Operating Systems and Networks

Networks Part 2: Physical Layer

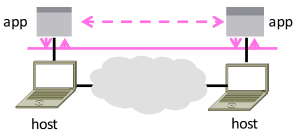
Adrian Perrig
Network Security Group
ETH Zürich

Overview

- Important concepts from last lecture
  - Statistical multiplexing, statistical multiplexing gain
  - OSI 7 layer model, interfaces, protocols
  - Encapsulation, demultiplexing
- This lecture
  - Socket programming overview
  - Physical layer
- Online lecture videos: http://computernetworks5e.org

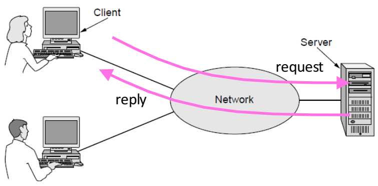
Network-Application Interface

- Defines how apps use the network
  - Lets apps talk to each other via hosts; hides the details of the network



Motivating Application

- Simple client-server connection setup



Motivating Application (2)

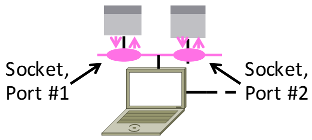
- Simple client-server connection setup
  - Client app sends a request to server app
  - Server app returns a (longer) reply
- This is the basis for many apps!
  - File transfer: send name, get file (§6.1.4)
  - Web browsing: send URL, get page
  - Echo: send message, get it back
- Let's see how to write this app …

Socket API

- Simple abstraction to use the network
  - The network service API used to write all Internet applications
  - Part of all major OSes and languages; originally Berkeley (Unix) ~1983
- Supports two kinds of network services
  - Streams: reliably send a stream of bytes
  - Datagrams: unreliably send separate messages. (Ignore for now.)

Socket API (2)

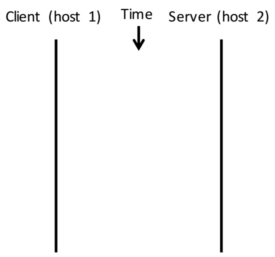
- Sockets let applications attach to the local network at different ports



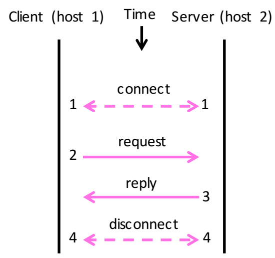
Socket API (3)

Primitive	Meaning
SOCKET	Create a new communication endpoint
BIND	Associate a local address with a socket
LISTEN	Announce willingness to accept connections; give queue size
ACCEPT	Passively wait for an incoming connection
CONNECT	Actively attempt to establish a connection
SEND	Send some data over the connection
RECEIVE	Receive some data from the connection
CLOSE	Release the connection

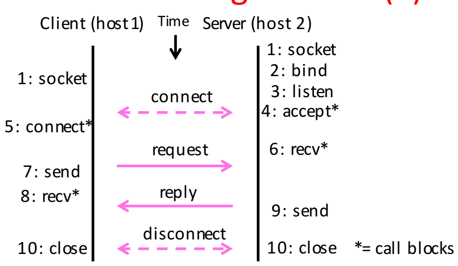
Using Sockets



Using Sockets (2)



Using Sockets (3)



Client Program (outline)

```
socket()        // make socket
getaddrinfo()   // server and port name
                // www.example.com:80
connect()       // connect to server [block]
…
send()          // send request
recv()          // await reply [block]
…               // do something with data!
close()         // done, disconnect
```

Server Program (outline)

```
socket()        // make socket
getaddrinfo()   // for port on this host
bind()          // associate port with socket
listen()        // prepare to accept connections
accept()        // wait for a connection [block]
…
recv()          // wait for request
…
send()          // send the reply
close()         // eventually disconnect
```

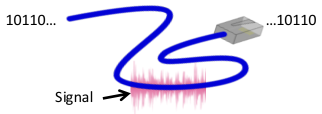

Where we are in the Course

- Beginning to work our way up starting with the Physical layer

Application
Transport
Network
Link
Physical

Scope of the Physical Layer

- Concerns how signals are used to transfer message bits over a link
  - Wires etc. carry <u>analog signals</u>
  - We want to send <u>digital bits</u>

