Digital Communication in the Modern World

Data Link Layer: Multi Access Protocols

http://www.cs.huji.ac.il/~com1

com1@cs.huji.ac.il

Some of the slides have been borrowed from:
Computer Networking: A Top Down Approach Featuring the Internet,
3rd edition.
Jim Kurose, Keith Ross
Addison-Wesley, July 2004.

Intro to the Data Link Layer

Travel analogy

- University sends you to a conference in India
  - taxi: Jerusalem to Ben-Gurion
  - plane: Ben-Gurion to Mumbai
  - plane: Mumbai to Chennai
  - rickshaw: Chennai to conference center
- student = datagram
- university = sending node (application layer)
- travel agent = routing algorithm (network layer)
- transportation mode = data link layer protocol, i.e. running an airline is different than a rickshaw
- travel segment = communication link (physical layer)

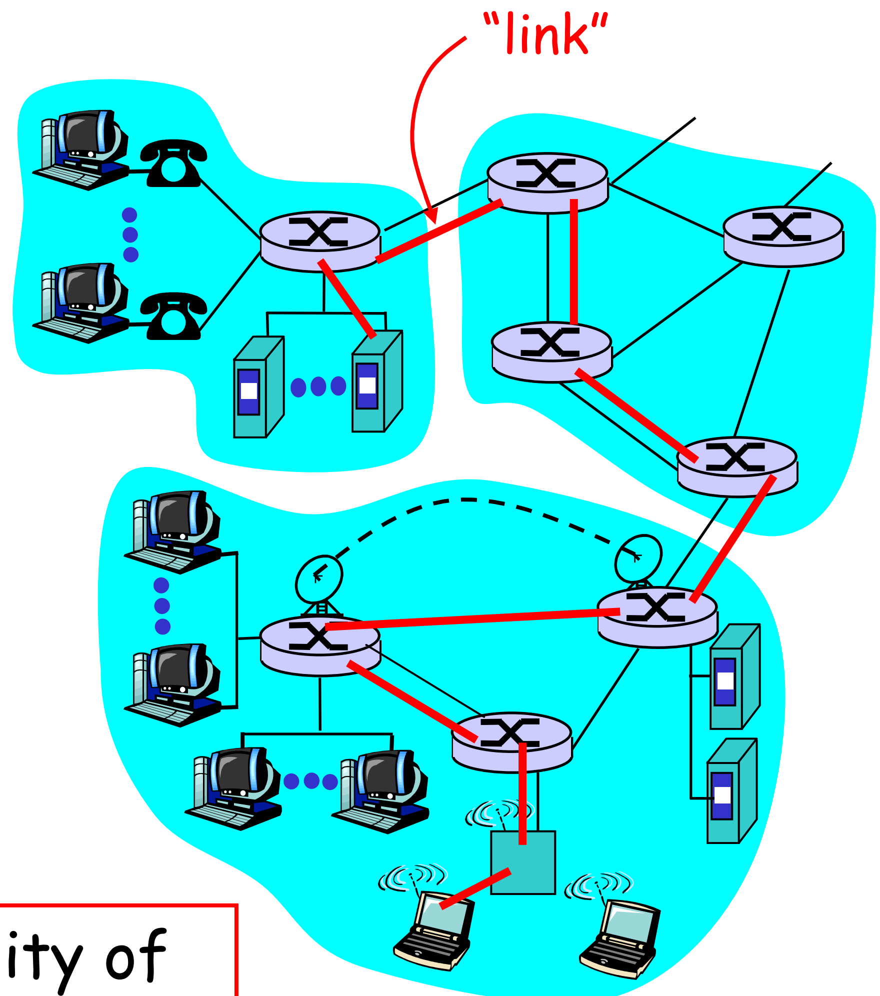
Link Layer

- Introduction and services
- Error detection and correction
- Multiple access protocols
- Link-Layer Addressing
- Ethernet
- Hubs and switches
- PPP

Link Layer

Some terminology:

- hosts and routers are **nodes**
- communication channels that connect adjacent nodes along communication path are **links**
  - wired links
  - wireless links
  - LANs
- layer-2 packet is a **frame**, encapsulates datagram



data-link layer has responsibility of transferring datagram from one node to adjacent node over a single link

