EECS 122, Lecture 24

Today's Topics:

Intro to the Telephone Network

Kevin Fall, kfall@cs.berkeley.edu

Problem Set 4

- Read: text 4.3
- Problems (chp. 4):
  - 2, 8, 9, 10, 11
  - due April 27

Telephone Net Concepts

- > 1 billion telephones, > 200 million calls a day just on one carrier (AT&T)
- Circuit switching
  - two-party, small end-to-end delay and jitter, reserved resources once call admitted
  - full duplex connections
- Intelligence placed within the network, not in end-systems (telephone sets) [contrast with Internet]

Recent History

- The important 1980's:
  - deployment of large digital switches
  - adaptation of computer-controlled switches to provide switching multiple data types
  - deployment of fiber optic transmission media
- The breakup (Jan 1, 1984)
  - AT&T -> 7 RBOCs plus AT&T and others
  - long-distance carriers (IXCs) open
  - local area (LATA) carriers (LECs) regulated

More Recent History

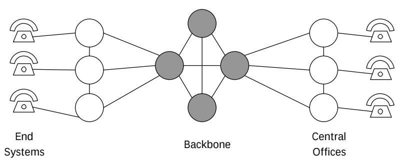
- The 1996 Telecom Act
  - removes numerous restrictions on LECs
  - LECs can provide long-distance and IXCs can provide local calling, if certain restrictions are met [like equal access to IXC, space sharing]
  - players:
    - ILECs (incumbent LECs; own COs and loops)
    - CLECs (competitive LECs)
    - "your competitor is your landlord"

Telephone Net Structure

- End systems (phones, faxes, etc)
- Central offices (COs)
  - local aggregation points for phone lines
  - wire pair (*local loop*) to each telephone
  - most are analog, provide A/D conversion
- Exchanges
  - switches connecting end systems
  - connect to back-bone (core) switches

Telephone Net Structure

- Backbone switches
  - (near) fully-connected set of switches
  - < 500 (vs 10000 exchanges)
- Simplified architecture picture:



Hierarchical Addressing Example

- To route: 512-224-3213
  - must pass through backbone handling 512 area code and 224 exchange
  - may also pass through others
- Special area codes (700,800,888,900)
  - number used as index into table
  - table contains actual telephone number
  - table may be modified by time of day, etc

The Details

- End systems
- Transmission
- Switching
- Signaling

End Systems

- Traditionally a telephone (POTS):
  - sound-to-electric transducer
  - electric-to-sound transducer
  - dialer, ringer, switch hook
- Echo issues:
  - with only 2 wires, side-tone (hearing yourself talk) must be limited but present
  - received sound may be echoed back (ok for small local delay, actively cancelled with circuitry at backbone switches [costly])

Newer End-Systems

- Digital local loops:
  - ISDN (BRI)
  - xDSL
- ISDN (BRI):
  - 2x64kb/s circuit channels
  - 1x16kb/s packet channel
- DSL:
  - up to 1Mb/s, possibly asymmetric, FDM with respect to POTS service

Transmission

- familiar characteristics: bandwidth, delay, attenuation
- attenuation addressed with regenerators:
  - with optical fiber, every 5000km
  - non-electric optical amplification possible
- digital multiplexing:
  - 8000 samples/sec at 256 levels=64kb/s
  - mu-law encoding in US, Japan

TDM Operation

- TDM muxing of digital voice streams
- Common service is T1(line), DS1(std):
  - 1.544Mb/s, 8000f/s at 193 bits/frame
  - 192/8 = 24 bytes(TDM'd calls)/frame + 1 bit
- Digital Signaling (DS) Hierarchy:
  - DS0 (64kb/s), DS1(1.544Mb/s), DS3(44.736Mb/s)
  - not exact multiples due to framing overheads

Plesiochronous Operation

- Almost synchronous: components generate data at nominally the same bit rate, but are allowed to vary by a bounded amount [used for DS2,3]
- Requires a good, but not perfect, clock
- Muxing uses bit interleaving; differences in clock rates are accommodated by *justification* or *bit stuffing*

Justification or (Pulse/Bit) Stuffing

- output channel rate higher than sum of input rates
- additional bits inserted to pad input rate
- allocations of input rates at output are at the minimum rate (no underflow), so slightly-fast inputs use up stuff bits
- need to read a whole frame to properly extract the individual inputs

Problems with Plesiochrony

- Each part of the world has its own (not directly inter-operable) format
- Justification spreads data from tributaries all across frame, making it difficult to add/drop data from a particular stream
- Hard to build switches that switch bundles of voice calls instead of individual ones [all must demux down to DS0 to find individual calls]

Synchronous Operation (SONET)

- If network was completely synchronous, no need for justification (in theory…)
- SONET defines a multiplexing hierarchy with exact multiples of data rates:
  - OC-3(155.52Mb/s), OC-12(622.08Mb/s), OC-24(1.24416Gb/s), OC-48(2.48832Gb/s)
- Assumes synchronized clock
- Uses byte interleaving across lesser-speed signals (tributaries)

Benefits of SONET

- Creates a standard muxing format for any number of 51.84Mb/s signals
- Creates an optical standard for interconnecting multiple-vendor equip.
- Creates standard operation, administration, and maintenance (OAM)
- Defines synchronous muxing format for lower-speed (DS1, 2, etc) signals

Complications of SONET

- SONET can handle aggregations of lower-speed plesiochronous signals, so still need a form of justification
- SONET has overhead:
  - about 27/810 bytes (about 3%)
- SONET is complicated:
  - recall 2-D frame format & pointer offset
- SONET requires good clock (+/-1 in 10^11)