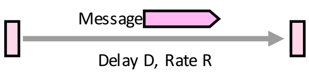


Topics

1. Properties of media
   - Wires, fiber optics, wireless
2. Simple signal propagation
   - Bandwidth, attenuation, noise
3. Modulation schemes
   - Representing bits, noise
4. Fundamental limits
   - Nyquist, Shannon

Simple Link Model

- We'll end with an abstraction of a physical channel
  - <u>Rate</u> (or bandwidth, capacity, speed) in bits/second
  - <u>Delay</u> or <u>Latency</u> in seconds, related to length



- Other important properties:
  - Whether the channel is broadcast, and its error rate

Message Latency

- <u>Latency</u> L: delay to send a message over a link
  - <u>Transmission delay</u>: time to put M-bit message "on the wire"

    T-delay = M (bits) / Rate (bits/sec) = M/R seconds

  - <u>Propagation delay</u>: time for bits to propagate across the wire

    P-delay = Length / speed of signals = Length / ⅔c = D seconds

  - Combining the two terms we have: **L = M/R + D**

Metric Units

- The main prefixes we use:

Prefix	Exp.	prefix	exp.
K(ilo)	10^3	m(illi)	10^{-3}
M(ega)	10^6	μ(micro)	10^{-6}
G(iga)	10^9	n(ano)	10^{-9}

- Use powers of 10 for rates, 2 for storage or data size
  - 1 Mbps = 1,000,000 bps, 1 KB = 2^{10} bytes
- "B" is for bytes, "b" is for bits

Latency Examples

- "Dialup" with a telephone modem:
  - D = 5 ms, R = 56 kbps, M = 1250 bytes
- Broadband cross-country link:
  - D = 50 ms, R = 10 Mbps, M = 1250 bytes

Latency Examples (2)

- "Dialup" with a telephone modem:
  D = 5 ms, R = 56 kbps, M = 1250 bytes
  L = 5 ms + (1250x8)/(56 x 10^3) sec = 184 ms!
- Broadband cross-country link:
  D = 50 ms, R = 10 Mbps, M = 1250 bytes
  L = 50 ms + (1250x8) / (10 x 10^6) sec = 51 ms
- A long link or a slow rate means high latency
  - Often, one delay component dominates

Bandwidth-Delay Product

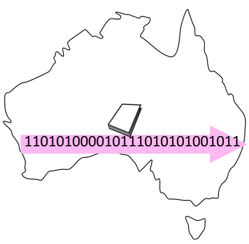
- Messages take space on the wire!

- The amount of data in flight is the <u>bandwidth-delay (BD) product</u>

  BD = R x D
  - Measure in bits, or in messages
  - Small for LANs, big for "long fat" pipes

Bandwidth-Delay Example

- Fiber at home, cross-country
  R=40 Mbps, D=50 ms
  BD = 40 x 10^6 x 50 x 10^{-3} bits
  = 2000 Kbit
  = 250 KB
- That's quite a lot of data "in the network"!



How "Long" is a Bit?

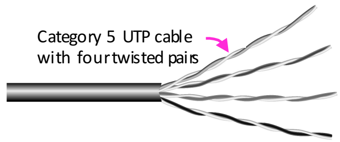
- Interesting trivia: how "long" is the representation of a bit on a wire?
- Considering a fiber optic cable
  - Signal propagation speed: 200'000'000 m/s
  - Sending rate: 1Gbps → duration of sending one bit: 1ns
  - Bit "length": 1ns * 200'000'000 m/s = 0.2 m
  - "Length" of a 1Kb packet: 0.2m * 8 * 2^{10} = 1.6km

Types of Media (§2.2, 2.3)

- Media propagate signals that carry bits of information
- We'll look at some common types:
  - Wires
  - Fiber (fiber optic cables)
  - Wireless

