Model Based Metrics

IPPM Working Group
IETF 92, Mar 2015

Matt Mathis <mattmathis@google.com>
Al Morton <acmorton@att.com>

Outline

- Document update
  - Mostly minor changes
  - Ready for WGLC?!?
- Revisit the larger problem
  - Why Model Based Metrics matter

Document changes -03 to -04

- Editorial cleanups
- Added several missing definitions
- Explicit linkage between network properties and tests (next slide)
- Dropped alternate method for computing run_length
- Dropped the CUBIC TCP model
    - Simplifying assumptions were not invertible
- Added a security section (oops!) and IANA considerations
- Updated references
- Passes id-nits
- Ready for WGLC
    - Have since found one editing oops
        - Truncated sentence, section 3 bullet 2

Required Network Properties

- Sufficient raw link data rate
  - Greater than target data rate by suitable overhead
- Sufficiently low packet loss
  - Less than bound computed by inverting performance model
- Sufficient buffering (queue) at dominant bottleneck
  - Smooth full window slowstart bursts
- Sufficient buffering to smooth sender interface bursts
  - Full window suggested but not required
- Bounded data and ACK interaction
  - Channel arbitration must minimally preserve self clock
- Progressive onset of loss/ECN when there is a standing queue
  - AQM of any sort

Any issues before WGLC?

- I think it is ready

Why Model Based Metrics?

The bigger picture

IPPM Working Group
IETF 92, Mar 2015

Matt Mathis <mattmathis@google.com>

Part 2: Why Model Based Metrics?

- History:
  - Could not debug long paths
  - The vBNS solicitation
  - 1993 Whiteboard SLA (non) design launched IPPM
- Why we were stumped
  - TCP equilibrium behavior causes circular dependencies
- Revisit the 1993 whiteboard
  - MBM fixes the problem

Typical 1990s debugging scenario^H^H^H^H^H nightmare

- Some (university research) end-to-end application fails
- Each provider (campus, backbone) tests their own section
  - All sections pass
- Some industrious users try “divide and conquer” debugging
  - Always “proved” that some other section was to blame
  - But were generally wrong
- Basic performance measurements are not useful diagnostics
  - Missing key repeatability properties
  - Extremely vantage point sensitive
  - No mapping from symptoms to root cause
    - Not actionable

The NSF drafts vBNS solicitation (1993?)

- NSP planned to build a separate R&E backbone
- But why build an independent network/service?
    - ANSnet transitioning from NSF (US gov) to commercial
    - MCI and others spinning up commercial backbones
- Why not a Service Level Agreement for commercial services?
    - Could have helped to bootstrap the commercial Internet
- Oh wait..... but this is hard
    - But it started me thinking about how to write a performance SLA
    - The following is reconstruction of an old whiteboard SLA design

1993 Whiteboard design: ISP scale performance SLA

- Wanted an "end point to cloud metric"
    - Estimate performance between one endpoint (user)
    - to some weighted aggregate of all other endpoints (content)
    - but partitioned by first ISP vs rest of path
        - To properly attribute responsibility

- And an "entire ISP cloud metric"
    - A weighted aggregate of the above for all ISP users
    - Summarize an ISP as a small set of metric values

But we knew we were missing some key parts

- We didn't know:
  - What metric(s) to use;
  - How to partition path properties;
  - Or how to pick endpoints;
  - Or how to do the (weighted) aggregation;
  - Or a whole bunch of other little pieces.

- This conversation led.....
  - me to push for the formation of IPPM and chair the 1st BOF
  - the start of several other instrumentation and measurement projects

A performance SLA is a very hard problem

- IPPM early years
  - "Analytic Framework" in [RFC 2330] never materialized
    - Wanted to develop an algebra on concatenated path metrics
    - More below
  - BTC framework [RFC 3148] hasn't really gone anywhere
    - Does not address non-repeatability issues
  - TCP Macroscopic Model [ACM CCR July 1997]
    - Launched TFRC and other myths
    - See my editorial "Reflections on the TCP Macroscopic Model" [ACM CCR Jan 2009]
  - Overlooked consequences of TCP bugs
    - Masked most of the real issues
- This kept us all stumped for two decades
  - I focused on TCP performance

IPPM wanted an algebra on TCP performance metrics

- Performance (BTC) measurements are vantage sensitive
- Can not "subtract" effects of testing tail circuits
    - Can't predict B-C from measures of A-B-C and A-B
- Can not predict measurements of concatenated paths
    - Can't predict A-C from measures of A-B and B-C
- We envisioned an algebra on metrics solving:
    - The university debugging problem
    - The property partitioning part of the SLA problem
- Motivated Analytic-Framework language in RFC 2330

Measurement issues were obscured by TCP issues

- Many "network" problems were actually TCP problems
  - Obscuring the real IP measurement problems
- All flows have bottlenecks at the edges
  - Either the access link or implicit to TCP

- Non-lame TCPs roll out: Linux (2005), Windows 7 (2009)
  - TCP works like the textbooks say it should

Other deep problems masked by lame TCPs (off topic)

- Bufferbloat
- TCP Rate Friendly does not work
- Lack of a real capacity allocation architecture for the Internet
  - All flows were bottlenecked at the edge (TCP or access link)
  - "Easy" to out build the load in the core
- The presumption that the Internet is fair by default, when it is not
  - Common situations where fair designs exhibit egregiously unfair behaviors

- Non-lame TCPs roll out: Linux (2005), Windows 7 (2009)
  - TCP works like the textbooks say it should

Better TCPs change the problem

- For all modern throughput maximizing protocols
  - Any long running bulk flow always raises RTT, loss or ECN marks
  - Between any pair of hosts anywhere in the entire internet
- Congestion control is an equilibrium process
  - Circular dependencies between data rate, RTT, loss and ECN
  - Any perturbation to any parameter changes all the rest
  - Any perturbation to any element changes everything
- Circular dependencies in TCP spoils IP measurement
  - Measured congestion is always a reflection of TCP's own CC
  - All network measurements depend on all components

Model Based Metrics put us back on the path forward

- Suppress equilibrium behavior by open looping TCP
  - IP test traffic mimics TCP independent of the network details
  - IP success criteria is based on TCP models
  - Eliminate circular interactions between RTT, packet loss and data rate
- We understand path composition for pure IP properties

- The performance singleton changes
  - From: How fast will TCP go?
  - To: Will TCP go fast enough for my application? (a predicate)

And suddenly the 1993 SLA metrics become clear

- We can now define the pieces
    - "Partitioned" is by apportioning the MBM loss budget
    - "Weighted Aggregate" is weighted probability of success
    - subpaths have to be edge-to-edge, beyond the edges
    - Type-P has to include all marking or classifications

- Ultimately the metric becomes:

"What is the estimated probability that an ISP can support my application at any given time to any remote server, under the assumption that the rest of the path is at least as good as the chosen ISP?"

Is this work of interest to the WG?

- Any interested co-authors?