Achievable Comprehensive Delay Reporting from Routers

Darryl Veitch† *

Collaboration with

Nicolas Hohn*, Konstantina Papagiannaki↑, Christophe Diot↑

† Sprint ATL, Burlingame CA.
* CUBIN, Dept. of Electrical & Electronic Engineering, University of Melbourne.
↑ Intel, Cambridge, England.

Web Page: *http://www.cubinlab.ee.mu.oz.au/∼darryl*

Sprint

Motivation

Packet Delay is an Important Metric :

- For real-time performance
- For SLA's
- Building block of end-to-end delay is through-router delay

Delay Measurement :

- Active: not suited for 100's of router interfaces
- Passive: even worse – expensive and inconvenient
- Router statistics: currently nothing measured, nothing reported

AIM

Measure and Report meaningful delay statistics via SNMP

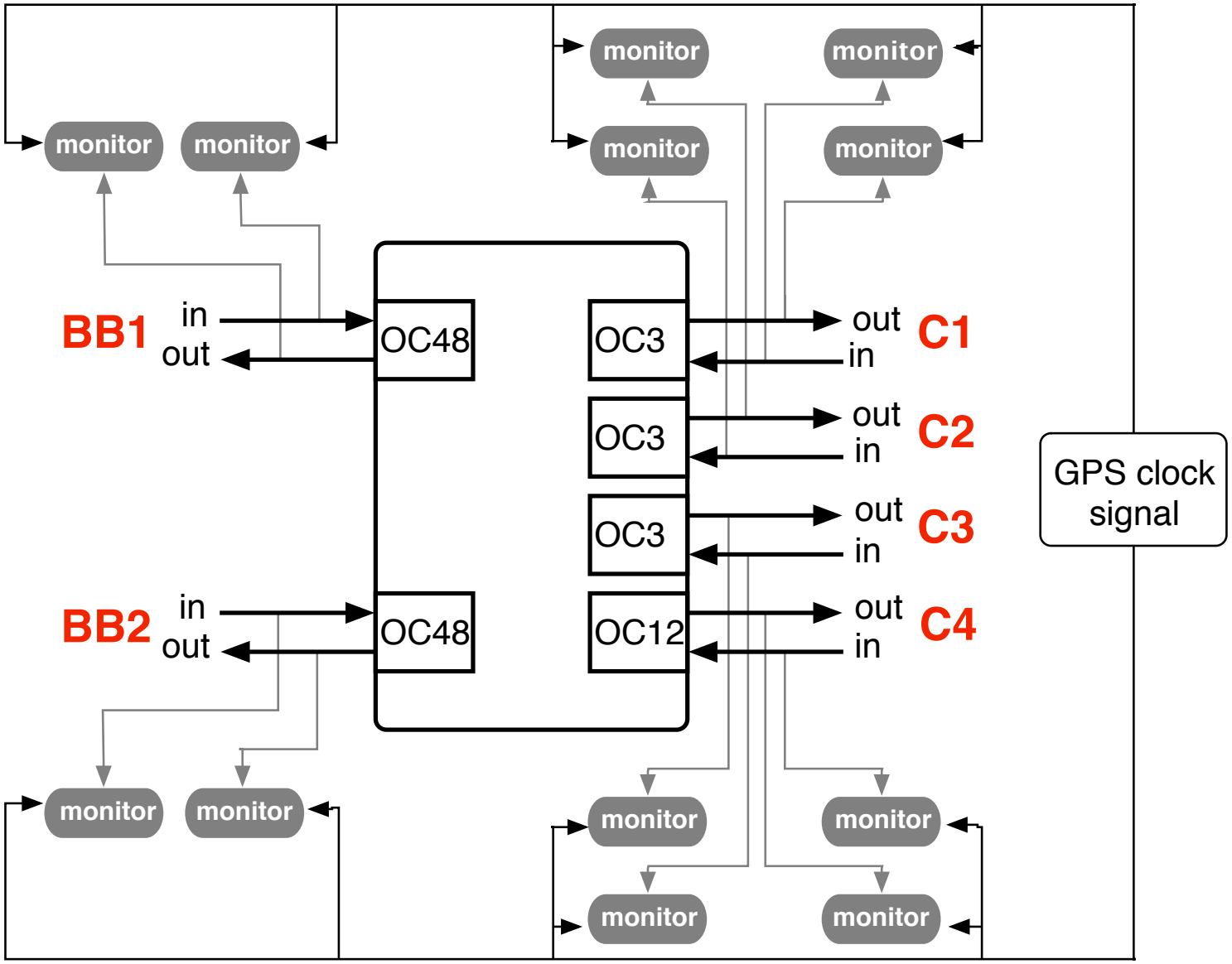
Problems To Solve

How measure raw delays inside a router? (per output interface) :

What statistics to take? (rich, but compact) :

How to report? (low volume) :

Full Router Monitoring: Experimental Setup



Sprint

Full Router Monitoring: Numbers

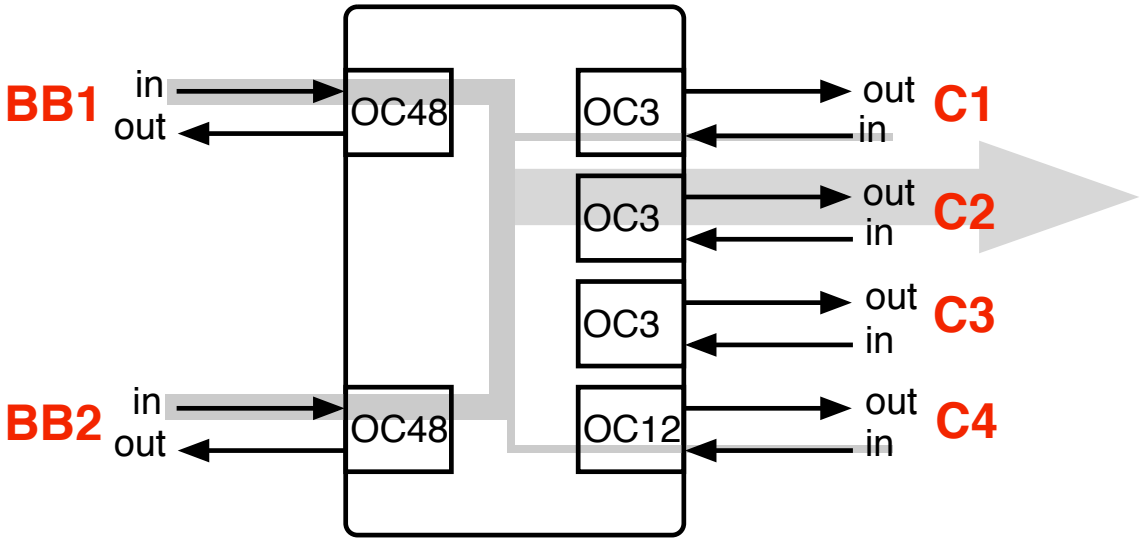
- Gateway router
- 13 hours of trace collection
- 7.3 billion packets
- 3 Terabytes of IP traffic