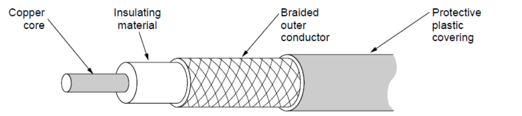
Wires – Twisted Pair

- Very common; used in LANs and telephone lines
  - Twists can reduce radiated signal or reduce effect of external interference signal



Wires – Coaxial Cable

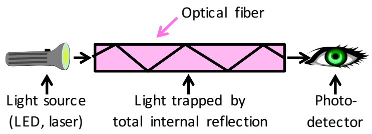
- Also common. Better shielding for better performance



- Other kinds of wires too: e.g., electrical power (§2.2.4)

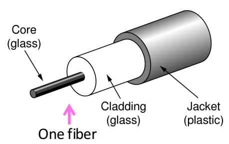
Fiber

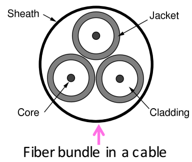
- Long, thin, pure strands of glass
  - Enormous bandwidth (high speed) over long distances



Fiber (2)

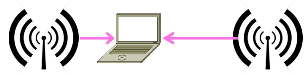
- Two varieties: multi-mode (shorter links, cheaper) and single-mode (up to ~100 km)

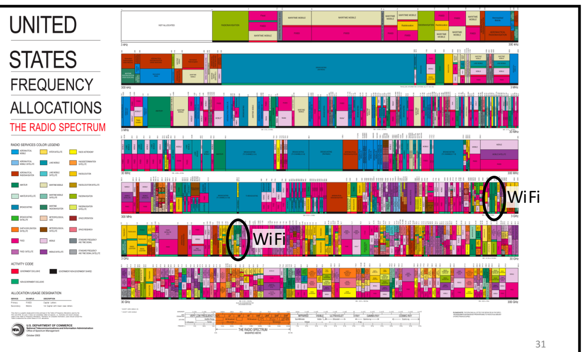




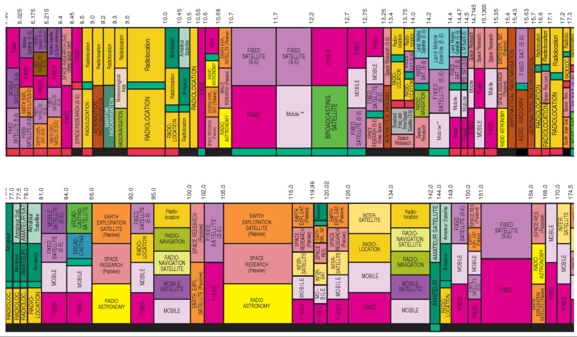
Wireless

- Sender radiates signal over a region
  - In many directions, unlike a wire, to potentially many receivers
  - Nearby signals (same freq.) interfere at a receiver; need to coordinate use



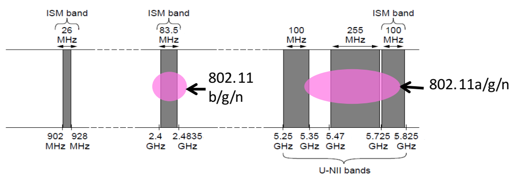


31



Wireless (2)

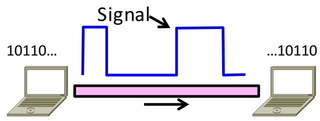
- Microwave, e.g., 3G, and unlicensed (ISM: Industry Science Medicine) frequencies, e.g., WiFi, are widely used for computer networking



33

Signals (§2.2, 2.3)

- Analog signals encode digital bits. We want to know what happens as signals propagate over media

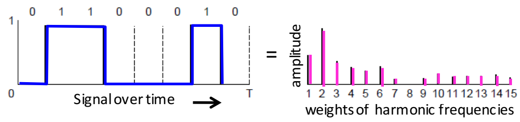


34

Frequency Representation

- A signal over time can be represented by its frequency components (called Fourier analysis)

