The War Between Mice and Elephants

Liang Guo, Ibrahim Matta

Presented by Vasilios Mitrokostas for CS 577 / EE 537

Images taken from Pankaj Didwania's 2013 presentation of this paper

An Issue of Fairness

Long connections are unintentionally favored over short connections by TCP congestion control algorithm

Mouse

Mouse

- Many connections, short traffic

Elephant

Elephant

- Few connections, large traffic
- 80-20 rule

The Elephant Wins

- Blame TCP; three main factors
  - Conservative ramp up of transmission rate
  - Painful packet loss for shorter connections
  - No packet samples for mice

TCP: Conservative Ramp Up

- Sending window starts at the smallest value
- This hurts many small connections which need to begin at this point each time

TCP: Painful Packet Loss

- A short connection's congestion window doesn't have enough packets to detect packet loss by duplicate ACKs
  - . . . so it's only detected by timeout, slowing the rate of data transmission

TCP: No Packet Samples

- TCP uses samples of packets to help determine timeout
  - . . . but each of the many, short connections lacks sampling data, so timeouts are set to conservative, large value

How to Combat Unfairness

- Guo and Matta's proposal; fight fire with fire
- Simulations say: give short connections preferential treatment to induce fairness
  - A weighted policy to classify TCP flows by size
  - RIO (RED with In and Out) queue management

Validating the Problem

- How did the authors draw these conclusions?
  - A study of short and long TCP flows
  - Previous papers highlight the uphill battle faced by mice . . . but their solutions modify TCP
  - Issue: isolating flows by class (short vs. long) may cause packet reordering, leading to poor performance
- Guo and Matta: place control inside the network with RIO

Proposed Solution

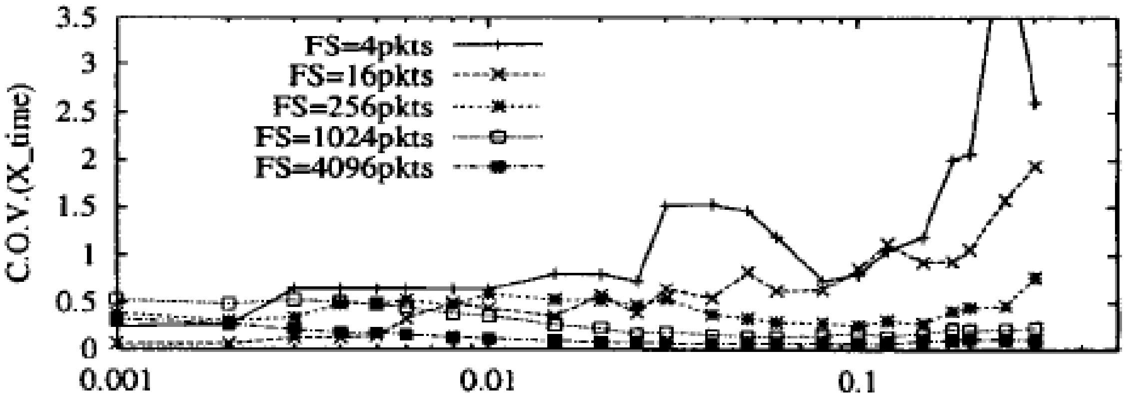
Mitigate packet loss by giving preferential treatment to short connections

RIO: Classify In or Out

- Classify packets as In or Out to determine size, allowing for preferential treatment
- Favor short connections at bottleneck link queues, so they experience fewer dropped packets

Why Is Packet Loss Critical?

- When loss rate is small, average transmission time is not greatly impacted
- When loss rate is large, time increases drastically (see TCP-Newreno test below, randomly dropped packets)