- Monitor more than 99.9% of traffic through router
- μs timestamp precision

Monitor moderately loaded **Access Link**

Full Capture Allows ‘Complete’ Packet Matching

Aim: Group records of same packet crossing different interfaces



Set	Link	Matched packets	% traffic on C2-out
C4	in	215987	0.03%
C1	in	70376	0.01%
BB1	in	345796622	47.00%
BB2	in	389153772	52.89%
C2	out	735236757	99.93%

Sprint

Anatomy of Through–Router Delay

- Store: full packet arrival to input linecard
- Forward: cross switch fabric to output linecard controller
- Output Queueing: queueing and serialisation at output linecard rate

Anatomy of Through–Router Delay

- Store: full packet arrival to input linecard — Exclude from system
- Forward: cross switch fabric to output linecard controller
- Output Queueing: queueing and serialisation at output linecard rate

Anatomy of Through–Router Delay

- Store: full packet arrival to input linecard
- Forward: cross switch fabric to output linecard controller
  - Model as packet size dependent minimum delay $\triangle(L)$
  - $\triangle(L)$ linecard & router dependent function – can be tabulated
- Output Queueing: queueing and serialisation at output linecard rate

Anatomy of Through–Router Delay

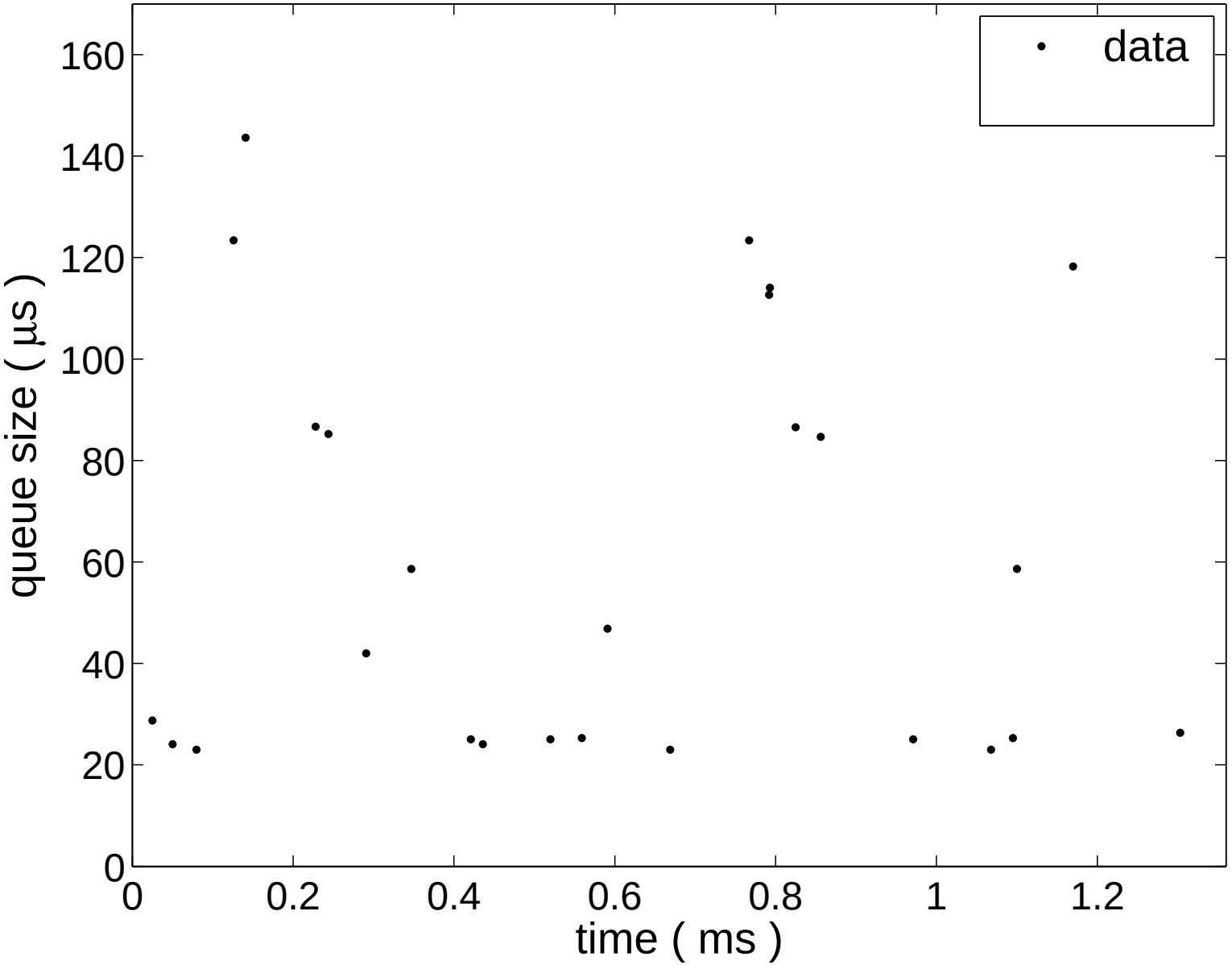
- Store: full packet arrival to input linecard
- Forward: cross fabric to output linecard controller
  - Model as packet size dependent minimum delay $\triangle(m)$
  - $\triangle(L)$ linecard & router dependent function – can be tabulated
- Output Queueing: queueing and serialisation at output linecard rate
  - Model as FIFO queue with deterministic service time

Anatomy of Through–Router Delay

- Store: full packet arrival to input linecard
- Forward: cross fabric to output linecard controller
  - Model as packet size dependent minimum delay $\triangle(m)$
  - $\triangle(L)$ linecard & router dependent function – can be tabulated
- Output Queueing: queueing and serialisation at output linecard rate
  - Model as FIFO queue with deterministic service time

