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)

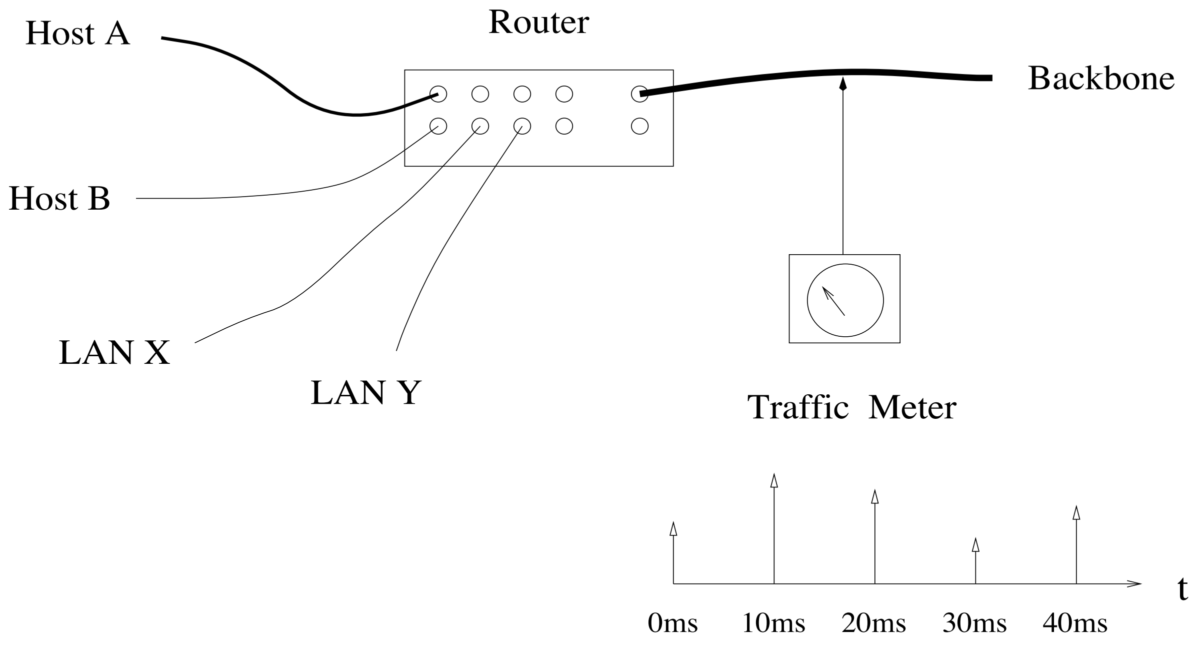
⟶ session arrivals: random

⟶ exponential interarrival time

⟶ swimming with the fishes: Poisson process

⟶ e.g., telephone calls, TCP connections, fast food

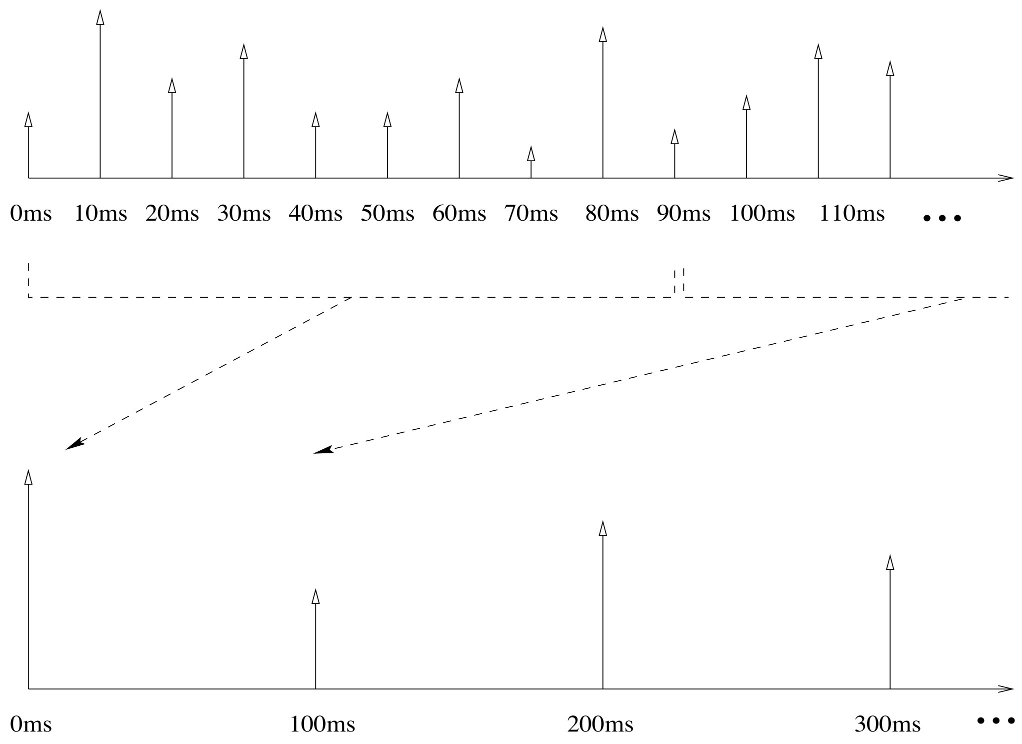
What does Internet traffic look like?



