Resource Provisioning and Network Traffic

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 and make money (or lose less money)?

Resource provisioning: two main resources

- Bandwidth allocation

  → primary

- Buffer allocation

  ⟶ resource dimensioning: long-term

  ⟶ also called network planning: months, years

  ⟶ on-demand resource allocation: short-term

  ⟶ i.e., second, minutes, hours

Turns out:

⟶ same principles apply to both

⟶ take ISP's viewpoint

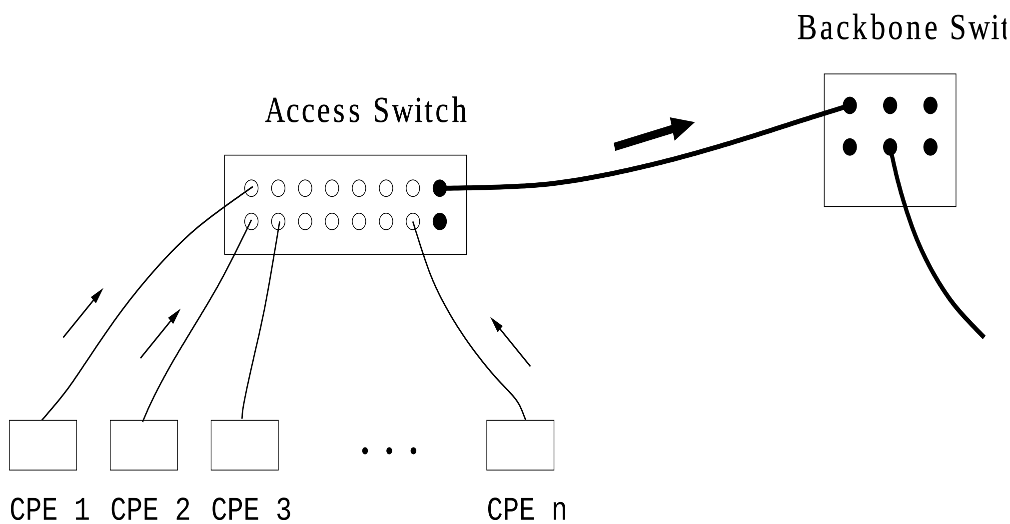
⟶ granularity: user-, session-, and packet-level

User- and Session-Level Resource Provisioning

Basic set-up:

→ aggregate demand at access switch

→ n users or CPE (customer premises equipment)



Set-up applies to:

- Telephone switch: TDM slot per session/user
- Dial-up modem pool: e.g., AOL Internet access
- Broadband access service: e.g., IP address pool

Basic building block: access switch

$\longrightarrow$ function: aggregation

$\longrightarrow$ performance benefit?

$\longrightarrow$ old banking trick: keep fraction of total deposit

$\longrightarrow$ observation: not all customers withdraw at once (?)

Networking: not all customers access network at once

$\longrightarrow$ affords efficiency & economy

- can keep fewer T1 lines
- can keep smaller modem pool
- can keep fewer IP addresses
- can keep less bandwidth

Note: a calculated risk

$\longrightarrow$ sometimes very many users connect at once

$\longrightarrow$ access denied: blocking

In what other major sector is "old banking trick" employed?

What makes old banking trick possible?

$\longrightarrow$ one of the few "laws of engineering"

Law of large numbers (LLN): the sum of many independent random variables concentrates around the mean

$\longrightarrow$ i.e., very few outliers

$\longrightarrow$ also, typically, mean $\ll$ maximum

Ex.: Suppose there are n users subscribing to Verizon in West Lafayette.

$\longrightarrow$ how many users will make a call at time t?

Assuming:

- $X_i(t) = 0$ if no call by user i at t, 1 if call
- $\Pr\{X_i(t) = 1\} = p$
- users make calling decisions independent of each other

  $\rightarrow$ i.e., $X_1(t), X_2(t), \ldots, X_n(t)$ are i.i.d.

  $\rightarrow$ note: same as coin tossing
- total calls at time t

  $\rightarrow$ $S_n(t) = X_1(t) + X_2(t) + \cdots + X_n(t)$
- average number of calls

  $\rightarrow$ $E[S_n(t)] = E[X_1(t)] + \cdots + E[X_n(t)] = np$

  $\rightarrow$ hence, $E[S_n(t)/n] = p$

  $\rightarrow$ independence needed?
- LLN: $\Pr\{|\frac{S_n(t)}{n} - p| > \varepsilon\} \rightarrow 0$ as $n \rightarrow \infty$ for any $\varepsilon > 0$

  $\rightarrow$ weak LLN

  $\rightarrow$ strong LLN?

Thus, for sufficiently large n deviation from the mean is rare.

$\longrightarrow$ Verizon can expect np calls at time t

$\longrightarrow$ with large n, very close to np calls

$\longrightarrow$ but, how large is "large"?

$\longrightarrow$ does WL have sufficiently many customers?