Link layer

- In the Link layer there are different link types and they have different protocols
- Frame transferred by different link protocols over different links (decided by the network layer):
  - e.g., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- Each link protocol provides different services
  - e.g., may or may not provide rdt over link; flow control; etc.
- Link types are grouped into:
  - point-to-point (e.g. between routers, host to ISP, etc.)
  - Broadcast (LAN, wireless)

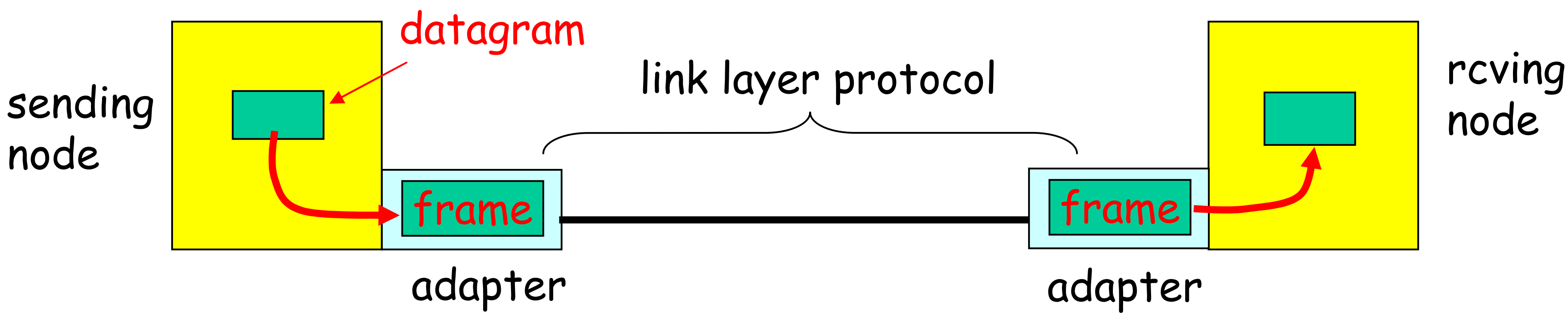
Link Layer Services

- Framing, link access:
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - "MAC" addresses used in frame headers to identify source, dest
    - different from IP address!
- Reliable delivery between adjacent nodes
  - similar service as in the transport layer
  - seldom used on low bit-error link (fiber, some twisted pair)
  - wireless links: high error rates
    - Q: why both link-level and end-to-end reliability?

Link Layer Services (more)

- *Flow Control:*
  - pacing between adjacent sending and receiving nodes
- *Error Detection*: (e.g. parity check, CRC)
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame
- Error Correction: (e.g. two Dimensional Bit Parity)
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission
- *Half-duplex and full-duplex*
  - with half duplex, nodes at both ends of link can transmit, but not at same time

Adaptors (aka NICs)

