Lecture 3-2

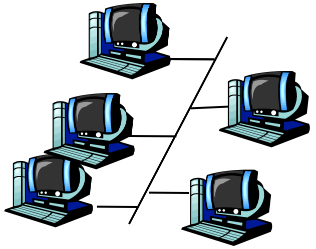
MAC, Random Access Protocols

ALOHA, CSMA, CSMA/CD

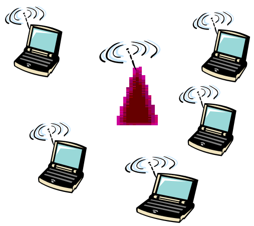
Multiple Access Links and Protocols

Two types of "links":

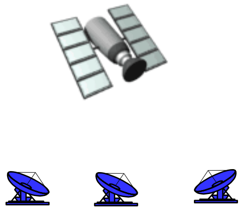
- point-to-point
  - PPP for dial-up access
  - point-to-point link between Ethernet switch and host
- broadcast (shared wire or medium)
  - old-fashioned Ethernet
  - upstream HFC
  - 802.11 wireless LAN



shared wire (e.g., cabled Ethernet)



shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

Multiple Access protocols

- single shared broadcast channel
- two or more simultaneous transmissions by nodes: interference
  - collision if node receives two or more signals at the same time

multiple access protocol

- distributed algorithm that determines how nodes share channel, i.e., determine when node can transmit
- communication about channel sharing must use channel itself!
  - no out-of-band channel for coordination

Ideal Multiple Access Protocol

Broadcast channel of rate R bps

1. when one node wants to transmit, it can send at rate R.
2. when M nodes want to transmit, each can send at average rate R/M
3. fully decentralized:
   - no special node to coordinate transmissions
   - no synchronization of clocks, slots
4. simple

MAC Protocols: a taxonomy

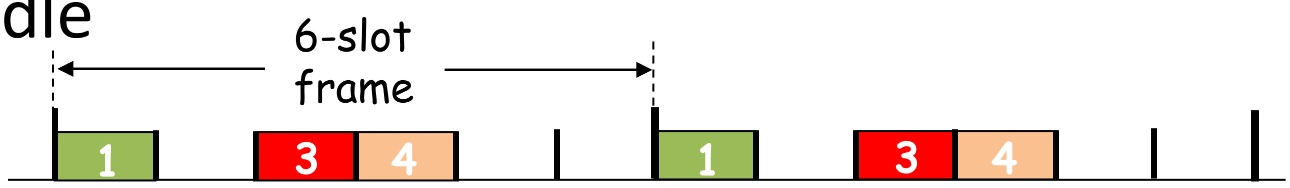
Three broad classes:

- Channel Partitioning
  - divide channel into smaller “pieces” (time slots, frequency, code)
  - allocate piece to node for exclusive use
- Random Access
  - channel not divided, allow collisions
  - “recover” from collisions
- “Taking turns”
  - nodes take turns, but nodes with more to send can take longer turns

Channel Partitioning MAC protocols: TDMA

TDMA: time division multiple access

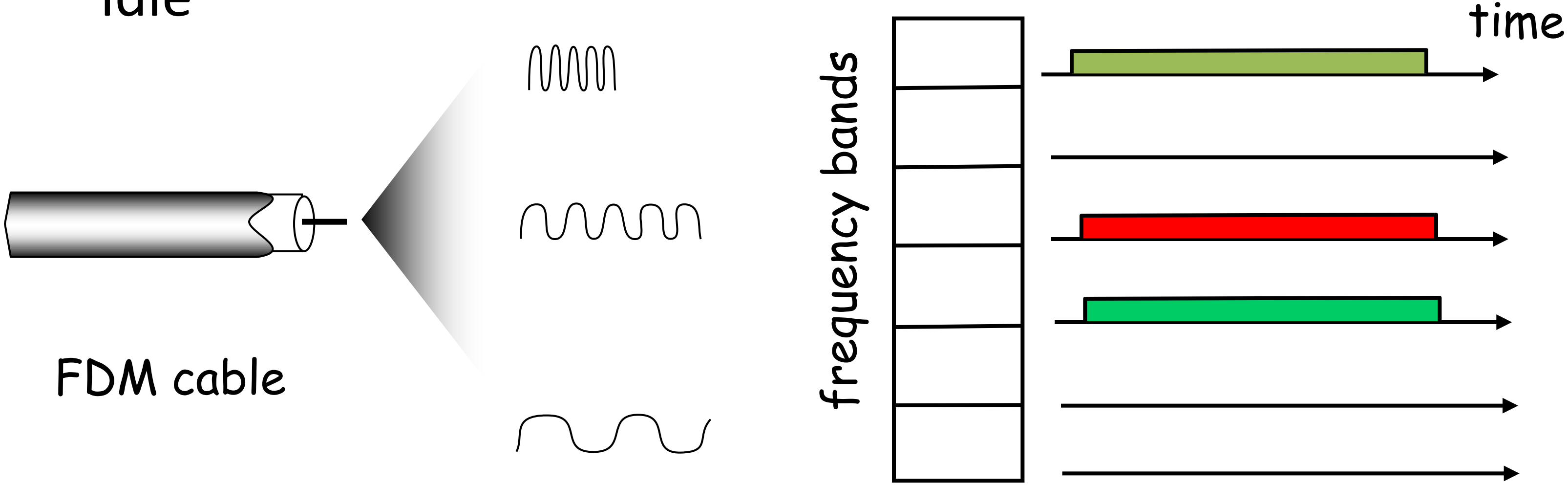
- access to channel in "rounds"
- each station gets fixed length slot (length = pkt trans time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle



Channel Partitioning MAC protocols: FDMA

FDMA: frequency division multiple access

- channel spectrum divided into frequency bands
- each station assigned fixed frequency band
- unused transmission time in frequency bands go idle
- example: 6-station LAN, 1,3,4 have pkt, frequency bands 2,5,6 idle

