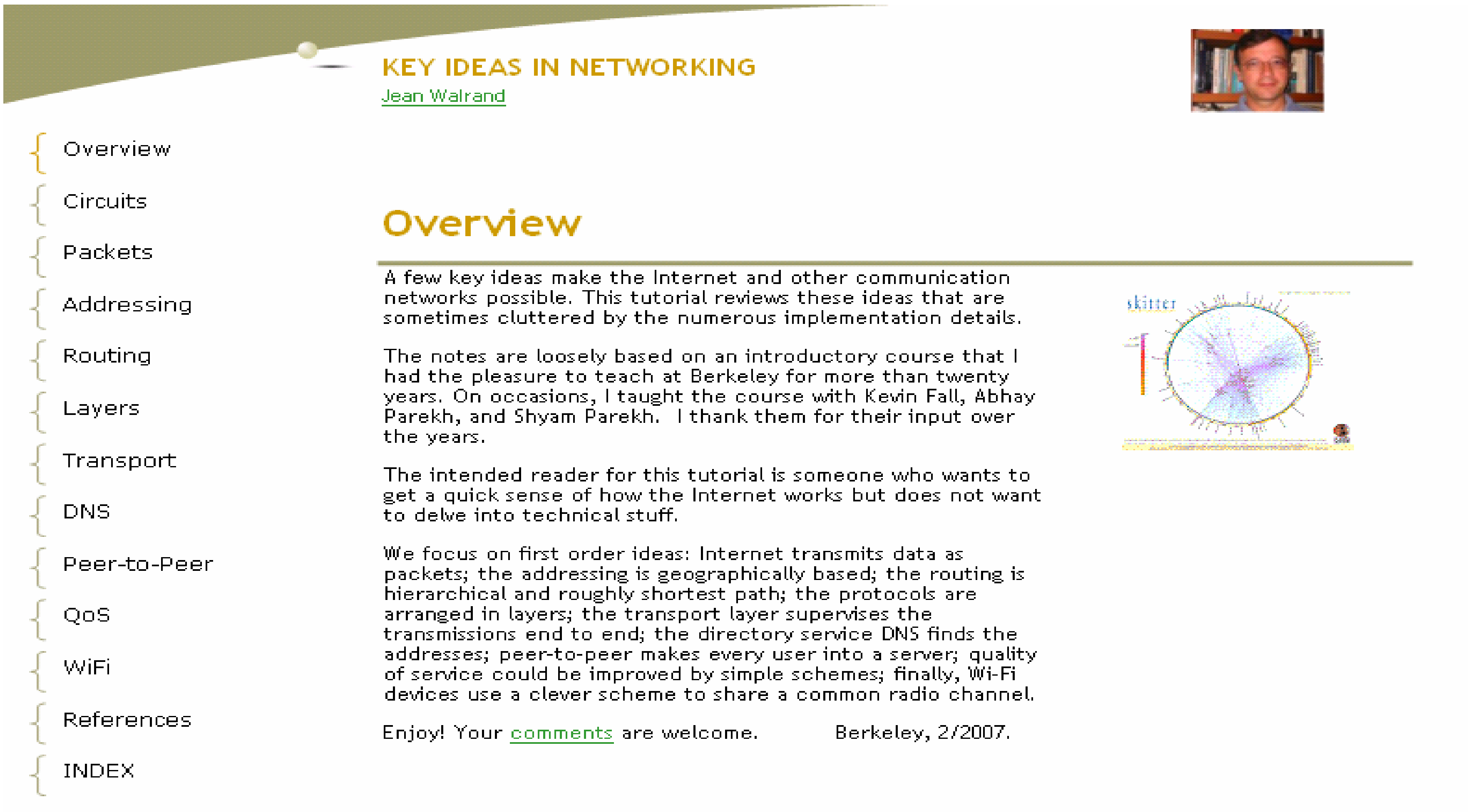
Highly Recommended Networking Tutorial…

- http://robotics.eecs.berkeley.edu/~wlr/Tutorials/nets.htm



KEY IDEAS IN NETWORKING

Jean Walrand

- Overview
- Circuits
- Packets
- Addressing
- Routing
- Layers
- Transport
- DNS
- Peer-to-Peer
- QoS
- WiFi
- References
- INDEX

Overview

A few key ideas make the Internet and other communication networks possible. This tutorial reviews these ideas that are sometimes cluttered by the numerous implementation details.

The notes are loosely based on an introductory course that I had the pleasure to teach at Berkeley for more than twenty years. On occasions, I taught the course with Kevin Fall, Abhay Parekh, and Shyam Parekh. I thank them for their input over the years.

The intended reader for this tutorial is someone who wants to get a quick sense of how the Internet works but does not want to delve into technical stuff.

We focus on first order ideas: Internet transmits data as packets; the addressing is geographically based; the routing is hierarchical and roughly shortest path; the protocols are arranged in layers; the transport layer supervises the transmissions end to end; the directory service DNS finds the addresses; peer-to-peer makes every user into a server; quality of service could be improved by simple schemes; finally, Wi-Fi devices use a clever scheme to share a common radio channel.

Enjoy! Your comments are welcome. Berkeley, 2/2007.

?

Unit 8

MACs for Local Area Networks

Acknowledgments: These slides were originally developed by Prof. Jean Walrand for EE122. The past and current EE122 instructors including Kevin Fall, Abhay Parekh, Shyam Parekh, and Adam Wolisz have contributed to their evolution.