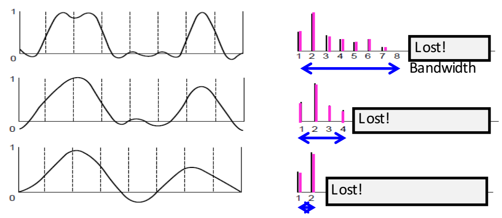
$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi nft) + \sum_{n=1}^{\infty} b_n \cos(2\pi nft)$$



35

Effect of Less Bandwidth

- Fewer frequencies (=less bandwidth) degrades signal



36

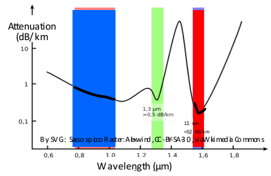
Signals over a Wire

- What happens to a signal as it passes over a wire?
  1. The signal is delayed (propagates at ⅔c)
  2. The signal is attenuated
  3. Frequencies above a cutoff are highly attenuated
  4. Noise is added to the signal (later, causes errors)

EE: Bandwidth = width of frequency band, measured in Hz
CS: Bandwidth = information carrying capacity, in bits/sec

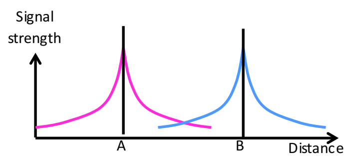
Signals over Fiber

- Light propagates with very low loss in three very wide frequency bands
  – Use a carrier to send information



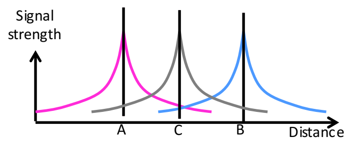
Signals over Wireless

- Travel at speed of light, spread out and attenuate faster than $1/dist^2$



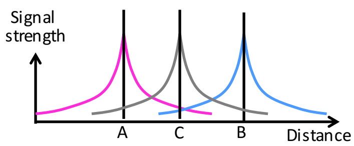
Signals over Wireless (2)

- Multiple signals on the same frequency interfere at a receiver



Signals over Wireless (3)

- Interference leads to notion of spatial reuse (of same freq.)

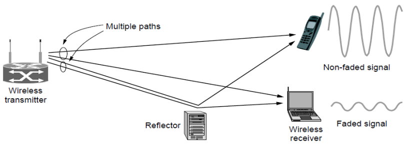


Signals over Wireless (4)

- Various other effects too!
  – Wireless propagation is complex, depends on environment
- Some key effects are highly frequency dependent
  – E.g., multipath at microwave frequencies

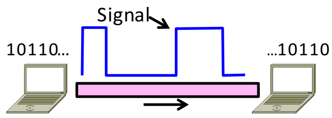
Wireless Multipath

- Signals bounce off objects and take multiple paths
  - Some frequencies attenuated at receiver, varies with location
  - Messes up signal; handled with sophisticated methods (§2.5.3)



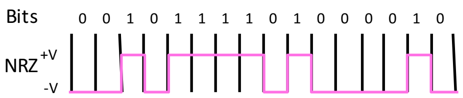
Modulation (§2.5)

- We've talked about signals representing bits. How, exactly?
  - This is the topic of modulation



A Simple Modulation

- Let a high voltage (+V) represent a 1, and low voltage (-V) represent a 0
  - This is called NRZ (Non-Return to Zero)



Many Other Schemes

- Can use more signal levels, e.g., 4 levels is 2 bits per symbol
- Practical schemes are driven by engineering considerations
  - E.g., clock recovery

Clock Recovery

- Um, how many zeros was that?
  - Receiver needs frequent signal transitions to decode bits

1 0 0 0 0 0 0 0 0 0 … 0

- Several possible designs
  - E.g., Manchester coding and scrambling (§2.5.1)

Clock Recovery – 4B/5B

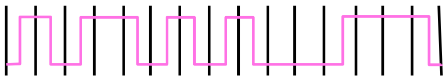
- Map every 4 data bits into 5 code bits without long runs of zeros
  - 0000 → 11110, 0001 → 01001, 1110 → 11100, … 1111 → 11101
  - Has at most 3 zeros in a row
  - Also invert signal level on a 1 to break up long runs of 1s (called NRZI, §2.5.1)