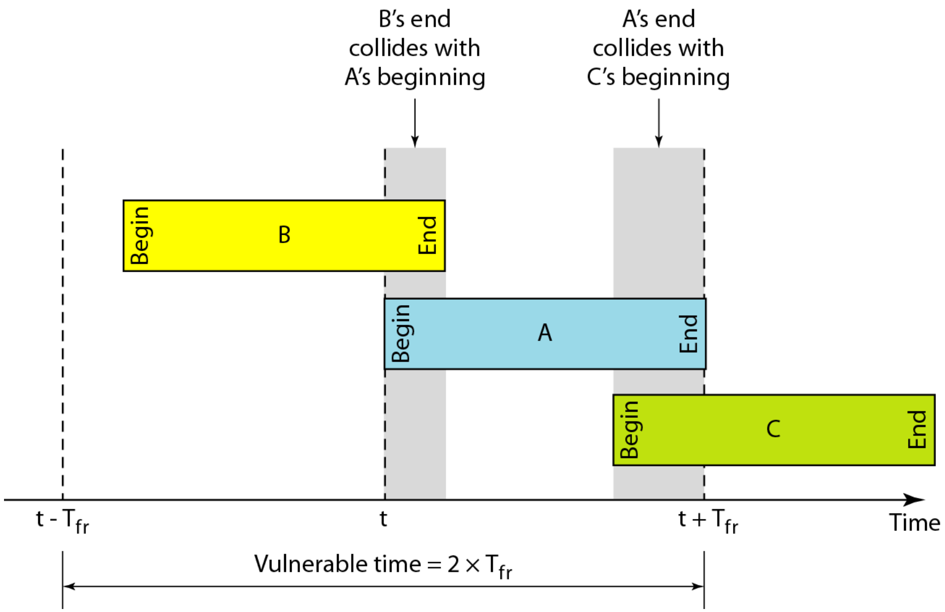


Random Access Protocols

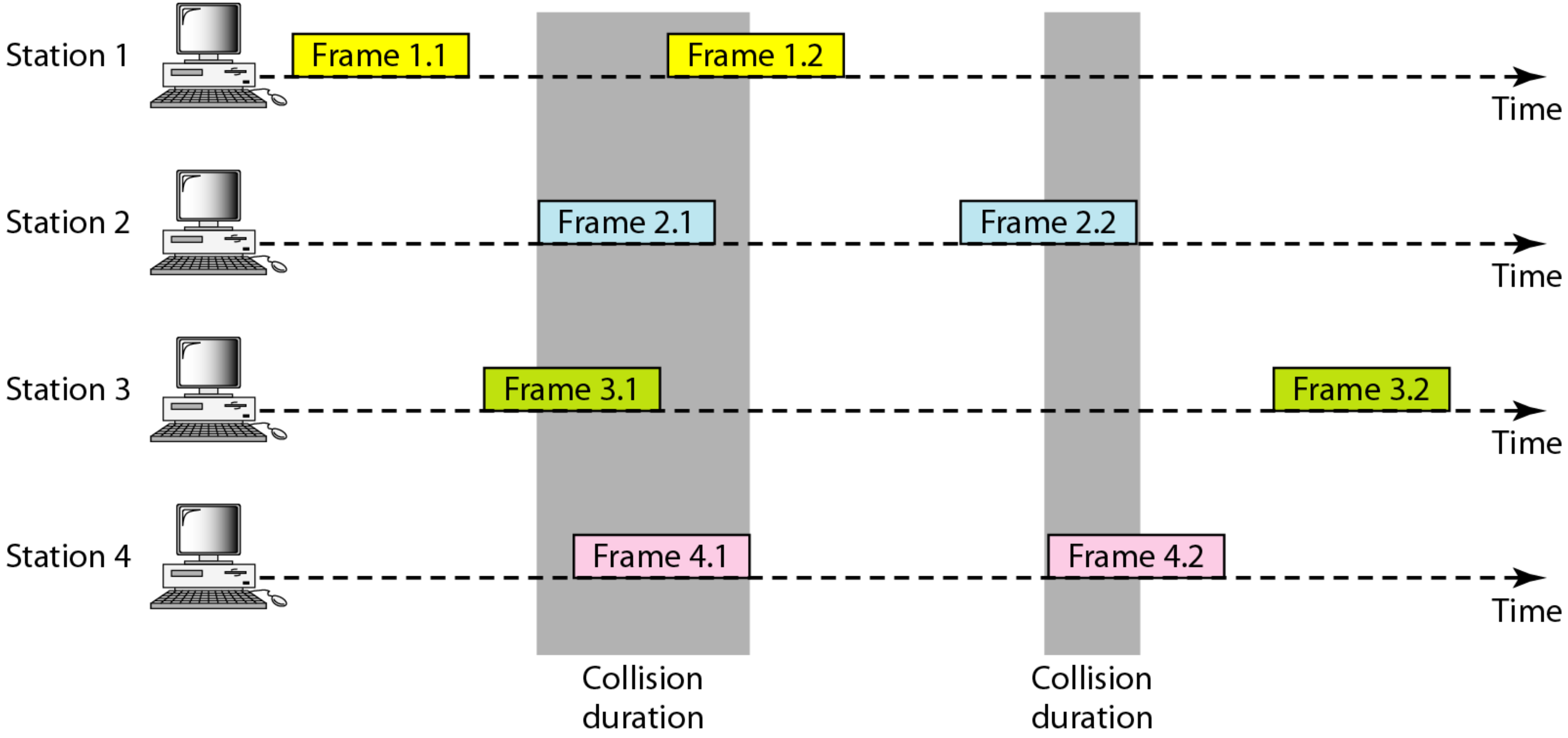
- When node has packet to send
  - transmit at full channel data rate R.
  - no *a priori* coordination among nodes
- two or more transmitting nodes "collision",
- random access MAC protocol specifies:
  - how to detect collisions
  - how to recover from collisions (e.g., via delayed retransmissions)
- Examples of random access MAC protocols:
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA

Pure ALOHA

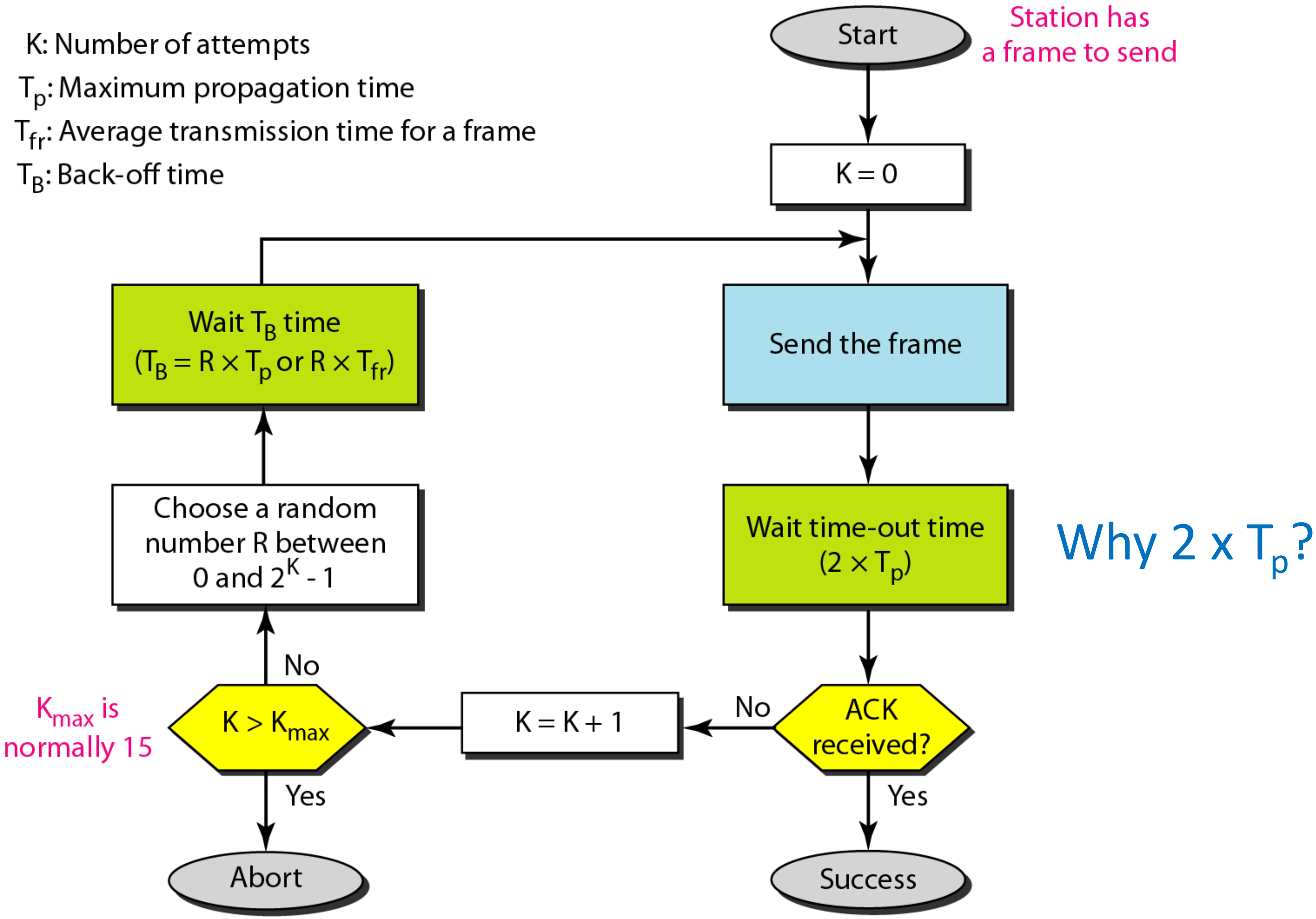
- Unslotted Aloha: simpler, no synchronization
- When frame first arrives, transmit immediately
- Collision probability increases:
  - A frame sent at time t_0 collides with other frames sent between $[t_0-T_{fr}, t_0+T_{fr}]$