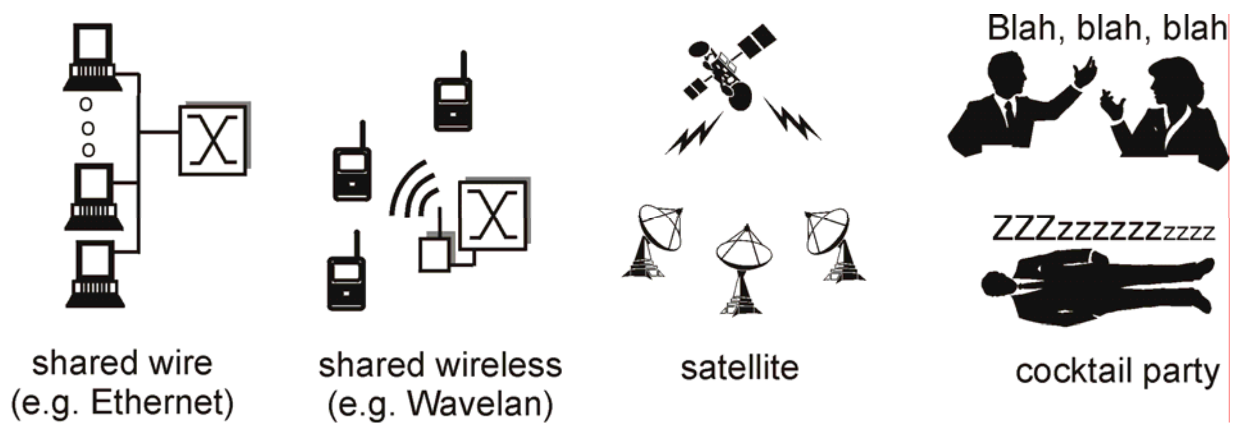


- link layer implemented in "adaptor"
  - Ethernet card, PCMCI card, 802.11 card
- sending side:
  - encapsulates datagram in a frame
  - adds error checking bits, rdt, flow control, etc.
- receiving side
  - looks for errors, rdt, flow control, etc
  - extracts datagram, passes to receiving node

Multiple Access Links and Protocols

Two types of "links":

- point-to-point <= *Single access*
  - PPP for dial-up access (between host and ISP)
  - point-to-point link between Ethernet switch and host
- broadcast (shared wire or medium) <= *Multiple Access*
  - traditional Ethernet
  - 802.11 wireless LAN



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
- "*Cocktail Party effect*" ("popolitika behavior")

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 bandwidth R bps

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

MAC Protocols: a taxonomy

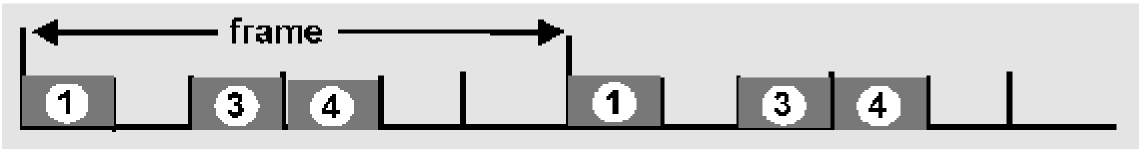
Three broad classes:

- 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
  - try to maximize probability that "everything will be ok"
- "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

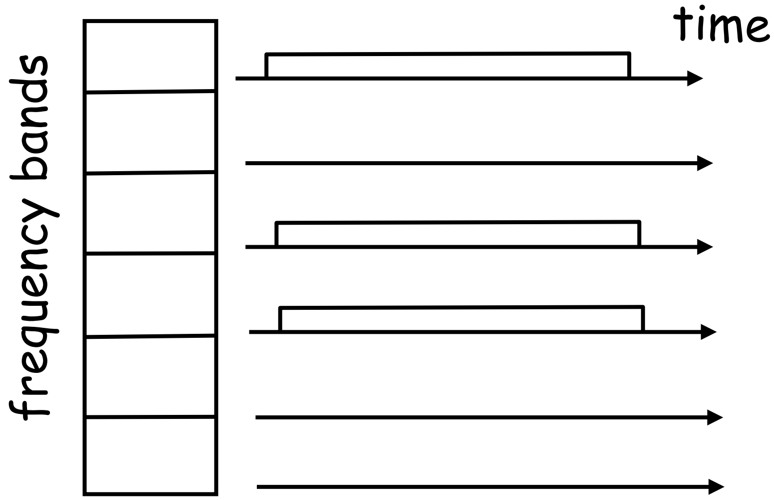
- access to channel in "rounds"
- each station gets fixed length slot in each round (length = packet transmission time)
- unused slots go idle
- example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle



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



Code Division Multiple Access (CDMA)