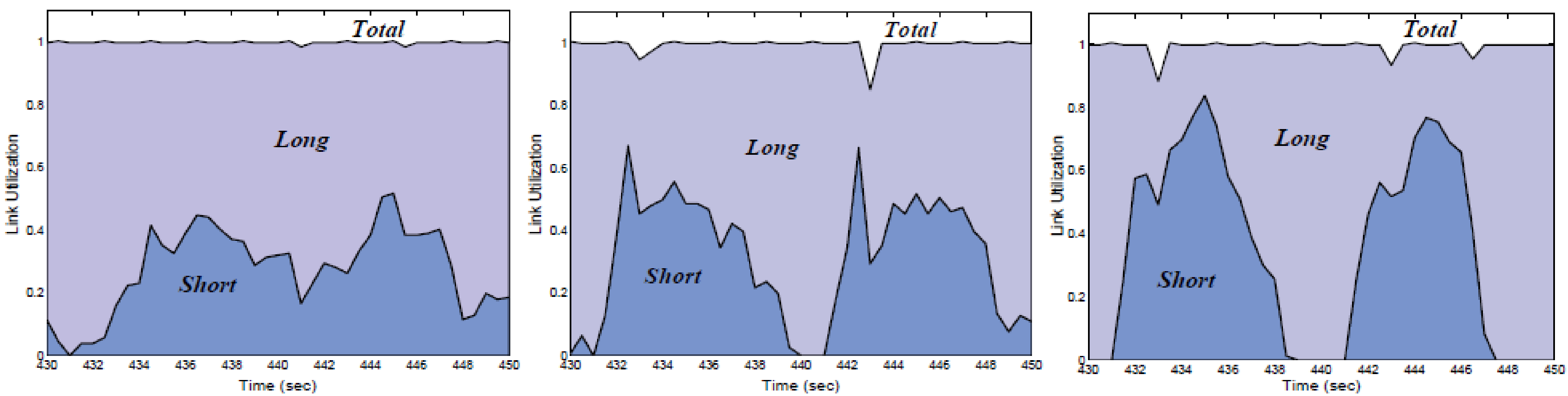
(b) Coefficient of Variation

Why Does Variability Happen?

- High loss rate = high chance for TCP to enter exponential backoff (congestion avoidance) phase, resulting in more variability
- Low loss rate = two options for TCP: transmit aggressively with slow-start or transmit in congestion avoidance phase, resulting in more variability (less consistency)
- First source of variability is on individual packets—greater impact on short flows due to number
- Second source of variability in end-phase—greater impact on long flows which finish beyond slow-start

Comparison by Simulation

- Network simulator *ns* by E. Amir et al.
- 10 long flows (100 packets) vs. 10 short flows (10,000 packets) (TCP-Newreno)
- 1.25Mbps link



Link utilization: (left) DropTail, (middle) RED, and (right) RIO-PS

Too Unfair to Elephants?

- RIO-PS (preferential treatment to short flows) graph shows short flows taking more of the total link utilization than long flows . . . unequal
- This is OK; early completion returns resources to long flows, so long-term goodput is maintained
- In fact, it results in a more stable environment for long flows because of fewer disturbances from short flows (once they finish)

Goodput Comparison

- Overall goodput for all flows remains stable
- 500 second simulation, note difference in load (RED and RIO-PS favor higher loads)

Link B/W	Flows	DropTail	RED	RIO-PS
1.25Mbps	All	153479	154269	154486
	Short	40973	49897	49945
	Long	112506	104372	104541
1.5Mbps	All	185650	184315	183154
	Short	43854	49990	49990
	Long	141796	134325	133164

TABLE I

NETWORK GOODPUT UNDER DIFFERENT SCHEMES

Implementation: Edge Routers

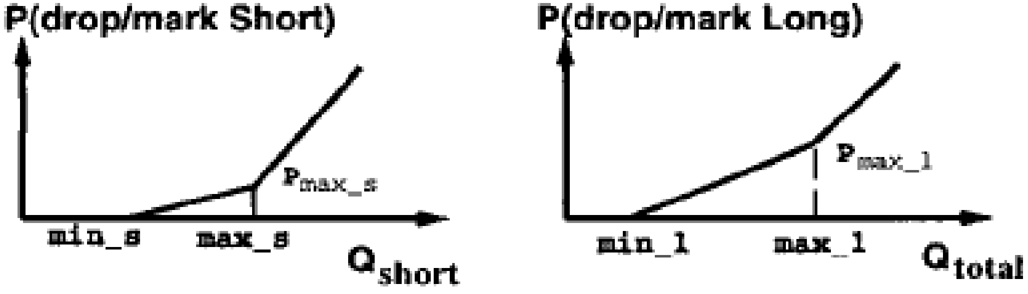
- Employ a Diffserv-like network architecture to differentiate between short and long TCP flows
- This is done through edge routers
    - Edge router tracks each flow, counting packets
    - Once a threshold L_t is met, flow is considered long (the first L_t packets of such a flow are considered short)
        - Authors claim this is OK because first few packets are vulnerable to packet losses, and this makes the system fair to all starting TCP connections
    - Every so often (T_u time units), flow is considered finished if no packets are observed in the period