Frames in a Pure ALOHA Network



Procedure for pure ALOHA protocol



Pure ALOHA Efficiency

P(success by given node)

= P(node transmits) ·

P(no other node transmits in [p_0-1,p_0] ·

P(no other node transmits in [$p_{0,}p_0$+1]

= $p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1}$

= $p \cdot (1-p)^{2(N-1)}$

... choosing optimum p and then letting N -> infinity ...

= 1/(2e) = .18

Slotted ALOHA

Assumptions:

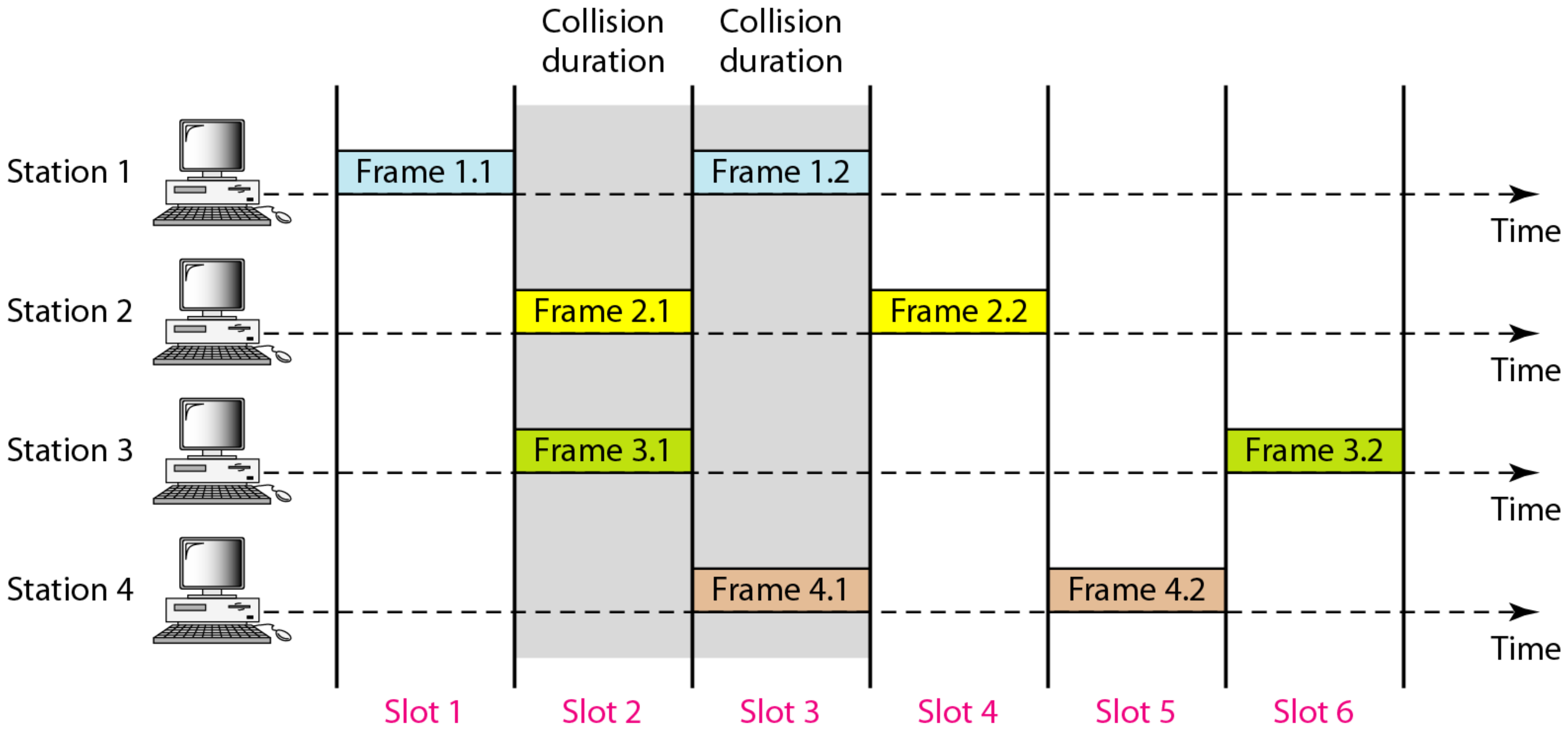
- All frames of the same size
- Time divided into equal size slots (time to transmit 1 frame)
- Nodes start to transmit only at the beginning of a slot
- Nodes are synchronized
- If 2 or more nodes transmit in the same slot, all nodes can detect a collision

Slotted ALOHA

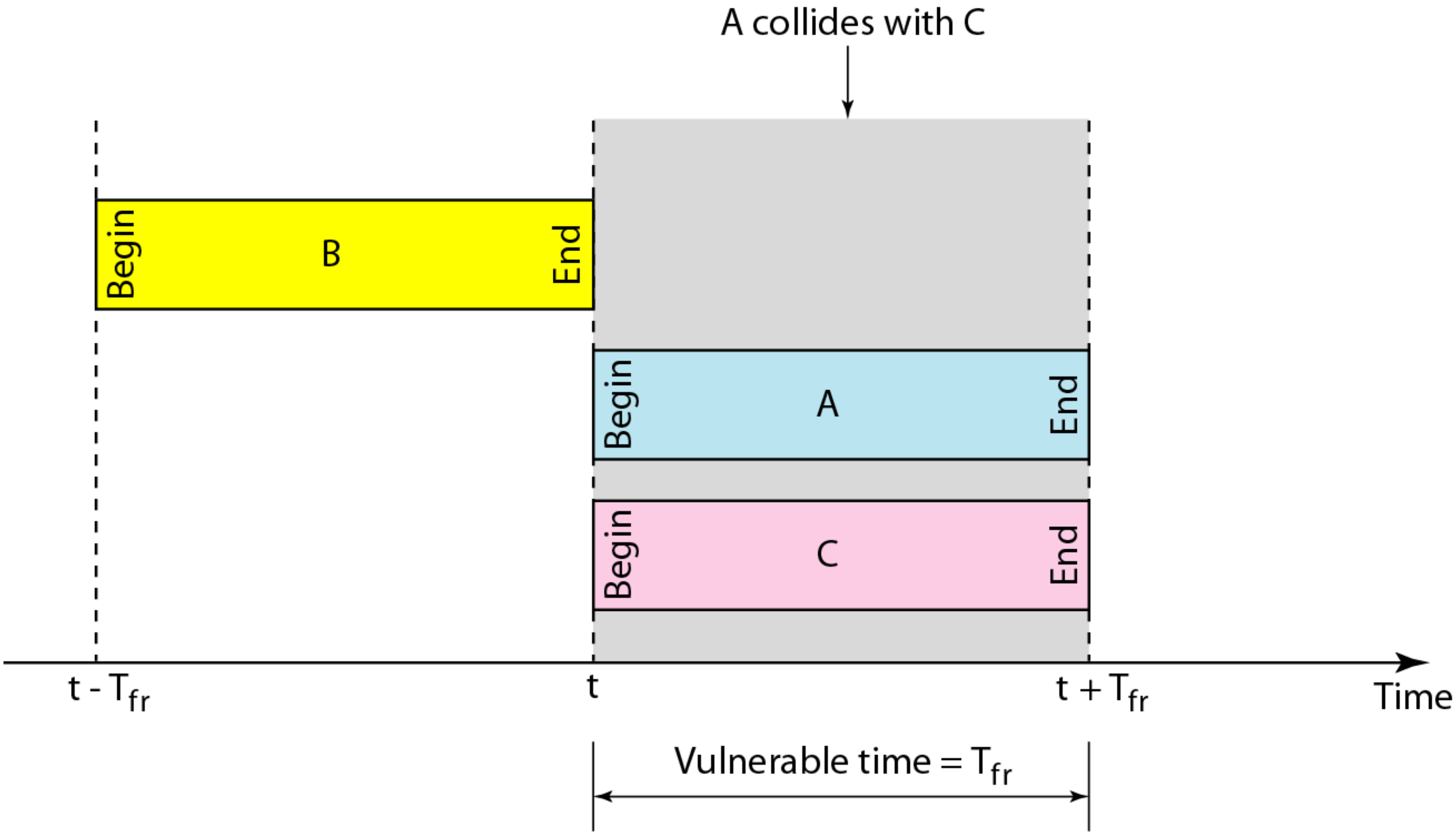
Operation:

