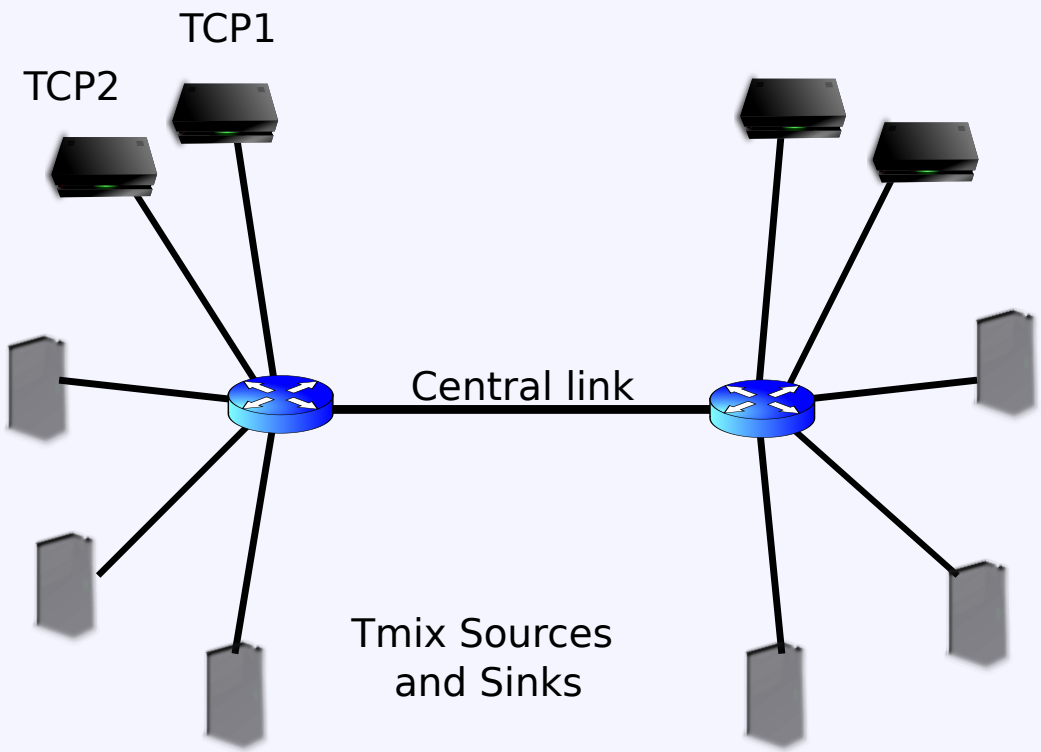
- **Delay/throughput tradeoff as function of queue size**
- Ramp up time: completion time of one flow
- Transients: release of bandwidth, arrival of many flows



Tests – cont.

Latency specific

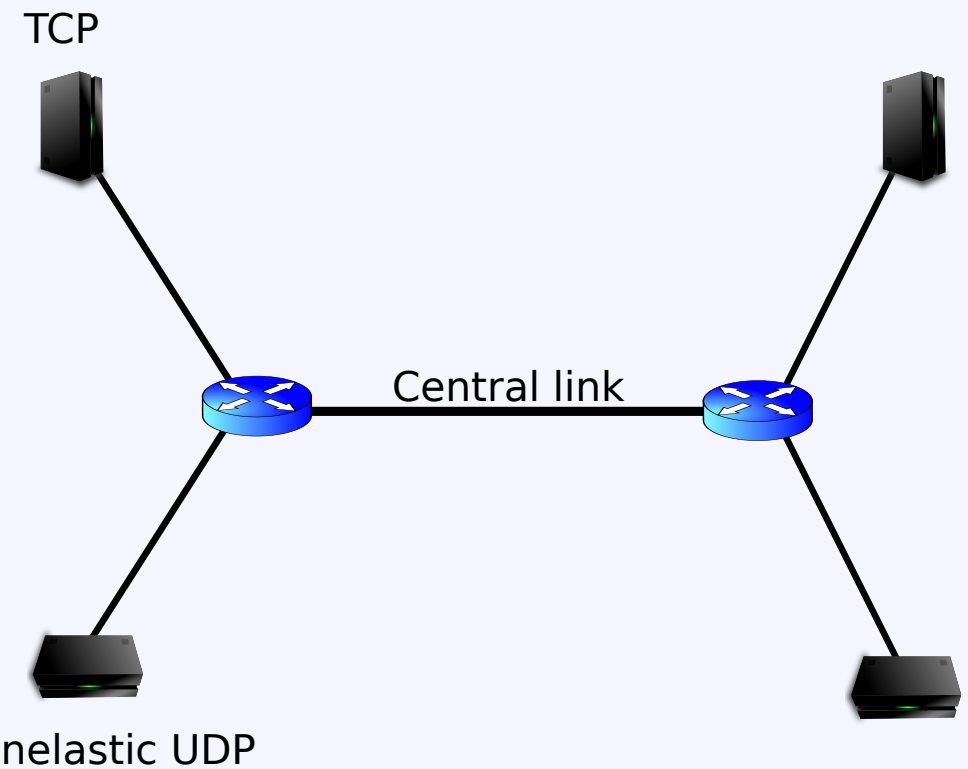
- Delay/throughput tradeoff as function of queue size
- **Ramp up time: completion time of one flow**
- Transients: release of bandwidth, arrival of many flows