Clock Recovery – 4B/5B (2)

- 4B/5B code for reference:
  - 0000→11110, 0001→01001, 1110→11100, … 1111→11101
- Message bits: 1 1 1 1 0 0 0 0 0 0 0 1

Coded Bits: 1 1 1 0 1 1 1 1 1 0 0 1 0 0 1

Signal:

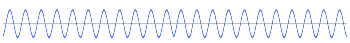


Passband Modulation

- What we have seen so far is baseband modulation for wires
  - Signal is sent directly on a wire
- These signals do not propagate well on fiber / wireless
  - Need to send at higher frequencies
- Passband modulation carries a signal by modulating a carrier

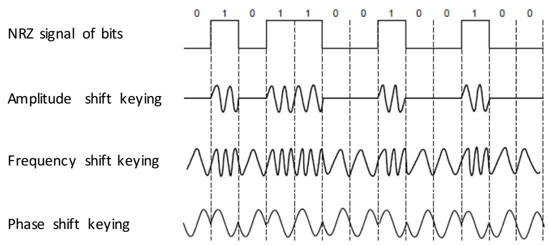
Passband Modulation (2)

- Carrier is simply a signal oscillating at a desired frequency:



- We can modulate it by changing:
  - Amplitude, frequency, or phase

Passband Modulation (3)

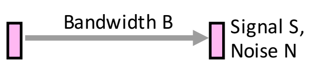


Fundamental Limits (§2.1)

- How rapidly can we send information over a link?
  - Nyquist limit (~1924)
  - Shannon capacity (1948)
- Practical systems are devised to approach these limits

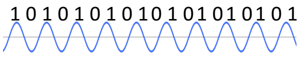
Key Channel Properties

- The bandwidth (B), signal strength (S), and noise strength (N)
  - B limits the rate of transitions
  - S and N limit how many signal levels we can distinguish



Nyquist Limit

- The maximum symbol rate is 2B

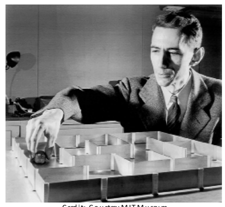


- Thus if there are V signal levels, ignoring noise, the maximum bit rate is:

$R = 2B \log_2 V$ bits/sec

Claude Shannon (1916-2001)

- Father of information theory
  - "A Mathematical Theory of Communication", 1948
- Fundamental contributions to digital computers, security, and communications

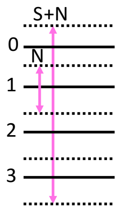


Electromechanical mouse that "solves" mazes!

Credit: Courtesy MIT Museum

Shannon Capacity

- How many levels we can distinguish depends on S/N
  - Or SNR, the Signal-to-Noise Ratio
  - Note noise is random, hence some errors
- SNR given on a log-scale in deciBels:
  - $SNR_{dB} = 10\log_{10}(S/N)$



Shannon Capacity (2)

- Shannon limit is for capacity (C), the maximum information carrying rate of the channel:

$C = B \log_2(1 + S/N)$ bits/sec

Wired/Wireless Perspective

- Wires and Fiber — Engineer SNR for data rate
  - Engineer link to have requisite SNR and B
  - →Can fix data rate
- Wireless — Adapt data rate to SNR
  - Given B, but SNR varies greatly, e.g., up to 60 dB!
  - →Can't design for worst case, must adapt data rate

Putting it all together – DSL

- DSL (Digital Subscriber Line, see §2.6.3) is widely used for broadband; many variants offer 10s of Mbps
  - Reuses twisted pair telephone line to the home; it has up to ~2 MHz of bandwidth but uses only the lowest ~4 kHz

DSL (2)

- DSL uses passband modulation (called OFDM §2.5.1)
  - Separate bands for upstream and downstream (larger)
  - Modulation varies both amplitude and phase (called QAM)
  - High SNR, up to 15 bits/symbol, low SNR only 1 bit/symbol

ADSL2: