SC/CSE 3213 Winter 2014

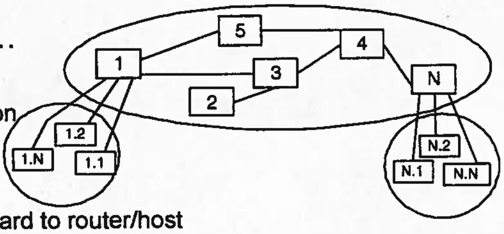
L2: Basic Networking Principles

Sebastian Magierowski
York University

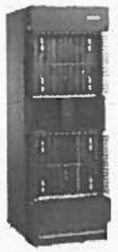
Outline

- Sharing
  - To reduce the cost of networking +1 billion devices you must share communication links
- Metrics
  - How do you measure and quantify the performance of a network? (Use some physics and use some math!)
- Scalability
  - The network is huge, organize it in a hierarchy such that changes only have local effects

General Structure

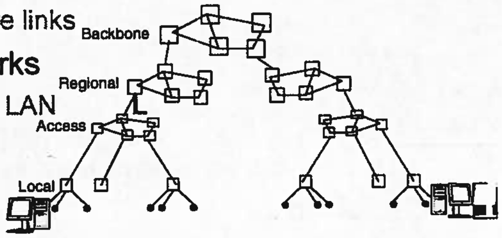


- Distributed collections of…
  - hosts
    - sources & sinks of information
    - at the edges
  - routers
    - receive information and forward to router/host
  - links
    - copper/cable/optical/wireless
- For this to work you need to think about arrangement more deeply



Sharing

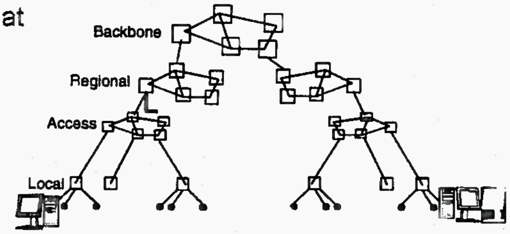
- Given the large amount of hosts…
  - You are going to have to share links
- Build a hierarchy of networks
  - Attach nearby computers in a LAN
  - Merge LANs with an access network
  - Then a regional network
  - Then a cross-country/globe backbone network
- Many computers share a link L



[©Morgan&Claypool, Walrand&Parekh]

Multiplexing Gain

- What makes sharing possible?
  - Computers don't all transmit at the same time
- If you send data into the network for 20 mins. over 8 hours
  - Then you are active only 20/(60•8) = 1/24 the time
  - For an L with transmission rate, R [bps], shared by A computers
    - We have to share A/24 computers
    - Thus each computer can in principle transmit at...
      - R/(A/24) = 24•R/A $= 6 \cdot \frac{R}{A}$
    - "Multiplexing Gain"



we might initially just expect $\frac{R}{A}$, until we take into account the avg. holding time of a unit

[©Morgan&Claypool, Walrand&Parekh]

Metrics

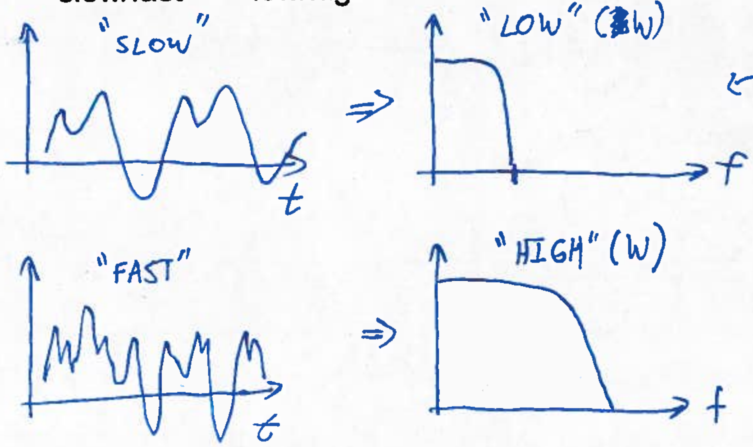
- To clarify network characteristics we must precisely define some basic measures of performance
  - Rate
  - Bandwidth
  - Capacity
  - Throughput
  - Delay
  - Delay Jitter
  - Queues

Rate

- How many bits a link can handle per unit time
  - DSL
    - downlink rate (768 kbps)
    - uplink rate (256 kbps)
- Very loosely a "broadband" link > 100 kbps
  - A non-universal value
- We'll (usually) use *R* to denote link rate

Bandwidth

- Spectral extent of a signal
- "slow/fast" -> "low/high"



← how do we obtain spectra?

