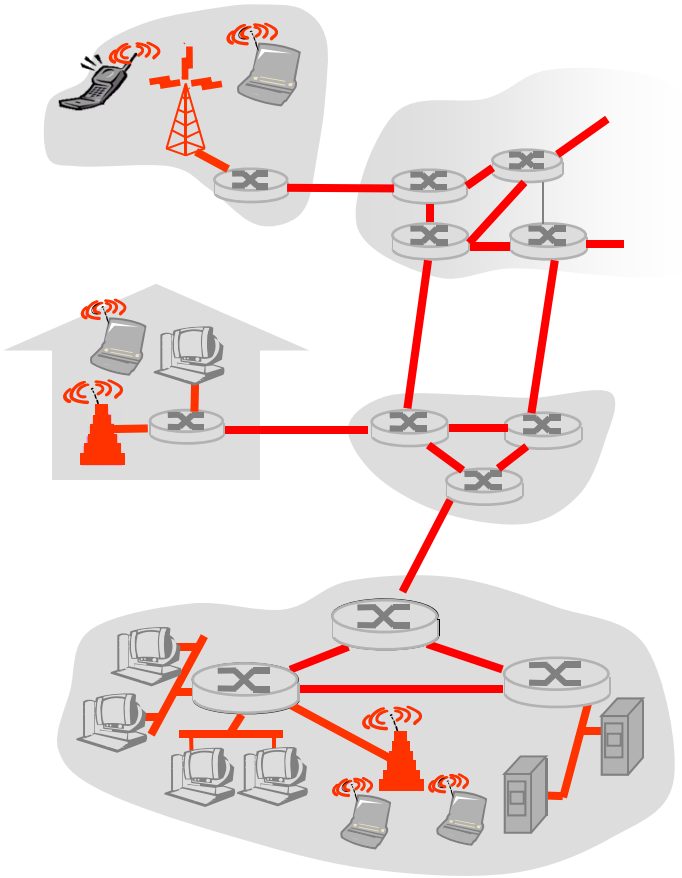
T4 Data Link Layer

Data Link Layer

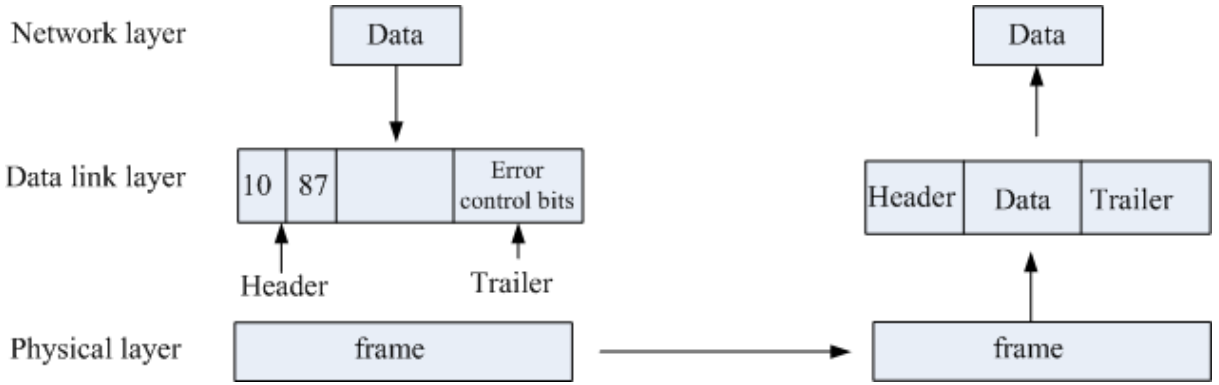
Objectives:

- understand principles behind data link layer services:
  - reliable data transfer
    - error detection and correction (by the receiver)
    - retransmission
  - link access by sharing a broadcast channel: multiple access
    - Instruct nodes **when** to transmit (… MAC protocols)



Data link layer

- The **data link layer** implements an error free packet delivery service between nodes that are attached to the same physical layer.
    1. It frames the packet into a specific format at the sending node and extracts the packet at the receiving node.
    2. It supervises the resulting packet delivery. Retransmission is requested if errors are detected.
- The **header** contains the physical address of the most recent node and the next intended node. The **trailer** contains error control information.