?

MACs for Local Area Networks

Medium Access Protocols

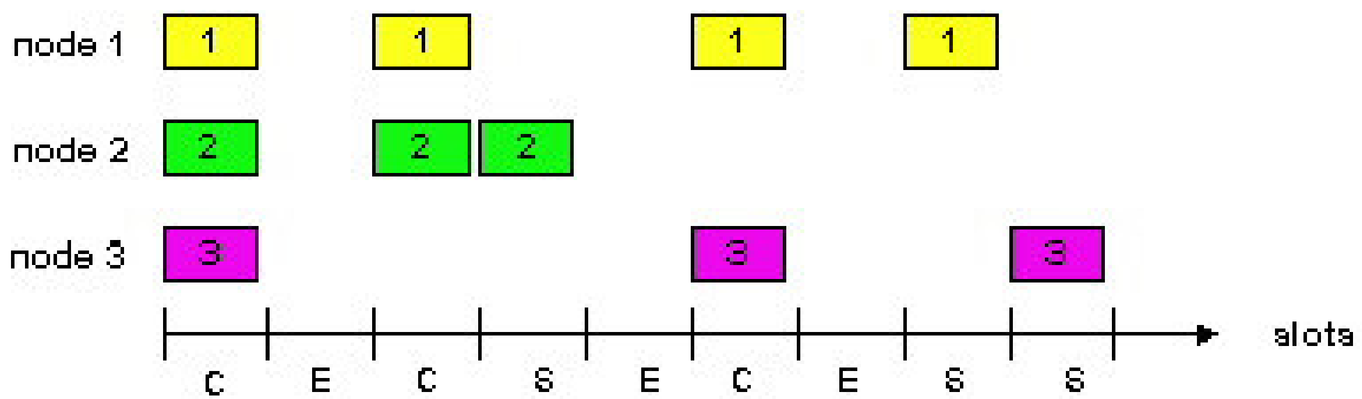
- Channel partitioning
  - Divide channel into smaller “pieces” (e.g., time slots, frequency)
  - Allocate a piece to node for exclusive use
- Random access
  - Allow collisions
  - “recover” from collisions
- “Taking-turns”
  - Tightly coordinate shared access to avoid collisions

Random Access protocols

- When node has packet to send
  - Transmit at full channel data rate R.
  - No *a priori* coordination among nodes
- Two or more transmitting nodes -> “collision”,
- Random access MAC protocol specifies:
  - How to detect collisions
  - How to recover from collisions
- Examples of random access MAC protocols:
  - Slotted ALOHA
  - CSMA and CSMA/CD

Slotted Aloha

- Time is divided into equal size slots (= packet transmission time)
- Node with new arriving pkt: transmit at beginning of next slot



Success (S), Collision (C), Empty (E) slots

Slotted Aloha: Efficiency

- What is the maximum fraction of successful transmissions?
- Suppose N stations have packets to send
  - Each transmits in slot with probability *p*
  - Prob. successful transmission *S* is (very approximated analysis!):
    - by a particular node: $S = p\,(1-p)^{(N-1)}$
    - by any of N nodes
  - S = Prob (only one transmits)

    $= N\,p\,(1-p)^{(N-1)} \approx 1/e = \mathbf{0.37}$

CSMA: Carrier Sense Multiple Access

- CS (Carrier Sense) means that each node can distinguish between an idle and a busy link
- Sender operations:
  - If channel sensed idle: transmit entire packet
  - If channel sensed busy, defer transmission
    - Persistent CSMA: retry immediately with probability p when channel becomes idle
    - Non-persistent CSMA: retry after a random time interval

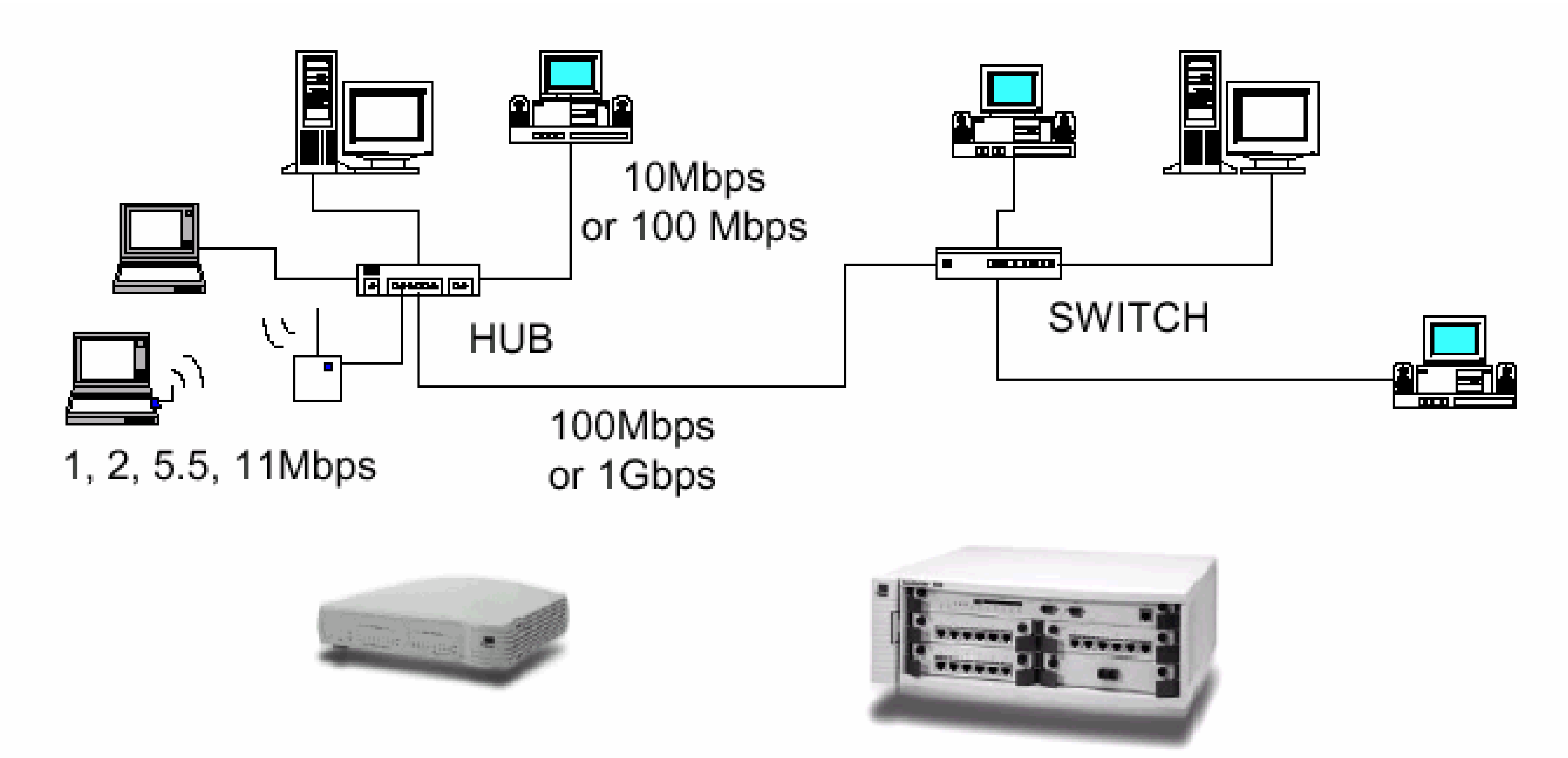
Ethernet Perspective

- Ethernet is wildly successful, partly due to low cost (compare with FDDI or Token Ring-- - see text book)
- Some issues:
  - nondeterministic service
  - no priorities
  - min frame size may be large

High-Level Goals

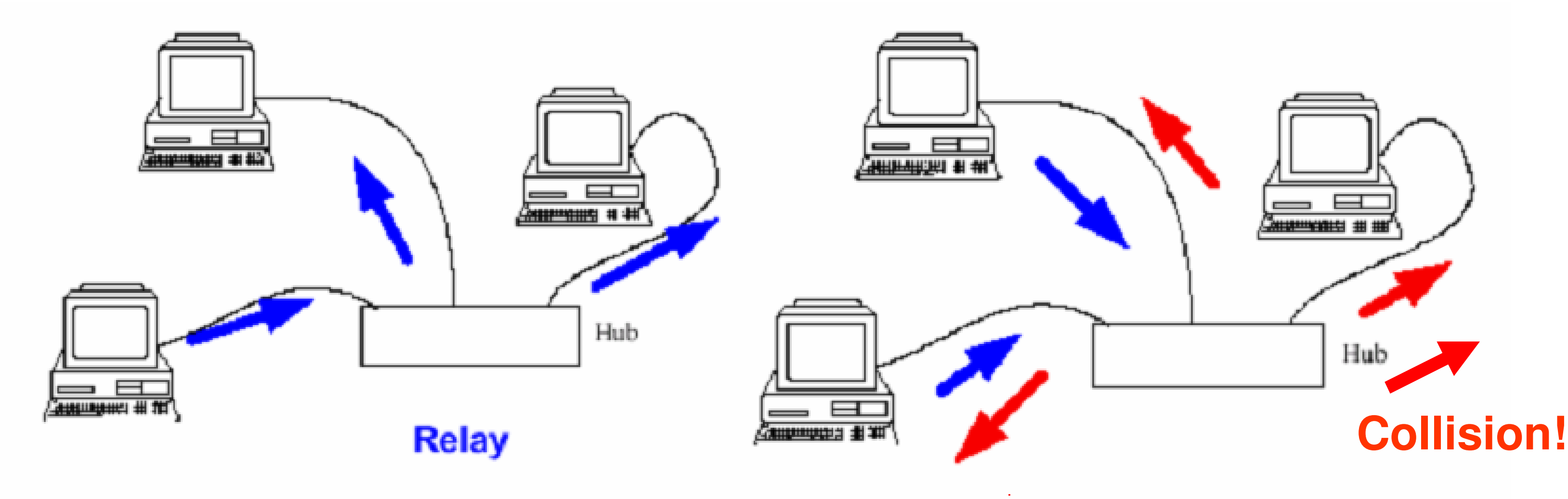
- Goal: share a communication medium among multiple hosts connected to it
- Problem: arbitrate between connected hosts
- Solution goals:
  - High resource utilization
  - Avoid starvation
  - Simplicity (non-decentralized algorithms)

Typical Ethernet Setup



Operations

- Hub: Single Collision Domain

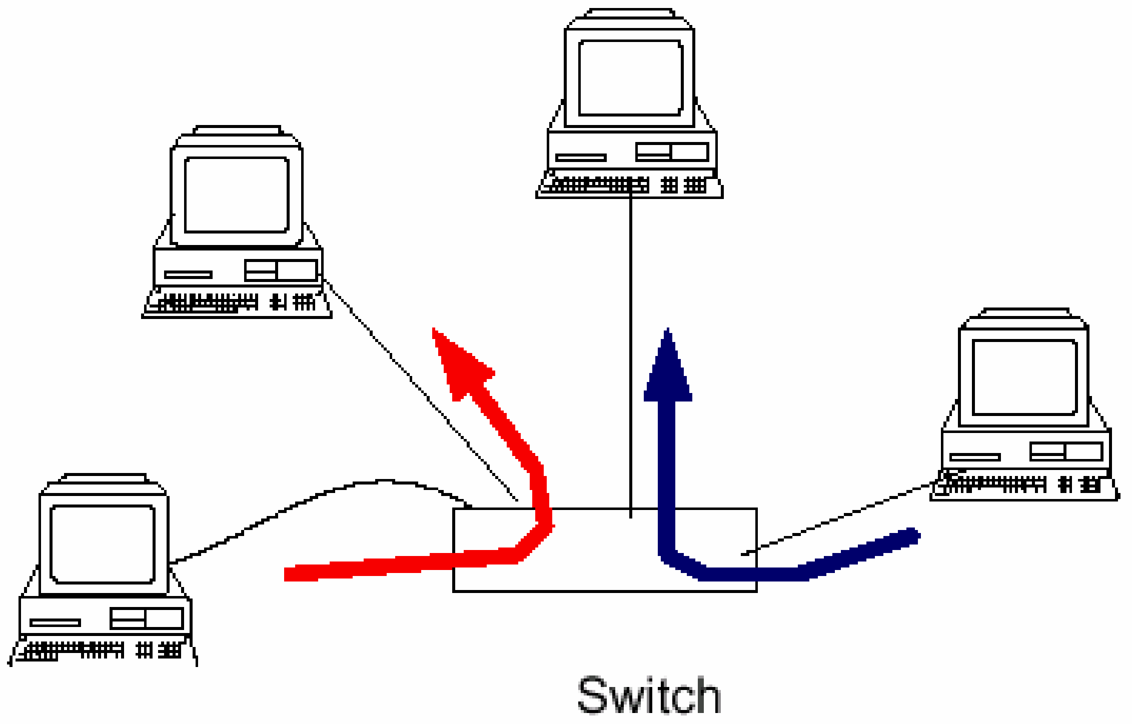


MAC Protocol: Wait until silent (carrier sense)
Transmit
If collision, wait random time & repeat

CSMA/CD

Operations

- Switch: No Collisions

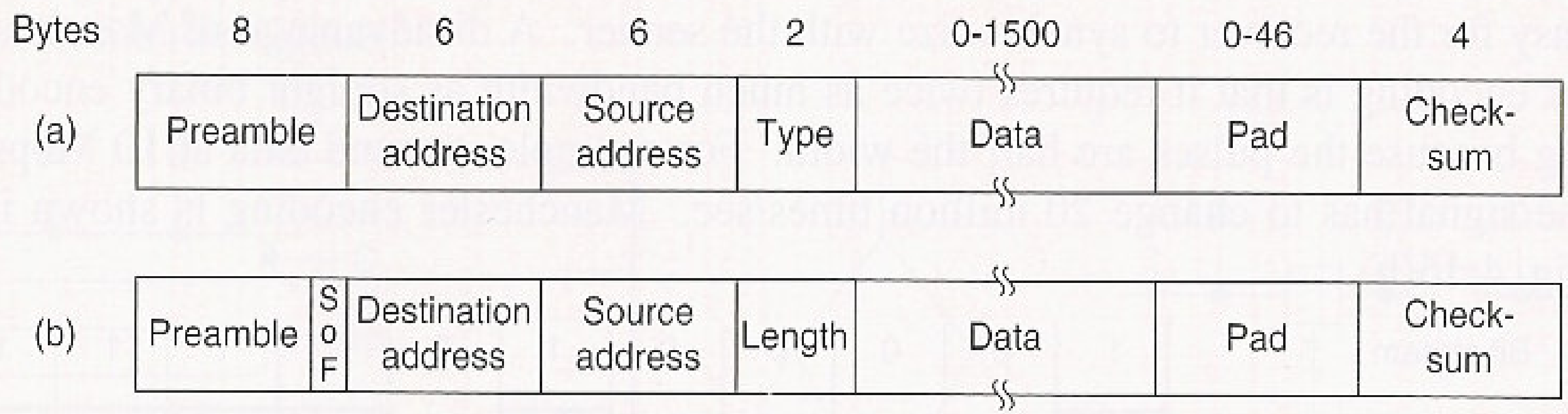


Multiple transmissions are possible
Switch stores packets that wait for same output

Physical Layer Basics

- Will discuss “classical” Ethernet primarily
- Single segments up to 500m; with up to 4 repeaters gives 2500m max length
- Baseband signals broadcast, Manchester encoding, 32-bit CRC for error detection
- Max 100 stations/segment, 1024 stations/Ethernet

Frame [Gast]



Frame formats. (a) DIX Ethernet. (b) IEEE 802.3.

- 8 byte preamble (DIX Ethernet): alternating 1/0 combination producing 10Mhz square wave [@ 10Mbps] for 6.4 μsec; used for receiver synchronization

Ethernet MAC Strategy

- Collisions *detected* within short time
- Colliding transmissions aborted, reducing channel wastage
- Easy in wired LANs: measure signal strengths, compare transmitted, received signals
- Difficult in wireless LANs

CSMA: collisions

Collisions *can* occur:

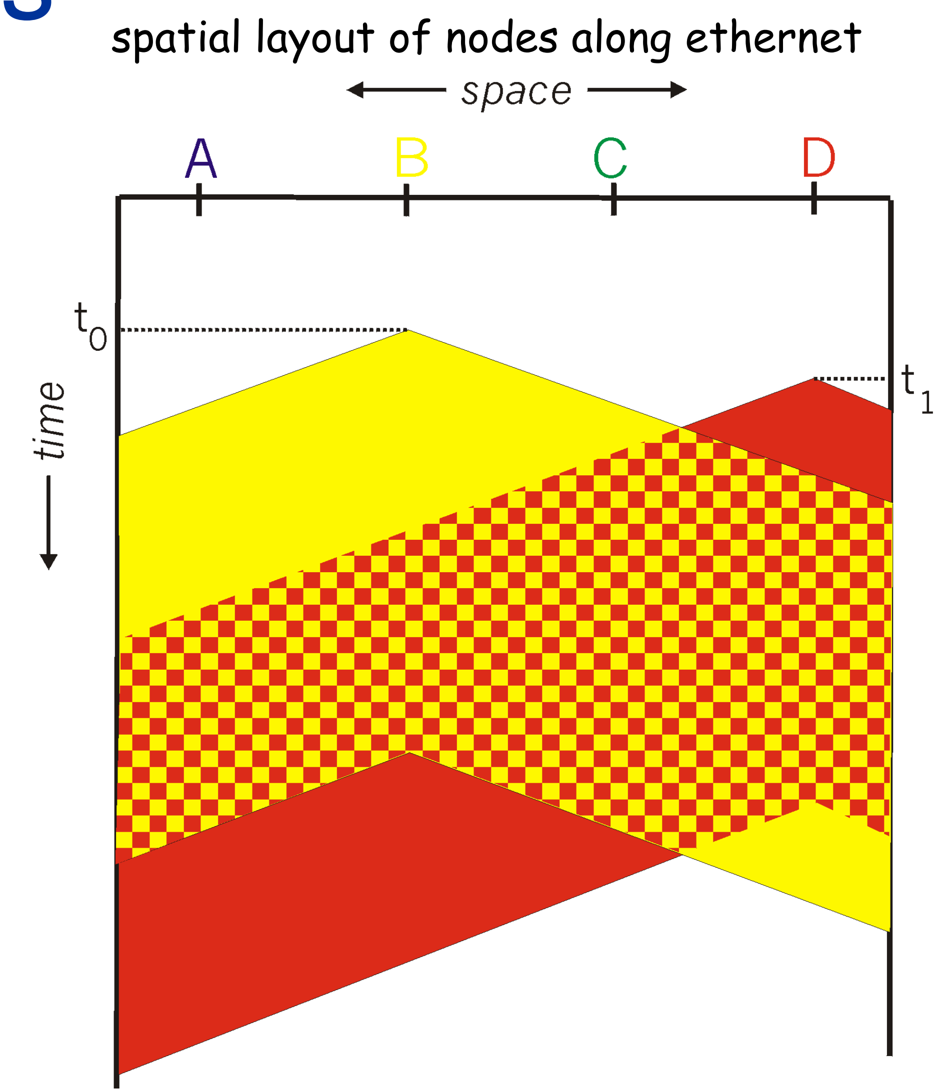
propagation delay means two nodes may not hear each other's transmission

Collision:

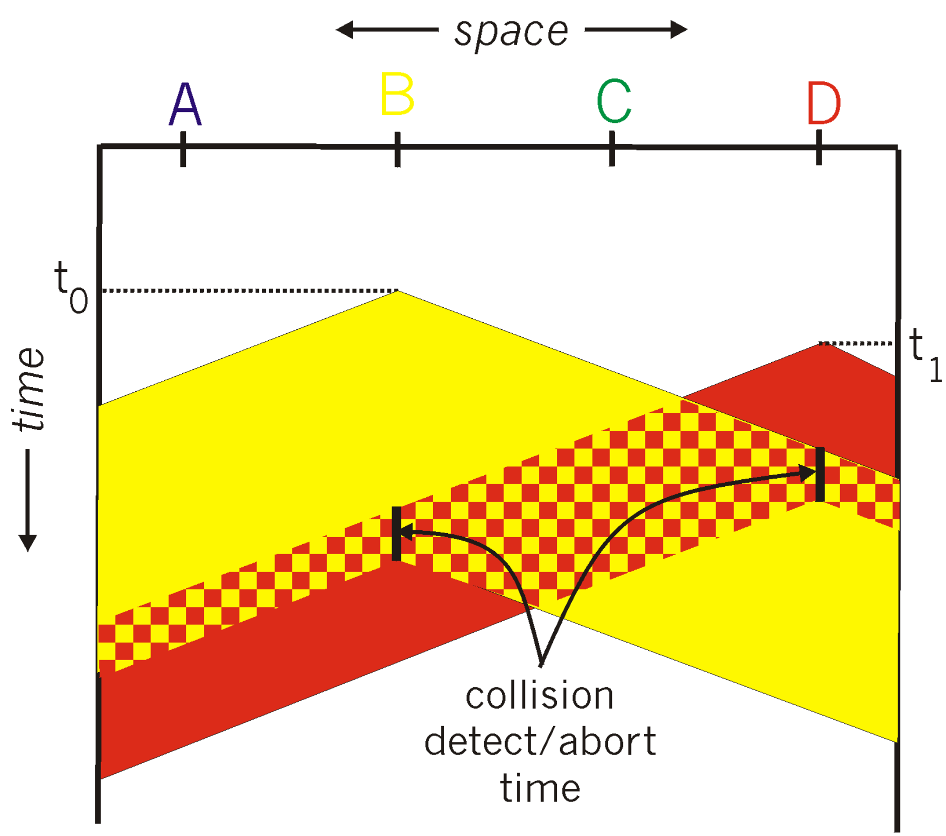
entire packet transmission time wasted

Note:

role of distance and propagation delay in determining collision prob.

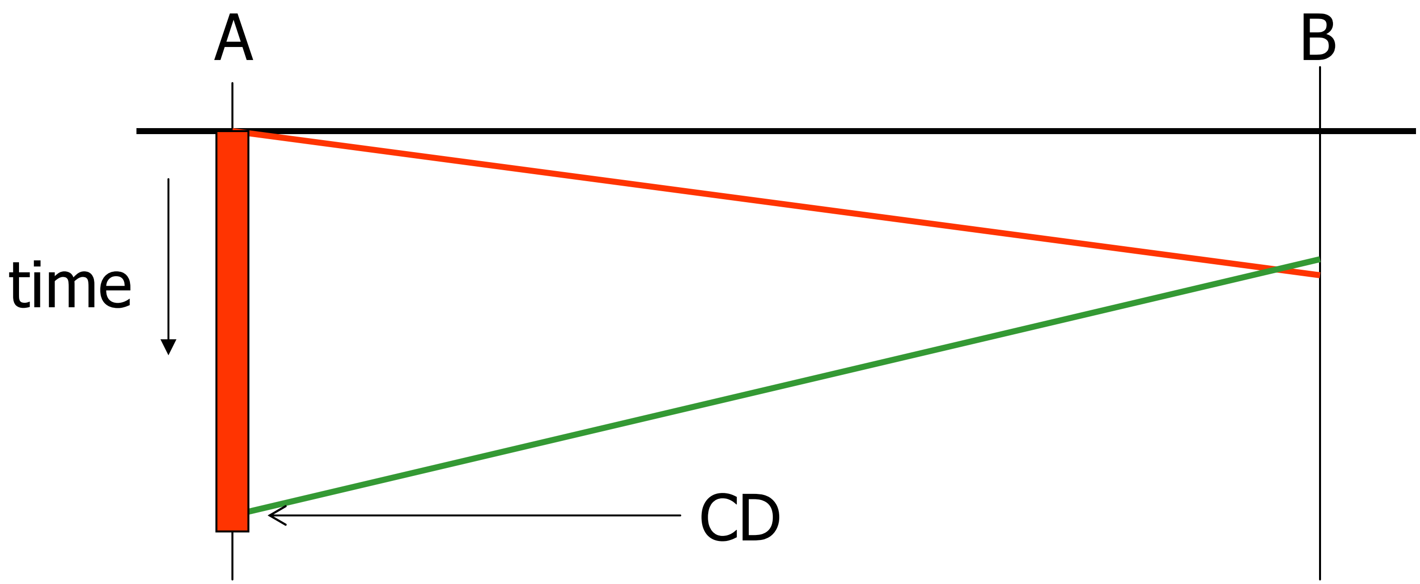


Timing



Collision Detection

- CD circuit operates by looking for voltage exceeding a transmitted voltage
- Want to ensure that a station does not complete transmission prior to 1st bit arriving at farthest-away station
- Time to CD can thus take up to 2x{max prop. delay}



Minimum Frame Size

- Speed of light is about 4μs/km in copper
- So, max Ethernet signal prop time is about 10 μsec, or 20μsec RTT
- With repeaters, etc. 802.3 requires 51μsec, corresponding to 512 bit-times
- Thus, minimum frame size is 512 bits (64 bytes); also called *slot time*

Maximum Frame Size

- 1500 byte limitation on maximum frame transmission size
- We will call this the *MTU*
- Limits maximum buffers at receiver
- Allows for other stations to send
  - Also requires 96 bit Inter-Packet-Gap (IPG)

Ethernet Operations

- When ready & line idle, await IPG (96 bit times) and send while listening (CD)
- If CD true, send max 48-bit jamming sequence and do exponential backoff
- Jamming sequence used to inform all stations that a collision has occurred

Ethernet Operations (2)

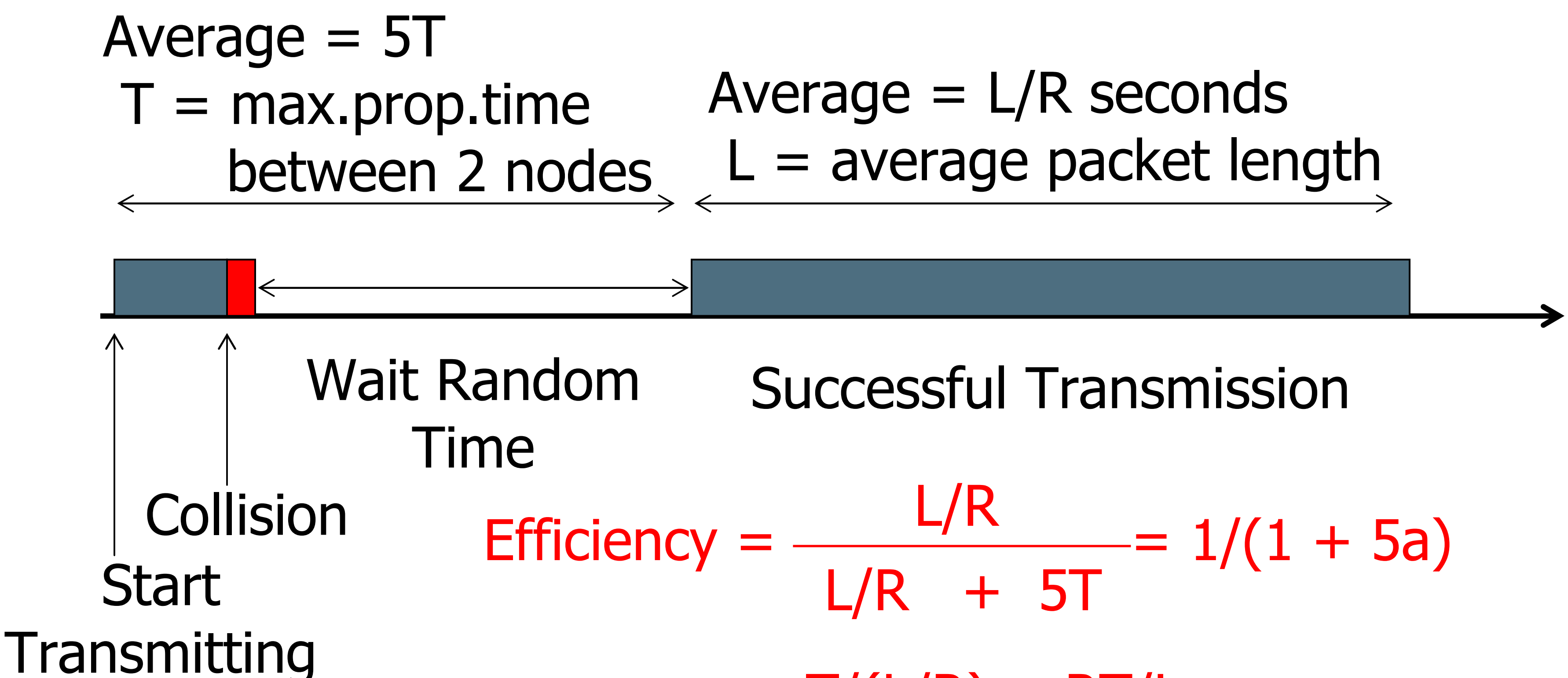
- For retransmission N (1<=N<=10)
  - choose k at random on U(0..2^N-1)
  - wait k * (51.2μsec) to retransmit
  - send on idle; repeat on another collision
  - for (11<=N<=15), use U(0..1023)
  - if N = 16, drop frame
- Longer wait implies lower priority (strategy is not “fair”)

Ethernet Capture Effect

- Given two stations A & B, unfair strategy can cause A to continue to "win"
- Assume A & B always ready to send:
  - if busy, both wait, send and collide
  - suppose A wins, B backs off
  - next time, B's chances of winning are halved

Ethernet Efficiency

- Typical Sequence of Events:



$$\text{Efficiency} = \frac{L/R}{L/R \; + \; 5T} = 1/(1 + 5a)$$

$$a = T/(L/R) = RT/L$$

Ethernet Efficiency (2)

- *a* impacts what happens during simultaneous transmission:
  - *a* small → early collision detection → Efficient
  - *a* large → late detection → Inefficient

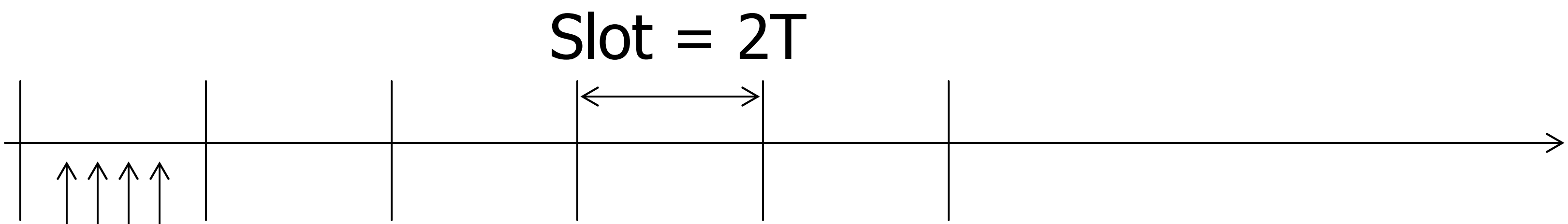
a = RT/L
eff = 1/(1 + 5a)

Example 1: 10Mbps, 1000m
=> T = (1km)(4μs/km) = 4μs; RT = 400 bits
L = 4000 bits, say
5a = 2000/4000 = 0.5 => efficiency = 66%

Example 2: 1Gbps, 200m
=> T = (0.2km)((4μs/km) = 0.8μs; RT = 800 bits
L = 4000 bits; 5a = 4000/4000 = 1 => efficiency = 50%

Ethernet Efficiency - Analysis

- Model:



N stations compete by transmitting with probability p, independently

If success => transmit L bits

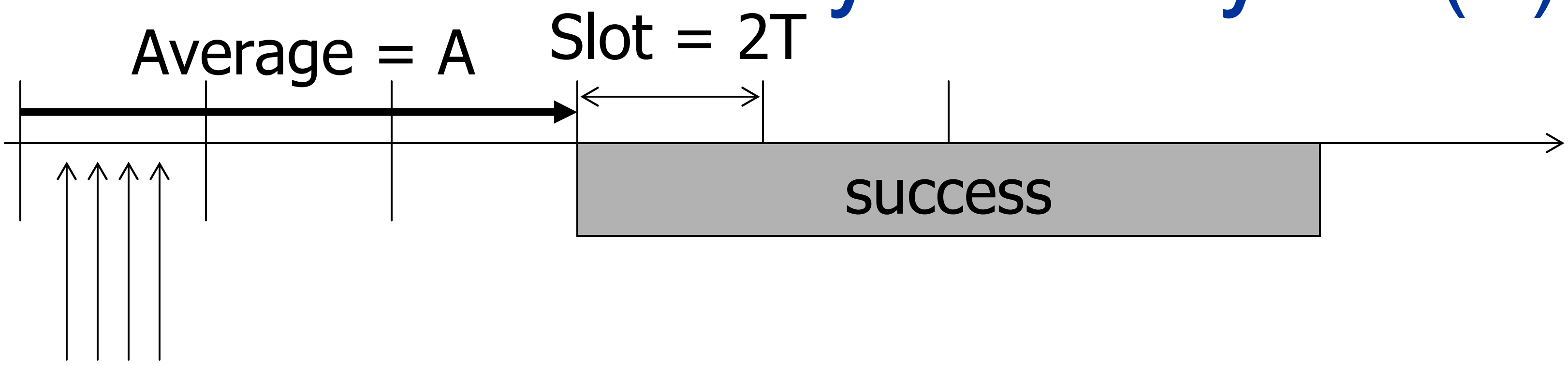
If failure (idle or collision), try next slot

P(success) = P(exactly 1 out of N transmits) = $Np(1 - p)^{N-1}$

Indeed: N possibilities of station that transmits

P(one given station transmits, others do not) = $p(1 - p)^{N-1}$

Ethernet Efficiency – Analysis (2)



$P(\text{success}) = Np(1 - p)^{N-1}$

Assume backoff algorithm results in best p = 1/N
=> $P(\text{success}) \approx 1/e = 0.36$

Average time until success:

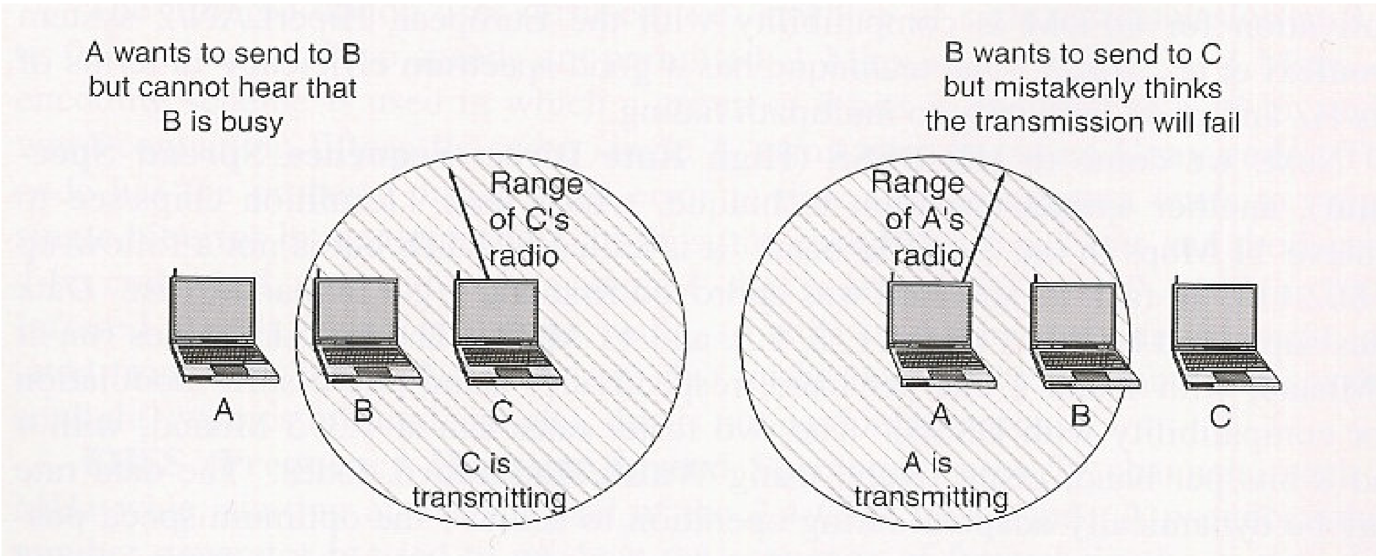
$$A = 0.36x0 + 0.64x(2T + A)$$

=> $A = 1.28T/0.36 = 3.5T$

In practice, backoff not quite optimal => 5T

802.11 MAC: Problems

- Hidden Terminal and Exposed Terminal problems



Solution for Hidden Terminal Problem: RTS/CTS Clearing

- RTS/CTS Clearing
- Used for frames larger than RTS/CTS threshold
- Tradeoff between overhead and retransmission costs