Choosing Variables

- Threshold L_t can be static or dynamic; can allow edge router to modify every T_c based on short and long flow counts . . . the Short-to-Long-Ratio (SLR)
- Choosing T_u and T_c needs further research (T_u = 1 sec, T_c = 10 sec in simulation)

Implementation: Core Routers

- Core routers give preferential treatment to short packets using RIO
- See packet dropping figure below; note that In (short) packet queuing is not affected by Out (long) packet arrivals



RIO queue with Preferential treatment to Short flows

Packet Reordering: Not a Problem

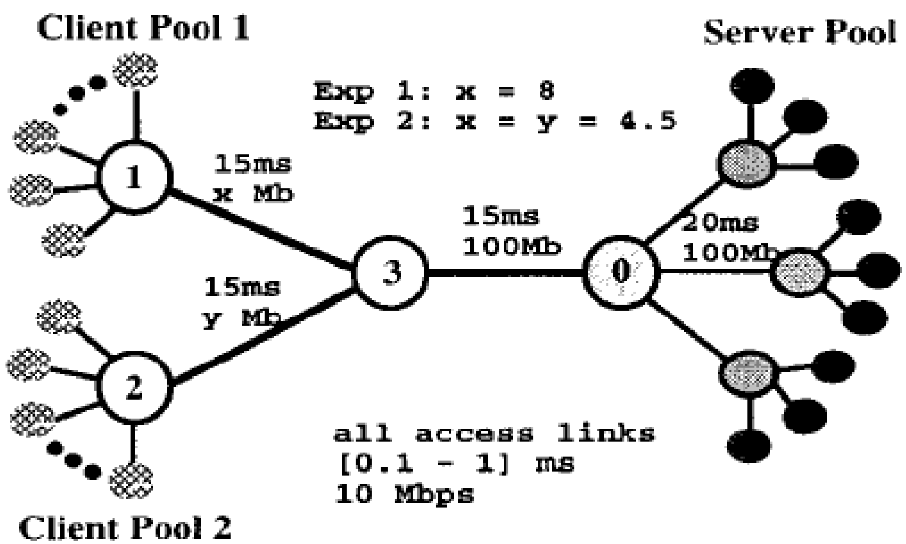
- Only one FIFO queue is used for all packets, short and long
  - No packet reordering even if same-flow packets are classified differently

Simulation

Is RIO-PS as beneficial as claimed?

Simulation of RIO-PS

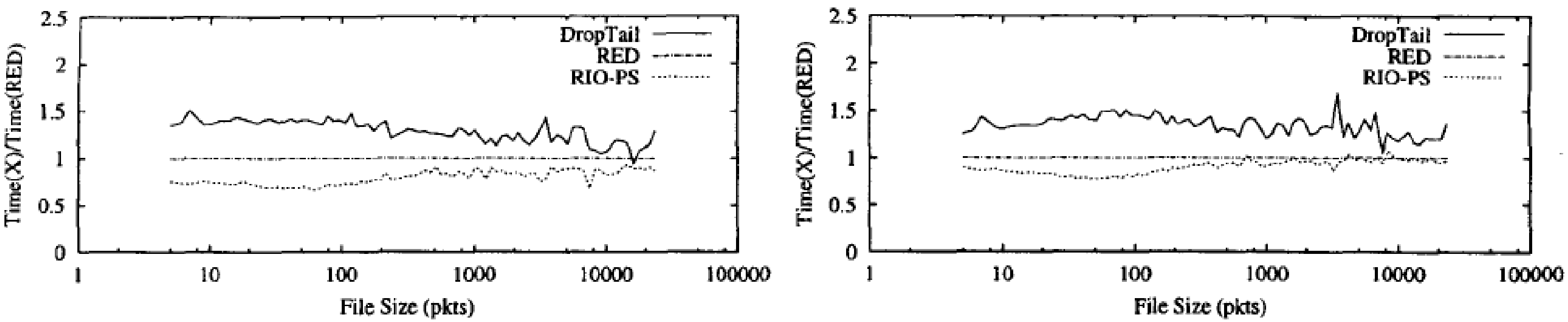
- Web traffic model; each page requires TCP connection
  - Tuned to maximize power, ratio between throughput and delay. High power implies high throughput and low delay