Tests – cont.

Latency specific

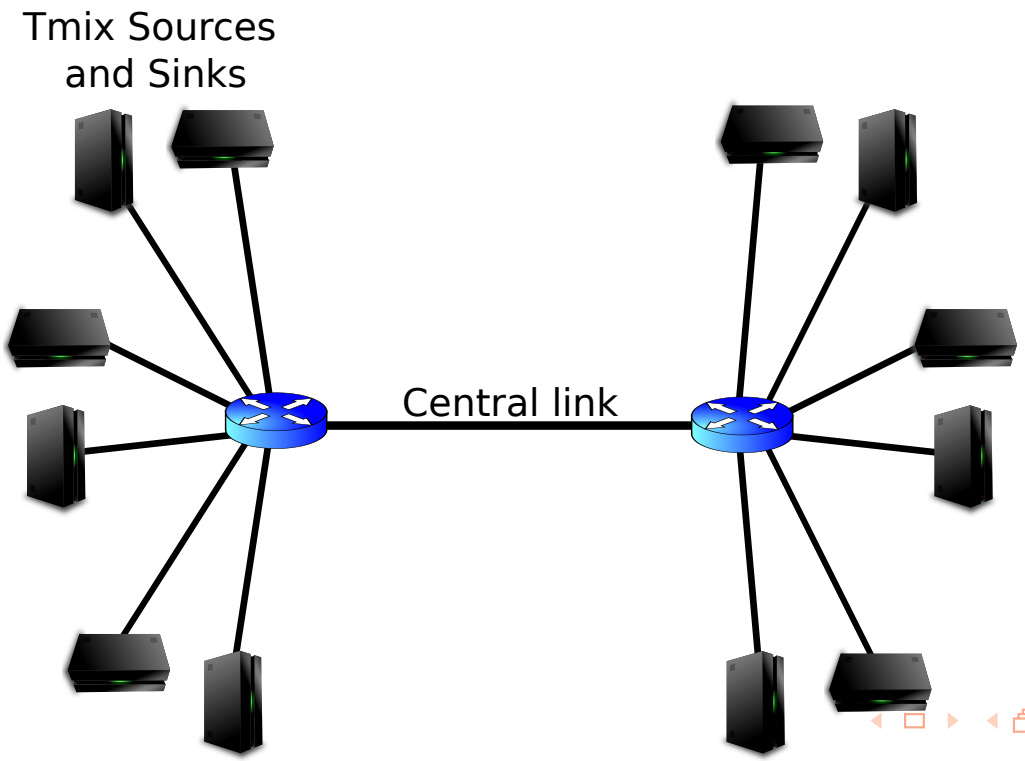
- Delay/throughput tradeoff as function of queue size
- Ramp up time: completion time of one flow
- **Transients: release of bandwidth, arrival of many flows**



Tests – cont.

