Outline for review

Chapter 1 :Introduction

- Internet overview
- what's a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
  - Internet structure
- performance: loss, delay, throughput
- layering, service models
- security
- history

Packet Delay

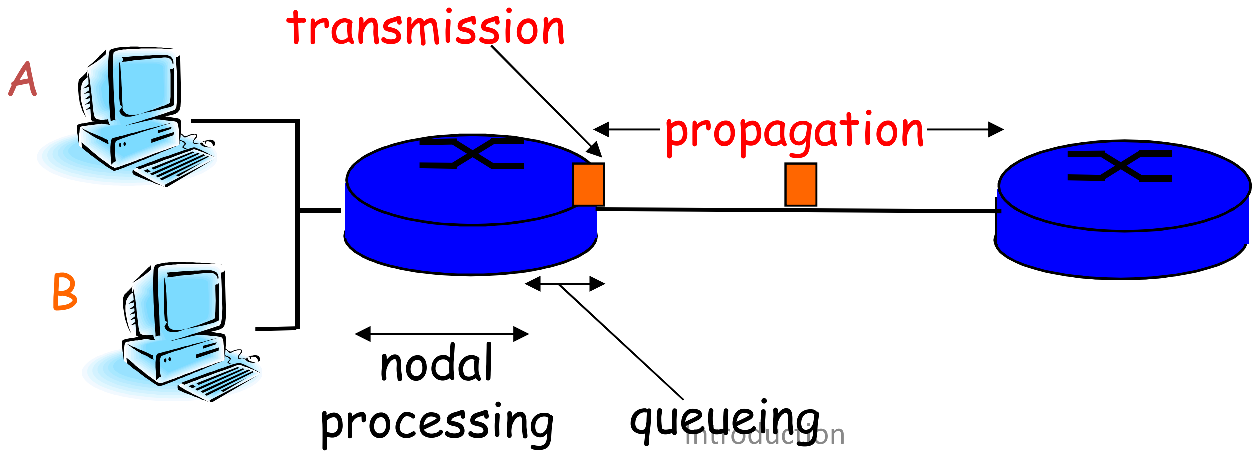
Transmission delay:

- R=link bandwidth (bps)
- L=packet length (bits)
- Time to send bits into link = L/R

Propagation delay:

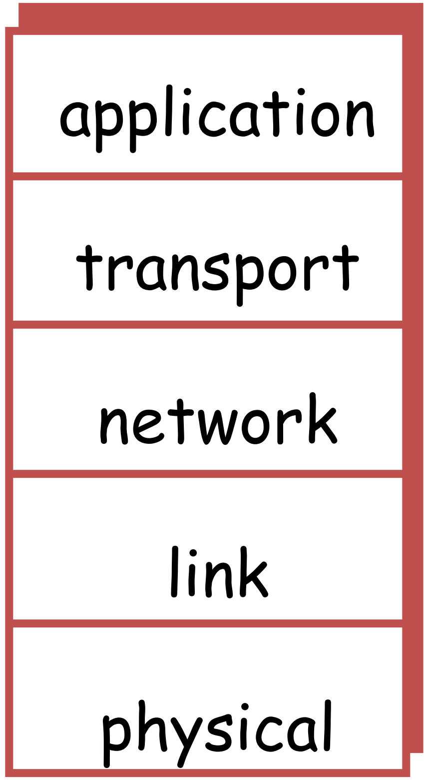
- d = length of physical link
- s = propagation speed in medium (~2×10^8 m/sec)
- propagation delay = d/s

Note: s and R are very different quantities!



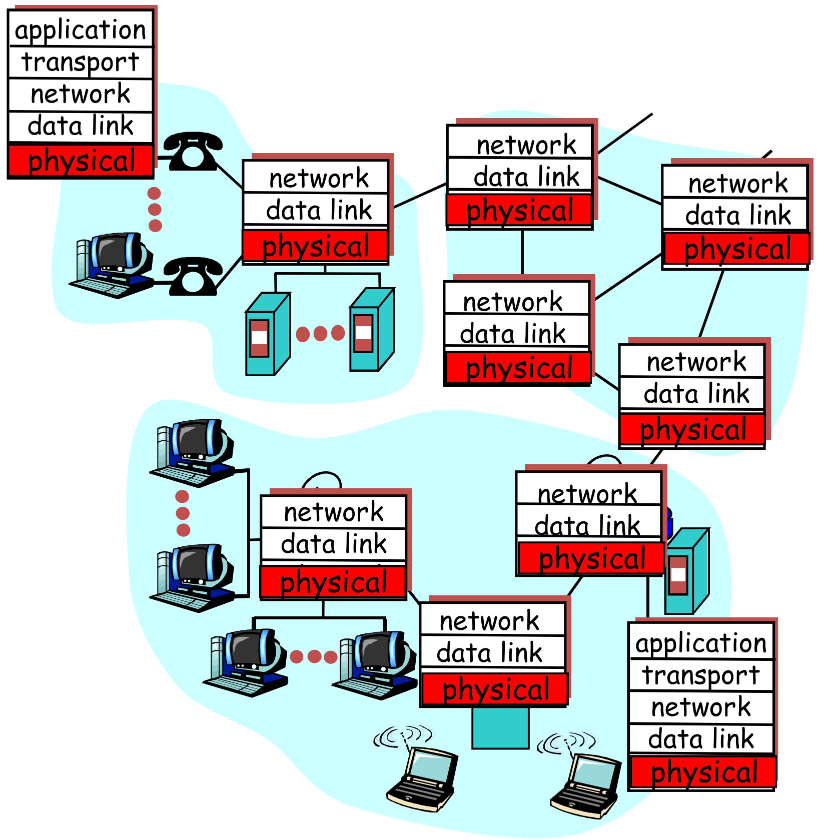
Internet Protocol Stack

- Application: supporting network applications
  - FTP, SMTP, HTTP
- Transport: process-process data transfer
  - TCP, UDP
- Network: routing of datagrams from source to destination
  - IP, routing protocols
- Link: data transfer between neighboring network elements
  - PPP, Ethernet
- Physical: bits “on the wire”



Chapter 2: Physical Layer

- The function of the physical layer and the issues to be considered
- Four important characteristics of the physical layer
- Some Basic Concepts in Data Communication
- Modulation technology and coding technology
- Nyquist's Law and Shannon's Formula
- Multiplexing technology and digital signal coding technology
- Commonly used transmission medias



Channel Characteristics

- Shannon's Formula: limit information transmission rate C of channel can be expressed as:

$$C = W\log_2(1+S/N)$$

  - W - Frequency bandwidth in Hz
  - S - Average signal power through the channel
  - N - Gaussian noise power through the channel
  - S/N – related to signal-to-noise ratio
    - Typically in db (decibels): $10\log_{10}S/N$
- Actual rate much lower than C due to signal loss
- Possible to achieve error-free transmission as long as transmission rate < C

Digital signal coding technology

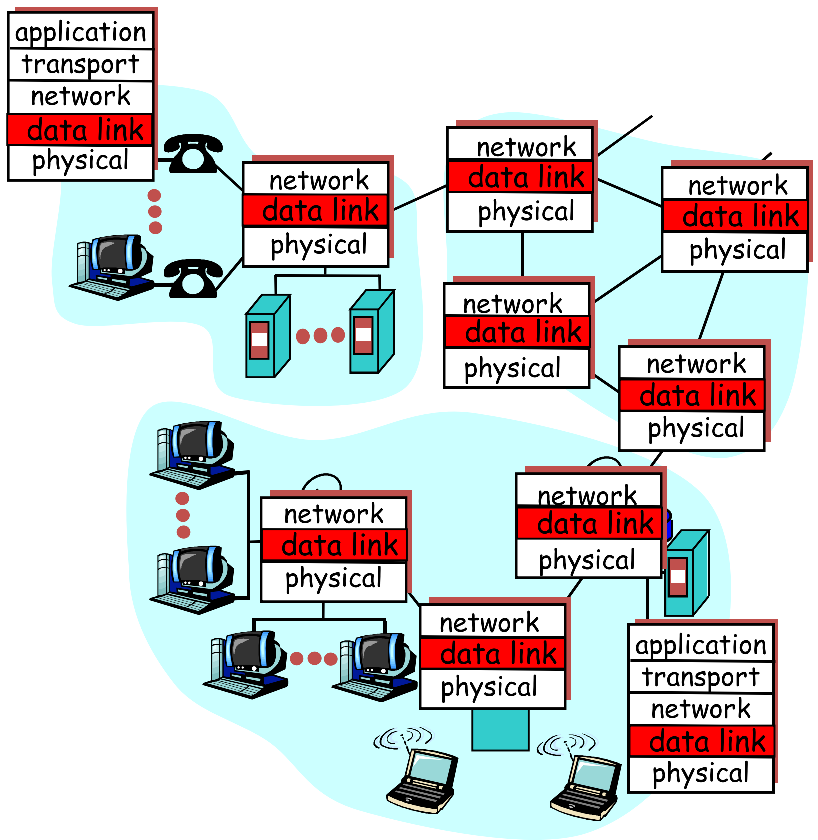
- Non-Retrun-to-Zero (NRZ)
- Manchester encoding
- Differential Manchester encoding

Exercise:

Draw the waveform of "001101" with NRZ, Manchester encoding, differential Manchester encoding.

Chapter 3: the data link layer

- principles behind data link layer services:
  - framing
  - error detection, correction
  - reliable data transfer
  - sharing a broadcast channel: media access control (MAC)

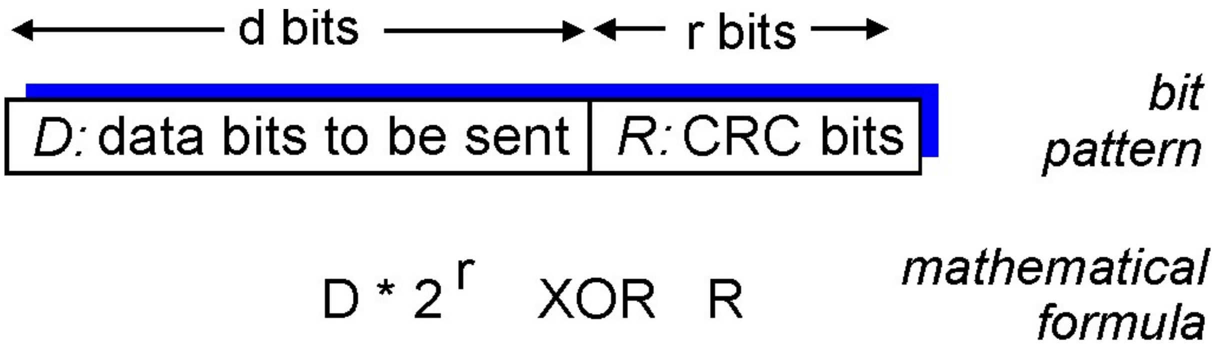


Framing

- Character count method :
  - encapsulate datagram into frame, adding header (character number)
- First and tail bound method based on character:
  - encapsulate datagram into frame, adding header, trailer (character)
- First and tail bound method based on bit:
  - encapsulate datagram into frame, adding header, trailer (bit sequence, e.g. 7EH)
- Physical layer coding violation method:
  - encapsulate datagram into frame without stuffing
  - Only be used in the networks with redundancy coding technology in the physical layer

Checksumming: Cyclic Redundancy Check

- view data bits, D, as a binary number
- choose r+1 bit pattern (generator), G
- goal: choose r CRC bits, R, such that
  - <D,R> exactly divisible by G (modulo 2)
  - receiver knows G, divides <D,R> by G. If non-zero remainder: error detected!
  - can detect all burst errors less than r+1 bits
- widely used in practice (ATM, HDCL)



Principles of reliable data transfer

- From rdt 1.0 to rdt 3.0: Assumptions and mechanisms
  - rdt 1.0: over a perfectly reliable channel
  - rdt2.x (rdt2.0, 2.1, 2,2): channel with bit errors
  - rdt3.0: channels with errors and loss
- Pipelined protocols
  - Go-back-N
  - Selective Repeat

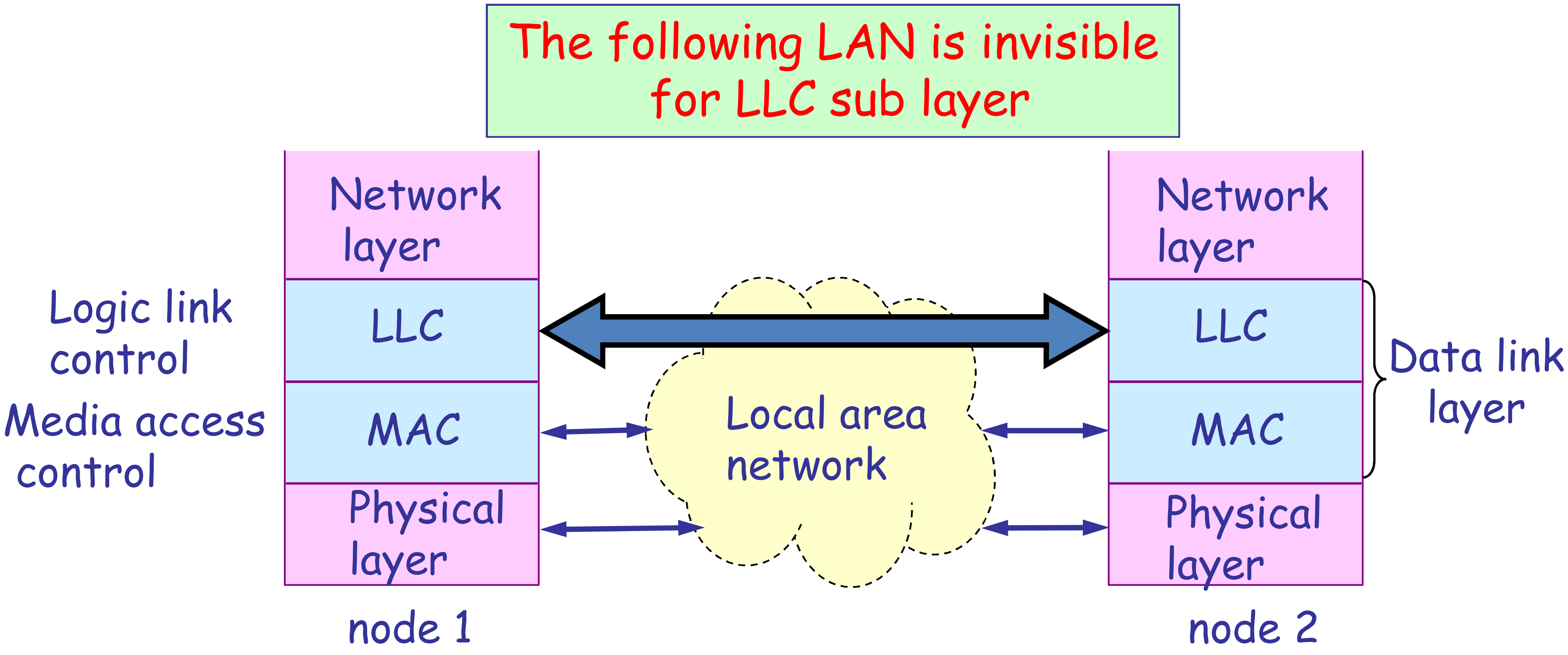
Summary of MAC protocols

- What do you do with a shared media?
  - Channel Partitioning, by time, frequency or code
    - Time Division,Code Division, Frequency Division
  - Random partitioning (dynamic),
    - ALOHA, S-ALOHA, CSMA, CSMA/CD
    - carrier sensing: easy in some technoligies (wire), hard in others (wireless)
    - CSMA/CD used in Ethernet
  - Taking Turns
    - polling from a central cite, token passing

Chapter 4: Local Area Network

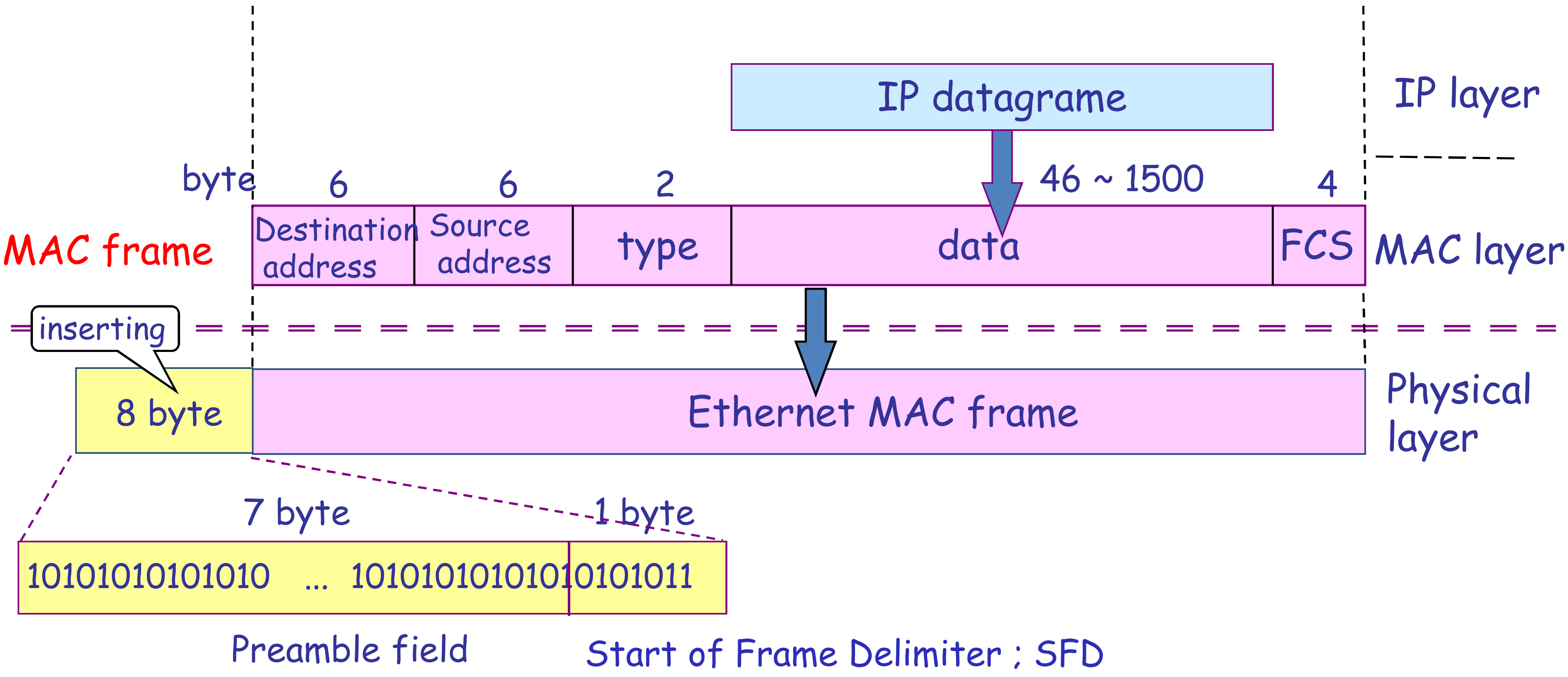
- LAN model
- Ethernet /IEEE 802.3(Ethernet V2 frame structure, CSMA/CD, MAC address)
- hubs, bridges, switches
- WLAN (IEEE 802.11, CSMA/CA, hidden terminal problem, RTS/CTS)

LAN model



For the same LLC, several MAC options may be provided.

Ethernet Frame Structure



Ethernet: uses CSMA/CD

A: sense channel, **if** idle

then {

transmit and monitor the channel;

If detect another transmission

then {

abort and send jam signal;

update # collisions;

delay as required by exponential backoff algorithm;

goto A

}

else {done with the frame; set collisions to zero}

}

else {wait until ongoing transmission is over and goto A}

Ethernet's CSMA/CD (more)

Exponential Backoff:

- *Goal*: adapt retransmission attemtps to estimated current load
  - heavy load: random wait will be longer
- first collision: choose K from {0,1}; delay is K x 512 bit transmission times
- after second collision: choose K from {0,1,2,3}…
- after ten or more collisions, choose K from {0,1,2,3,4,…,1023}

Hubs, switches and routers

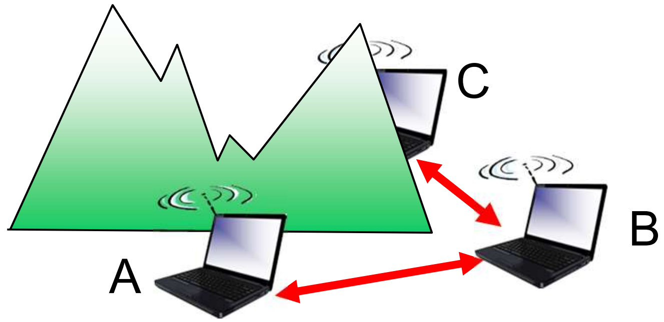
- Hubs: Physical Layer devices, do not isolate collision domains, (results in no increase in max throughput),cannot connect different Ethernet types (e.g., 10BaseT and 100baseT)
- Bridges: Link Layer devices, isolates collision domains since it buffers frames; bridges filter packets,do not isolate broadcast domains; maintain filtering tables, implement filtering, learning and spanning tree algorithms
- routers: network layer devices (examine network layer headers), isolates broadcast domains ; maintain routing tables, implement routing algorithms

802.11 MAC

- MAC layer covers three functional areas:
  - Reliable data delivery
    - ACK-based scheme for reliability (receiver sends ACK after each successful transmission)
  - Medium access control
    - CSMA/CA; collision avoidance, not collision detection, Why? How?
  - Security
    - Wired Equivalent Privacy (WEP),WEP relies on a secret key being shared by end hosts and APs

