INTERNET TRAFFIC AND QoS

Simplest of all: constant bit rate (CBR)

→ flat is good
→ because predictable
→ e.g., telephone call, real-time MP3 audio

Internet data traffic: variable bit rate (VBR)

→ session arrivals: random
→ exponential interarrival time
→ swimming with the fishes: Poisson process
→ e.g., telephone calls, TCP connections, fast food
What does Internet traffic look like?

Host A → Host B

LAN X → LAN Y

Router → Backbone

Traffic Meter

0ms 10ms 20ms 30ms 40ms t

→ logging

→ traffic time series (at 10ms granularity)
Aggregation (time):

→ analogous to computing sample mean
→ aggregation over multiple time scales
→ what to expect?
Aggregated traffic becomes flat

→ “flat is good” rule for QoS provisioning
→ bandwidth dimensioning
→ technically: law of large numbers in action
→ not correlated in time
→ efficient and happy customers

Also aggregation over multiple users

→ called statistical multiplexing
→ assuming independence between different users
→ nice normal distribution shape
→ allows non-peak reservation
→ consider peak-to-mean ratio: less costly
• LLN: principal engineering tool used by large transit providers and large access providers

→ “largeness” is key

→ even though components are random, system is well-behaved and predictable

→ apply at ingress/egress and backbone links

→ measurement-based tool: traffic matrix
Network engineering:

- Feedback traffic control
  - closed-loop control ("adaptive")
  - small time scale: msec
  - mainly by end systems
  - e.g., congestion control

- Resource provisioning
  - open-loop control ("in advance")
  - large time scale: seconds, minutes, and higher
  - mainly by service providers

Question: what do ISPs do to keep customers happy
QoS policy:

- Per-user (or flow) reservation for super-quality service
  → guaranteed service

- Shared service classes (platinum, gold, silver, bronze)
  for good service
  → differentiated service

Internet standards:

- IETF IntServ
  → RSVP protocol
  → analogous to leasing a line

- IETF DiffServ
  → different types of router behavior
  → AF, EF, Cisco’s LLQ for VoIP
Cisco 7206VXR router: packet loss rate

→ 8 classes

→ OC-3 link

→ varying offered load
Back to Internet traffic:

→ what does it look like?

→ doesn’t become flat with time aggregation

→ stays bursty!

→ in a peculiar fashion: self-similar or fractal
Some fractal objects:

Menger sponge (picture from www.ics.uci.edu/∼eppstein):

Fractal fern:

—are fractal objects random?
Internet: self-similar

Telephony: Poisson-like

peak
avrg
Consequences:

→ cannot use “flat is good” method anymore
→ intrinsic trade-off between QoS and efficiency
→ bad news for QoS provisioning
→ traffic must be correlated in time (why?)
Ex.: on/off model

\[ X_1(t) \]
\[ X_2(t) \]
\[ X_3(t) \]
\[ X(t) \]

\[ \text{on-period: TCP file transfer} \]
\[ \text{on-period length: file transfer completion time} \]
\[ \text{ignore internal details within on-period: sawtooth} \]
\[ \text{on-period could be VoIP session: CBR} \]
\[ \text{not exactly: a user talks only 40\% of the time} \]
\[ \text{approximate view: ok by Amdahl’s law} \]
\[ \text{“don’t fret about small things”} \]
We know session arrivals are (approximately) Poisson; what about session lifetimes?

Important fact: TCP session lifetimes are heavy-tailed

\[ \Pr\{Z > x\} \approx x^{-\alpha} \]

\[ \rightarrow \text{ as opposed to: } \Pr\{Z > x\} \approx e^{-bx} \]

\[ \rightarrow \text{ exponent: } 1 < \alpha < 2 \text{ (closer to 1)} \]

\[ \rightarrow \text{ note: different from Internet connectivity power-law} \]

\[ \rightarrow \text{ much more likely session will last a long time} \]

\[ \rightarrow \text{ has finite mean but infinite variance} \]

\[ \rightarrow \text{ cat has a very fat tail ("too fat to carry")} \]

Why would TCP session lifetimes be heavy-tailed?

\[ \rightarrow \text{ TCP traffic makes up bulk of Internet traffic} \]

\[ \rightarrow \text{ greater than 80\%} \]
Tale of elephants and mice:

\[\rightarrow\] UNIX and WWW file systems

\[\rightarrow\] many small ones (mice)

\[\rightarrow\] a few very large ones (elephants)

\[\rightarrow\] 10% consumes 90% of bandwidth
How to check if files sizes are heavy-tailed?

Since $\Pr\{Z > x\} \approx x^{-\alpha}$, take logarithm on both sides:

$\rightarrow \log \Pr\{Z > x\} \approx -\alpha \log x$

$\rightarrow$ linear function with negative slope $-\alpha$

$\rightarrow$ holds true for large $x$

$\rightarrow$ what’s the slope $\alpha$?

$\rightarrow$ we don’t care about details of small sizes (why?)
Elephants in action:

$\rightarrow$ at backbone router
Can the problem be solved?

→ no: as long as elephants and mice holds

Turns out to be a wide-spread phenomenon is sociology, networks, and elsewhere

→ size (population) of cities
→ popularity
→ frequency of words in books
→ etc.

In the real world:

→ norm: skewed distribution of sizes
→ power-law
Sample mean convergence rate: exponential vs. Pareto

Exponential: rate=1/60

Pareto: alpha=1.2, b=10
Real-time VBR source profile

Consider traffic profile of compressed video (e.g., MPEG)

→ periodic real-time application

→ period $T = 1/f$ ($f$: frame rate)
Burstiness structure:

→ burstiness persists across multiple time scales
Possible causes:

- Heavy-tailed scene durations
  \[\rightarrow\] facilitates inter-frame compression

- Repeating GOP pattern
  \[\rightarrow\] e.g., I B B P B B

![Scene Duration vs. New Scene](chart.png)