Bit Error Rate

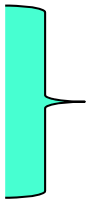
- Digital transmission systems introduce errors
- Errors occur due to noise and/or interference on a communication channel, e.g., Bit Error Rate (BER) = 10^{-7}

BER = # of bit errors / Total # of bits transmitted
= probability that a bit is received incorrectly

Types of errors:

Single bit error: one bit per frame has an error
Multiple bit error: two or more non-consecutive bits have errors
Burst error: two or more consecutive bits have errors

BER (wireless): 10^{-5}
BER (copper): 10^{-8}
BER (fibre optics): 10^{-12}



Significantly impacts link layer design

Packet Error Rate

Packet Error Rate (PER): For an N-bit packet, the probability that at least one of the N bits is received incorrectly. ε is the BER.

$$PER = 1 - (1 - \varepsilon)^N \approx N\varepsilon$$

Given the fact that

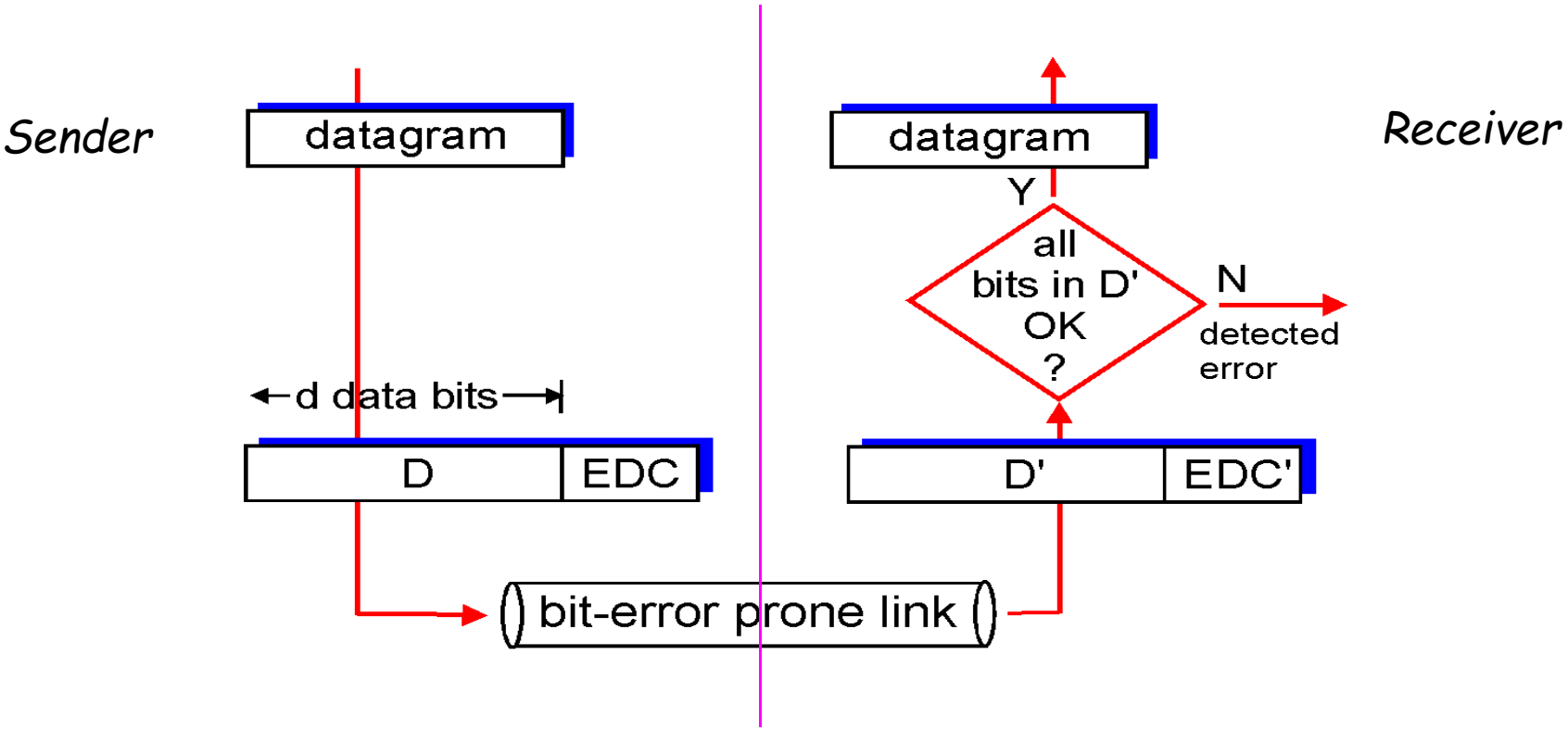
$$(1 - \varepsilon)^N \approx 1 - N\varepsilon$$

Error Detection (general idea)

D = Data (header + data)
EDC = Error Detection and Correction bits (redundancy/trailer)