Throughput- and fairness-related

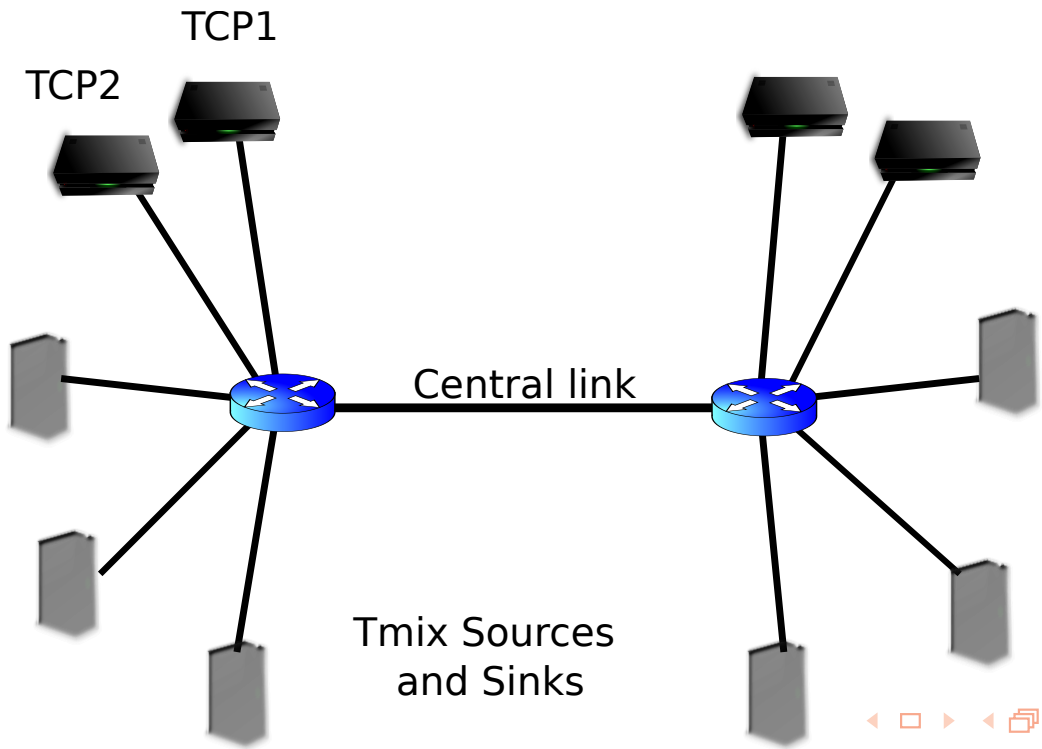
- **Impact on standard TCP traffic**
- Intra-protocol and inter-RTT fairness
- Multiple bottlenecks



Tests – cont.

Throughput- and fairness-related

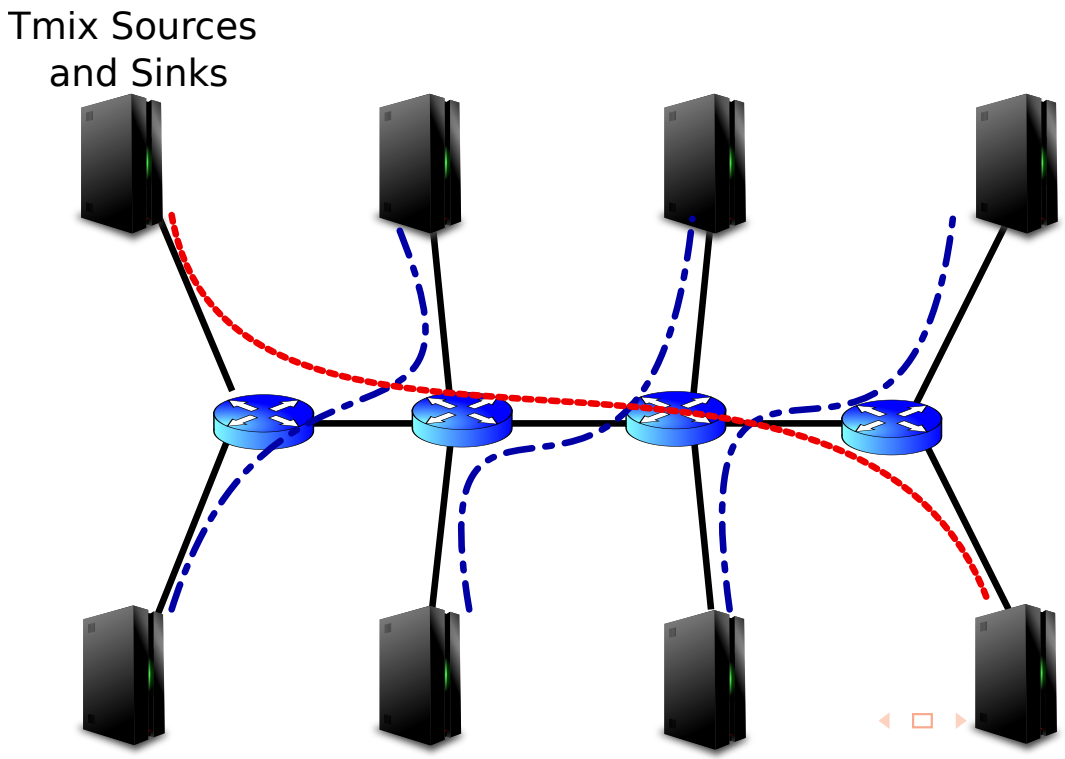
- Impact on standard TCP traffic
- **Intra-protocol and inter-RTT fairness**
- Multiple bottlenecks



Tests – cont.

Throughput- and fairness-related

- Impact on standard TCP traffic
- Intra-protocol and inter-RTT fairness
- **Multiple bottlenecks**



Traffic

Tmix

- Constructed from UNC trace captures
- File of connection vectors

Traffic

Tmix

- Constructed from UNC trace captures
- File of connection vectors

Connection Vector (CV)

- models application layer interaction on a TCP flow

Traffic

Tmix

- Constructed from UNC trace captures
- File of connection vectors

Connection Vector (CV)

- models application layer interaction on a TCP flow

Resulting traffic

- Realistic traffic
- Thousands of concurrent TCP connections

Changes since the last draft

`http://riteproject.eu/ietf-drafts`

- tmrg $\rightarrow$ iccrg
- Use of Tmix and traffic generation specifications
- Change in load specification – percent loss $\rightarrow$ offered load (as percent of bottleneck capacity)
- Refinement of scenarios based on implementation exercise
- Removal of most of the more academic discussion of alternatives.
- Progressing implementation in NS2 – help and contributors welcome!

Future extensions

Latency metrics

- Many of the metrics are throughput- and efficiency-based
- Latency is increasing in importance
  - Per-packet delay distribution
  - (Application-level) Flow completion times

Making traffic suitable for tests

Dealing with non-stationarity

- The trace traffic is not stationary
- While realistic, it makes determining test parameters difficult
- Solution adopted is to shuffle the CV start times
  - bin sizes relative to bottleneck capacity
  - 5 s for 100 Mbps
  - Fisher-Yates shuffle

Making traffic suitable for tests – cont.

Scaling

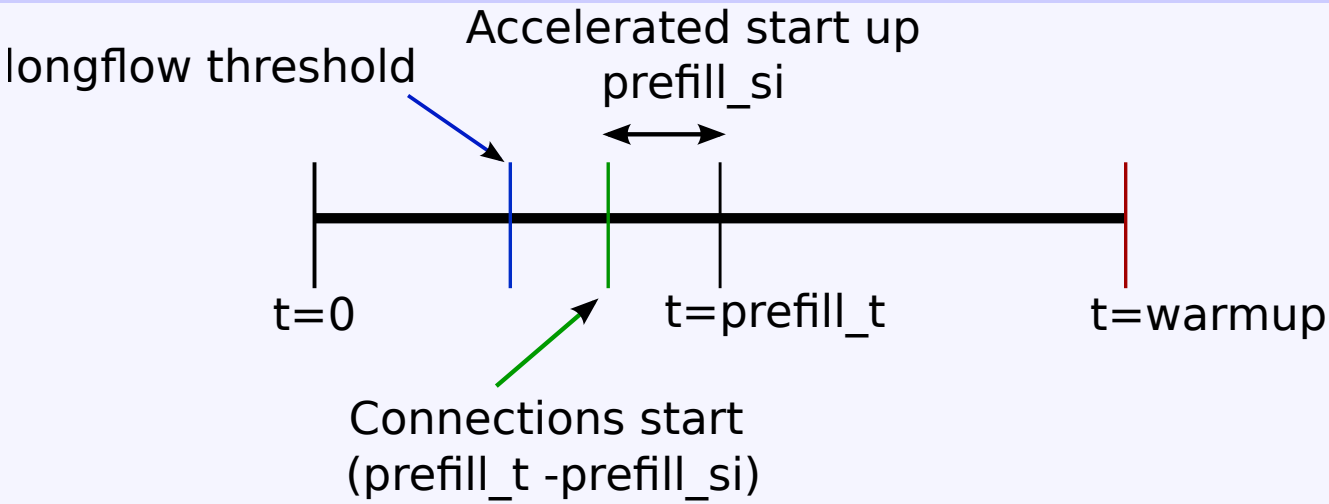
experiment_cv_start_time = scale * cv_start_time

Making traffic suitable for tests – cont.

Scaling

experiment_cv_start_time = scale * cv_start_time

Pre-filling to accelerate start up



Making traffic suitable for tests – cont.

"Steady state"

The following approach is taken

- Simulation time is long enough so the offered load in the last and second last third is equal (within 5%)
- Warm up time is calculated based on NS2 simulation of NewReno
  - time until the load (and queue stats) after warm up are equal in the first and second halves are equal (within 5%)