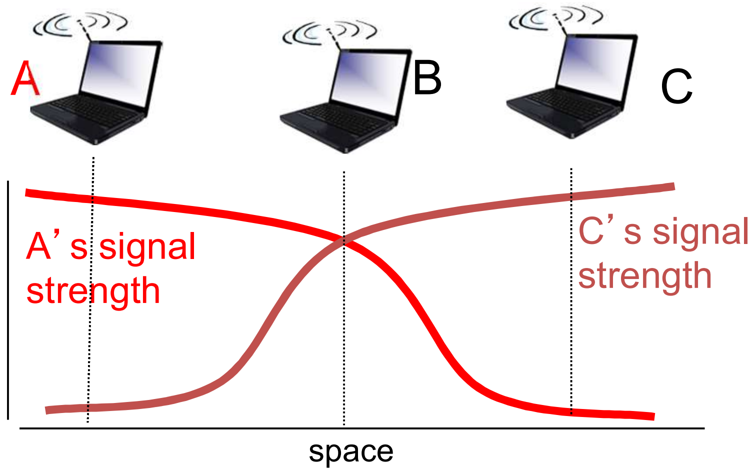
Wireless network characteristics

Multiple wireless senders and receivers create additional problems (beyond multiple access):



Hidden terminal problem

- B, A hear each other
- B, C hear each other
- A, C can not hear each other means A, C unaware of their interference at B



Signal attenuation:

- B, A hear each other
- B, C hear each other
- A, C can not hear each other interfering at B

Avoiding collisions (more)

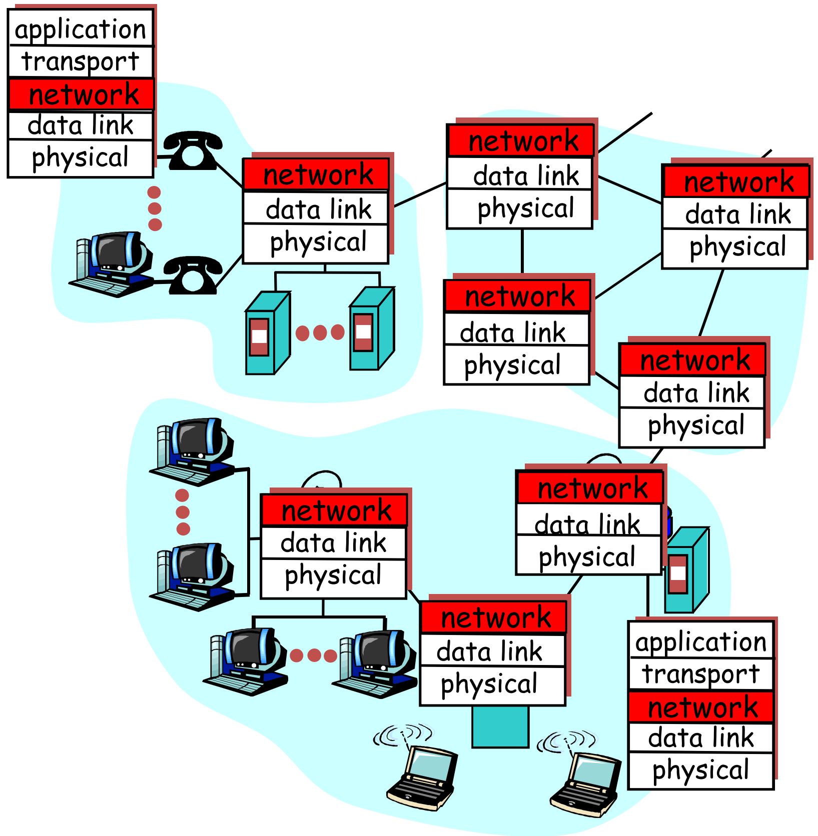
idea: allow sender to "reserve" channel rather than random access of data frames: avoid collisions of long data frames

- sender first transmits *small* request-to-send (RTS) packets to BS using CSMA
  - RTSs may still collide with each other (but they're short)
- BS broadcasts clear-to-send CTS in response to RTS
- CTS heard by all nodes
  - sender transmits data frame
  - other stations defer transmissions

avoid data frame collisions completely using small reservation packets!

Chapter 5: Network Layer

- Virtual circuit and datagram networks
- Routing algorithms:
  - Dijkstra's algorithm
  - Broadcast routing
  - Link state
  - Distance vector ("count to infinity" problem)
- Routing in the Internet (RIP, OSPF, BGP)
- IP: Internet protocol
  - IPv4 Datagram format
  - IP fragment
  - IPv4 addressing
  - NAT
  - ARP
  - ICMP
  - IPv6
  - From IPv4 to IPv6



Link state algorithm

Broadcast routing + Dijkstra's algorithm

- Having each node broadcast link-state packets to all other nodes → all nodes have an identical and complete view of the network.
- Using Dijkstra's algorithm compute the least-cost path from one node to all other nodes in the network
- If a link cost changes, re-broadcast and re-compute

Dijkstra's algorithm

```
Initialization:
  N' = {u}
  for all nodes v
    if v adjacent to u
        then D(v) = c(u,v)
    else D(v) = ∞

Loop
  find w not in N' such that D(w) is a minimum
  add w to N'
  update D(v) for all v adjacent to w and not in N' :
     D(v) = min( D(v), D(w) + c(w,v) )
  /* new cost to v is either old cost to v or known
   shortest path cost to w plus cost from w to v */
until all nodes in N'
```

Distance vector algorithm

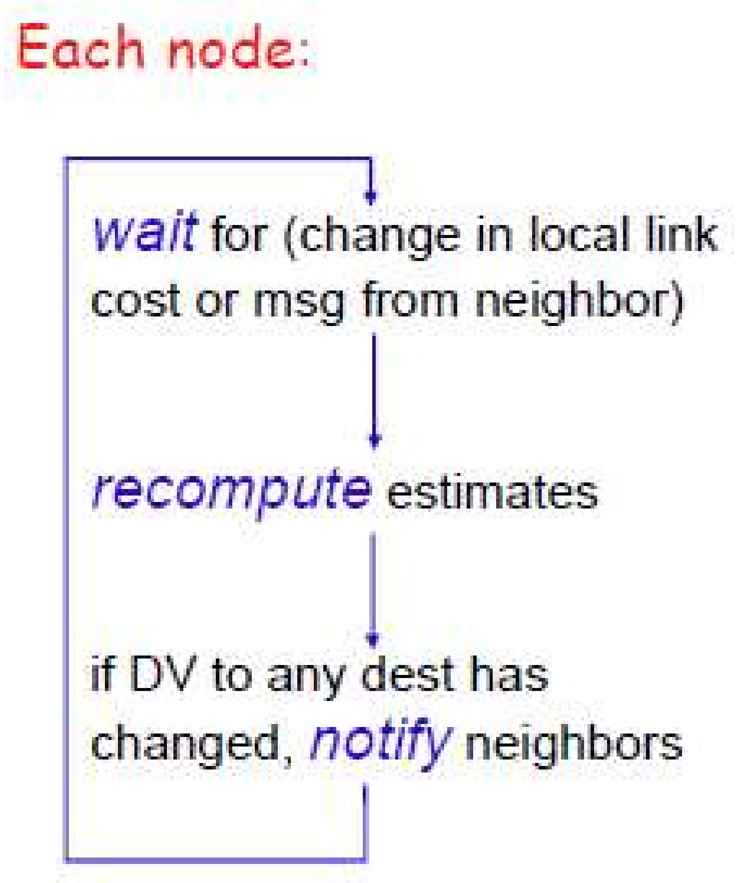
Iterative, asynchronous:

Each local iteration caused by:

- Local link cost change
- DV update message from neighbor

Distributed:

- Each node notifies neighbors only when its DV change
- Neighbors then notify their neighbors if necessary



Count to infinity problem

Link cost changes:

- good news travels fast
- Bad news travels slow – "count to infinity" problem!