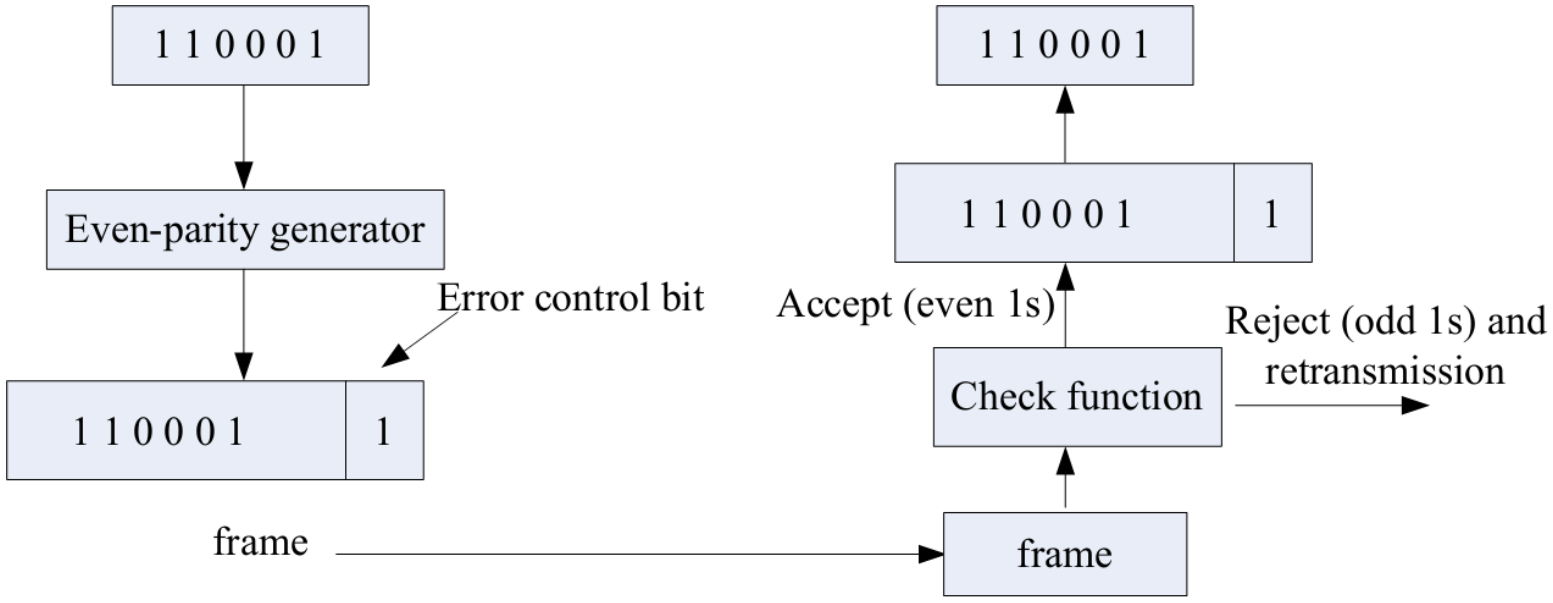
Error detection **not** 100% reliable!

- protocol may miss some errors, but rarely
- larger EDC field yields better detection and correction, but introduce more overhead



Parity Checking

A redundant bit called a **parity bit** is appended to every data packet unit so that the total number of ones becomes either **even** or **odd**. Both the sending and receiving systems must use the same type of parity. If an even parity unit is transmitted and an odd parity unit is received, the data unit won't be accepted.



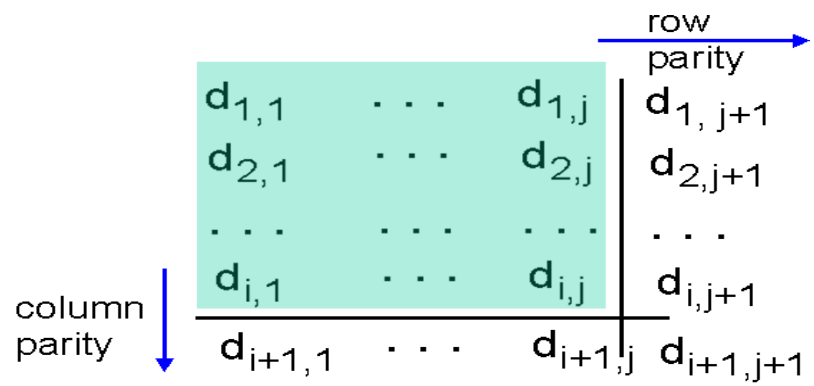
Parity Checking

Generating function checks all the bit "1"s. If the number of bit "1"s is an odd number, add bit "1"; an even number, add bit "0".

The receiver puts the entire frame through a check function. If the received bit stream passes the checking criteria, the data portion is accepted and the redundant bit is discarded.