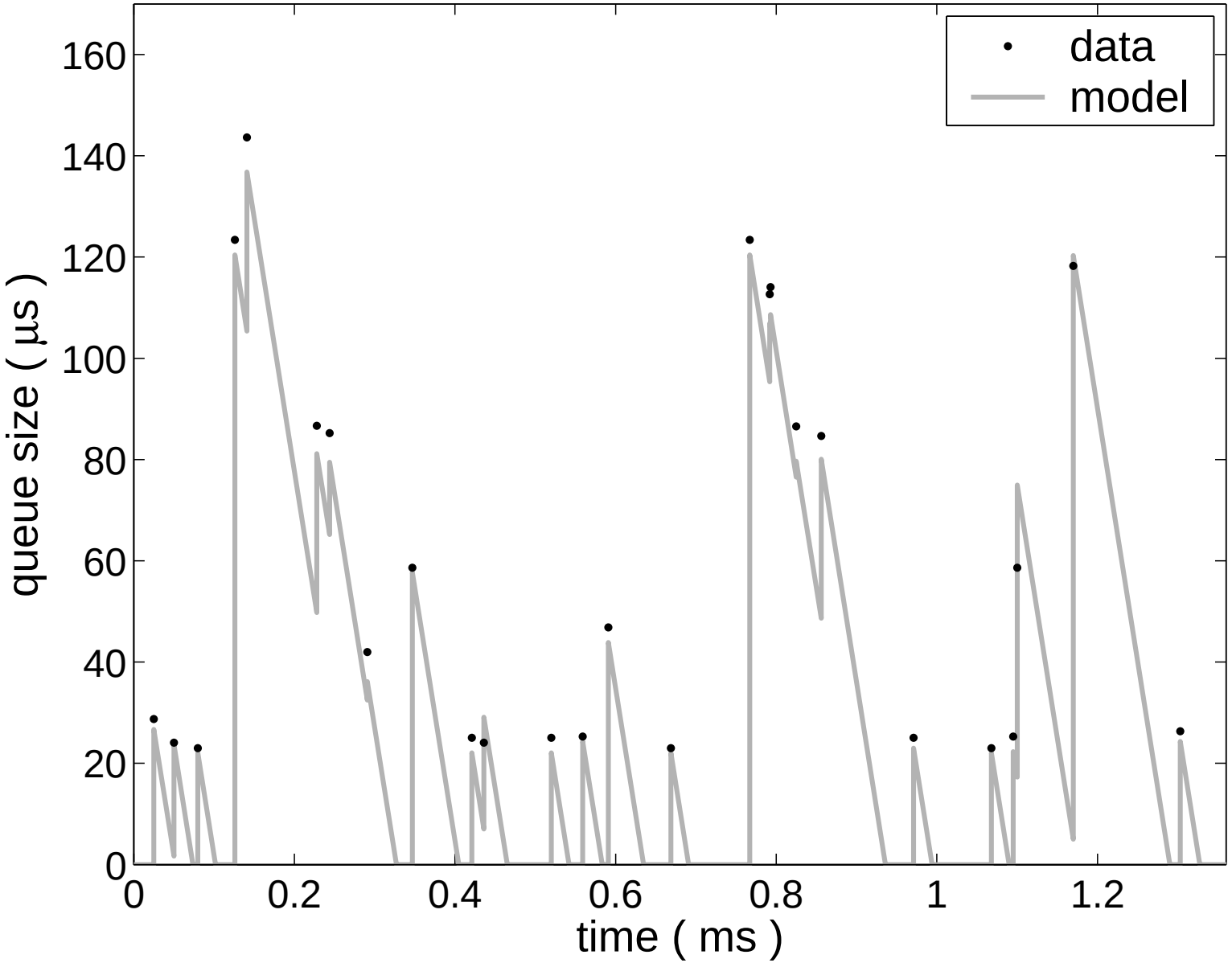
(simple delay at front end) + (output queue)

Some Measured Delays



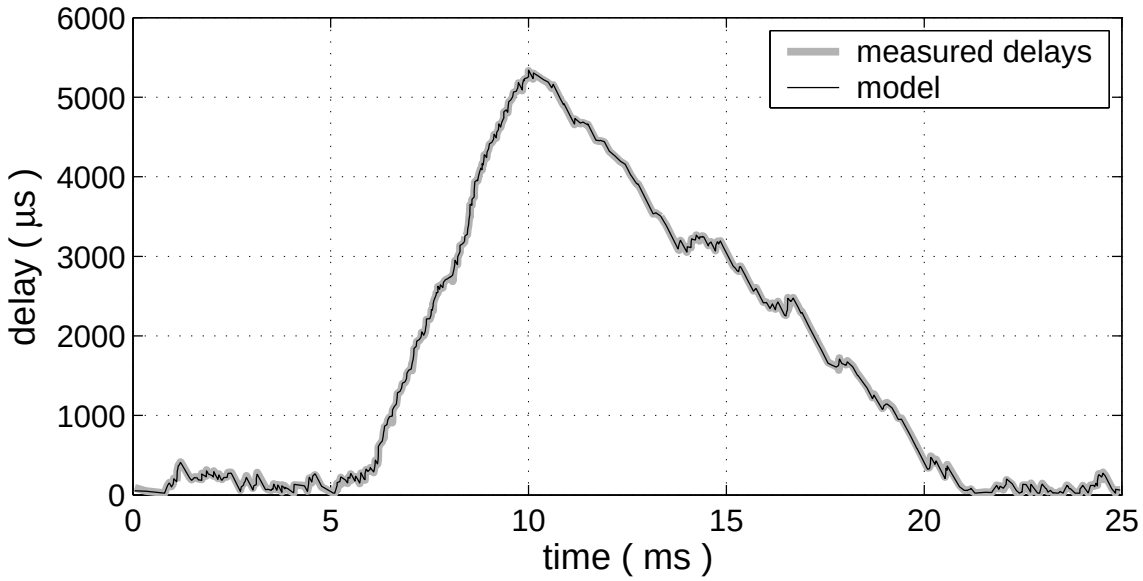
Sprint

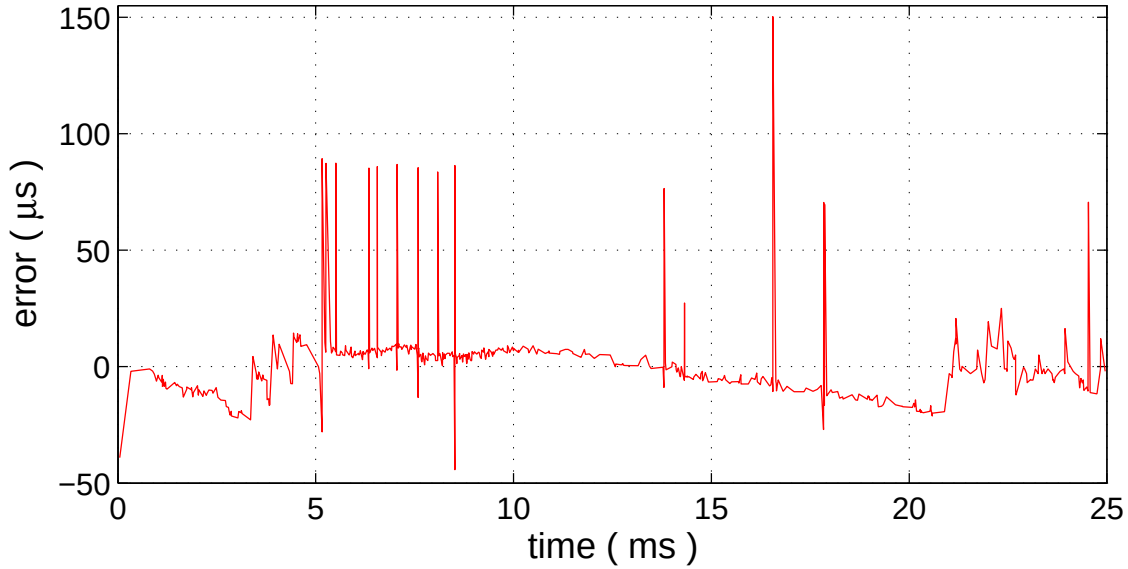
Model Validation



Sprint

The Model Works Well!