- When a node obtains a fresh frame, transmits in the next slot
- *if no collision:* node can send a new frame in the next slot
- *if collision:* node retransmits the frame in each subsequent slot with prob. p until success

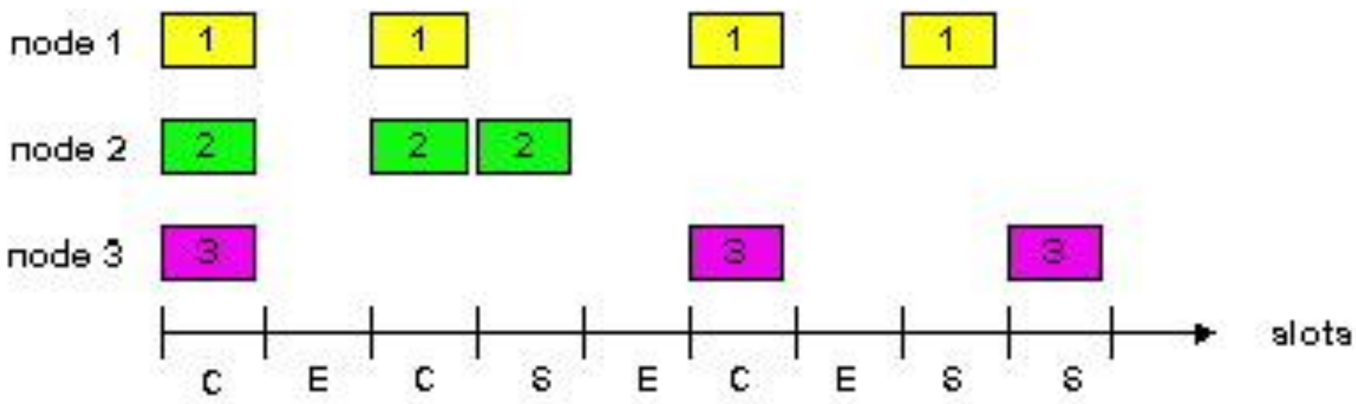
Frames in a Slotted ALOHA Network



Vulnerable Time for Slotted ALOHA



Slotted ALOHA



Pros

- A single active node can continuously transmit at a full rate of a channel
- Highly decentralized: only slots in nodes need to be in sync
- Simple

Cons

- Collisions, wasting slots
- Idle slots
- Nodes may be able to detect collision in less than time to transmit packet
- Clock synchronization

Slotted Aloha efficiency

Efficiency : long-run fraction of successful slots (many nodes, all with many frames to send)

- *suppose:* N nodes with many frames to send, each transmits in slot with probability *p*
- prob that given node has success in a slot = $p(1-p)^{N-1}$
- prob that *any* node has a success = $Np(1-p)^{N-1}$

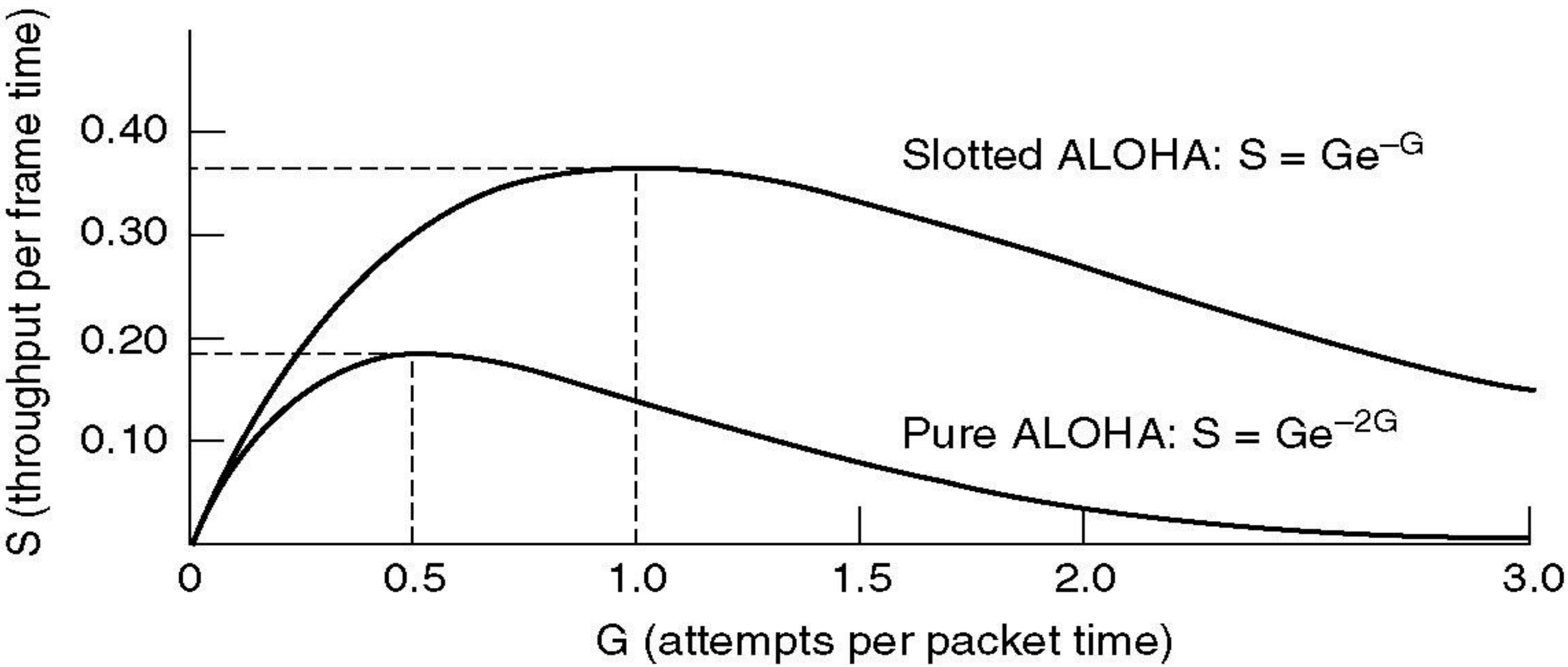
- max efficiency: find p* that maximizes $Np(1-p)^{N-1}$
- for many nodes, take limit of $Np^*(1-p^*)^{N-1}$ as N goes to infinity, gives:

Max efficiency = 1/e = .37

At best: channel used for useful transmissions 37% of time!