↙ (for the purpose of quantification)

Fourier Series

- Finding the spectrum of a periodic signal

$$x(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(2\pi f_0 n t) + \sum_{n=1}^{\infty} b_n \sin(2\pi f_0 n t)$$

$$f_0 = \frac{1}{T} \qquad a_n = \frac{2}{T}\int_{-T/2}^{T/2} x(t)\cos(2\pi f_0 n t)\,dt$$

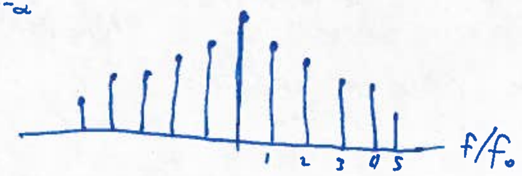
$$b_n = \frac{2}{T}\int_{-T/2}^{T/2} x(t)\sin(2\pi f_0 n t)\,dt$$

$$\sin\theta = \frac{e^{j\theta} - e^{-j\theta}}{2}$$

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{i2\pi f_0 n t} \qquad |c_n| = \sqrt{a_n^2 + b_n^2} \qquad \theta_n = -\operatorname{atan}\left(\frac{b_n}{a_n}\right) \qquad c_n = \frac{1}{T}\int_{-T/2}^{T/2} x(t)e^{-j2\pi f_0 n t}\,dt$$

- projection on y-axis
- but θ could be spinning in any dir.

to genera the.t! (this fn. exposes that)

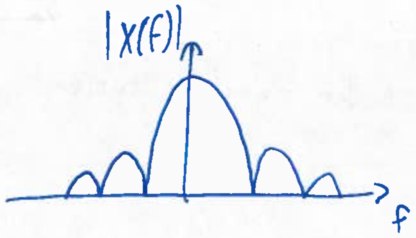


Fourier Transform

- Spectrum of non-periodic signals

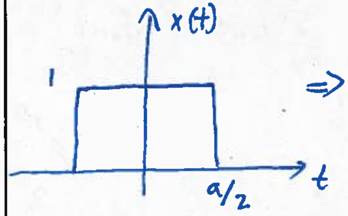
$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi f t}\,dt$$

$$X(f) = \int_{-\infty}^{+\infty} x(t)\,e^{-j2\pi f t}\,dt$$

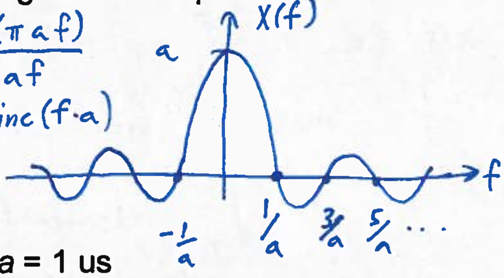


Spectrum of a Pulse

- A pulse in the time domain give a sinc spectrum



$$\Rightarrow \quad a\cdot\frac{\sin(\pi a f)}{\pi a f} = a\cdot \text{sinc}(f\cdot a)$$



- Imagine a bit stream with *a* = 1 us

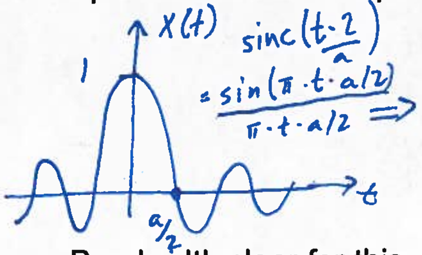
$\therefore$ $R = 1\,\text{Mbps}$ $W \approx 1\,\text{MHz}$ ← or something like it