Sprint

Which Statistics?: Key is Busy Periods

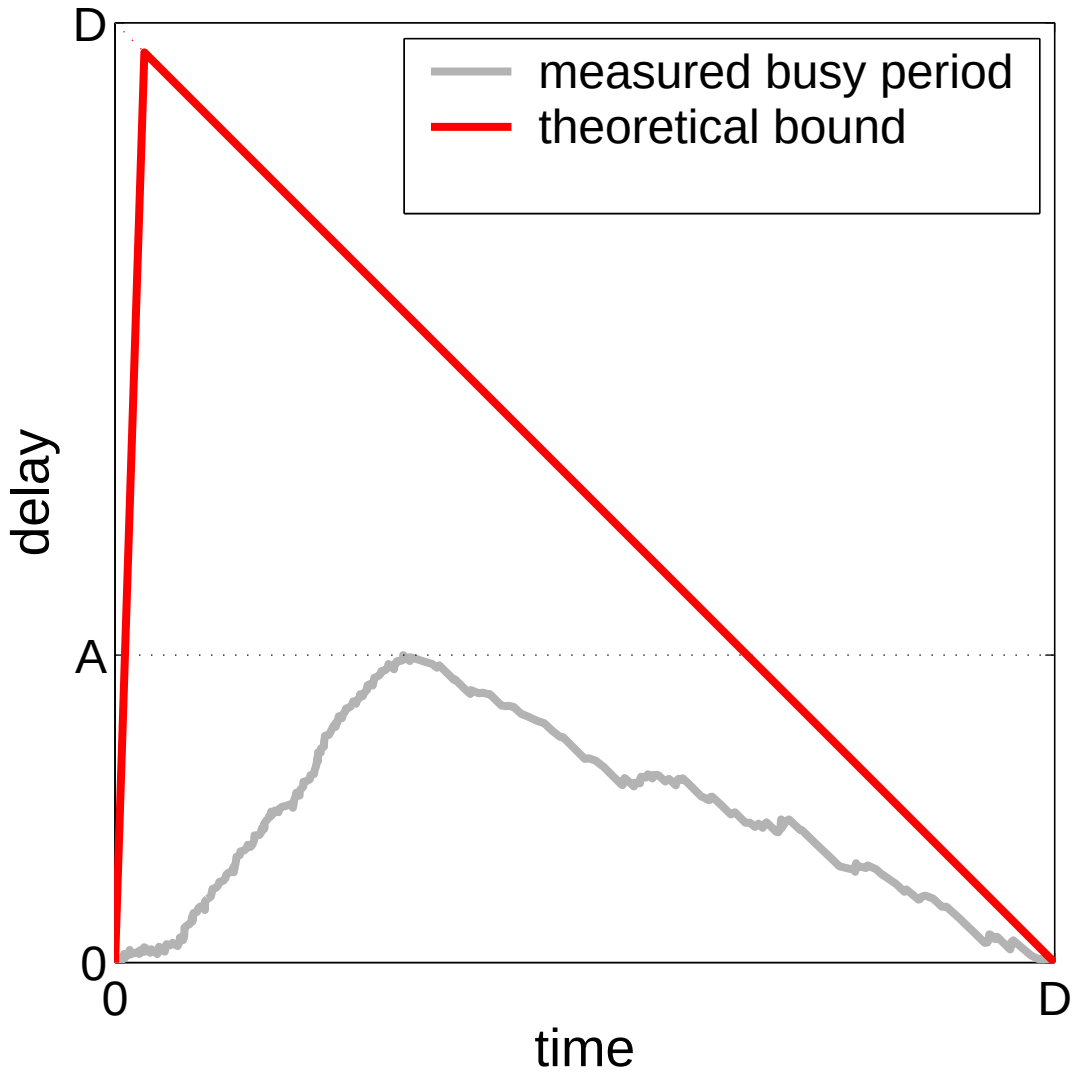
The model defines Busy Periods (pkt in system) and Idle Periods

Why focus on BPs?

- Input timestamps unavailable in routers:
  – but inside non-trivial BP's, Δ doesn't matter!
  – queue content tells all, measurable in routers
- BP's structure contains everything!

Anatomy of a Busy Period

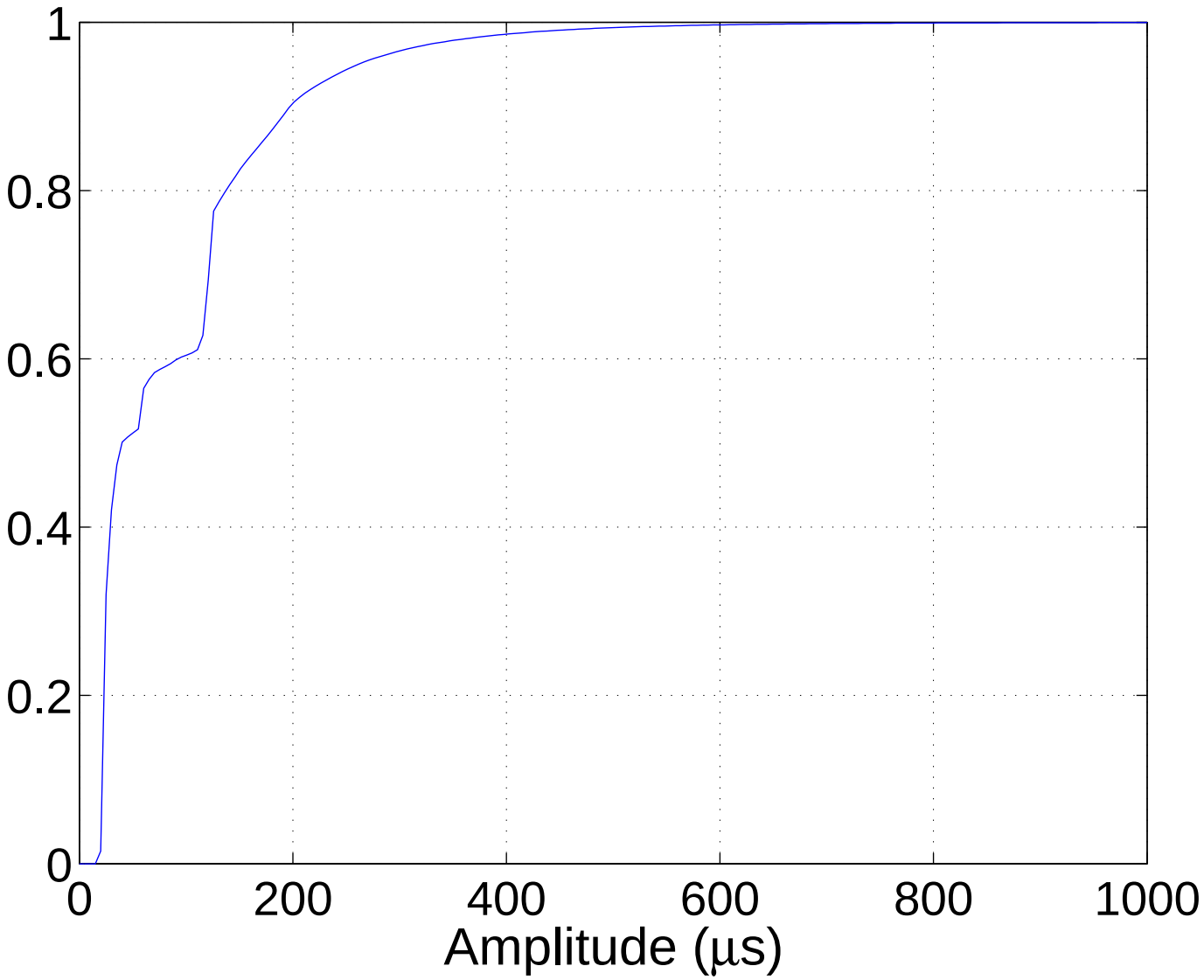
Amplitudes and Durations are important descriptors



Sprint

Busy Period Amplitudes

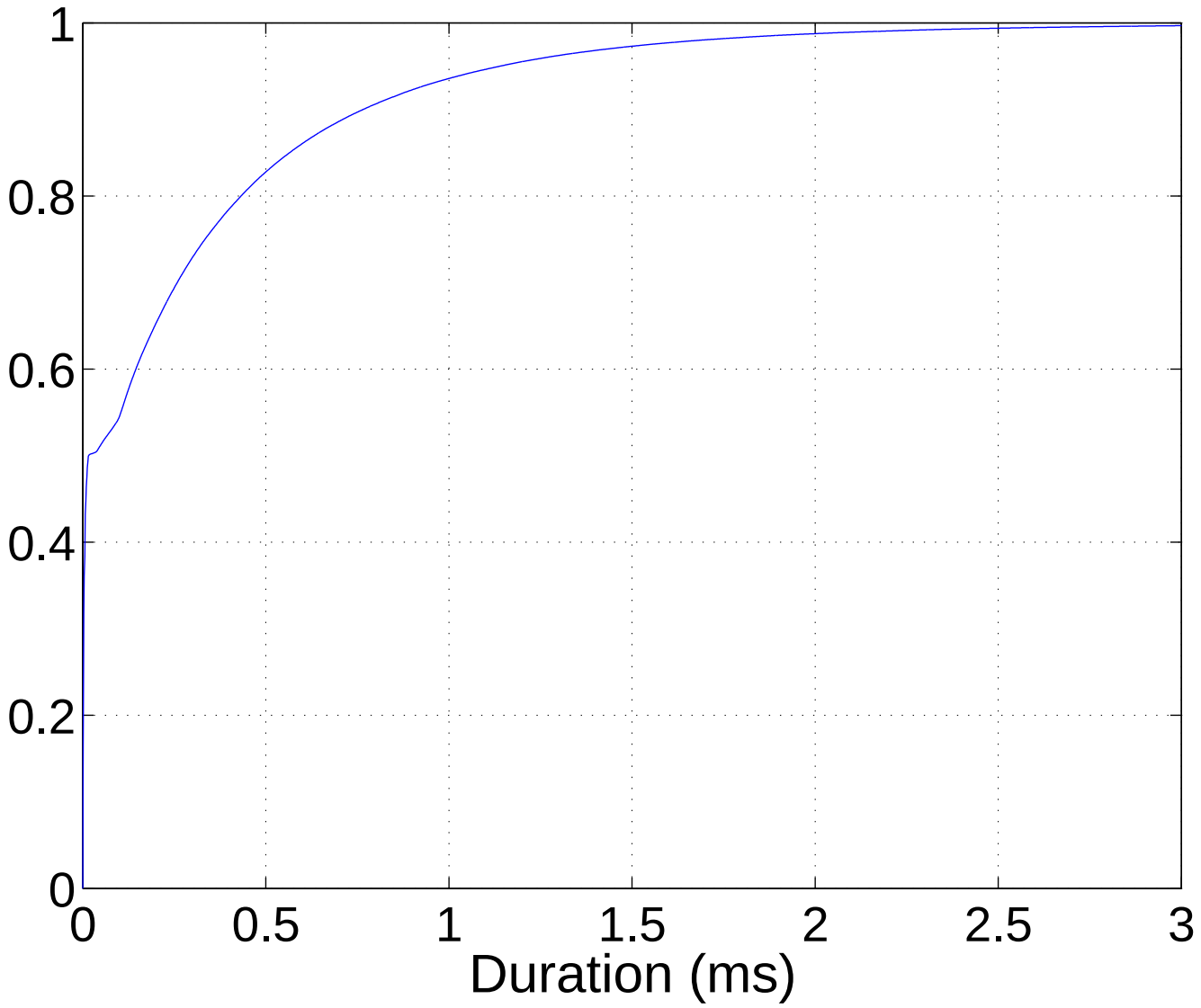
A measure of worst delay in congestion episode