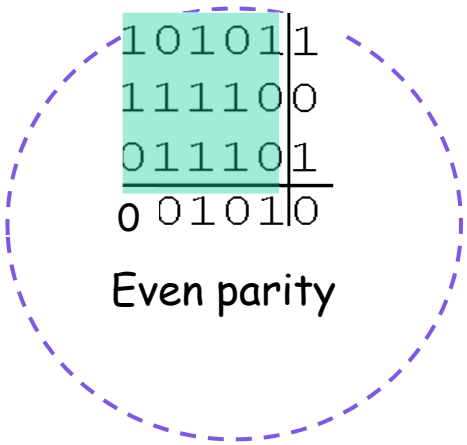
Reliability:
PC can detect all single-bit errors, but cannot locate errors
It can detect multiple or burst errors only if the total number of errors is an odd number.
It is the most common and least expensive mechanism for error detection.

Parity Checking (Two dimensional)



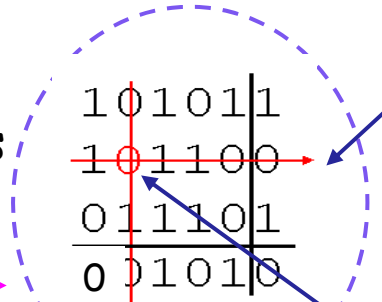
Detect **and** correct **single bit errors**

To be sent (use even parity): 1 0 1 0 1 1 1 1 1 0 0 1 1 1 0



Send this

… but this is what you receive



Inconsistent Row….

Inconsistent column….

Fix this error 0 → 1.

e.g. like this ….

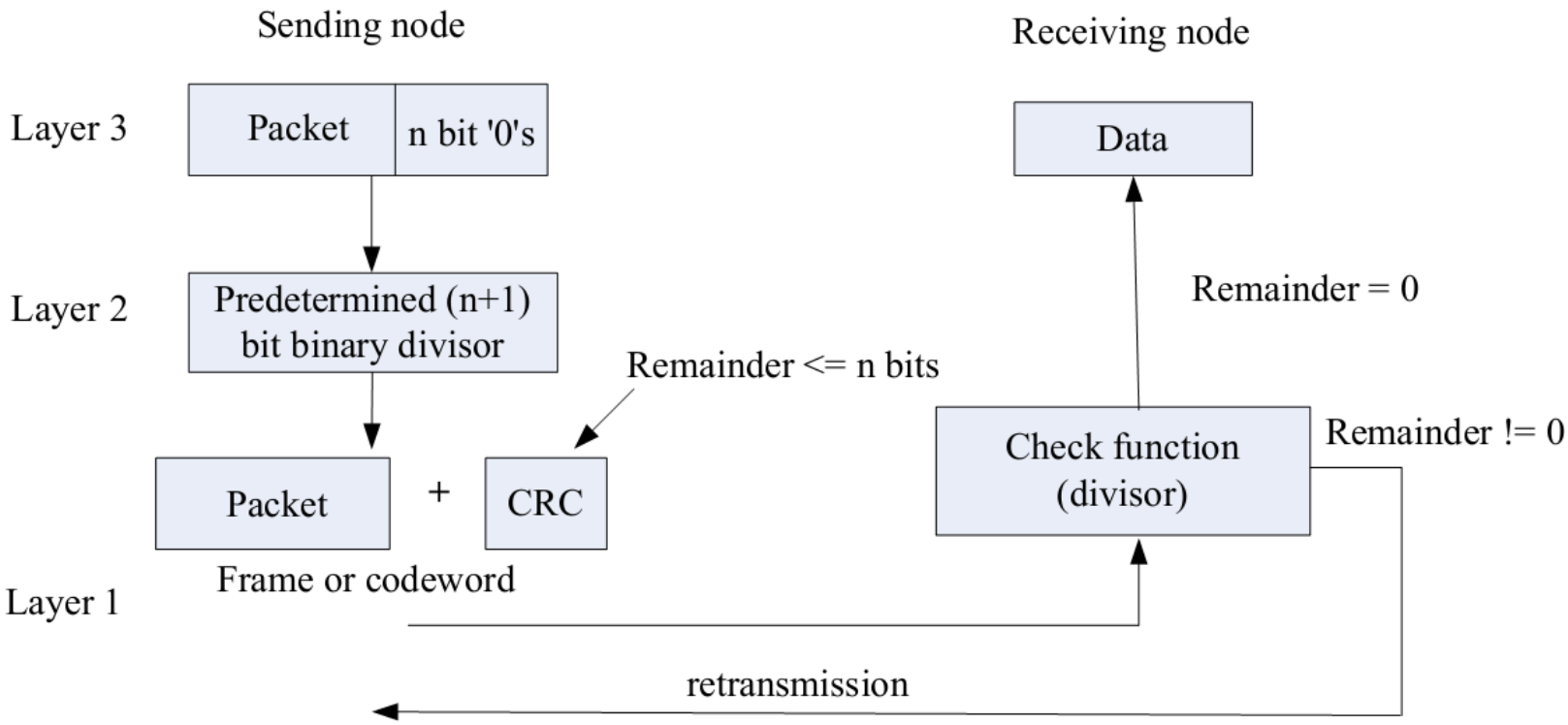
1 0 1 0 1 1 1 1 1 0 0 1 1 1 0 1 0 1 0 0 1 0 1 0

Cyclic Redundancy Code (CRC)

- CRC has been used by almost all communication networks. It is based on "binary division". In CRC, instead of adding bit together to achieve a desired parity, a sequence of redundant bits, called the CRC or the CRC remainder, is appended to the end of a data unit so that the resulting data unit (frame) becomes exactly divisible by a predetermined binary number.

- To be valid, a CRC must have two qualities: it must have exactly one less bit than the divisor, and appending it to the end of the data string must make the resulting bit sequence exactly divisible by the divisor.

Cyclic Redundancy Code (CRC)



The redundancy bits used by CRC are derived by dividing the data unit by the predetermined divisor; the remainder is the CRC.

Cyclic Redundancy Check (CRC)

- view data bits, D, as a binary number
- choose r+1 bit generator G (leftmost and rightmost bits are both 1)
- Append r bit "0" to D
- Then, Divide it by G. The remainder is CRC code (r bits).
- Send <D,R> codes to the receiver
- The receiver divides the received bit sequence by G
- If non-zero remainder: error detected!
- However, **a remainder of zero means**:

  i. no error;

  ii. error cannot be detected

CRC Example:

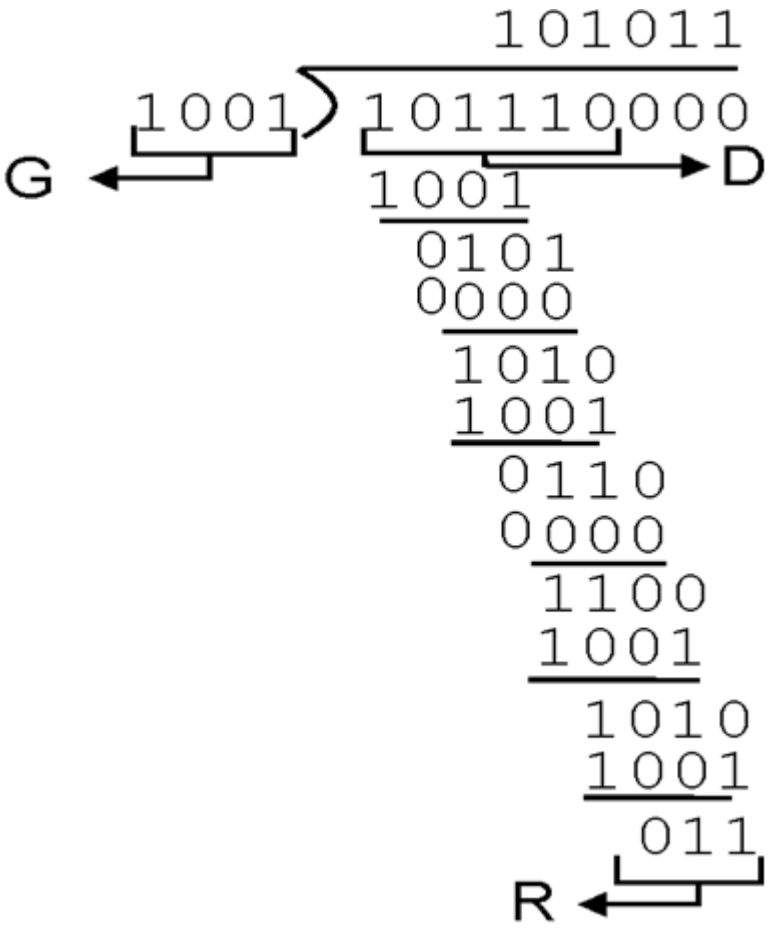
D = 101110 G = 1001

- All CRC calculations are done in modula-2 arithmetic without carries in addition or borrows in subtraction, which is equivalent to bitwise exclusive-or (XOR)

1 XOR 1 = 0	0 XOR 0 = 0
1 XOR 0 = 1	0 XOR 1 = 1

- The leading zero of the remainder (for each step) is dropped off

- When the leftmost bit of the remainder is zero, use 0000 instead of the original divisor.



Transmitted frame: 1 0 1 1 1 0 0 1 1

Cyclic Redundancy Check (CRC)

Reliability:
CRC will detect all possible errors except those that change the bit value by exactly the value of the divisor. Popular CRC divisors, use 13, 17 and 33 bits bringing the likelihood of an undetected error to almost zero.

The CRC generator (the divisor) is most often represented not as a string of 1's and 0's but as an **algebraic polynomial**, so that it is readily implemented using shift-register circuit.

It also provides a more compact framework and can be used **to mathematically prove the concept**.

Cyclic Redundancy Check (CRC)

The algebraic polynomial is called polynomial code.
We treat a k-bit frame as polynomial with coefficients of 0 and 1 only with terms from X^{k-1} to X^0

e.g., 110001 : $1 \cdot X^5 + 1 \cdot X^4 + 0 \cdot X^3 + 0 \cdot x^2 + 0 \cdot X^1 + 1 \cdot x^0 = x^5 + x^4 + 1$

- Standard polynomial of divisor:

CRC-12 (13 bits): $G(X) = x^{12} + x^{11} + x^3 + x + 1$

1100000001011

CRC-16 (17 bits): $G(X) = x^{16} + x^{15} + x^2 + 1$

Cyclic Redundancy Check (CRC)

- At the sender, the message is represented by m bits.

$$U(x) = U_{m-1}x^{m-1} + U_{m-2}x^{m-2} + \ldots + U_0 x^0 \quad U_j = \{0,1\} \quad j\{0,\ldots,m-1\}$$

- The generator polynomial has degree r or has (r+1) bits.

$$G(x) = g_r x^r + g_{r-1}x^{r-1} + \ldots + g_0$$

For a **binary divisor**: $g_0 = g_r = 1$

- U(x) is appended with r zeros.

$$x^r U(x) = U_{m-1}x^{n-1} + U_{m-2}x^{n-2} + \ldots + U_0 x^r \qquad n = m + r$$

- Divide $X^rU(x)$ by G(x) to obtain the remainder:

$x^r U(x) = G(x)Q(x) + C(x)$ where Q(x) is the quotient.

- Add C(x) to $X^rU(x)$ in order to get the codeword T(x)

$$T(x) = x^r U(x) + C(x) = G(x)Q(x)$$

Cyclic Redundancy Check (CRC)

- On the receiver side:

$$\hat{T}(x) = T(x) + E(x) \longleftarrow \text{Due to noise and interference}$$

Note that each error in the frame (codeword) corresponds to a non-zero coefficient in E(x). In other words; x^i indicates an error occurs at the $i+1^{th}$ bit.

- If there are no **errors** then

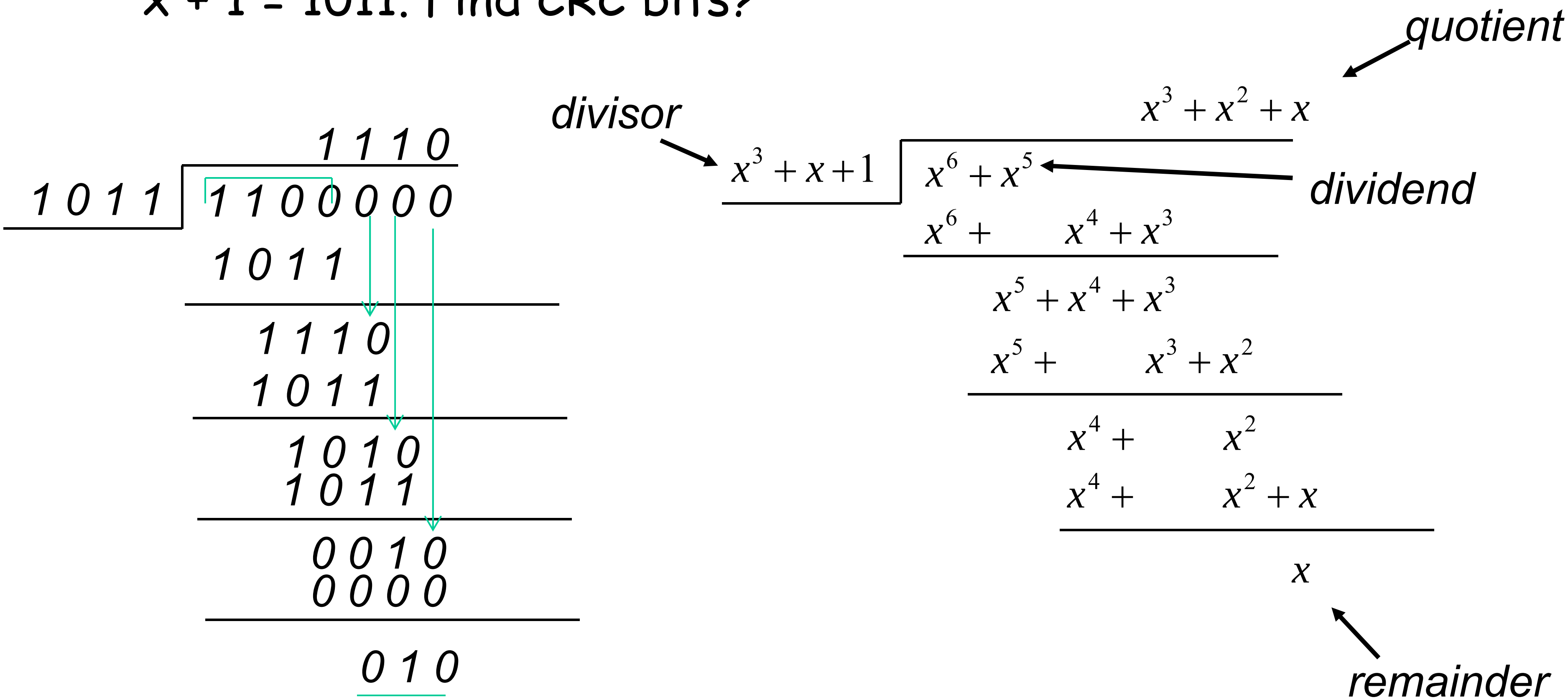
$$E(x) = 0$$

- Divide $\hat{T}(x)$ by $G(x)$

1. If the remainder ≠ 0, then *E(x)* ≠ 0
2. If the remainder = 0, then $G(x) | E(x)$ which implies:
   a. E(x) = 0, there is **no error**, or
   b. E(x) ≠ 0, but E(x) is divisible by G(x), there is an **undetected error**

Cyclic Redundancy Check (CRC)

- The data is 1100 and the generator polynomial $G(x) = x^3 + x + 1 = 1011$. Find CRC bits?



Properties of CRC

1. All single bit errors can be detected.
   For a single bit error, $E(x) = x^i$, $0<=i<=n-1$, $n=m+r$.
   Note that each error in the frame (codeword) corresponds to a non-zero coefficient in E(x). In other words; x^i indicates an error occurs at the $i+1^{th}$ bit.

   Since G(x) has at least two nonzero terms (x^r and 1)

$$G(x) = x^r + g_{r-1}x^{r-1} + \ldots + g_1 x^1 + 1$$

   E(x) must have at least two nonzero terms in order to be divisible by G(x). Therefore, for single bit errors, the error can be detected.

Properties of CRC

2. Double bit errors (two isolated single bit errors) separated by k-1 bits, can be detected if $G(x) \nmid (x^k + 1)$

   Since $E(x) = x^j + x^i = x^i (x^k + 1), \quad k = j - i$ and $G(x) \nmid x^i$
   Then, $G(x) \nmid E(x)$ if $G(x) \nmid (x^k + 1)$

3. Single burst errors of length k <= r can be detected. Burst length refers to the number of bit positions from the first error to the last error inclusive.

   Burst errors of length k can be given by

$$x^i (x^{k-1} + \ldots + 1)$$

   From 1, $G(x) \nmid x^i$

   If k-1 < degree of G(x) (i.e., k <= r) , then G(x) can never divide E(x)

Properties of CRC

4. The probability that a burst error of length r+1 is undetected is $2^{-(r-1)}$, (assuming all error patterns are equally likely).