- As *R* increases *W* increases

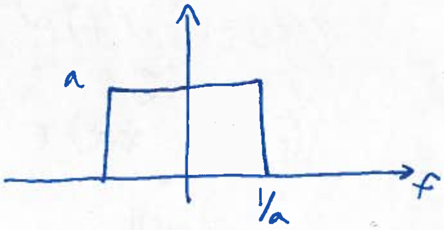
$R \propto W$ } ← at least this is true

Spectrum of a Sinc

- Spectrum of different pulse



$$\text{sinc}\left(\frac{t\cdot 2}{a}\right) = \frac{\sin(\pi\cdot t\cdot a/2)}{\pi\cdot t\cdot a/2} \Rightarrow$$

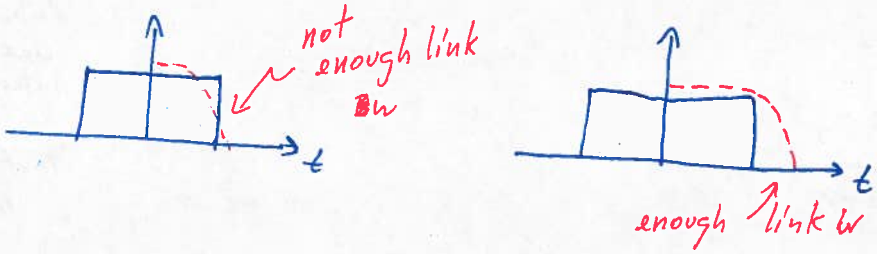


- Bandwidth clear for this

→ why is this a better signalling system

Link Bandwidth

- Links with sufficient bandwidth, *W*, pass spectral components of a signal without excessive attenuation



Link Capacity

- What is the biggest R I can stuff through a link…
- …and completely recover ALL the data I send through it
  - i.e. zero error communication
- $R_{max}|_{no\ error} = C = W \cdot \log_2(1 + SNR)$

channel BW

how much stronger signal is over noise

$SNR = \frac{A^2}{\sigma^2}$

60dB $10^3 = 2^{10}$

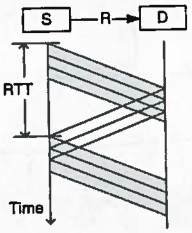
$W = 1 MHz$ $SNR = 10^6 \approx 2^{20}$

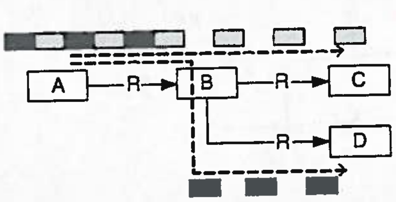
$C = 10^6 \cdot \log_2(2^{20}) = 20 Mbps$ ← thru only 1MHz (if you have 60dB SNR)

Throughput

- Rate of data transfer but…
  - …accounts for overhead of communication and network details

wait for ack & time delay





data was interleaved with other traffic (-OR- shared links with other traffic

- 3 MB file takes 2 minutes to download
  - *T* is? $\frac{8\cdot 3\cdot 2^{20}}{2\text{ min}} \approx \frac{24\times 10^{6}}{120\text{ sec}} = 2\times 10^{5}$ bps = 200 kbps

---

Delay

- Time elapsed between two points of interest
- Consists of
  - transmission time : amt of time to get packet into link L/R
  - propagation time : time for physical signal to propogate $d/v = (d/c)\sqrt{\epsilon_r}$
  - queuing time : waiting time at node before transmission
  - processing time : time to perform required operation on a packet

Delay Jitter

- Delay varies
- Delay jitter, J = max delay – min delay
- Important for streaming or real-time apps
- Buffer for at least J

Queue Delay Estimation

- λ: packets arriving per second
- μ: packets processed per second
- ρ: utilization = λ/μ assume < 1 (things not overloading system)
- Rate at which a queue is emptied? : $\mu - \lambda$ $\underset{\text{in rate}}{\lambda} \to \square \to \mu$ (out rate)
- Avg. time to empty a queue? $T = \frac{1}{\mu-\lambda} = \frac{1}{\mu}\left(\frac{1}{1-\rho}\right) \approx \frac{1}{\mu}$ iff ρ is low
- Avg. wait time to get ~~out of~~ to front of a queue? → $T - \frac{1}{\mu}$: I arrive, must wait for T, less the time to process me
- Avg. number of packets in queue? → $L = \lambda \cdot T = \frac{\lambda}{\mu-\lambda}$
- R = 10 Mbps, P = 1 KB, λ = 1000
  - avg. delay, T? (transmission and queuing components?)
  - avg. queue length, L?
  - delay jitter ~ 3x average delay

1/1250

3.2 ms queuing time + 0.8 ms transmission time

$\lambda \doteq 1000$ packets per sec.

$P = 1$ KB, R = link rate

$\mu = \frac{10^7}{8 \times 10^3} = \frac{R}{P} = 1250$ ↑ packets per sec.

1 KB = 2^{10}

$T = \frac{1}{\mu - \lambda}$

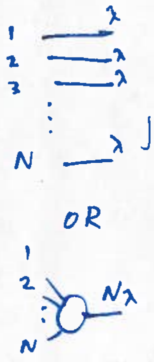
$= \frac{1}{1250-1000} = 4$ ms

$L = \frac{1000}{1250-1000} = 4$

jitter 12ms

Congestion

- If λ goes up to 1150 (from 1000)
  - T now? $\frac{1}{1250-1150} = 10$ ms
  - L now? $\frac{1150}{1250-1150} = 11.5$
  - J now? $\longrightarrow$ 30 ms
- Average packet delay and jitter increase quickly as λ approaches μ



Link Sharing

- What if we have N computers share a link of speed $N\lambda$
  - $T_N = T/N = \frac{1}{N\mu - N\lambda} = \frac{T}{N}$
    - Our average delay (and jitter drops)
  - $L_N = L$ $L_N = N \cdot \lambda \cdot \frac{T}{N} = \lambda \cdot T$
    - Same number of average packets queued
- Sharing a faster link (instead of using a slower dedicated link) ↖ through switches
  - reduces delay
  - transmission time is only 1/μN

Little's Result

another Interesting result from queuing theory

- $N = \lambda \cdot T$
- N: avg. number of packets in the system (i.e. network)
- T: avg. time spent in the system
- λ: avg. packet arrival per second in the system
- How many packets in Internet?
  - 1B users, 100MB/day, 0.1 seconds in Internet

$$N = \overbrace{10^9 \cdot \frac{8\times10^7}{24\times60\times60}}^{\lambda} \times \overset{T}{0.1} = 920\text{billion bits}$$

↑ in fiber & routers

$$\frac{920\times10^6}{1500\times8} = 76,666 \text{ pack}$$

- How many bits stored in a fiber?
  - 2.4 Gbps, 20% capacity, 100 km long

$$N = 2.4\times10^9 \times 0.2 \cdot \frac{100\times10^3}{3\times10^8/(2/3)} = \underline{240\text{ kb}}$$

$$\approx \underline{30\text{KB}}$$

- How many bits in a router?
  - 16 1-Gbps ports, 10% capacity, each bit in router for 5 ms

$$16\times10^9 \cdot 0.1 \cdot 5\times10^{-3} = 8\text{Mb} \approx 1\text{MB}$$

$$8\text{Mb} = 8\times10^6\text{b}$$

$$\frac{8\times10^6}{2^{20}}$$

$$= 7.6\text{MB}$$

Fairness

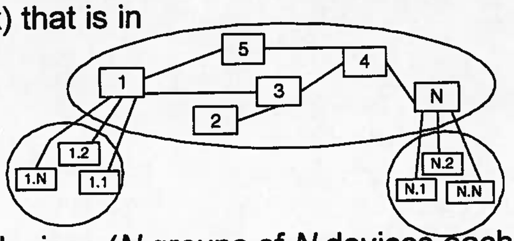
- What rates should be allocated to different flows
- For example…
  - 1 Mbps link
  - Two flows: 1.6 Mbps and 0.4 Mbps
  - How do you partition the resource?
    - In proportion to desired throughput?
      - 0.8 Mbps/0.2 Mbps
      - Fair to 0.4 Mbps? Can't get through even though it asks for less?
    - To the lowest request?
      - 0.6 Mbps/0.4 Mbps
      - "max-min allocation"
- Achieving fairness complex issue in real networks

Scalability

- For a network to grow large in size modifications must have only limited effect
- Assuming 50% of the 1B Internet computers added in the last 10 years
  - ~100,000 computers added every day!
- ARPANET routers stored the address of each computer on the network
- In light of scaling rates this was unsustainable
- A number of other basic network functions need to account for scaling…

Location-Based Addressing

- Define address of a computer in terms of the group (network) that is in



- $M = N^2$ devices (N groups of N devices each)
- How many devices does each router have to be aware of? N
- What's it look like as you scale? ?

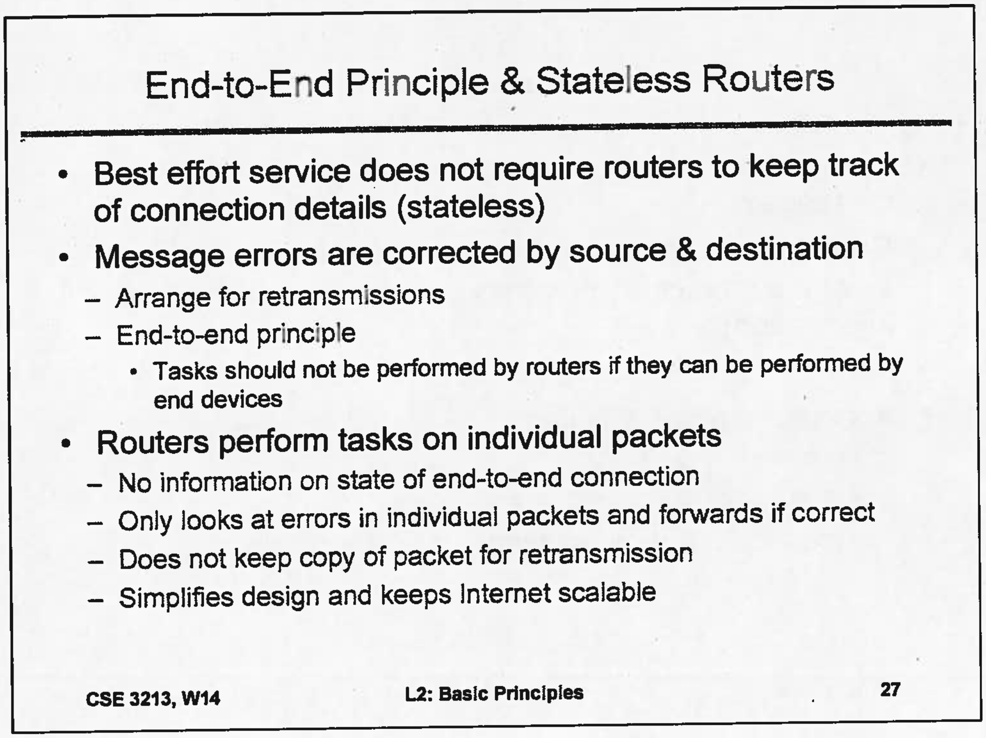
rises linearly with networks?

Two-Level Routing

- How do you actually learn the best way to route a message?
- Routers have to exchange messages with each other to learn the metrics of their links (and keep updating this information)
- For *N* routers, *N* messages must be sent
- If routers are partition into…
  - √*N* domains and
  - √*N* sub-domains (inside domains)
- …much less routing messages are required

Best Effort Service

- Internet does not guarantee any precise property (e.g. delay, throughput, etc.)
- Provides best effort service
  - Try to deliver packets as well as possible
- As technology improves more and more demanding applications become supported
- Applications adapt to changing quality of services



End-to-End Principle & Stateless Routers

- Best effort service does not require routers to keep track of connection details (stateless)
- Message errors are corrected by source & destination
  - Arrange for retransmissions
  - End-to-end principle
    - Tasks should not be performed by routers if they can be performed by end devices
- Routers perform tasks on individual packets
  - No information on state of end-to-end connection
  - Only looks at errors in individual packets and forwards if correct
  - Does not keep copy of packet for retransmission
  - Simplifies design and keeps Internet scalable