!

Pure vs. Slotted ALOHA Performance



CSMA (Carrier Sense Multiple Access)

CSMA: listen before transmit:

If channel sensed idle: transmit entire frame

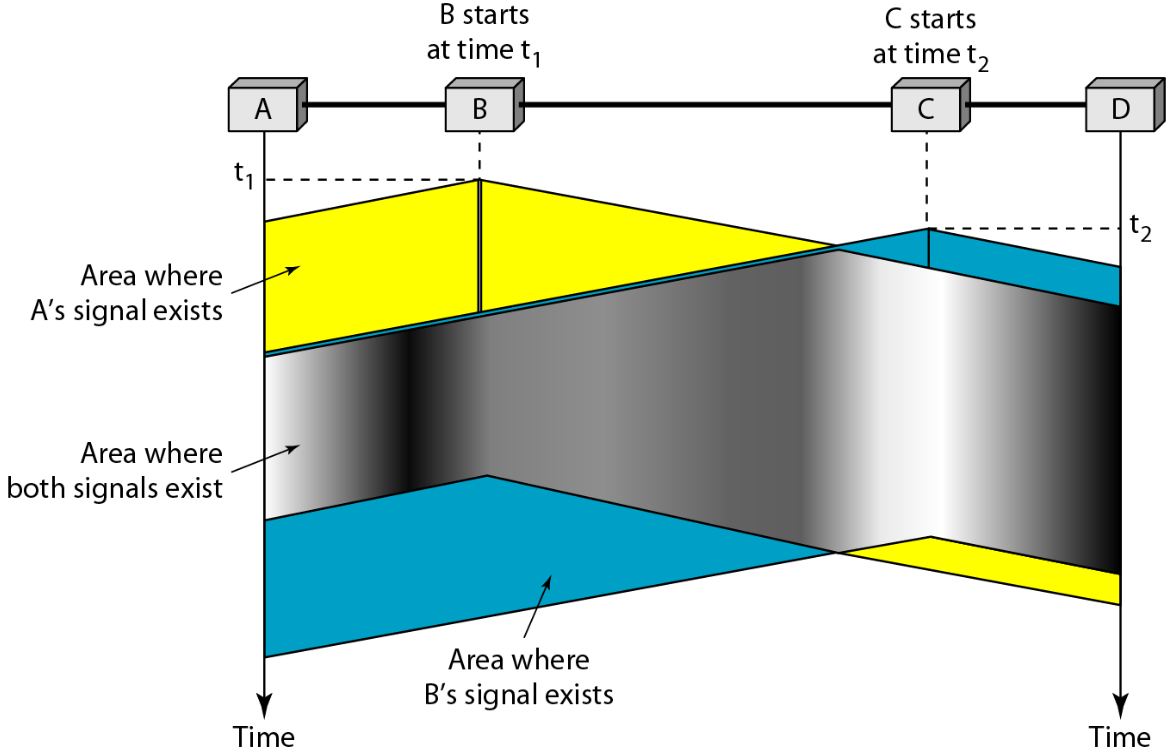
- If channel sensed busy, defer transmission

- human analogy: don't interrupt others!

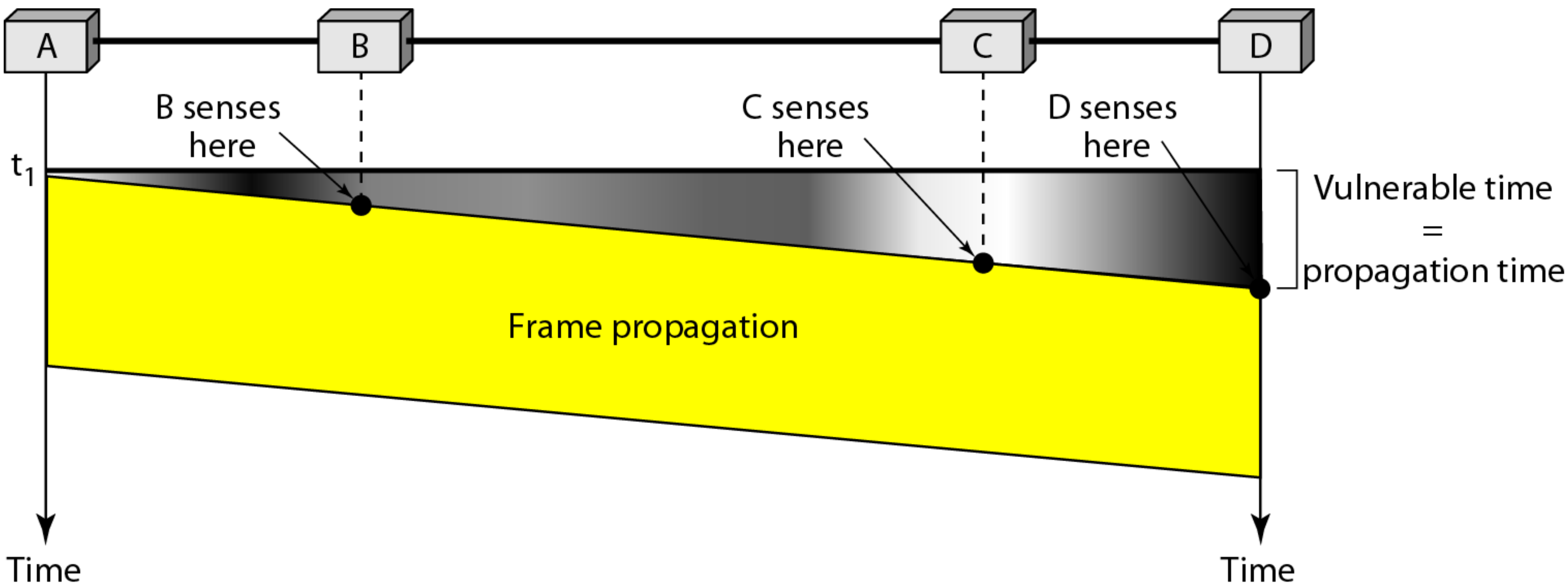
CSMA collisions

collisions *can* still occur: propagation delay means two nodes may not hear each other's transmission

collision: entire packet transmission time wasted



Vulnerable time in CSMA



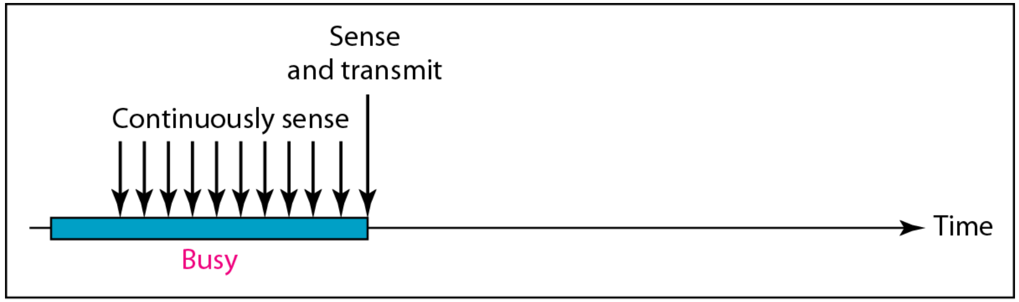
note: role of distance & propagation delay in determining collision probability