Simulation Topology

Single Client Experiment

- 4,000-second simulation
  - (2,000-second warm-up)
- Record response time using preferential treatment
  - What about initial timeout (ITO) from 3 seconds to 1 second? Authors warn unnecessary retransmissions may lead to congestion collapse (slow links or high round-trip delay), but plot results anyway (donkey)

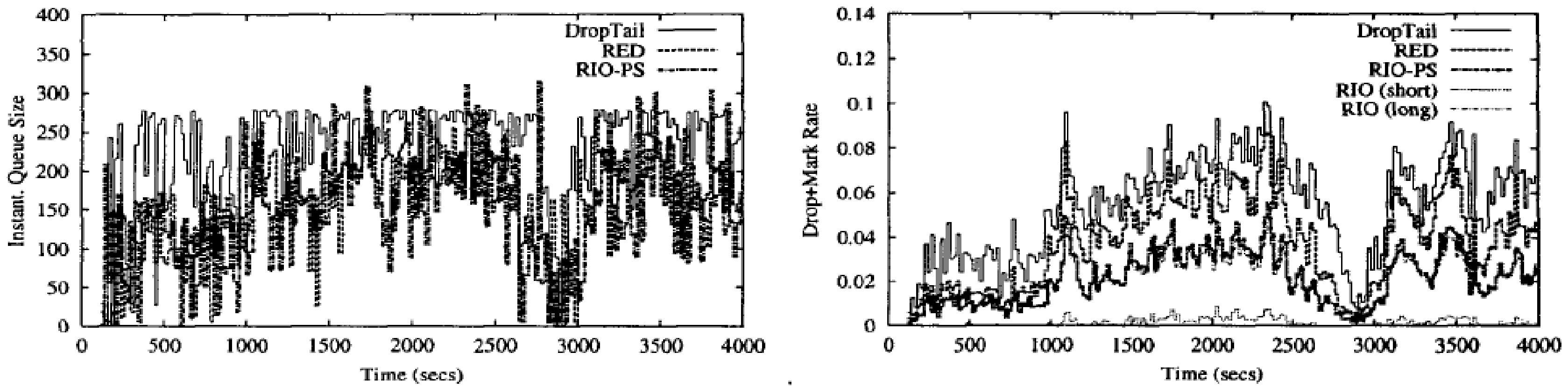


(a) Initial Retransmission Timer 3 seconds (b) Initial Retransmission Timer 1 second

Average response time relative to RED

Advantage

- Performance improvements; reduction on overall mark/drop rate without risk of queue overload at the bottleneck link
  - Why? Short flows now have fewer packet drops, which means fewer congestion notifications



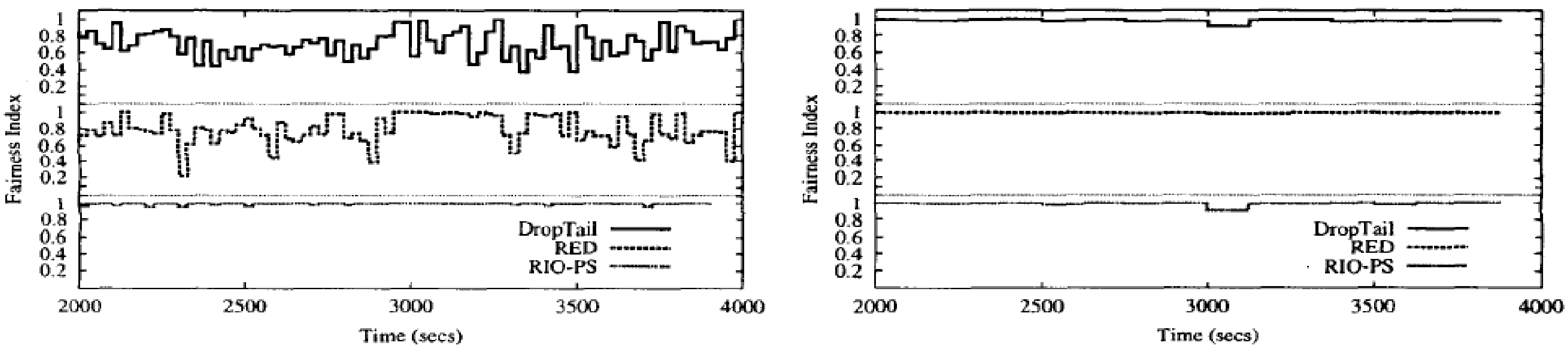
(a) Instantaneous Queue Size (b) Instantaneous Drop/Mark Rate