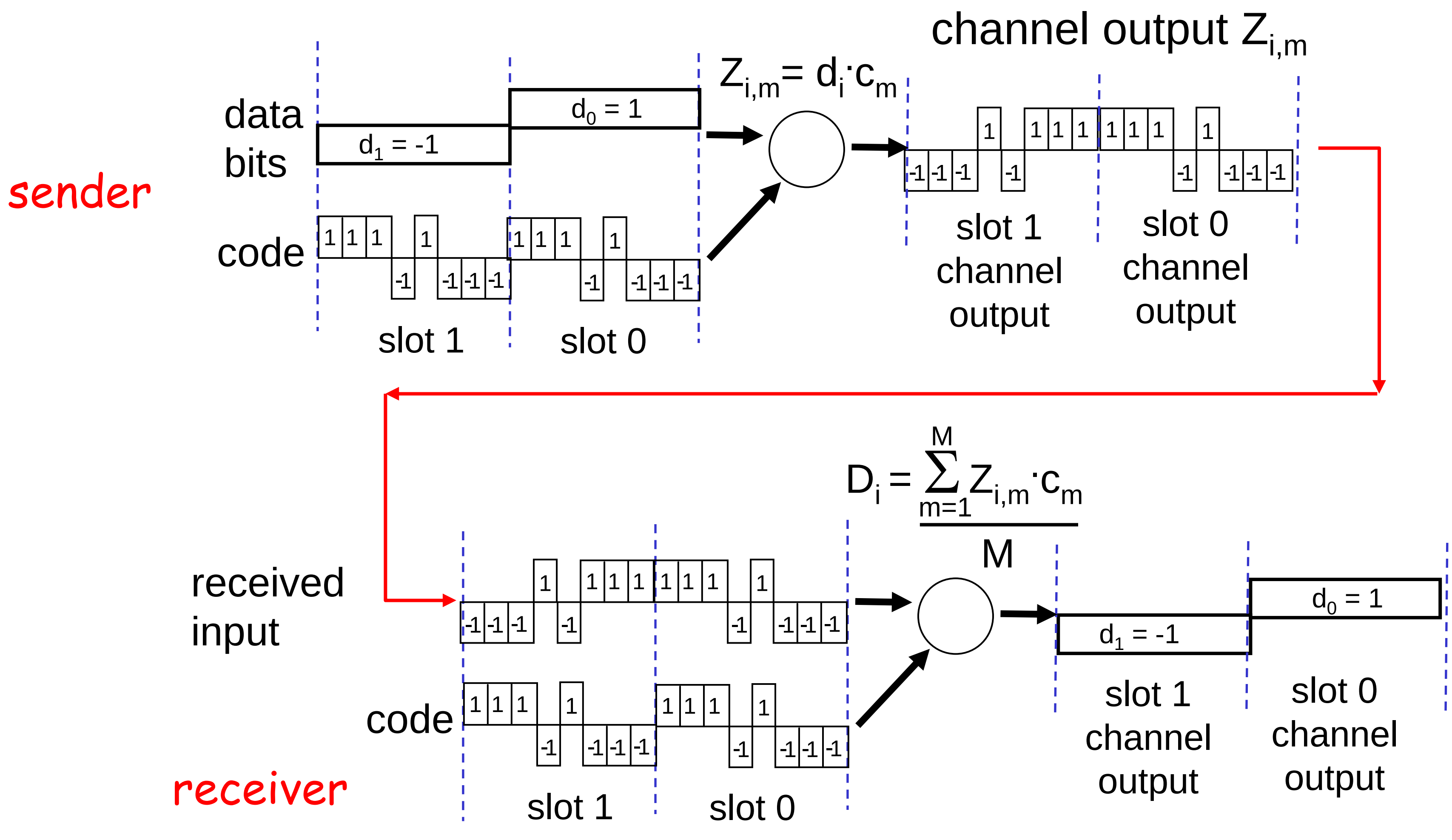
- used in several wireless broadcast channels standards (cellular, satellite, etc)
- unique “code” assigned to each user; i.e., code set partitioning
- all users share same frequency, but each user has own “chipping” sequence (i.e., code) to encode data
- *encoded signal* = (original data) X (chipping sequence)
- *decoding:* inner-product of encoded signal and chipping sequence
- allows multiple users to “coexist” and transmit simultaneously with minimal interference (if codes are “orthogonal”)

Code Division Multiple Access (CDMA)

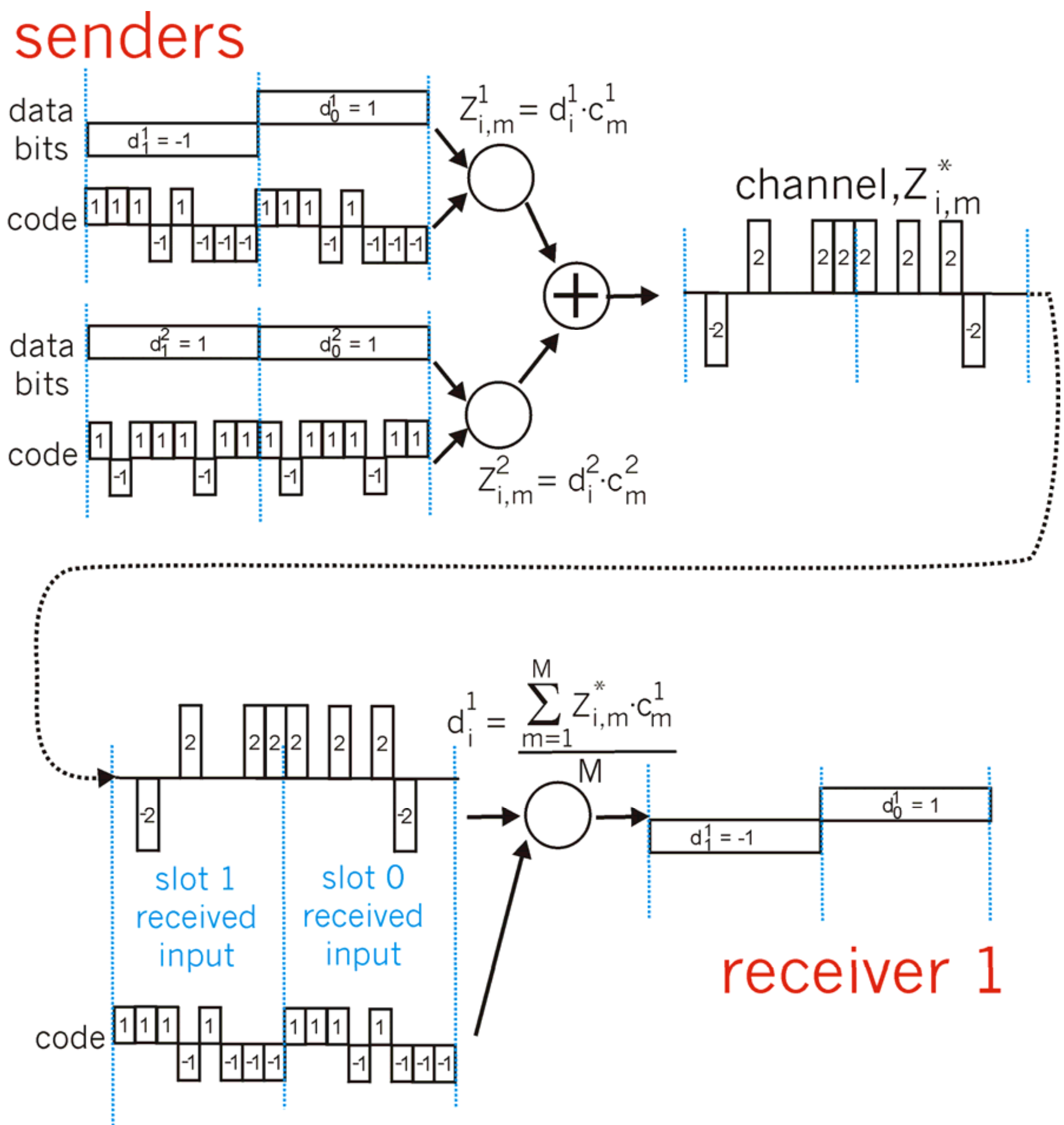
- Code vector $\vec{C} = (c_1, c_2, \ldots, c_m, \ldots, c_M)$; $c_i \in \{1,-1\}$
- Let d_i denote data bit *i* and c_m code bit *m*,
- Let $Z_{i,m} := d_i \cdot c_m$
- Then:

$$d_i = \frac{1}{M}\sum_{m=1}^{M}(d_i \cdot c_m)\cdot c_m = \frac{1}{M}\sum_{m=1}^{M} Z_{i,m}\cdot c_m$$