Characteristics of CSMA Protocols

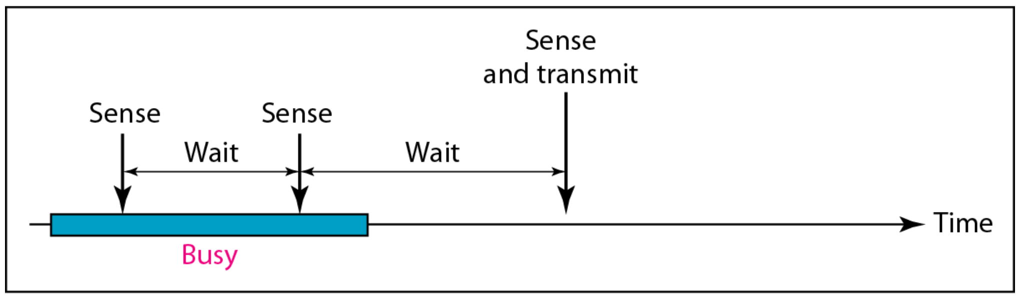
Characteristics of the three basic CSMA protocols

CSMA Protocol	Characteristics
Nonpersistent	If medium is idle, transmit. If medium is busy, wait random amount of time and resense channel.
1-persistent	If medium is idle, transmit. If medium is busy, continue listening until channel is idle; than transmit immediately.
p-persistent	If medium is idle, transmit with probability *p*. If medium is busy, continue listening until channel is idle; then transmit with probability *p*.