To be useful for engineering, we need to know more

$\longrightarrow$ rate of convergence

Large deviation bound:

$\longrightarrow$ $\Pr\{|\frac{S_n(t)}{n} - p| > \varepsilon\} < e^{-an}$

$\longrightarrow$ constant a depends on ε

$\longrightarrow$ exponential decrease in n

$\longrightarrow$ also, holds for all n

$\longrightarrow$ engineering: blocking probability

Large deviation bound gives simple prescription for resource provisioning:

- measure p (historical data); ISP knows n
- determine acceptable blocking probability δ

  $\rightarrow$ e.g., $\delta = 0.00001$

  $\rightarrow$ i.e., one in 10000 calls gets blocked

- find ε such that

$$\begin{aligned} \Pr\{|\frac{S_n(t)}{n} - p| > \varepsilon\} &= \Pr\{|S_n(t) - np| > n\varepsilon\} \\ &< e^{-an} = \delta \end{aligned}$$

  $\rightarrow$ note: ε determines excess capacity allocated

  $\rightarrow$ recall: a depends on ε (called rate function)

  $\rightarrow$ one of the main tools used by ISPs/telcos

From ISP's perspective, is this enough for making resource provisioning decision?

$\longrightarrow$ what crucial element may be missing?

Connection lifetime or duration

$\longrightarrow$ also called call holding time

$\longrightarrow$ for how long a resource (e.g., modem) is busy

$\longrightarrow$ if fixed, then previous formula holds

$\longrightarrow$ user session property

$\longrightarrow$ in general: connection lifetime is variable

$\longrightarrow$ e.g., average telephone call: 7 minutes

Let L denote connection lifetime (assuming i.i.d. across all users)

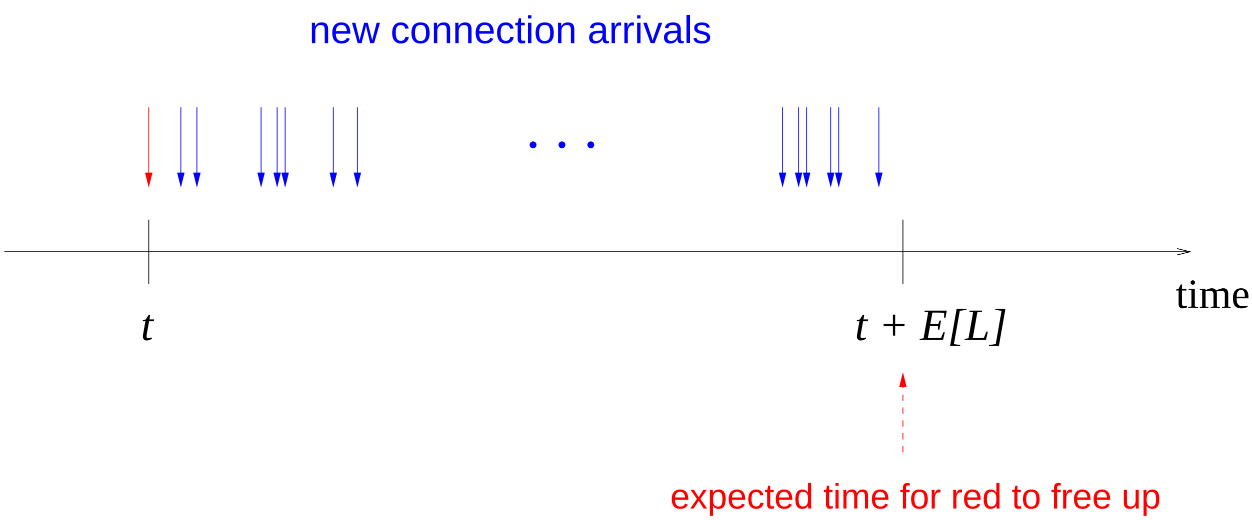
$\longrightarrow$ by measurement, ISP knows its distribution

$\longrightarrow$ consider average lifetime $E[L]$

$\longrightarrow$ consider two time instances t and $t + E[L]$

$\longrightarrow$ what to do?

View system in terms of time granularity $E[L]$:



- use large deviation formula to estimate connection arrivals during time window $[t, t + E[L])$

  $\longrightarrow$ excess capacity $n\varepsilon$ above and beyond mean np

- use distribution of L to estimate $\Pr\{L > E[L]\}$

  $\longrightarrow$ may refine ε to ε' $(\varepsilon < \varepsilon')$

  $\longrightarrow$ for $E[L]$ not-too-small may not be needed (why?)

Remarks:

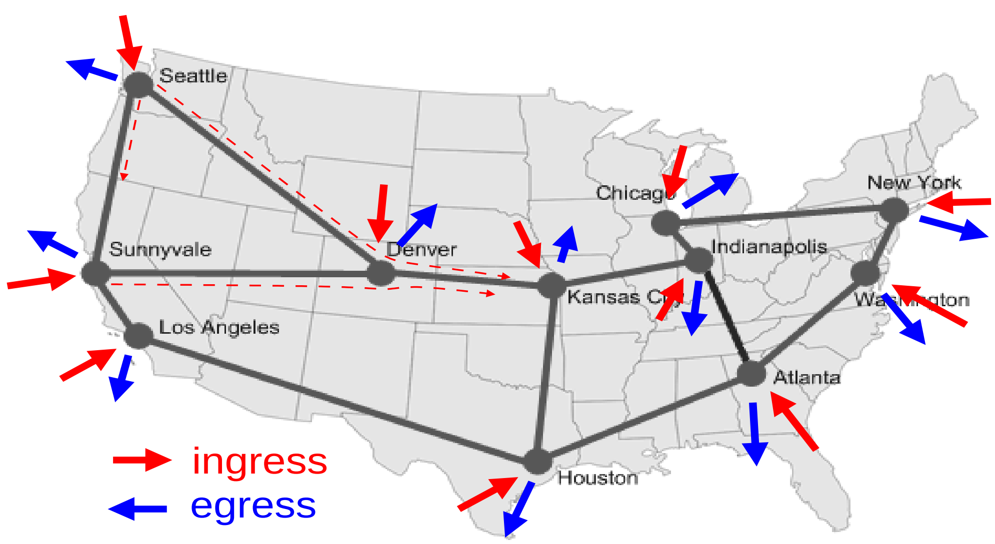
- LLN: principal engineering tool used by large transit providers and large access providers

  → "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



- sometimes can apply central limit theorem (CLT): aggregate has Gaussian (normal) distribution

  $\longrightarrow$ in practice: not very useful

  $\longrightarrow$ e.g., tail of Gaussian: not very accurate

  $\longrightarrow$ deviation estimate valid only for moderate ε

  $\longrightarrow$ may not even look Gaussian!

  $\longrightarrow$ needs very large n

  $\longrightarrow$ large deviation bound: holds for all n

- aggregation over time window $[t, t + E[L])$

  $\longrightarrow$ a single user can have 2 or more sessions

  $\longrightarrow$ may violate independence assumption (across users)

  $\longrightarrow$ independence over time: separate matter

- we assumed discrete number of resources

  → e.g., 10000 modems, 50000 IP addresses, 1000 T1 lines, etc.

  → valid viewpoint at user/session granularity

  → also applies to packet granularity

  → as long as independence over time holds

How does session arrival for a single user over time look like?

⟶ aggregation over time

⟶ resource provisioning: buffering

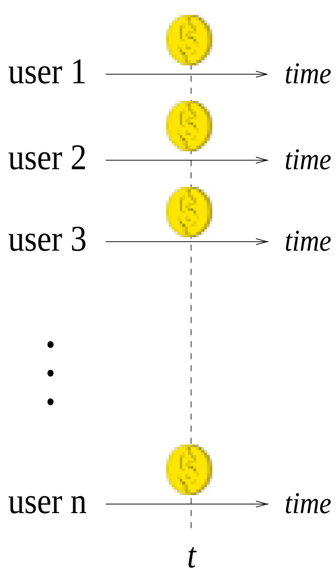
⟶ vs. aggregation over users (bandwidth)

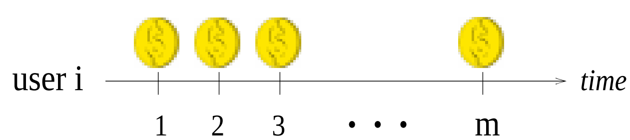
Session arrivals over time:

- apply coin tossing idea over time
- before: one coin per user
- now: one user has multiple coins

  $\longrightarrow$ coins are assumed to be i.i.d. with probability p

  $\longrightarrow$ apply LLN over time!





LLN over users

LLN over time

Over discrete time window $[1, m]$, same bounds apply; for user i:

$\longrightarrow$ $\mathcal{S}_i(m) = X_i(1) + X_i(2) + \cdots + X_i(m)$

$\longrightarrow$ $\Pr\{|\frac{\mathcal{S}_i(m)}{m} - p| > \varepsilon\} < e^{-am}$

Thus: if we have $mp + m\varepsilon$ resources (e.g., buffer), then can buffer user i's service requests (could be even packets) over time $[1, m]$ without "loss"

$\longrightarrow$ loss probability $< e^{-am}$

$\longrightarrow$ before: blocking probability

$\longrightarrow$ in practice: m can't be too high

$\longrightarrow$ buffering $\Rightarrow$ delay penalty

$\longrightarrow$ some applications require quick response time

One refinement: what does the time spacing between successive arrivals look like?

$\longrightarrow$ prob. session will arrive after k steps: $(1-p)^k p$

$\longrightarrow$ called geometric distribution (where did we see it?)

$\longrightarrow$ most important: exponentially decreasing in k

Corresponding distribution in continuous time:

$\longrightarrow$ be^{-bt} (t in place of k)

$\longrightarrow$ exponential distribution

$\longrightarrow$ essentially equivalent to geometric distribution

$\longrightarrow$ important property: memoryless