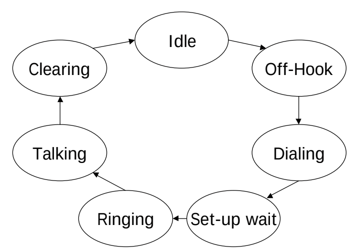
Switching

- Telephone switch is actually two parts:
  - switching hardware (moves data)
  - switch controller (handles set up/clearing)
  - (we covered most of the issues already)
- Controllers known as *overlay network*
- Messaging between controllers form signaling network, with its own protocol

Simple Signaling

- Tones or pulses from end system interpreted at switch controller
- If intra-exchange call, rings bell on receiver, sets up billing record
- If inter-exchange, sends set-up message to switch controller on nearest backbone
- Controller not directly involved in the forwarding of voice samples (control versus data plane)

Simplified Controller FSM



Simplified… actual #5 ESS switch control software is 15 million lines of C code (much of which is this stuff)

Common Channel Inter-office Signaling (CCIS)

- Older phone network used in-channel & in-band signaling between controllers using tones (discovered by *phone phreaks*)
- Current system uses out-of-band signaling
  - more secure and flexible
  - uses packet switching
  - messages use SS7 protocol

Signaling System 7

- Covers call establishment, routing and enhanced services (conference calls, etc)
- Entire protocol stack
  - SCCP (analogous to TCP)
  - MTP-3 (analogous to IP)
  - MTP-{2,1} (datalink, physical)
  - predates ISO; hard to extend
- Q.931 standard defines ISDN-UP semantics (call control, admission, etc)

Routing Structure

- calls are routed as closely as possible (within exchanges, between exchanges in same area, or through backbone if necessary)
- (near) fully-connected backbone makes routing decision fairly straightforward
- hierarchical area/prefix/number format provides global uniqueness and scaling

Telephone Network Routing

- COs or tandem switches connect to [multiple] core switches (toll switches). Multiple cores from various IXCs.
- Dense connectivity within core provides for reasonably simple routing:
  - if src/dst in same CO, connect them
  - if src/dst in same LEC, use 1-hop path between COs
  - otherwise, call to (one of) the core(s)

Internet vs Telco Routing

- Phone call traffic relatively easy to predict (both load and time), so can pre-select paths
- Telephone switches/links very reliable
- Centralized control over core
- Highly connected with multiple equal-cost paths
- QoS for each path (but same for all)

Dynamic Non-Hierarchical Routing (DNHR)

- 10 time periods each day
- each toll switch assigned a primary (1-hop) path to another toll switch and a list of alternate (2-hop) paths [by time]
- try 1-hop path first, then try others in order (called *crankback*)
- crankback useful when routing supports QoS but wants small connection rejection rate

The Erlang Map

- used to compute blocking probability on a trunk group given load, capacity
- DNHR assigns alternate paths to toll switches to minimize blocking probability
- So, path depends on expected load which depends on the path selection!
- How to deal with this:
  - system of equations (Erlang Map)
  - unique fixed-point solution (Erlang fixed point)

Erlang Formula

- *B(k)*=blocking probability on trunk k
- *B(k)=E(L(k),C(k)), E()* is Erlang formula
- *L(k)*=load on link k, *C(k)* is capacity link k
- *r*=set of links, *v(r)*=load on route r
- Approximate L(k) by:

$$L(k) = \sum_{r:k\in r} v(r) \prod_{j\in r-\{k\}} (1-B(j))$$

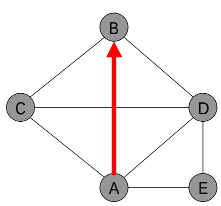
Intuition

- v(r) is the intrinsic load on router r
- each (1-B(j)) is a "thinning" of the load, so load on trunk k is just the "thinned" sum of the loads on all routes that share trunk k
- B(k) depends on some other B(j)'s through the B()=E() equation and eqn.
- So, each B(k) is implicitly defined by the others, forming the Erlang map

Metastability in DNHR

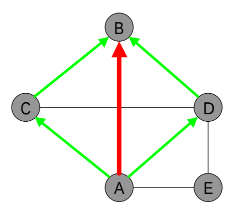
- Solution of Erlang map is the long-term mean blocking probability
- This mean is usually the mean of two values b_{high} and b_{low}
- Two values more representative of operating states of the network using DNHR. Why?

Metastability in DNHR



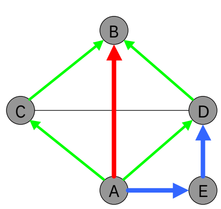
•A->B calls saturate 1-hop link

Metastability in DNHR



•New calls routed thru {ACB} & {ADB}

Metastability in DNHR



•A->D call won't fit across {A,D} link

Metastability in DNHR

- System can reach a stable point in which most or all traffic is taking 2-hop paths when it could all be taking 1-hop paths
- undesirable because can lead to high blocking probability with moderate load
- can be prevented by reserving part of each link bandwidth for 1-hop calls (called trunk reservation). May increase individual blocking probability, but leads to overall smaller blocking probability

Other Techniques

- TSMR (Trunk Status Map Routing)
  - in DNRH, alternative paths updated about once a week based on traffic studies
  - each switch measures load, tells central computer
  - periodic alternate path recomputation by central computer

Other Techniques

- RTNR (Real-Time Network Routing)
  - current-generation routing algorithm (replaced DNHR in ATT switches in 1991)
  - distributed control
  - each switch measure load on all outgoing trunks
  - to make decision, originator asks destination for its trunk loading list, takes logical AND of the two lists and chooses path (which is symmetric)
  - about 1 or 2 blocked in core (of 260 million)