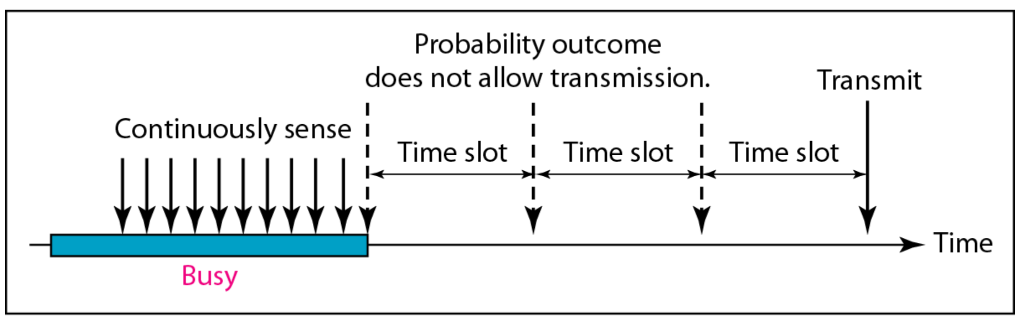
Three Persistence Methods



a. 1-persistent

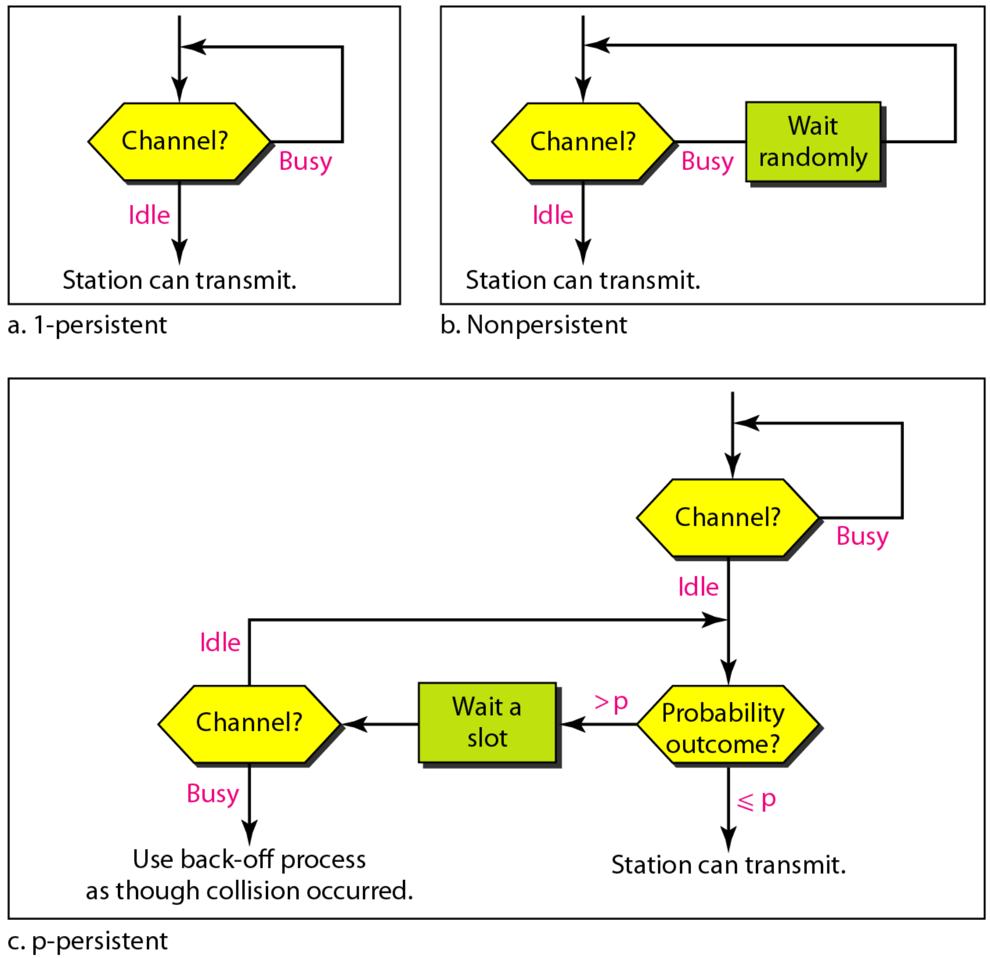


b. Nonpersistent



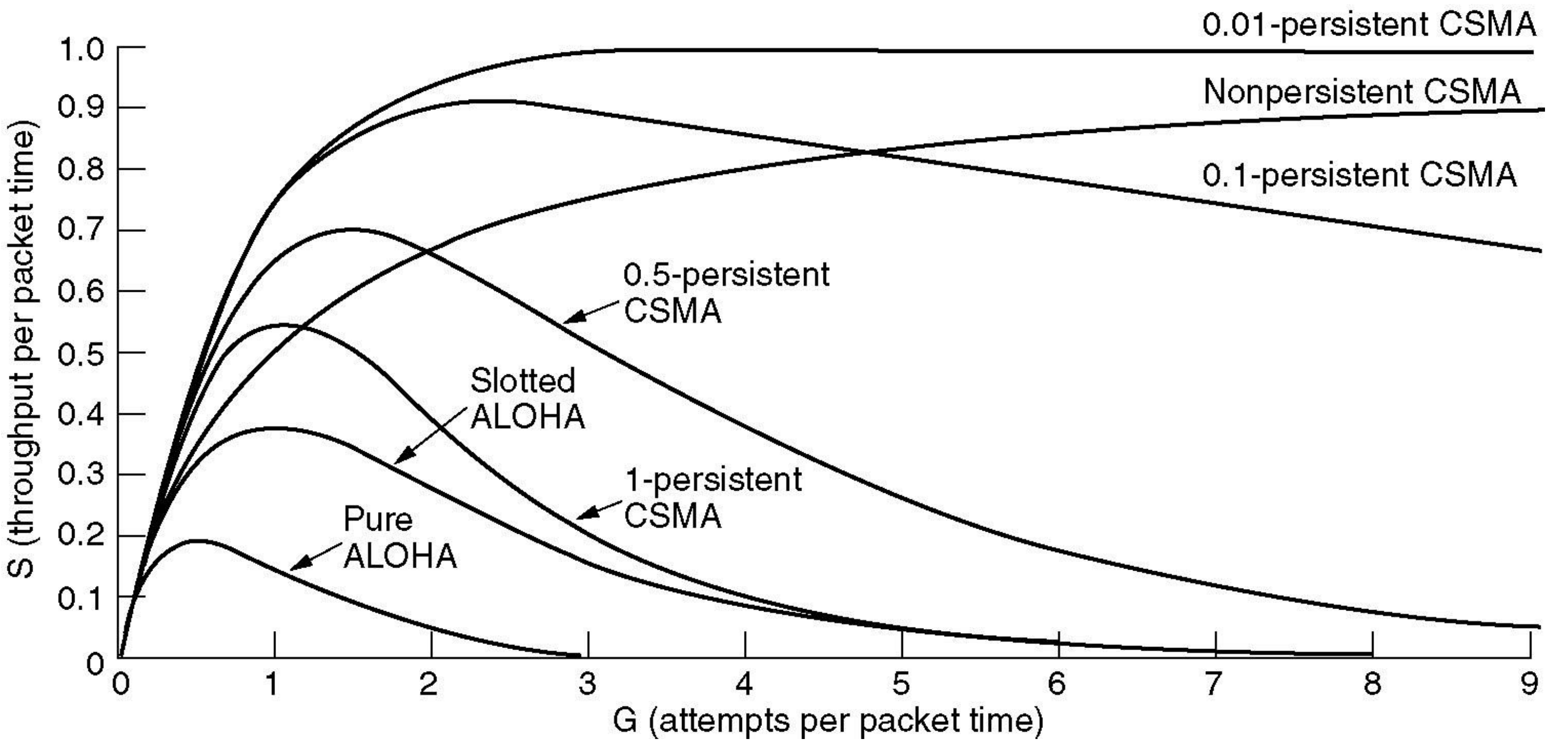
c. p-persistent

Flow diagram for three persistence methods



Persistent and Nonpersistent CSMA

Comparison of the channel utilization versus load for various random access protocols.



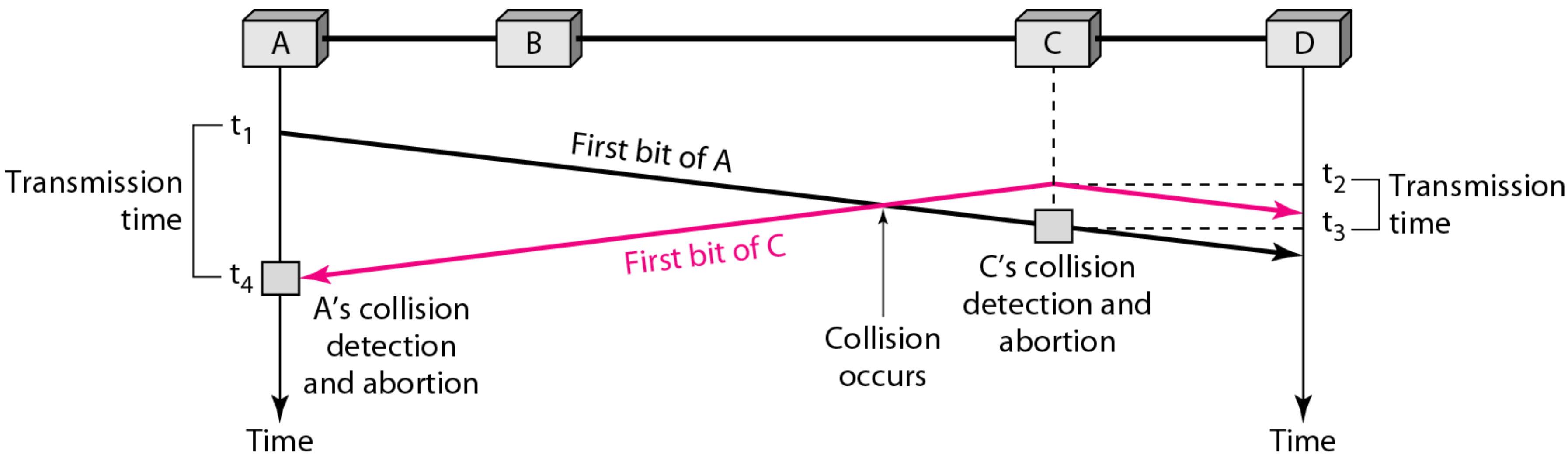
CSMA/CD (Collision Detection)

CSMA/CD: carrier sensing, deferral as in CSMA

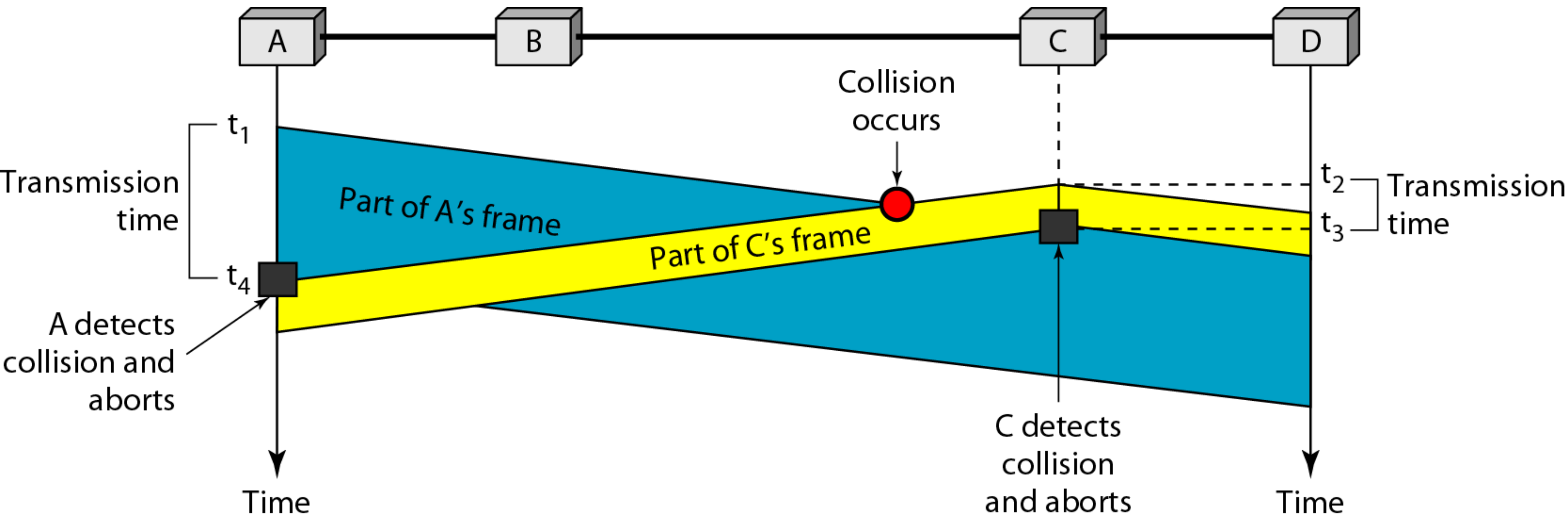
- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage

- collision detection:
  - easy in wired LANs: measure signal strengths, compare transmitted, received signals
  - difficult in wireless LANs: received signal strength overwhelmed by local transmission strength
- human analogy: the polite conversationalist

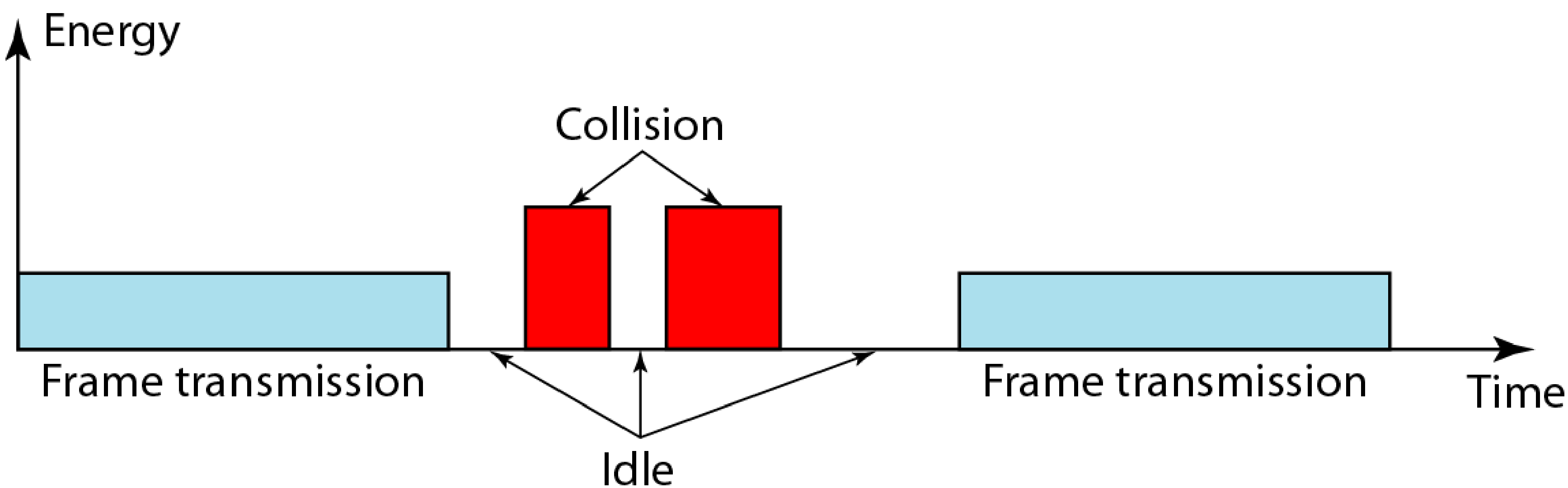
Collision of the first bit in CSMA/CD



Collision and abortion in CSMA/CD



Energy level during transmission, idleness, or collision



“Taking Turns” MAC protocols

channel partitioning MAC protocols:

- share channel *efficiently* and *fairly* at high load
- inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node!

Random access MAC protocols

- efficient at low load: single node can fully utilize channel
- high load: collision overhead

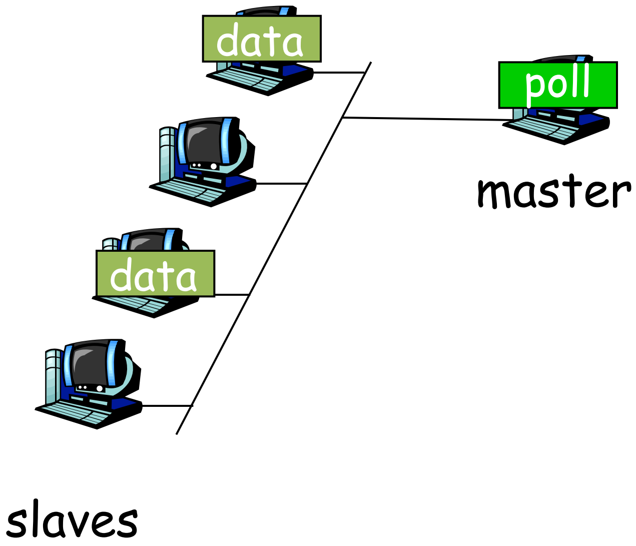
“taking turns” protocols

- look for best of both worlds!

"Taking Turns" MAC protocols

Polling:

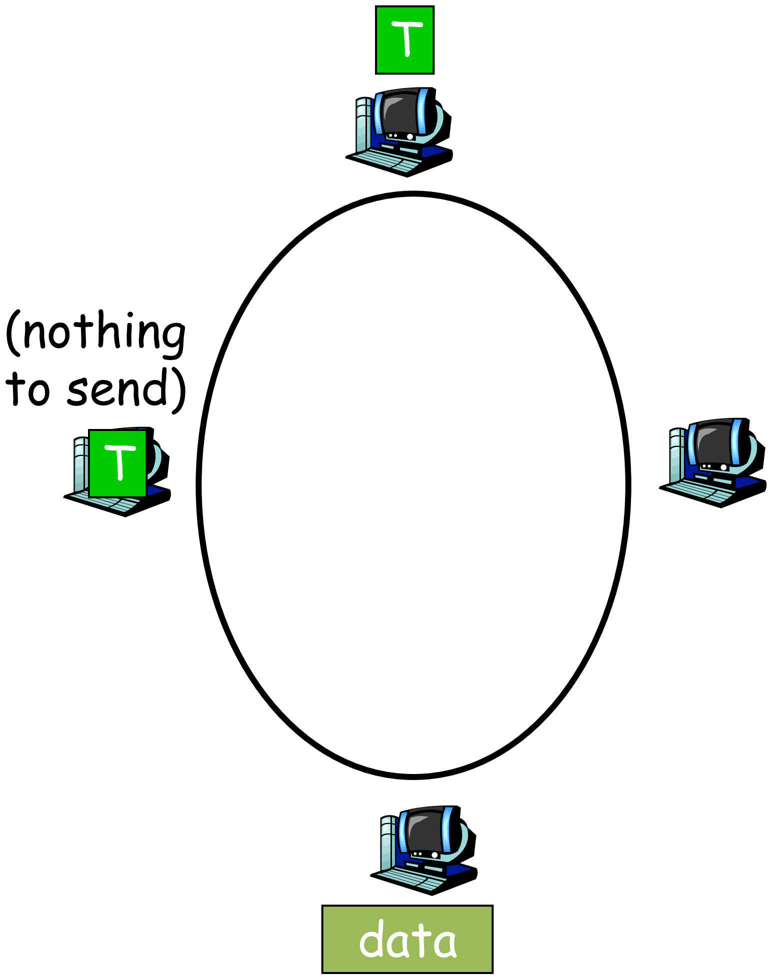
- master node "invites" slave nodes to transmit in turn
- typically used with "dumb" slave devices
- concerns:
  - polling overhead
  - latency
  - single point of failure (master)



"Taking Turns" MAC protocols

Token passing:

- control **token** passed from one node to next sequentially.
- token message
- concerns:
  - token overhead
  - latency
  - single point of failure (token)



Summary of MAC protocols

- *channel partitioning,* by time, frequency or code
  - Time Division, Frequency Division
- *random access* (dynamic),
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - carrier sensing: easy in some technologies (wire), hard in others (wireless)
  - CSMA/CD used in Ethernet
  - CSMA/CA used in 802.11
- *taking turns*
  - polling from central site, token passing
  - Bluetooth, FDDI, IBM Token Ring