How to solve the count to infinity problem:

- Poisoned reverse

"bad news travels slow"

"count to infinity" problem

node x table

from \ cost to	x	y	z
x	0	4	5
y	4	0	1
z	5	1	0

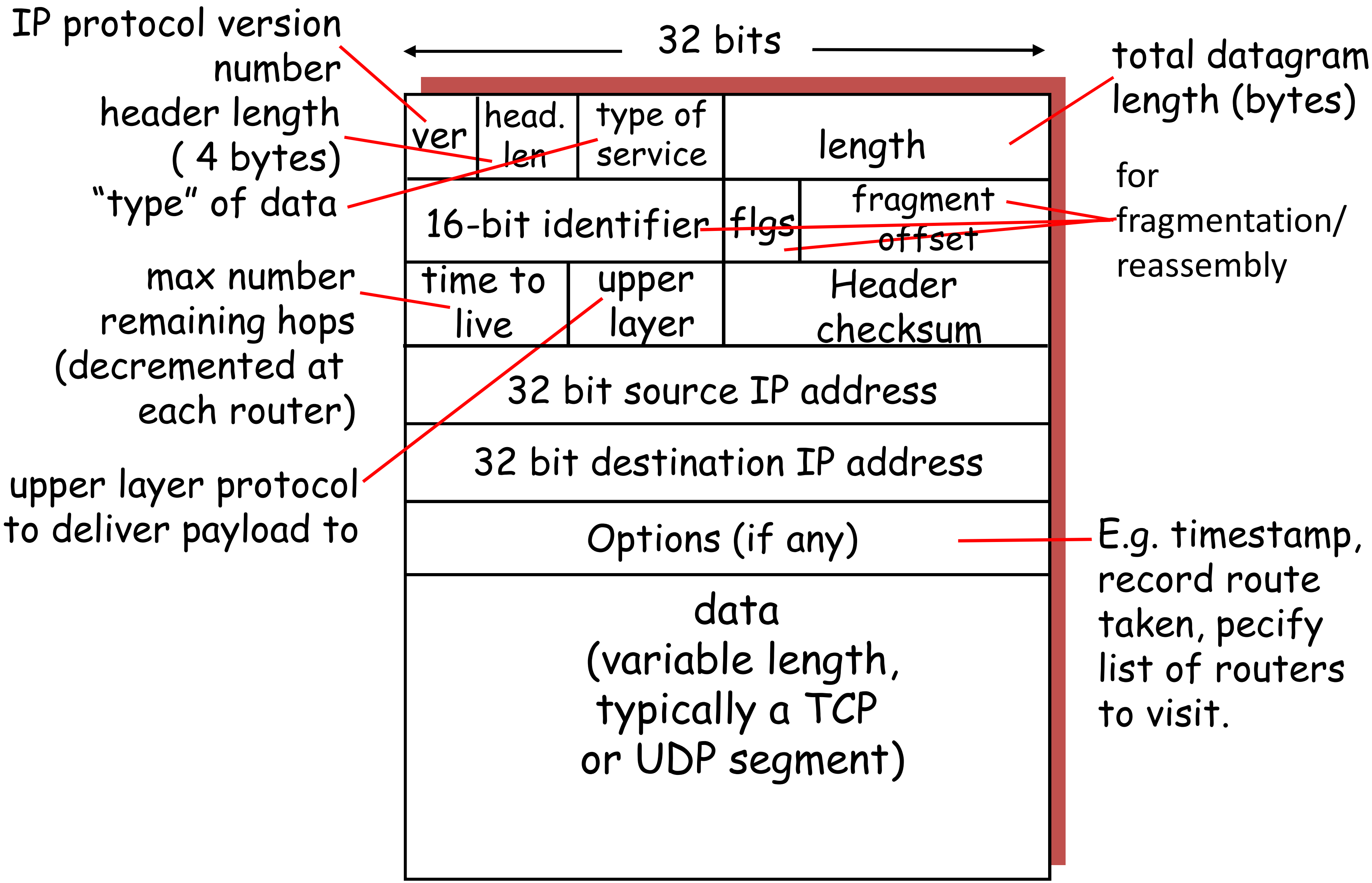
node y table

from \ cost to	x	y	z
x	0	4	5
y	4	0	1
z	5	1	0

node z table

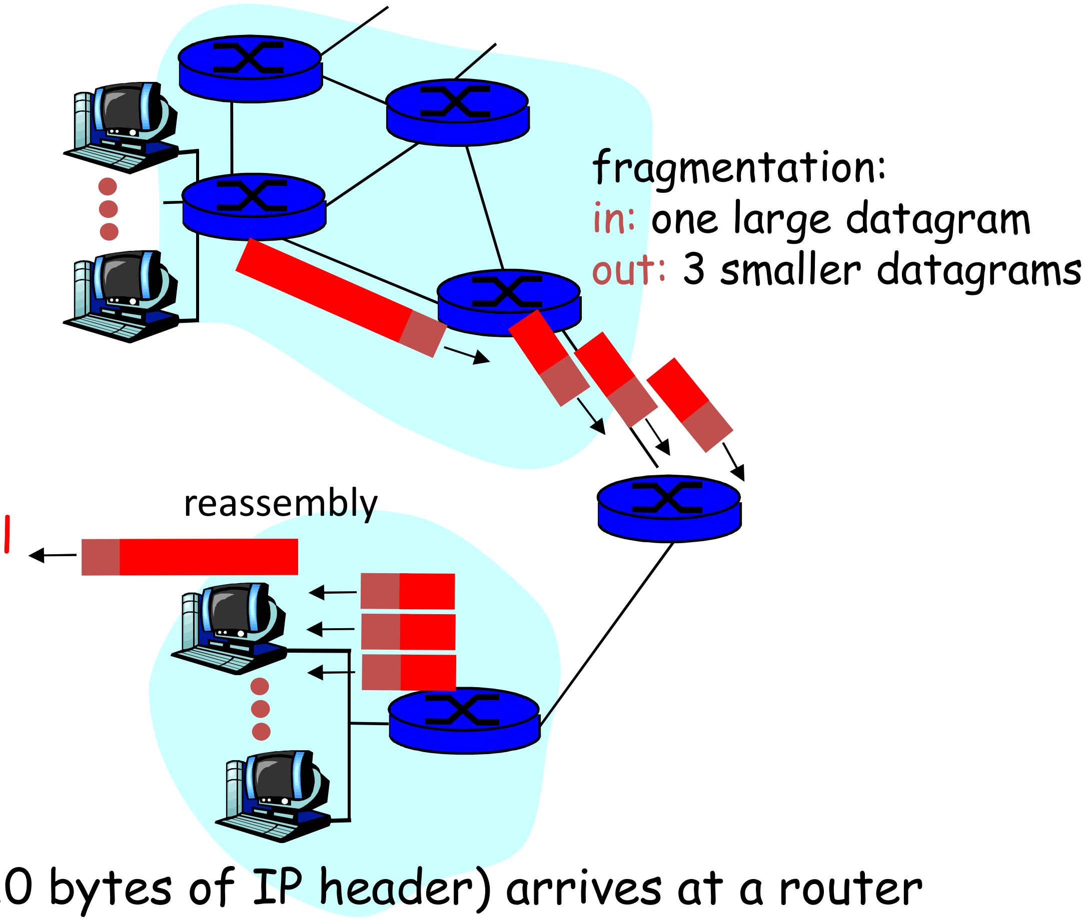
from \ cost to	x	y	z
x	0	4	5
y	4	0	1
z	5	1	0

IP datagram format



IP Fragmentation & Reassembly

- network links have MTU (max.transfer size) - largest possible link-level frame.
  - different link types, different MTUs
- large IP datagram divided ("fragmented") within net
  - one datagram becomes several datagrams
  - "reassembled" only at final destination
  - IP header bits used to identify, order related fragments



Q: A datagram of 4000 bytes (20 bytes of IP header) arrives at a router and must be forwarded to a link with an MTU of 1500 bytes. How to do?