CDMA Encode/Decode



CDMA: two-sender interference



Random Access Protocols

- When node has some 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

Slotted ALOHA

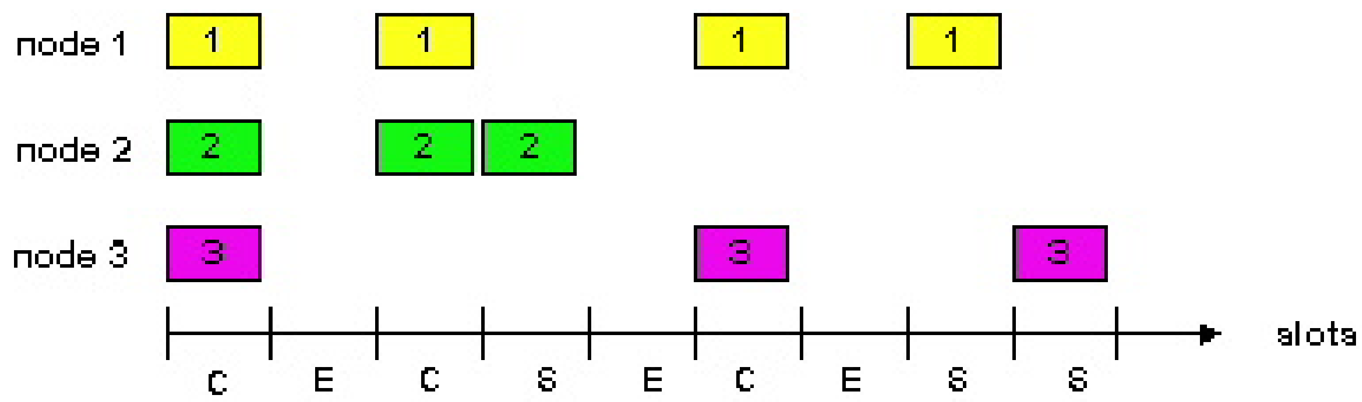
Assumptions

- all frames same size
- time is divided into equal size slots = time to transmit 1 frame
- nodes start to transmit frames only at beginning of slots
- nodes are synchronized
- if 2 or more nodes transmit in slot, all nodes detect collision

Operation

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

Slotted ALOHA



Pros

- single active node can continuously transmit at full rate of channel
- decentralized: although 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 is the long-run fraction of successful slots when there are many nodes, each with many frames to send

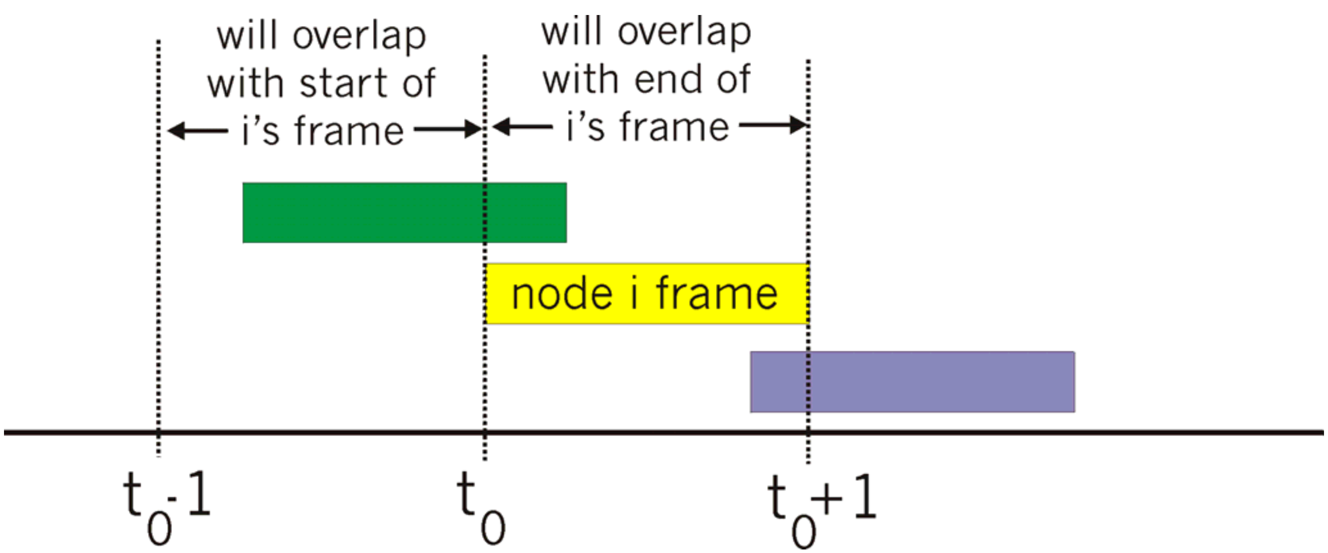
- Suppose N nodes with many frames to send, each transmits in slot with probability *p*
- prob that node "#1" has success in a slot $= p(1-p)^{N-1}$
- prob that any node has a success $= Np(1-p)^{N-1}$

- For max efficiency with N nodes, 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 $p^* = 1/e \approx 0.37$

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

Pure (unslotted) ALOHA

- unslotted Aloha: simpler, no synchronization
- when frame first arrives
  - transmit immediately
- collision probability increases:
  - frame sent at t_0 collides with other frames sent in $[t_0-1,t_0+1]$



Pure Aloha efficiency

P(success by given node) = P(node transmits) $\cdot$

P(no other node transmits in $[t_0-1,t_0]$ $\cdot$

P(no other node transmits in $[t_0,t_0+1]$

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

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

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

Even worse !

$p^* = 1/(2e) \approx 0.18$

CSMA (Carrier Sense Multiple Access)

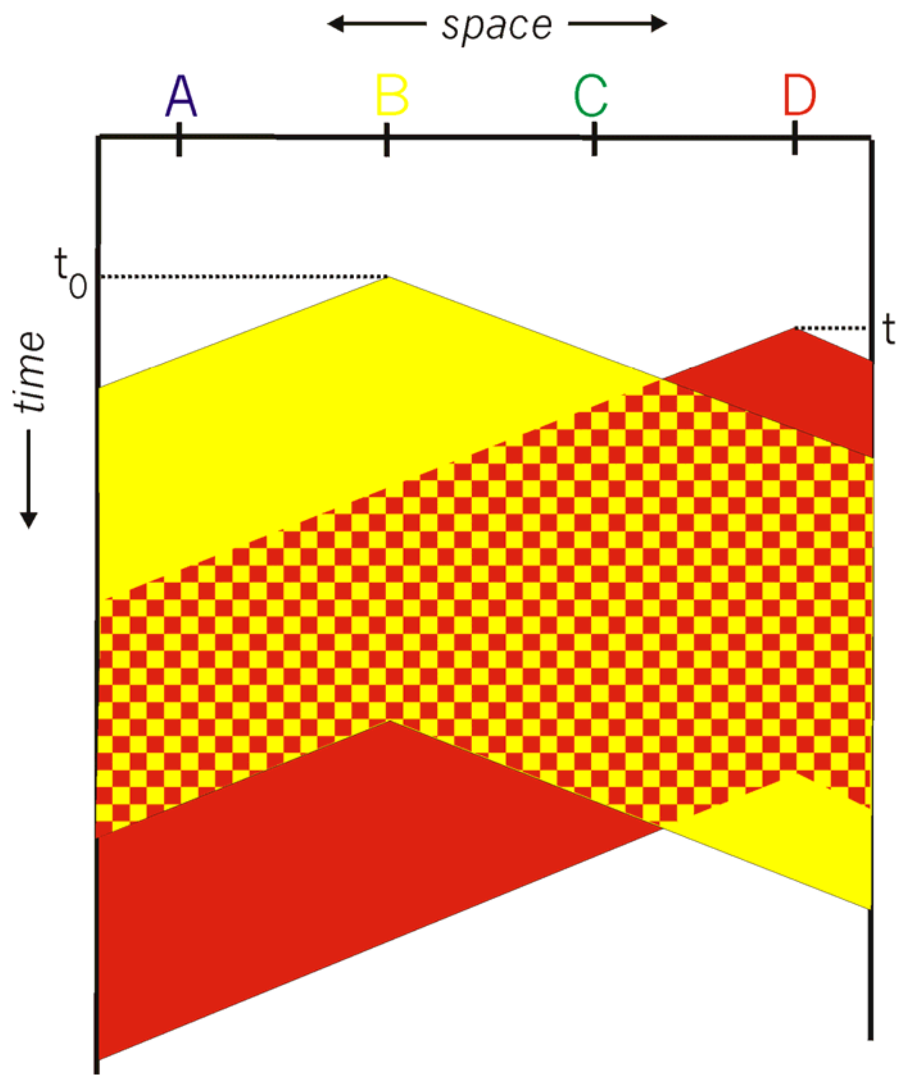
CSMA: listen before transmit:

If channel sensed idle: transmit entire frame

- If channel sensed busy, defer transmission
- Human analogy: don't interrupt others that already talk

CSMA collisions

spatial layout of nodes



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

collision:
entire packet transmission time wasted

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

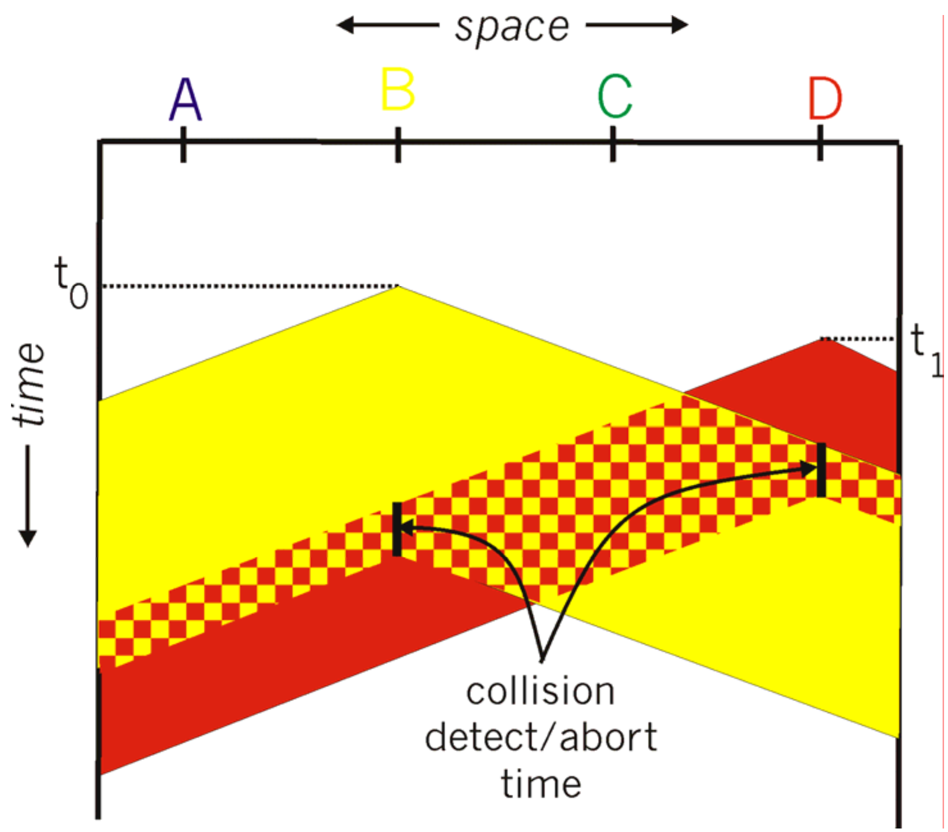
CSMA/CD (Collision Detection)

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: receiver shut off while transmitting
- human analogy: the polite conversationalist

CSMA/CD collision detection



“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!

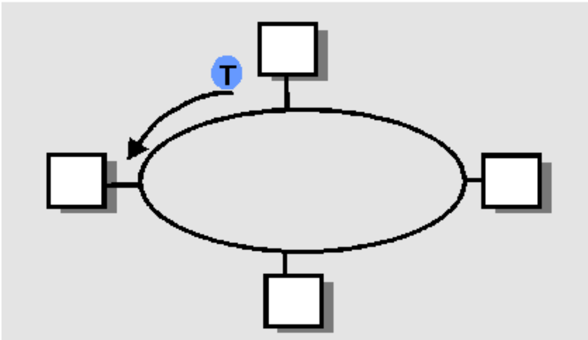
"Taking Turns" MAC protocols

Polling:

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

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

- What do you do with a shared media?
  - Channel Partitioning (a priori division):
    - Time Division, Frequency Division, Code division
  - Random partitioning (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 a central site
    - token passing