Revealing the secret of better performance

Fairness Index

- Computed using a fairness index formula based on response time T_i

$$FI = \frac{(\sum_{i=1}^{10} T_i)^2}{10 \sum_{i=1}^{10} T_i^2},$$

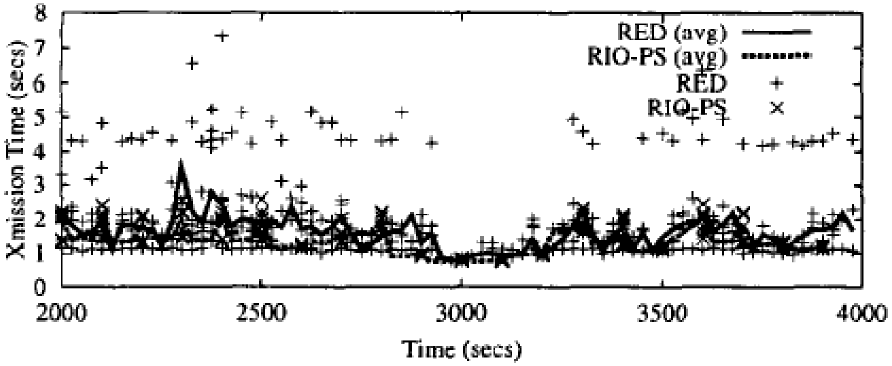


(a) Fairness Index of Short Connections (b) Fairness Index of Long Connections

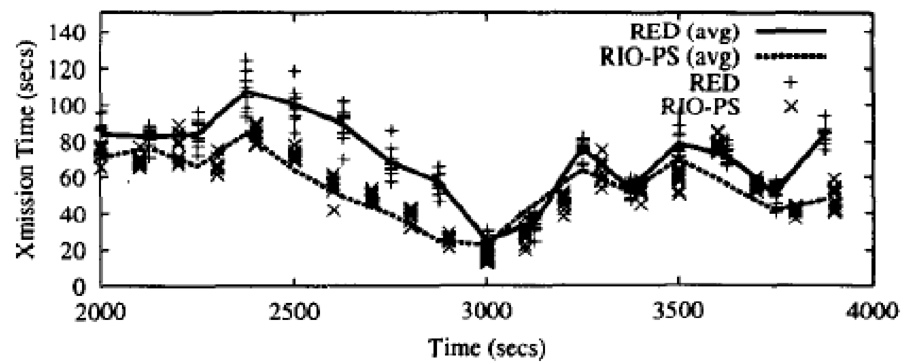
Fairness of Transmission Time

Fairness Index Continued

- Transmission times and goodput



(a) Transmission Time of Short Connections



(b) Transmission Time of Long Connections

Transmission Time of Foreground Traffic

Scheme	DropTail	RED	RIO-PS
Exp1 (ITO=3sec)	4207841	4264890	4255711
Exp1 (ITO=1sec)	4234309	4254291	4244158

NETWORK GOODPUT OVER THE LAST 2000 SECONDS

Unbalanced Requests Experiment

- Paper suggests preferential treatment still helps, but results are captured in another paper due to space limitation

Evaluation

Does the model hold in real-world cases?
Can it be feasibly deployed?

Evaluating the Simulation Model

- The web traffic model used for simulation is the "Dumbbell and Dancehall" one-way traffic model
- Guo and Matta claim that the RIO-PS scheme still grants an advantage when reverse traffic is present
  - Why? Short exchanges due to control packet handling on the client side are protected by this scheme (due to the preferential treatment)
- Authors also say simulation results mean RIO-PS works in extremely unbalanced cases, so odd traffic topologies would not be a problem (is this true?)

Evaluating Deployment

- A paper on edge devices is referenced to show that per-flow state maintenance (In vs. Out) and per-packet processing does not significantly impact end-to-end performance (sounds nebulous)
- RIO-PS only needs to be implemented at busy bottleneck links

Conclusions

- RIO-PS benefits short connections, which represent the majority of TCP flows
    - Long flows are thus minimally impacted
- Goodput is either the same or improved, depending on the network
- Flexible architecture; only edge routers need to be tuned