IPv4 addressing

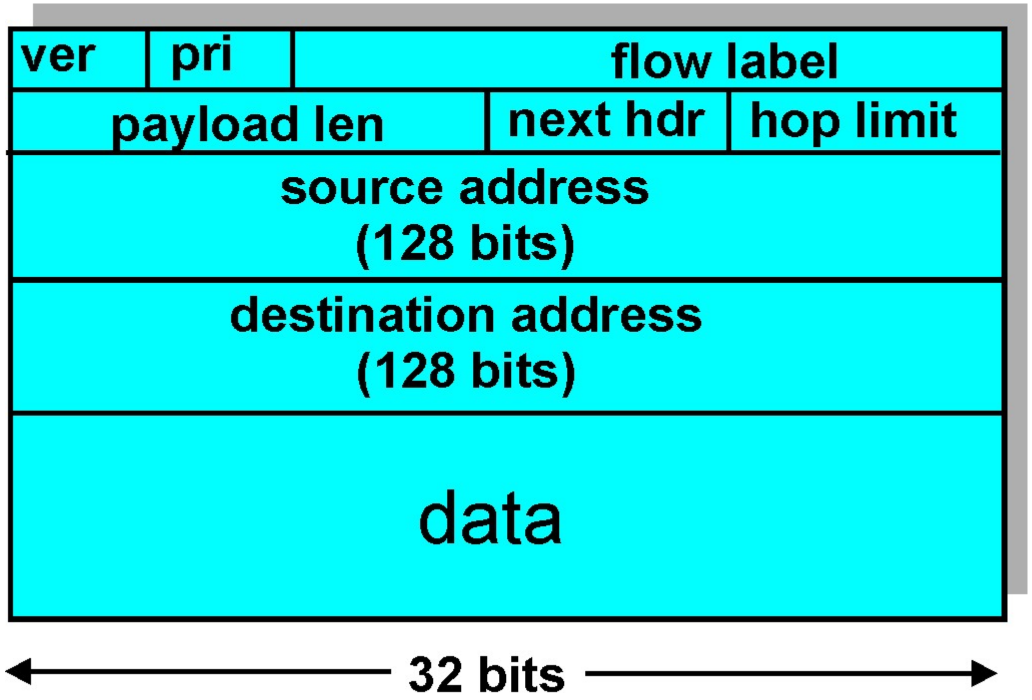
- Dotted-decimal notation:193.32.216.9
- classful addressing:
- Subnetting and subnet mask
- CIDR: Classless InterDomain Routing
  - network portion of address of arbitrary length
  - address format: a.b.c.d/x

network part | host part

11001000 00010111 00010000 00000000

200.23.16.0/23

IPv6 Header



ver
pri
flow label
payload len
next hdr
hop limit
source address
(128 bits)
destination address
(128 bits)
data
32 bits

IPv6 address

- Three types: unicast, multicast, anycast
- Colon hexadecimal notation:

68E6:8C64:FFFF:FFFF:0:1180:960A:FFFF

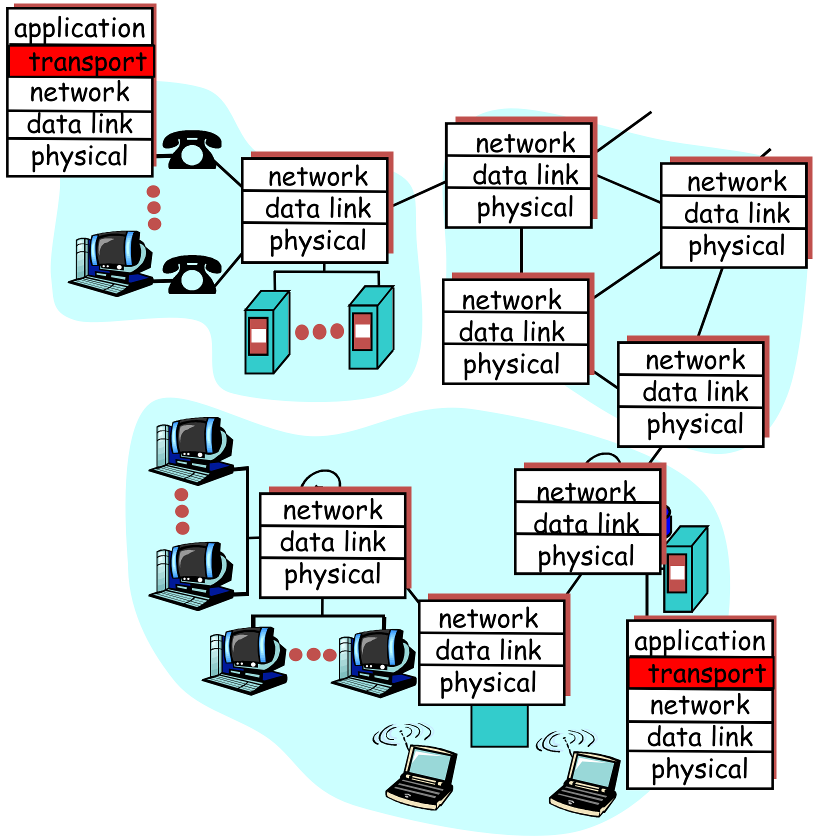
- Zero compression:

FF05:0:0:0:0:0:0:B3 == FF05::B3 ;

0:0:0:0:0:0:128.10.2.1 == ::128.10.2.1

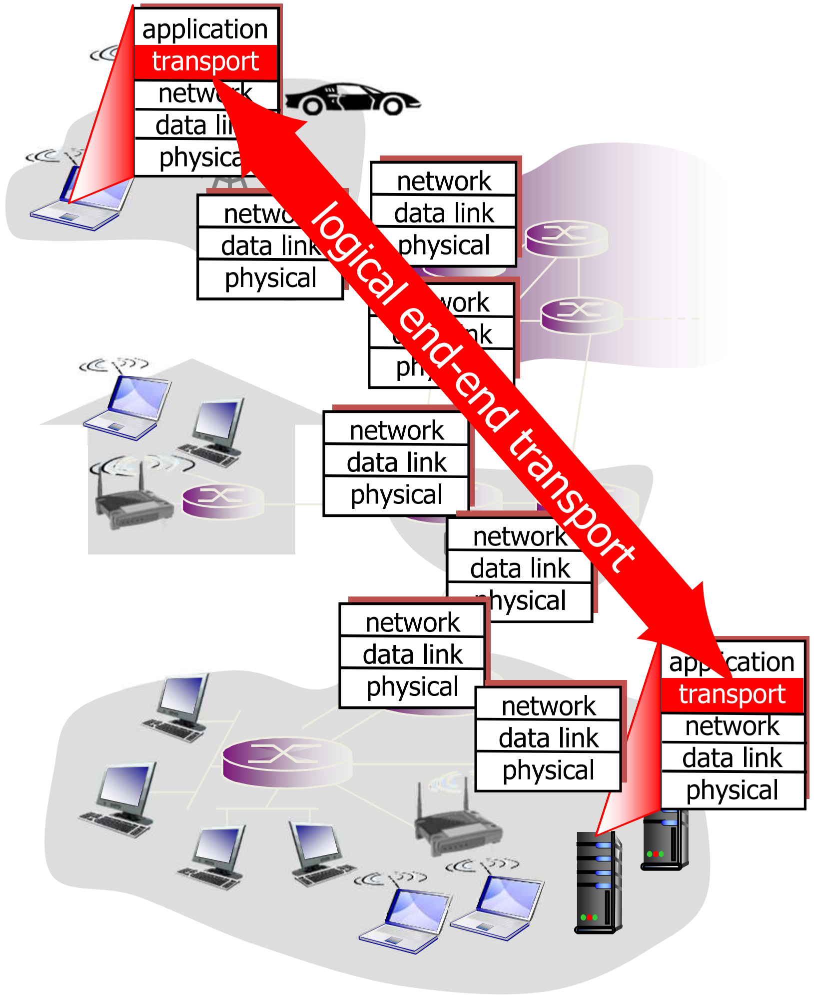
Chapter 6: Transport Layer

- Principles behind transport layer services:
  - Multiplexing, demultiplexing: port numbers and sockets
  - Reliable data transfer
  - Flow control
  - Congestion control
- Two transport layer protocols in the Internet
  - TCP, UDP
  - How to choose?



Internet transport-layer protocols

- reliable, in-order delivery (TCP)
  - congestion control
  - flow control
  - connection setup
- unreliable, unordered delivery: UDP
  - no-frills extension of "best-effort" IP
- services not available:
  - delay guarantees
  - bandwidth guarantees



TCP: Overview

- point-to-point:
  - one sender, one receiver
- reliable, in-order *byte stream:*
  - no “message boundaries”
- pipelined:
  - TCP congestion and flow control set window size
- full duplex data:
  - bi-directional data flow in same connection
  - MSS: maximum segment size
- connection-oriented:
  - handshaking (exchange of control msgs) inits sender, receiver state before data exchange
- flow controlled:
  - sender will not overwhelm receiver
- Congestion controlled

TCP seq. numbers, ACKs

sequence numbers:

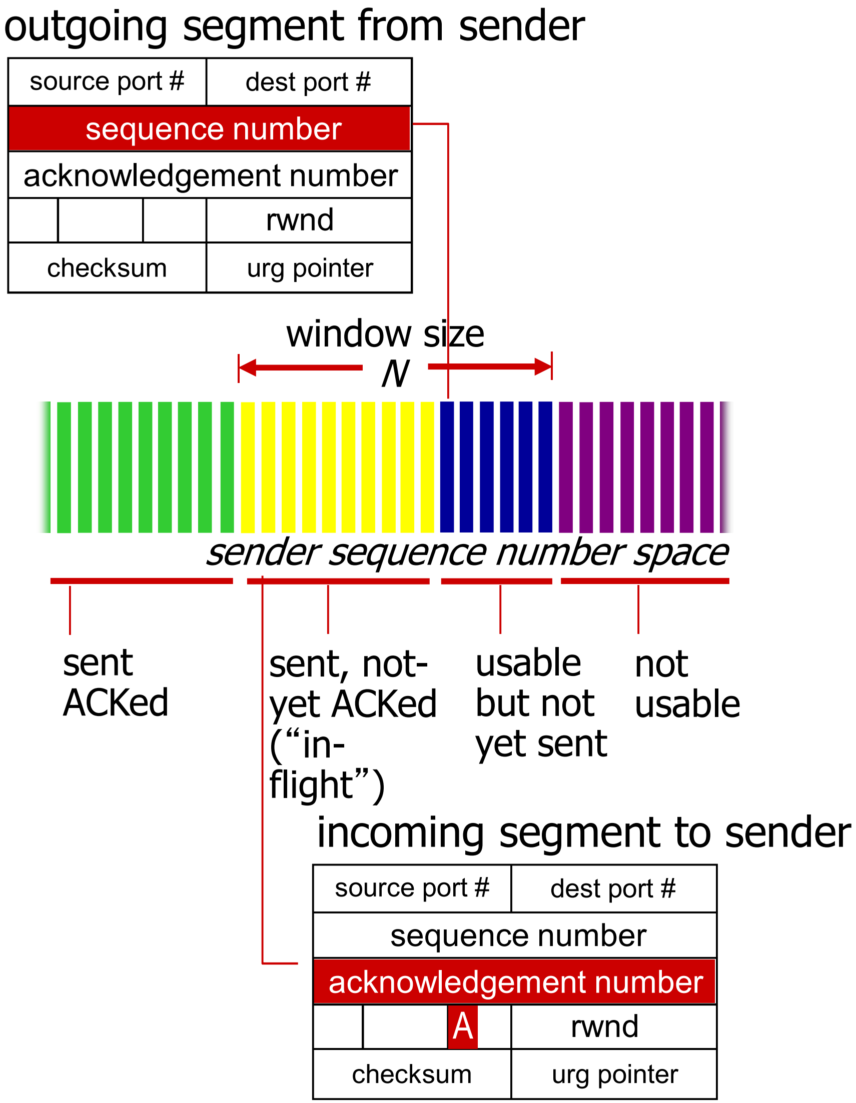
- byte stream "number" of first byte in segment's data

acknowledgements:

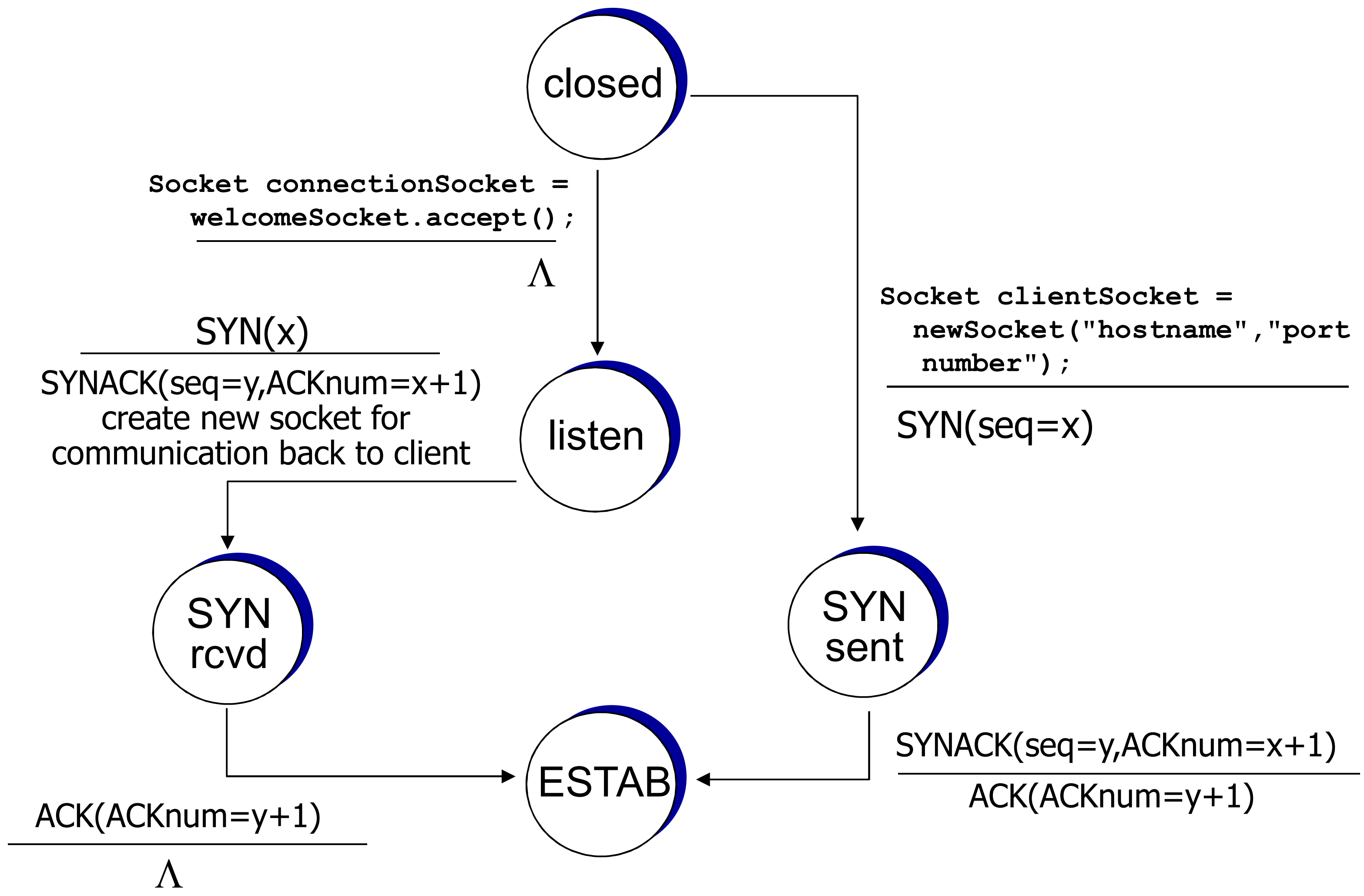
- seq # of next byte expected from other side
- cumulative ACK

Q: how receiver handles out-of-order segments

- A: TCP spec doesn't say, - up to implementor



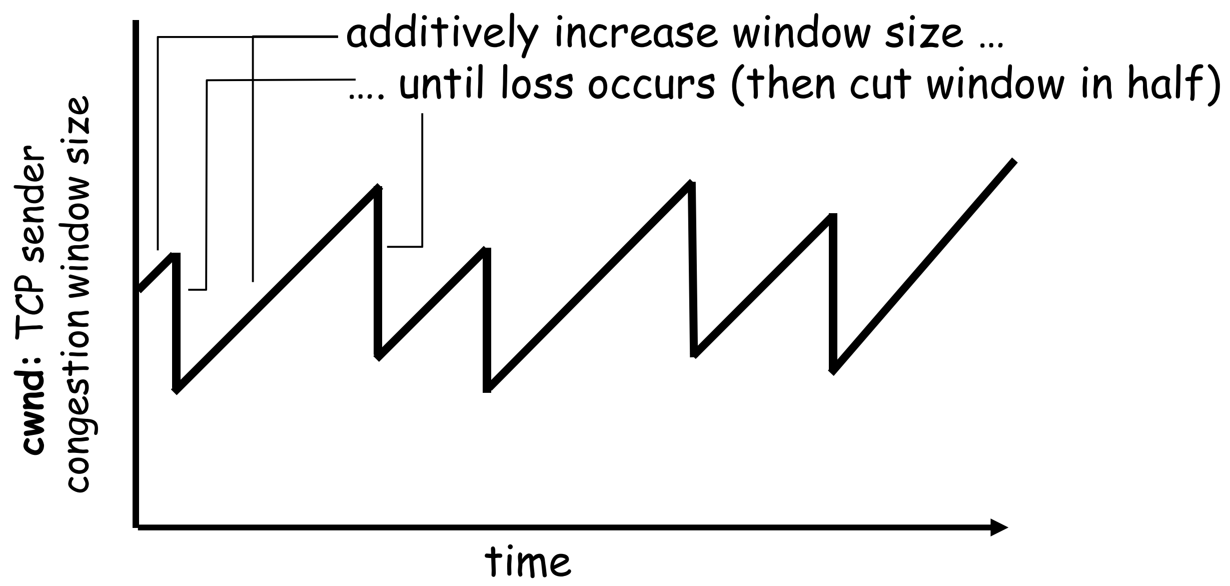
TCP 3-way handshake: FSM



TCP congestion control:
additive increase multiplicative decrease

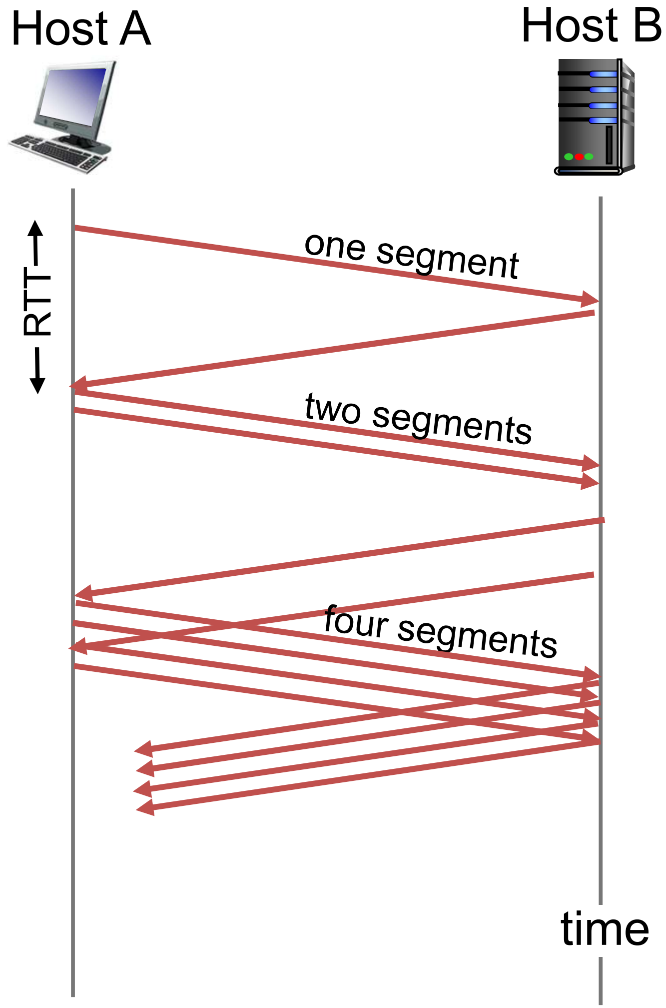
- *approach:* sender increases transmission rate (window size), probing for usable bandwidth, until loss occurs
  - *additive increase:* increase **cwnd** by 1 MSS every RTT until loss detected
  - *multiplicative decrease*: cut **cwnd** in half after loss

AIMD saw tooth behavior: probing for bandwidth



TCP Slow Start

- when connection begins, increase rate exponentially until first loss event:
  - initially **`cwnd`** = 1 MSS
  - double **`cwnd`** every RTT
  - done by incrementing **`cwnd`** for every ACK received
- *summary:* initial rate is slow but ramps up exponentially fast



TCP: detecting, reacting to loss

- loss indicated by timeout:
  - **`cwnd`** set to 1 MSS;
  - window then grows exponentially (as in slow start) to threshold, then grows linearly
- loss indicated by 3 duplicate ACKs: TCP RENO
  - dup ACKs indicate network capable of delivering some segments
  - **`cwnd`** is cut in half window then grows linearly
- TCP Tahoe always sets **`cwnd`** to 1 (timeout or 3 duplicate acks)