⟶ logging

⟶ traffic time series (at 10ms granularity)

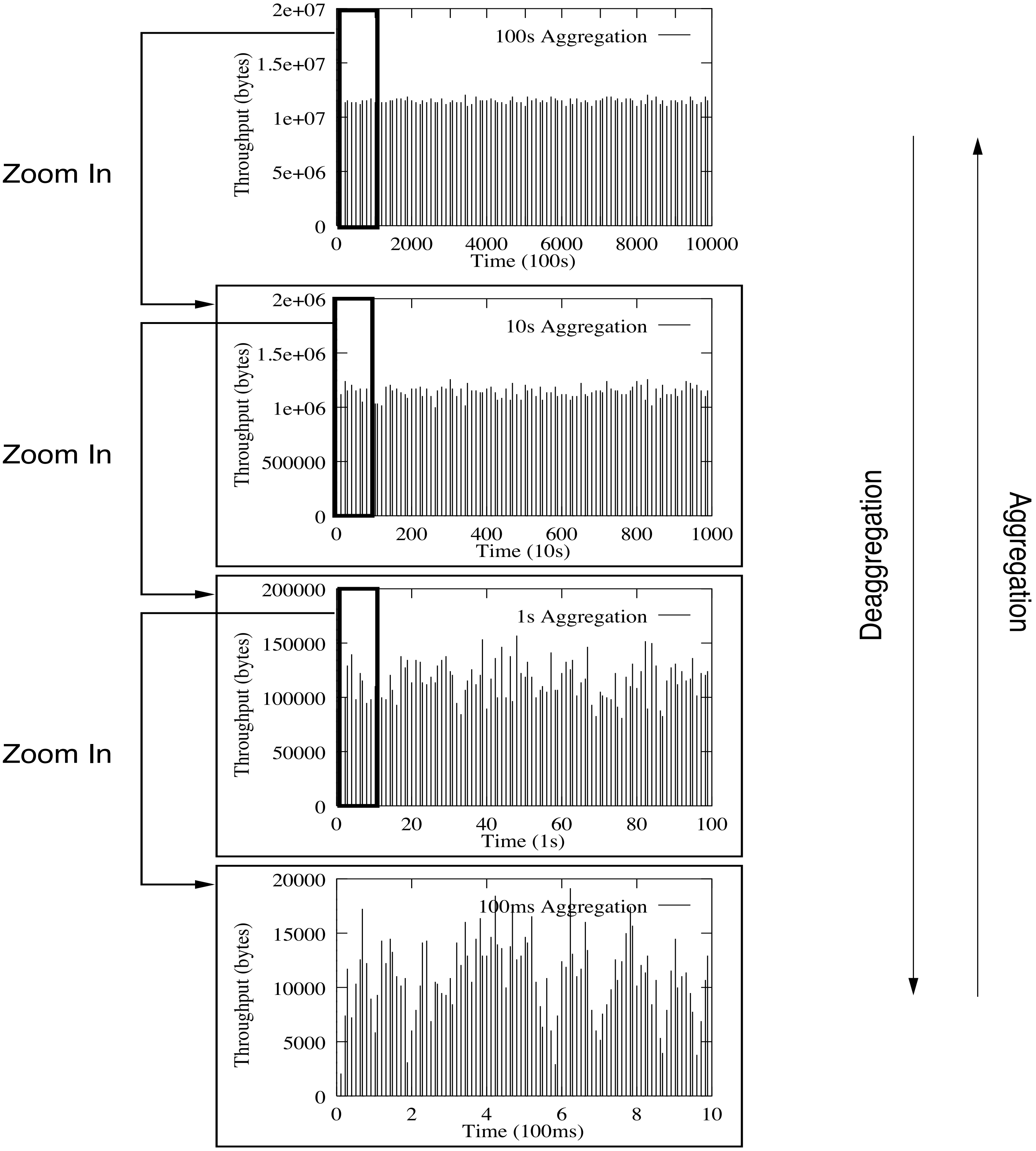
Aggregation (time):



→ analogous to computing sample mean

→ aggregation over multiple time scales

→ what to expect?


Zoom In
Zoom In
Zoom In
Deaggregation
Aggregation
100s Aggregation
Throughput (bytes)
0
5e+06
1e+07
1.5e+07
2e+07
0
2000
4000
6000
8000
10000
Time (100s)
10s Aggregation
Throughput (bytes)
0
500000
1e+06
1.5e+06
2e+06
0
200
400
600
800
1000
Time (10s)
1s Aggregation
Throughput (bytes)
0
50000
100000
150000
200000
0
20
40
60
80
100
Time (1s)
100ms Aggregation
Throughput (bytes)
0
5000
10000
15000
20000
0
2
4
6
8
10
Time (100ms)

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

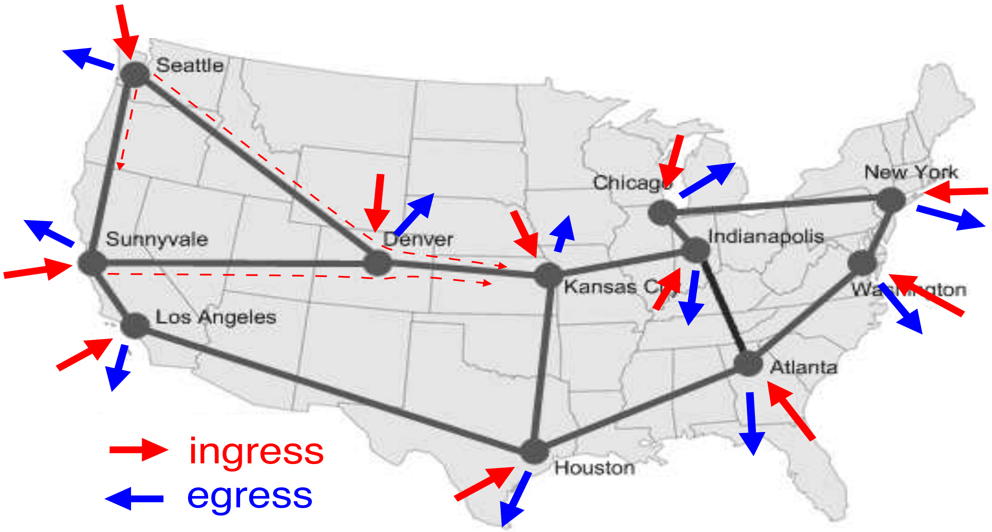
- LLN: principal engineering tool used by large transit providers and large access providers

  → "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

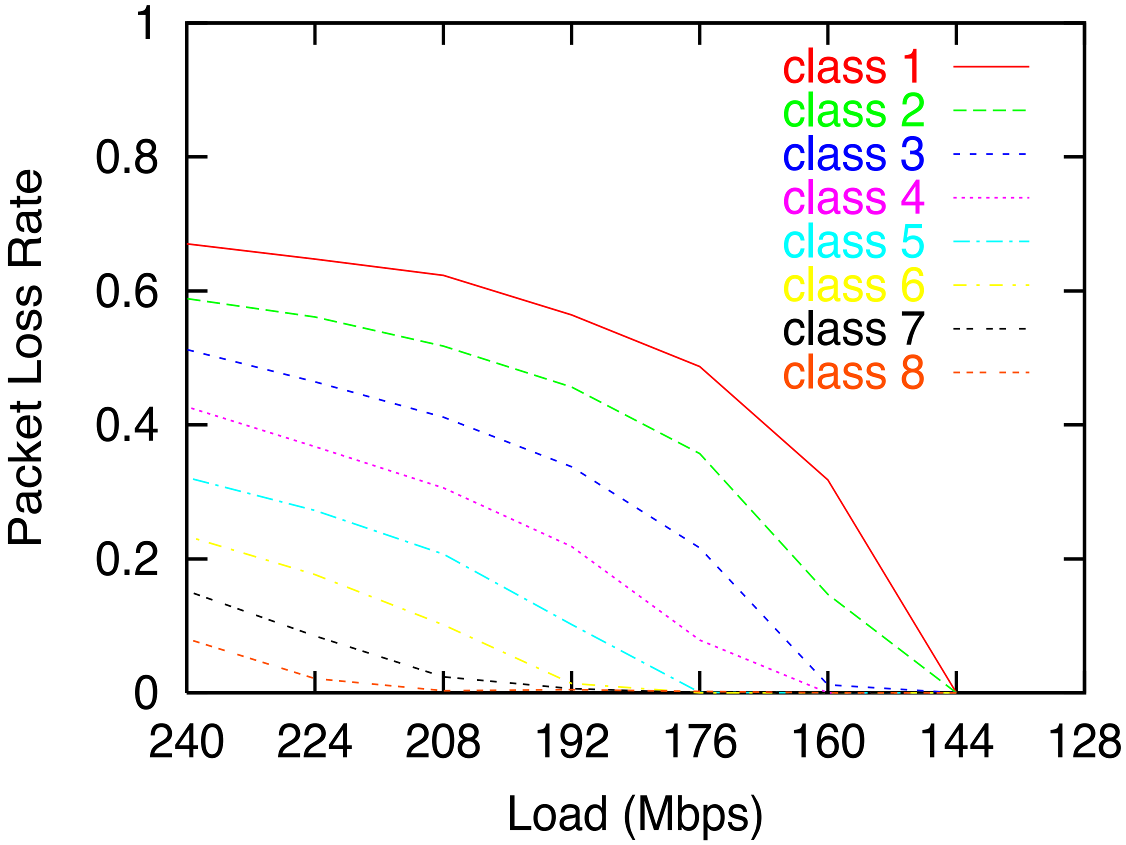
  → analogous to leasing a line

- 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

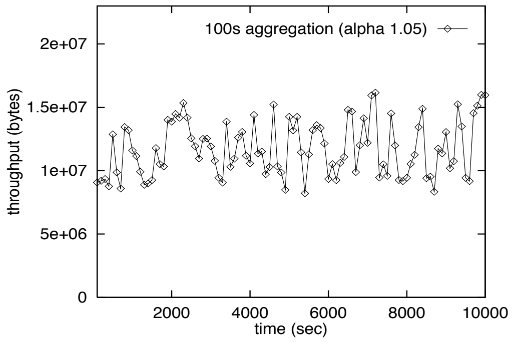
Back to Internet traffic:

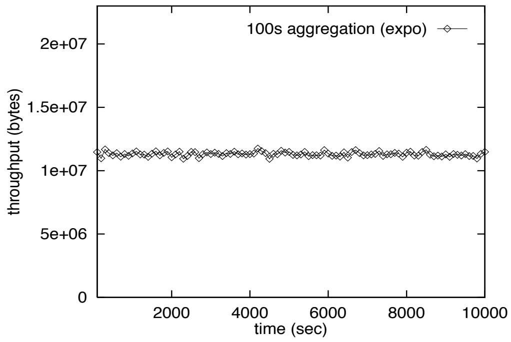
⟶ what does it look like?

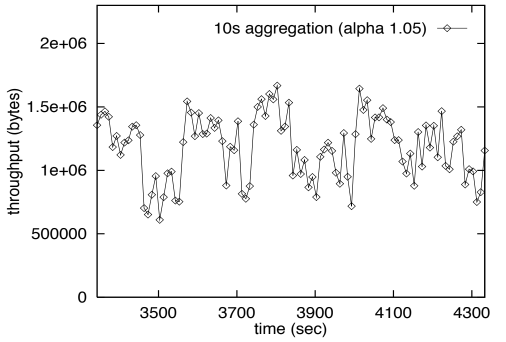
⟶ doesn't become flat with time aggregation

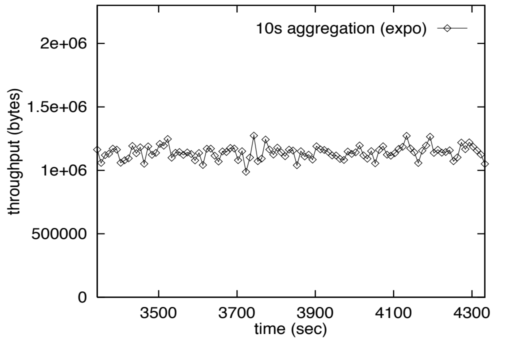
⟶ stays bursty!

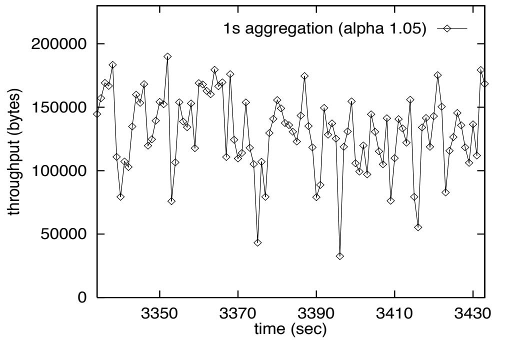
⟶ in a peculiar fashion: self-similar or fractal

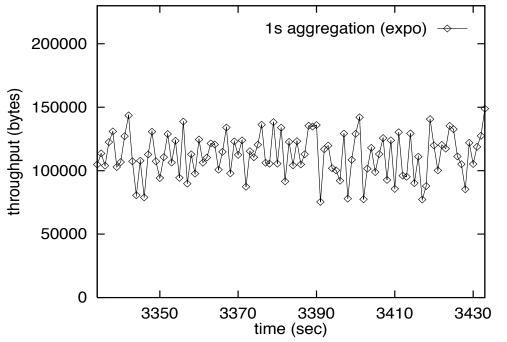

100s aggregation (alpha 1.05)
throughput (bytes)
0
5e+06
1e+07
1.5e+07
2e+07
2000
4000
6000
8000
10000
time (sec)

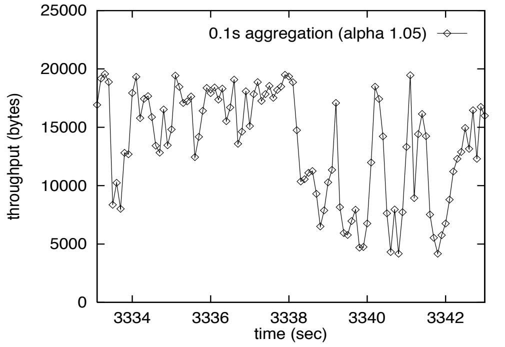

100s aggregation (expo)
throughput (bytes)
0
5e+06
1e+07
1.5e+07
2e+07
2000
4000
6000
8000
10000
time (sec)

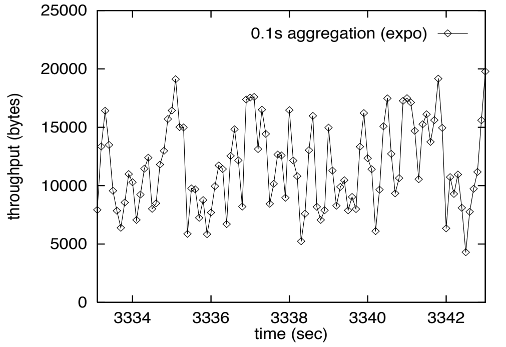

10s aggregation (alpha 1.05)
throughput (bytes)
0
500000
1e+06
1.5e+06
2e+06
3500
3700
3900
4100
4300
time (sec)

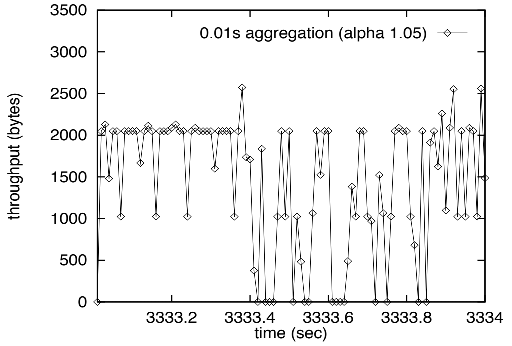

10s aggregation (expo)
throughput (bytes)
0
500000
1e+06
1.5e+06
2e+06
3500
3700
3900
4100
4300
time (sec)

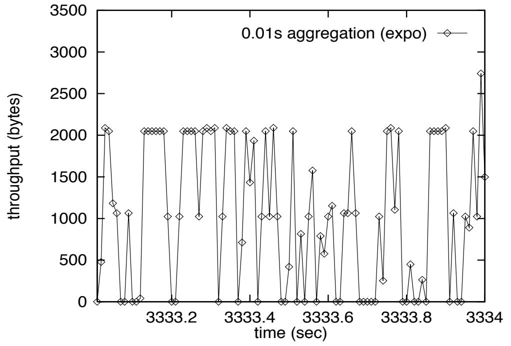

1s aggregation (alpha 1.05)
throughput (bytes)
0
50000
100000
150000
200000
3350
3370
3390
3410
3430
time (sec)


1s aggregation (expo)
throughput (bytes)
0
50000
100000
150000
200000
3350
3370
3390
3410
3430
time (sec)


0.1s aggregation (alpha 1.05)
throughput (bytes)
0
5000
10000
15000
20000
25000
3334
3336
3338
3340
3342
time (sec)

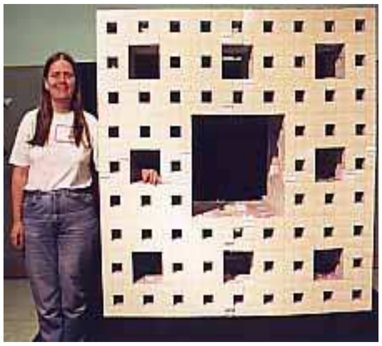

0.1s aggregation (expo)
throughput (bytes)
0
5000
10000
15000
20000
25000
3334
3336
3338
3340
3342
time (sec)


0.01s aggregation (alpha 1.05)
throughput (bytes)
0
500
1000
1500
2000
2500
3000
3500
3333.2
3333.4
3333.6
3333.8
3334
time (sec)


0.01s aggregation (expo)
throughput (bytes)
0
500
1000
1500
2000
2500
3000
3500
3333.2
3333.4
3333.6
3333.8
3334
time (sec)

Some fractal objects:

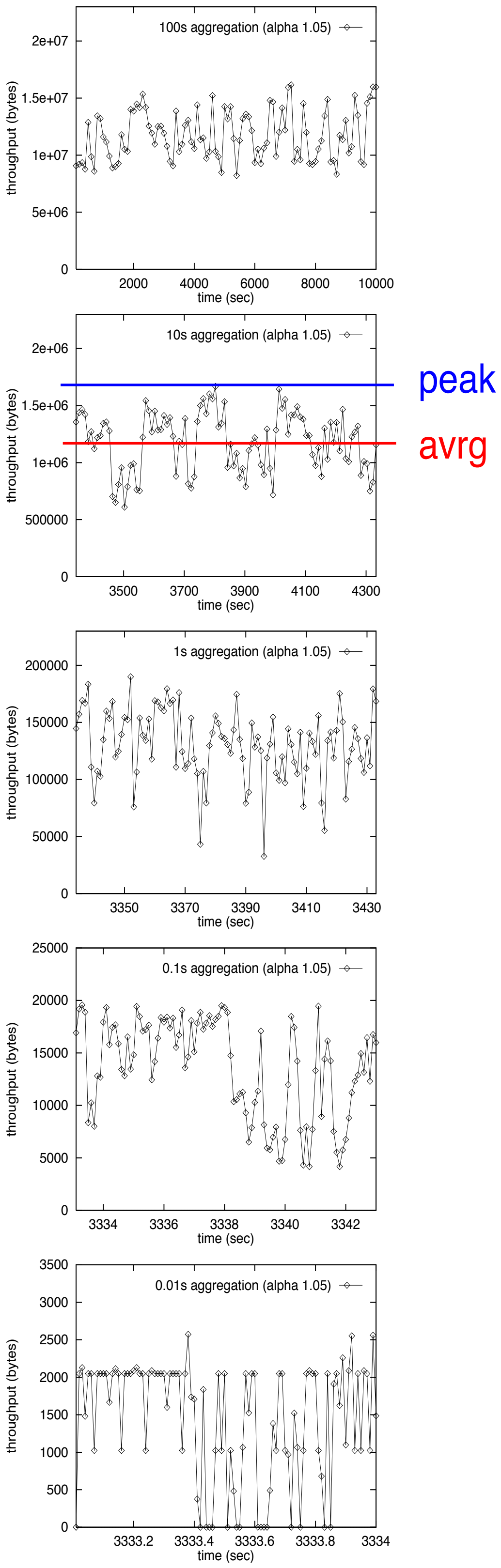
Menger sponge (picture from `www.ics.uci.edu/∼eppstein`):



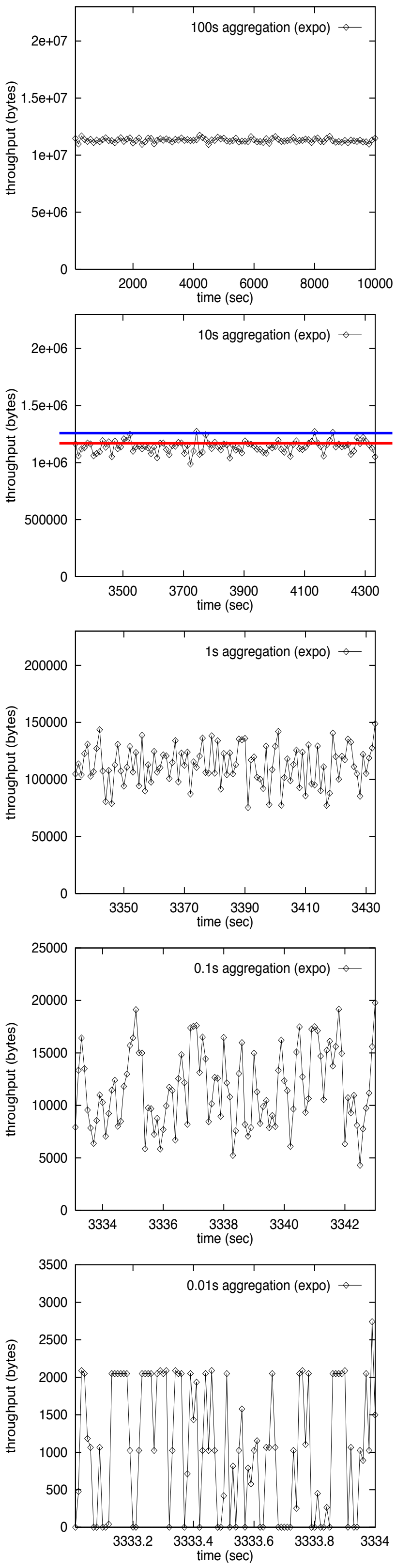
Fractal fern:

⟶ are fractal objects random?

Internet: self-similar



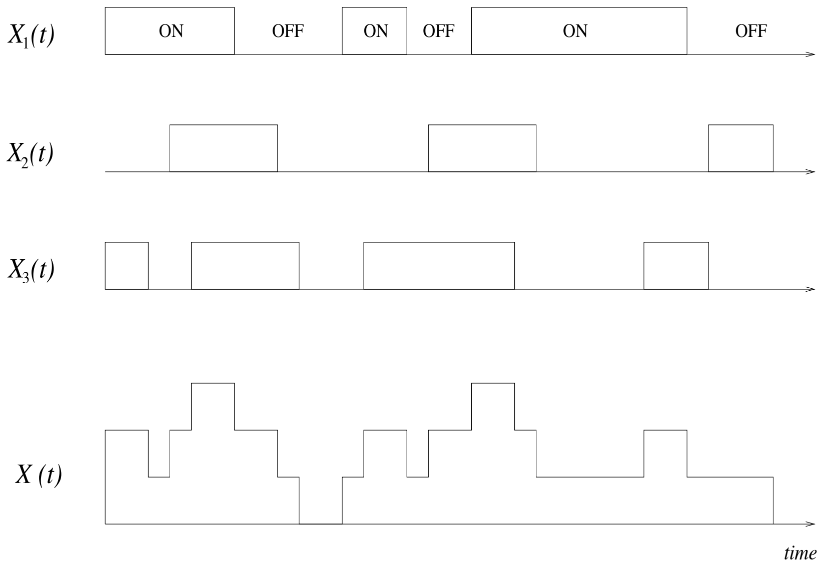
Telephony: Poisson-like



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



⟶ on-period: TCP file transfer

⟶ on-period length: file transfer completion time

⟶ ignore internal details within on-period: sawtooth

⟶ on-period could be VoIP session: CBR

⟶ not exactly: a user talks only 40% of the time

⟶ approximate view: ok

We know session arrivals are (approximately) Poisson; what about session lifetimes?

Important fact: TCP session lifetimes are heavy-tailed

$\longrightarrow$ $\Pr\{Z > x\} \approx x^{-\alpha}$

$\longrightarrow$ as opposed to: $\Pr\{Z > x\} \approx e^{-bx}$

$\longrightarrow$ exponent: $1 < \alpha < 2$ (closer to 1)

$\longrightarrow$ note: different from Internet connectivity power-law

$\longrightarrow$ much more likely session will last a long time

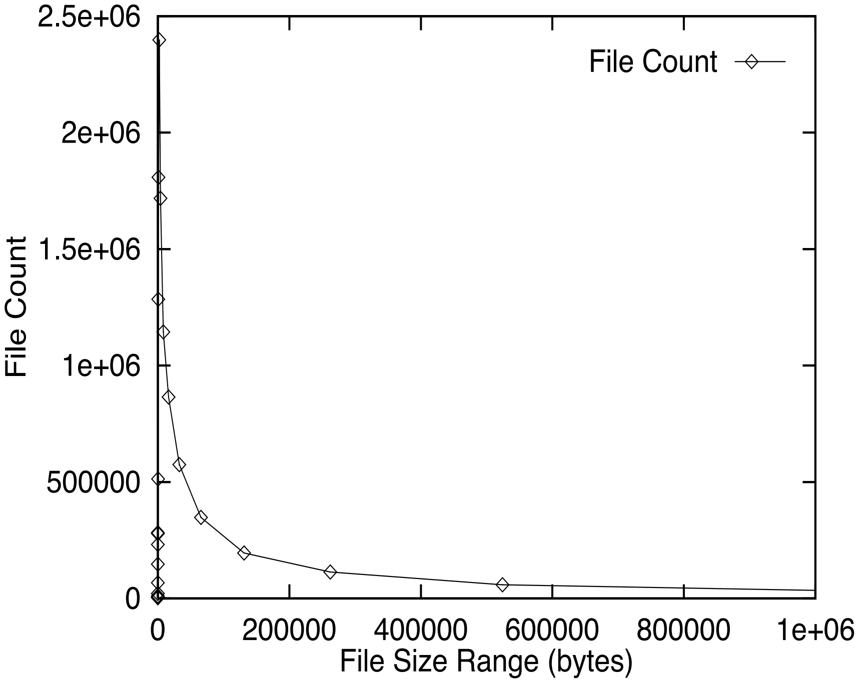
$\longrightarrow$ has finite mean but infinite variance

Why would TCP session lifetimes be heavy-tailed?

$\longrightarrow$ TCP traffic makes up bulk of Internet traffic

Tale of elephants and mice:

⟶ UNIX and WWW file systems



⟶ many small ones (mice)

⟶ a few very large ones (elephants)

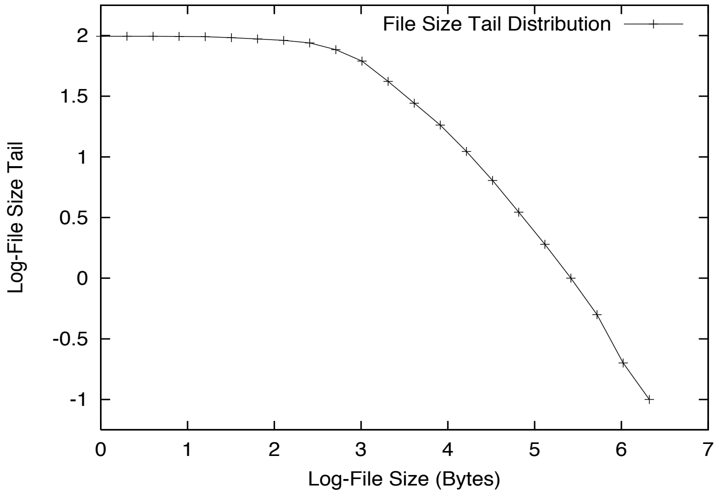
⟶ 10% consumes 90% of bandwidth

How to check if files sizes are heavy-tailed?

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

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

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



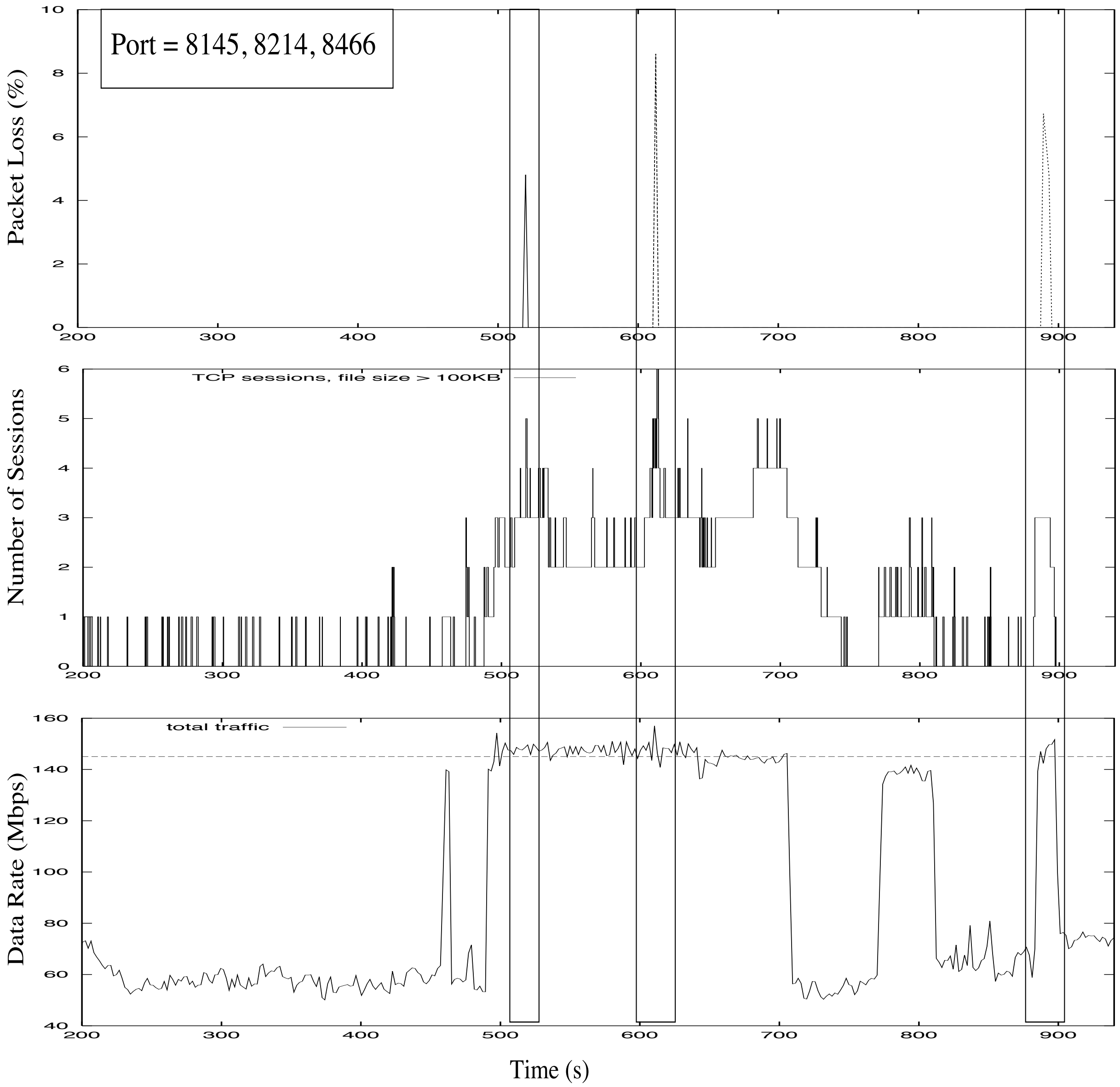
$\longrightarrow$ holds true for large x

$\longrightarrow$ what's the slope α?

$\longrightarrow$ we don't care about details of small sizes (why?)

Elephants in action:

⟶ 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:

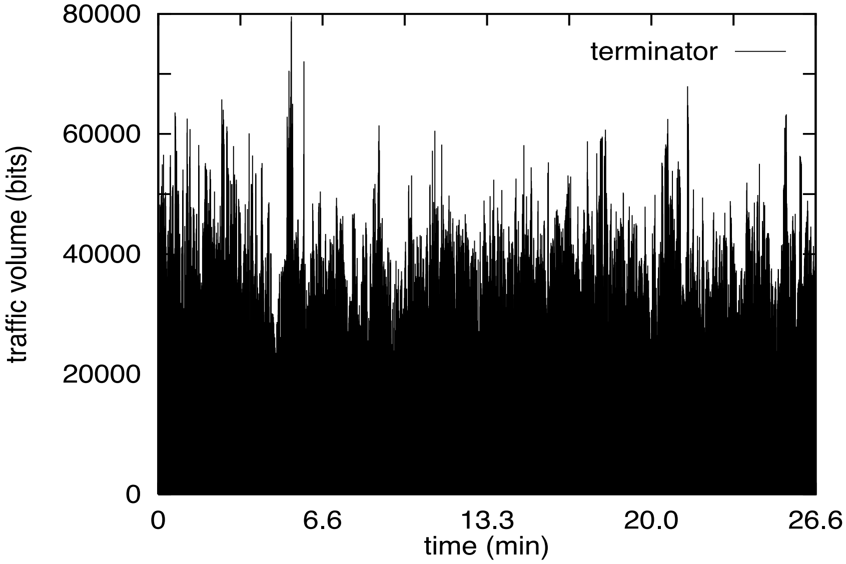
⟶ norm: skewed distribution of sizes

⟶ power-law

Bandwidth allocation rule of thumb: 50% overprovision

Real-time VBR source profile

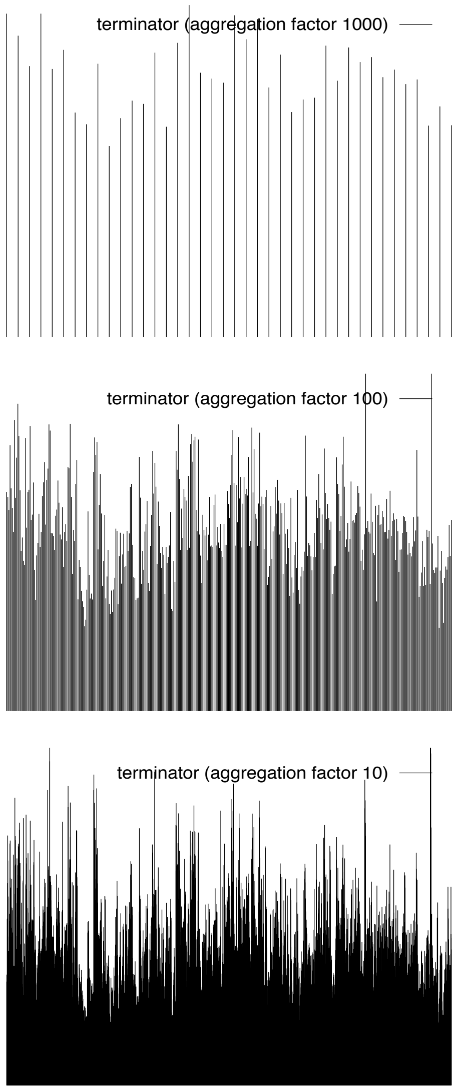
Consider traffic profile of compressed video (e.g., MPEG)



⟶ periodic real-time application

⟶ period $T = 1/f$ (f: frame rate)

Burstiness structure:



⟶ burstiness persists across multiple time scales

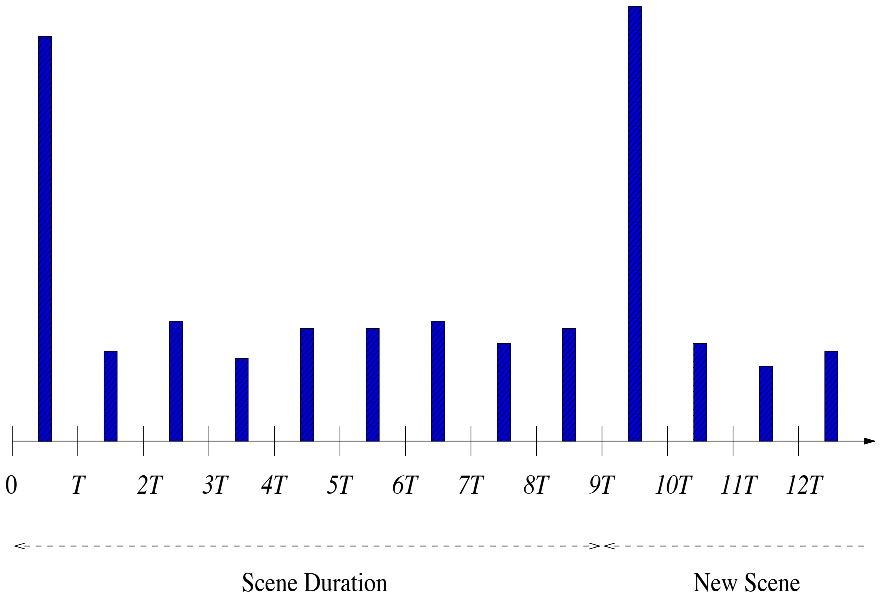
Possible causes:

- Heavy-tailed scene durations

  ⟶ facilitates inter-frame compression

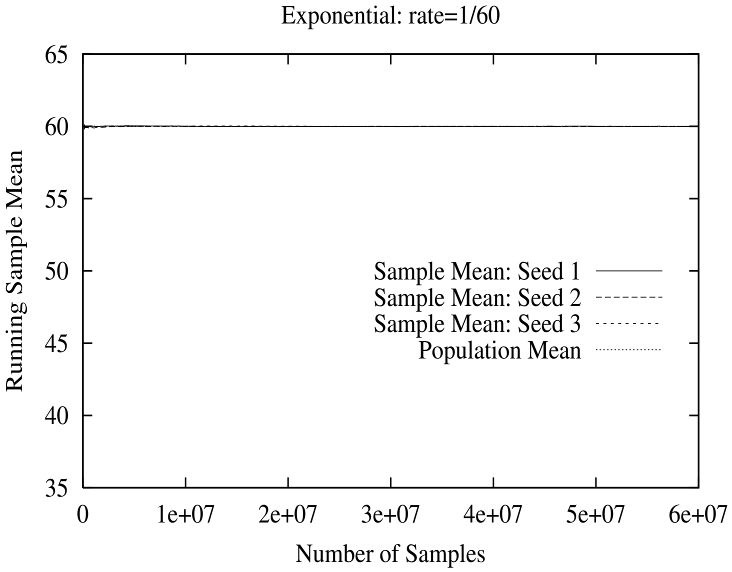
- Repeating GOP pattern

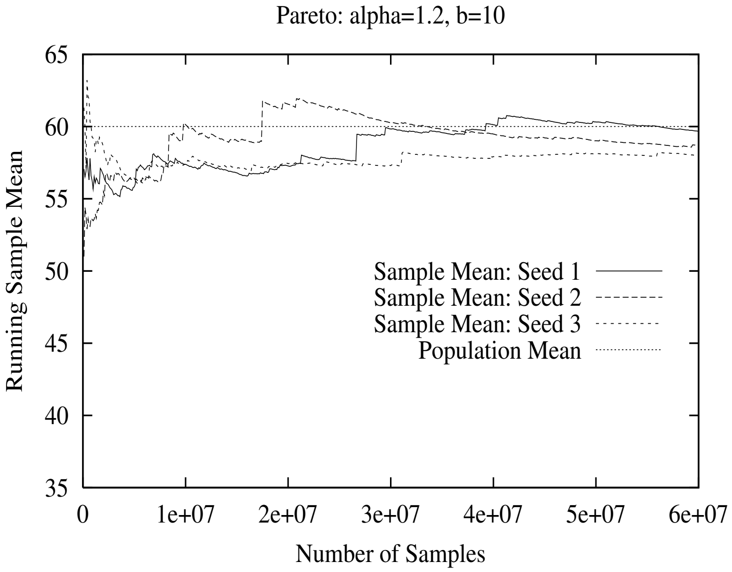
  ⟶ e.g., I B B P B B



Note on sampling and convergence.

Sample mean convergenge rate: exponential vs. Pareto





extremely slow convergence: concern when generating artificial workload