Sprint

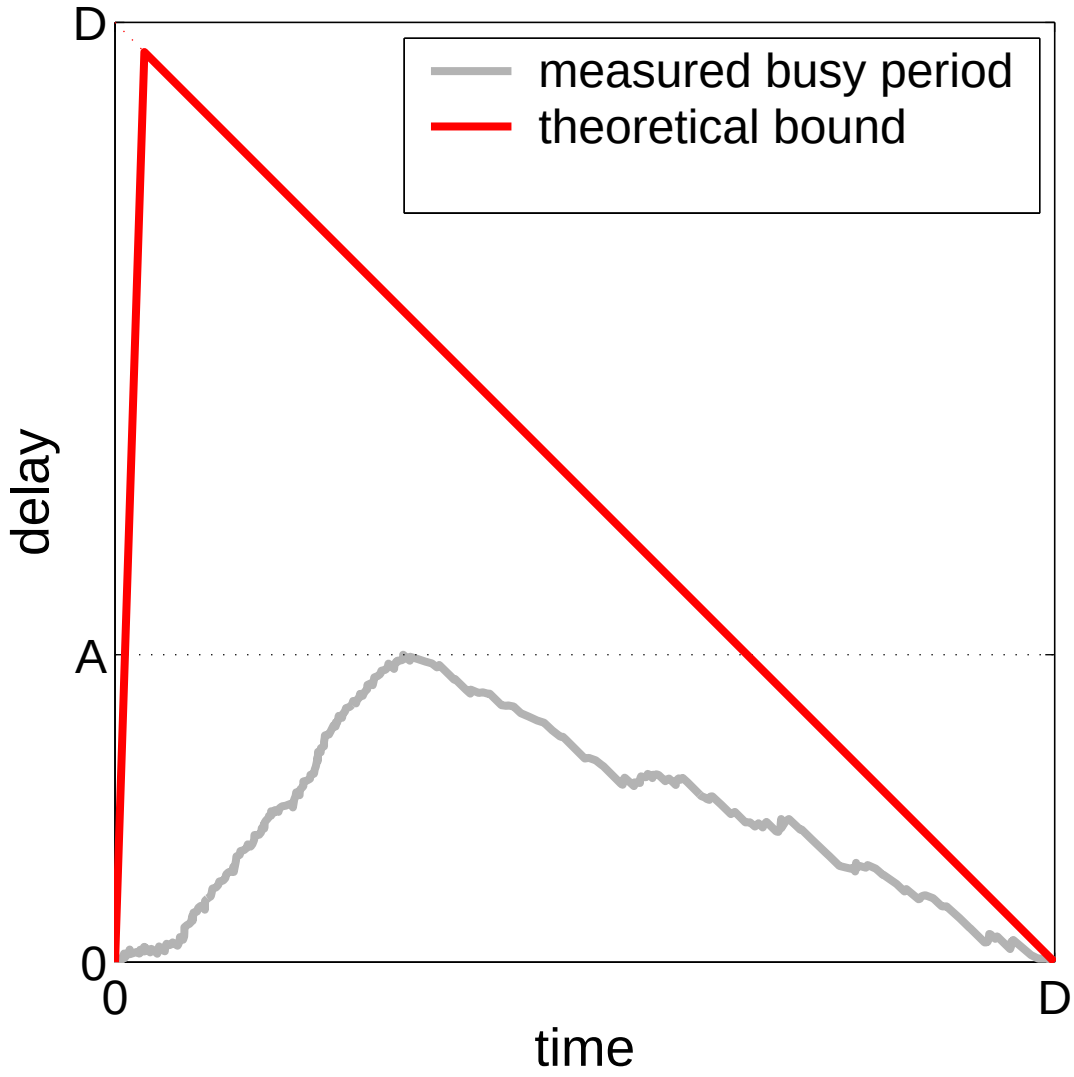
Busy Period Durations

A measure of duration of congestion episode



Sprint

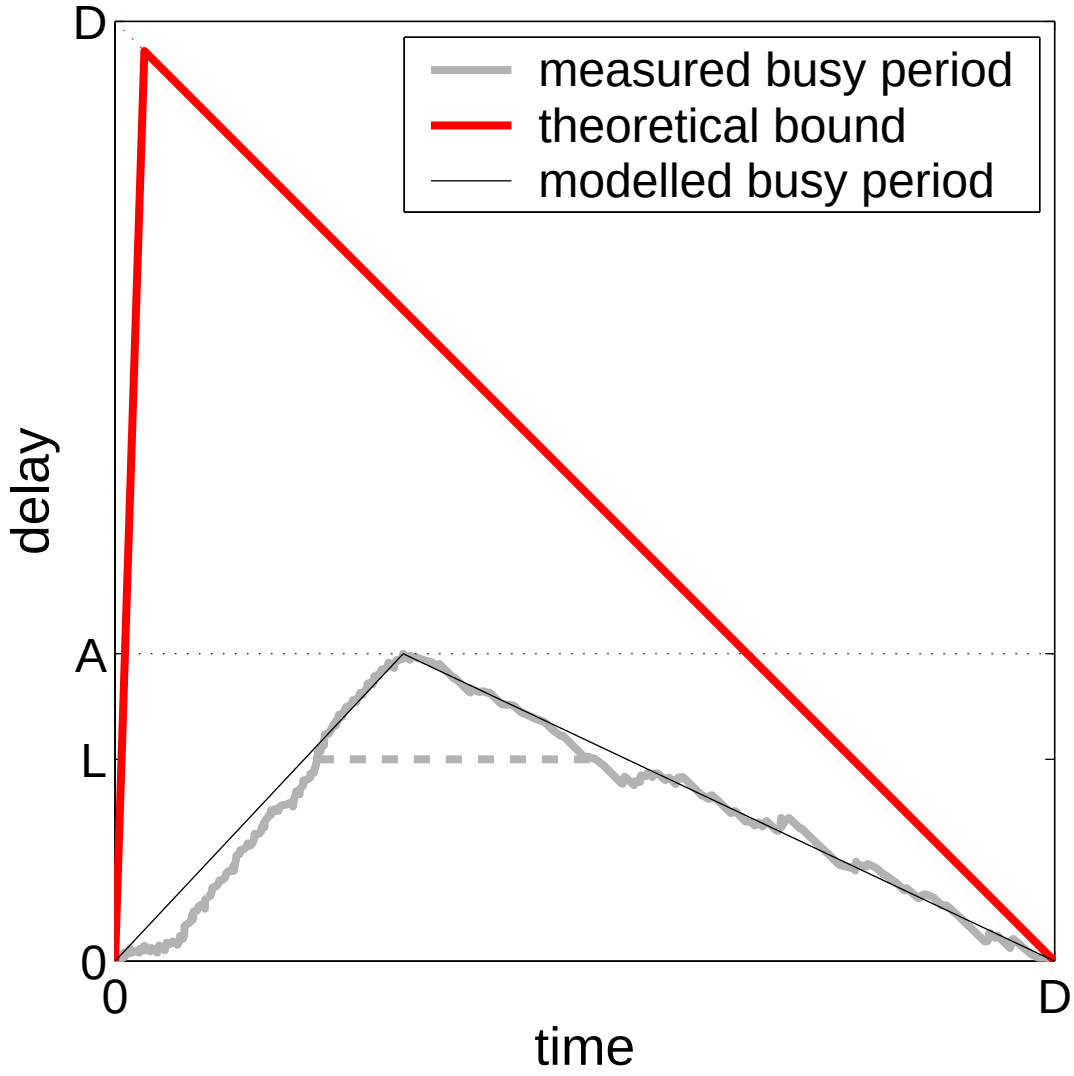
But what about BP Shape? and Why Bother?



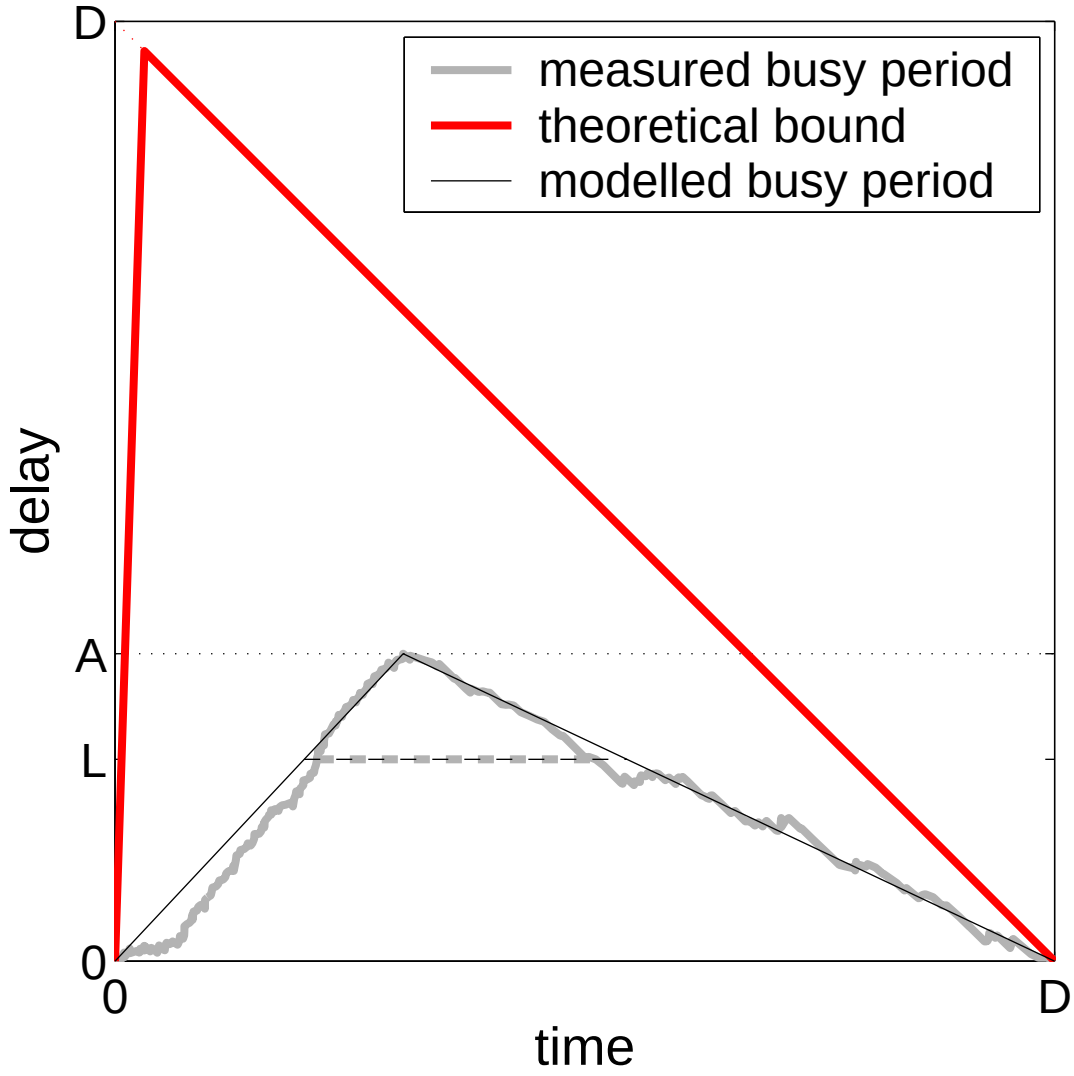
Rich Raw stats: Many more, detailed statistics accessible from postprocessing

Sprint

Duration of congestion episode of size L



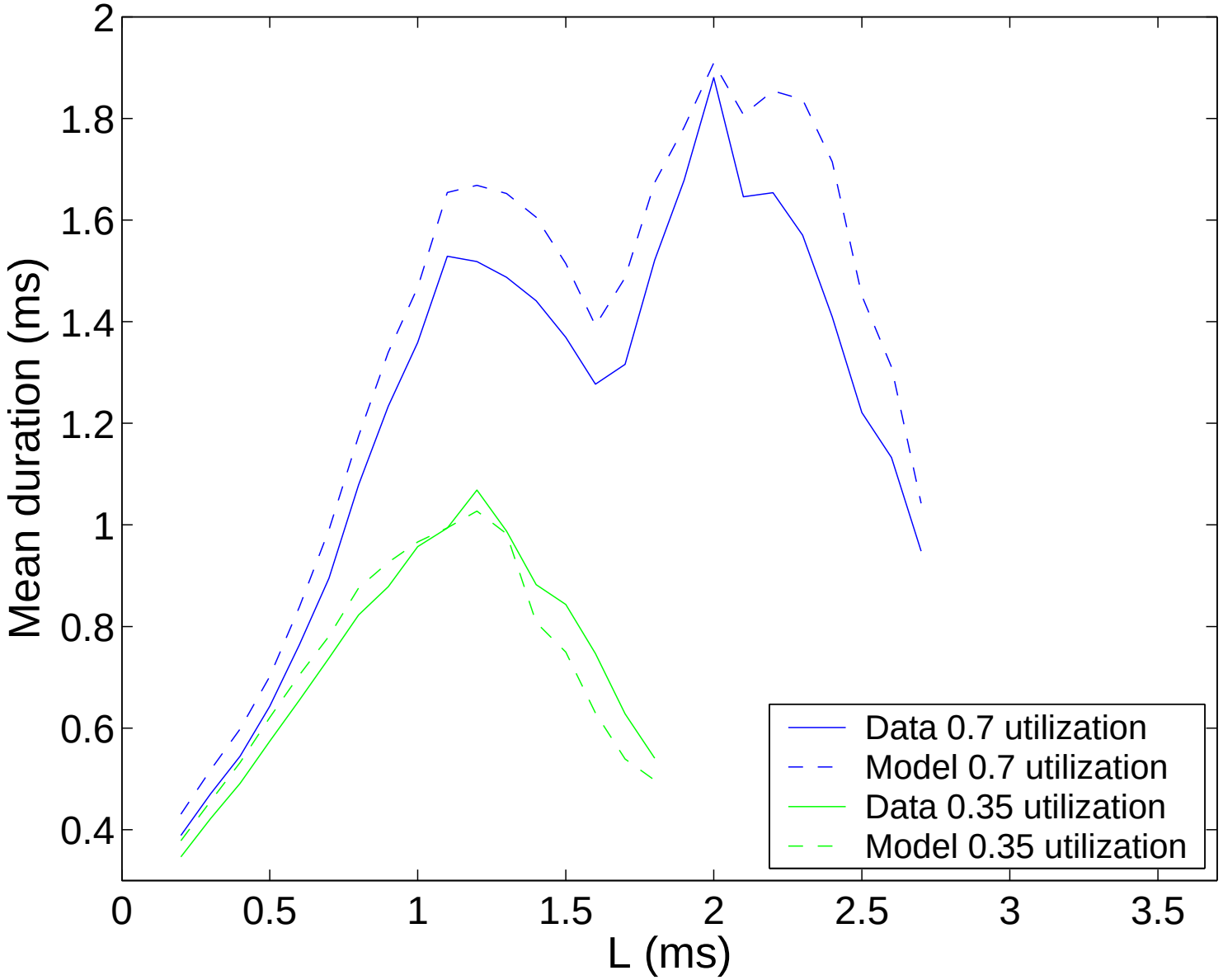
Use Simple Triangle Shape Model



Predicted duration: $d^{(T)}_{L,A,D} = D\left(1 - \frac{L}{A}\right)$ if $A \geq L$

Sprint

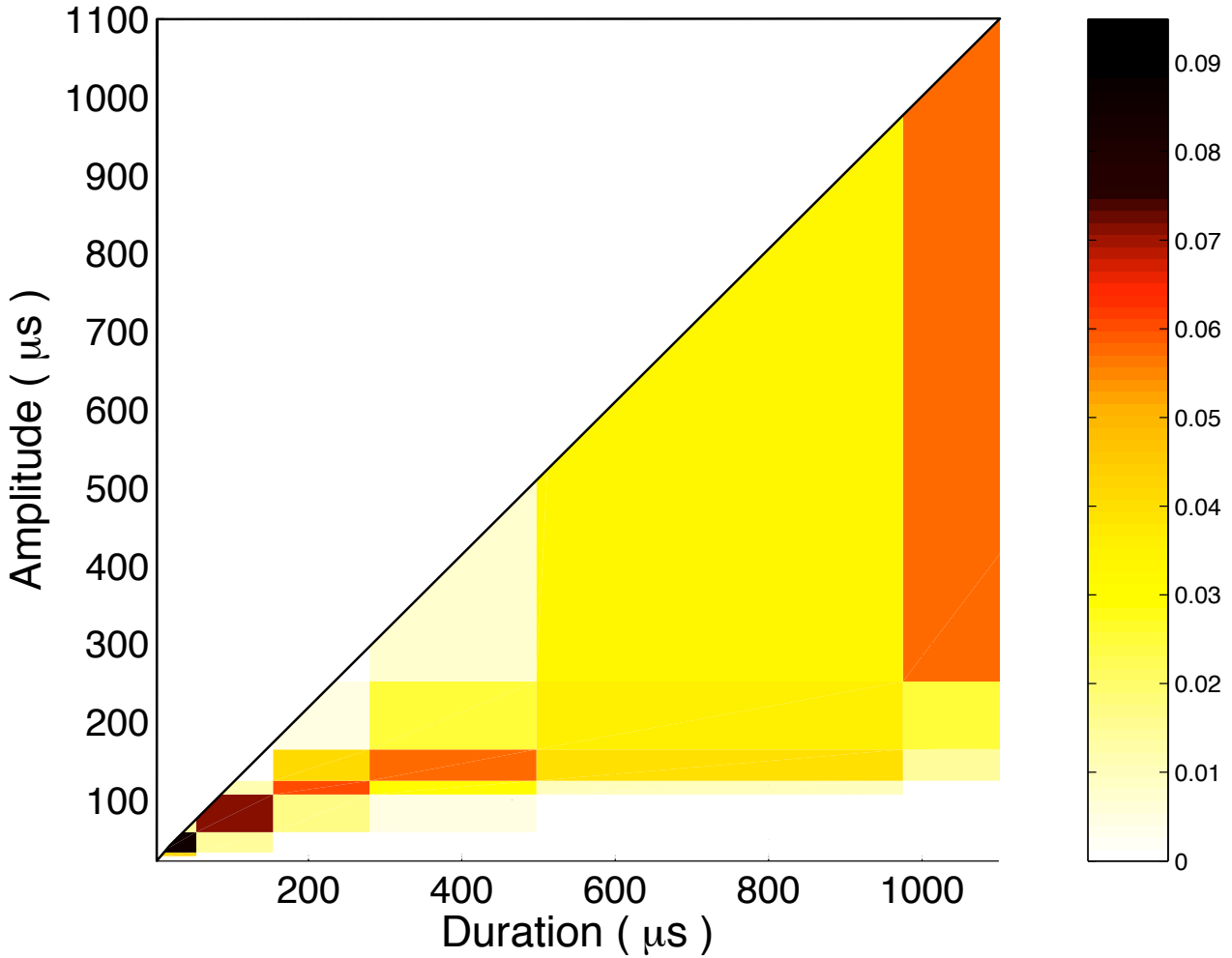
Congestion Duration Estimate: Performance



Sprint

How to Capture Shape? : BP (Duration,Amplitude) Distribution

A measure of shape of congestion episode – more complete picture



Select bins via quantiles: automatically adjusts to where action is.

Sprint

Reporting Statistics: Router

BP Based Algorithm

- timestamp BP start (queue moves from empty)
- within BP, track maximum queue size $q^* = \max(q^*, q_i)$
- end of BP:
  - timestamp BP end
  - calculate BP duration
  - store duration and amplitude $d^* = q^*/\mu$
- use high resolution histogram to limit memory (many single pkt BP's)

Exporting Statistics: SNMP

Determine number n of low resolution histogram bins from bandwidth

Every 5 minutes

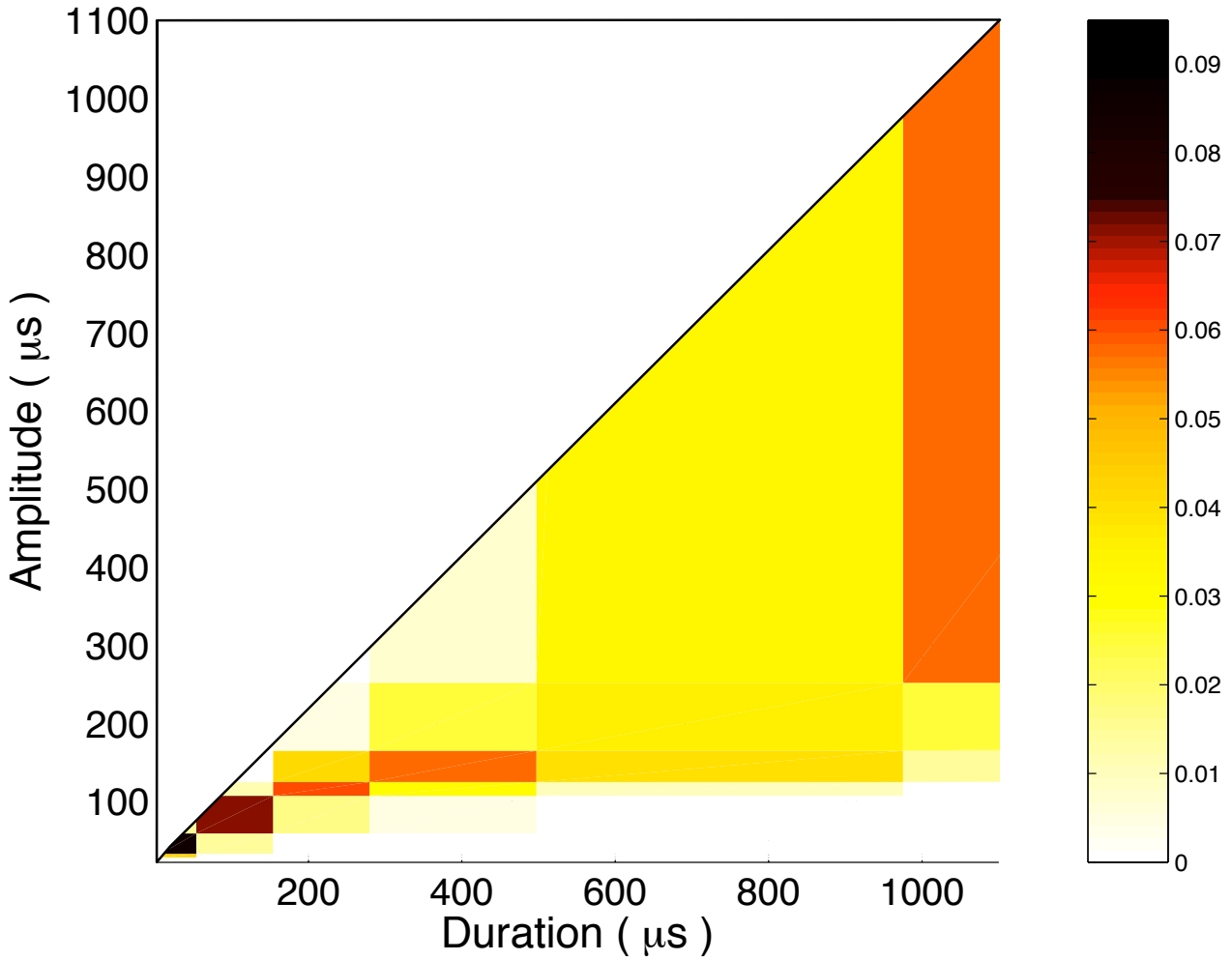
- form 1D histograms of duration and amplitude: n equally populated bins – automatically adapts to data, puts resolution where action is
- bin boundaries define $n^2/2$ 'boxes' of 2D histogram
- calculate 2D histogram, and export
  (two 1D histograms naturally included)

Also need/could

- report queue distribution (not BP based)
- basic Idle Period statistics, # packets etc.

An Example Exported Discretised 2D Distribution

Shows popular shapes, amplitudes, and durations



Sprint

Conclusion

- Accurate model for through–packet delay in S&F routers
- Delays can be measured by router
- Busy periods contain full set of delay and utilisation information
- Rich, Raw summary is possible via BP amplitude and duration
- Summary is computationally light, memory feasible
- Reporting with controlled volume through percentile based quantisation
- 2D distribution allows detailed and basic delay metrics
- Unforseen metrics derivable from raw data outside router!

Publications (accepted Sigmetrics preprint and Tech reports):
http://ipmon.sprint.com