Let $E(x) = x^i \left(x^r + e_{r-1}x^{r-1} + \ldots + e_1 x + 1 \right)$

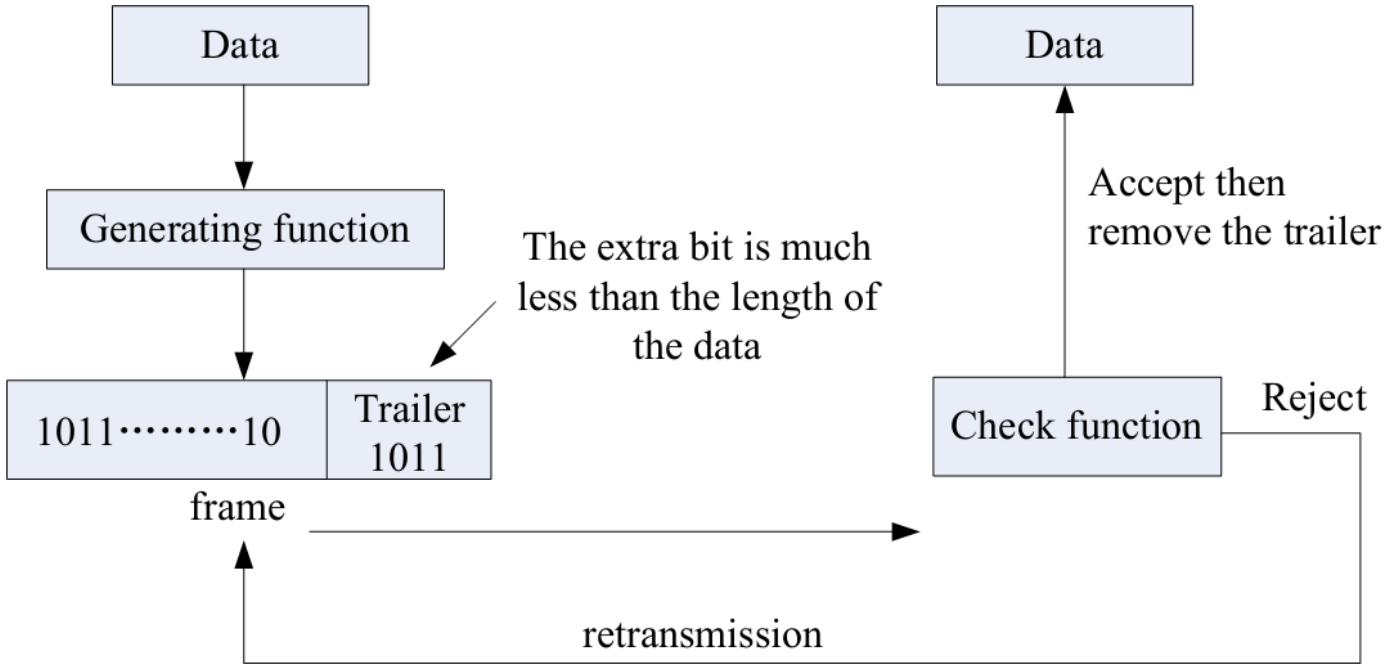
where $e_i \in \{0,1\}, \quad 0 \le i \le m-1$

$G(x) \mid E(x)$ if and only if: $e_{r-1} = g_{r-1}, \ldots, e_1 = g_1$

Since there are 2^{r-1} equally likely patterns, the probability of undetected burst errors of length r+1 is $2^{-(r-1)}$

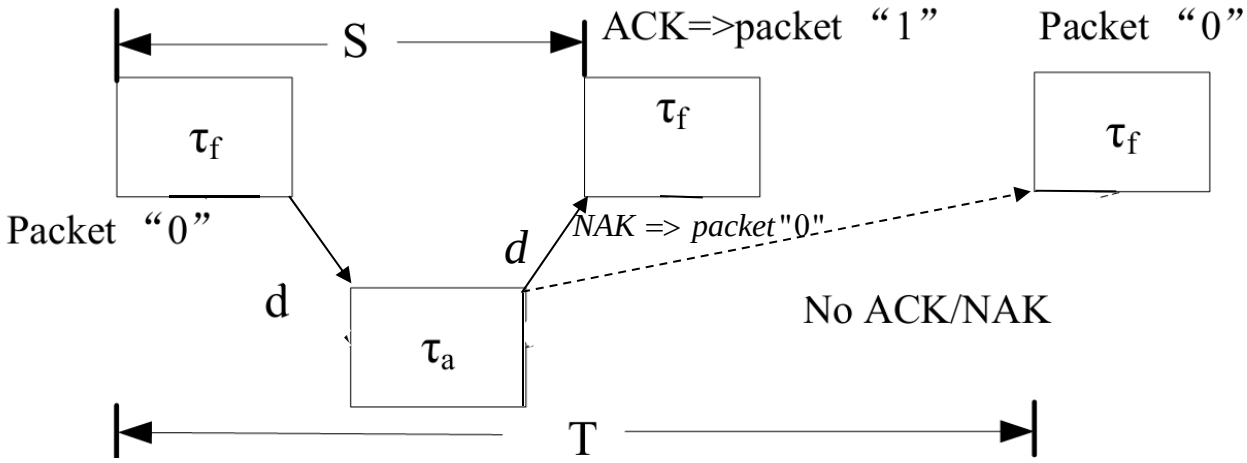
Recap: Error Detection and Retransmission

Error detection means adding extra bits for detecting errors at the destination. If there is error, the receiver arranges for the transmitter (sender) to send another copy of the same packet by using the retransmission protocol for supervising the retransmission.



Stop-and-Wait Protocol

- Retransmission in the data link layer is based on automatic repeat request (ARQ). All retransmission protocols are based on the same mechanisms: timers and acknowledgements.
- Stop and wait protocol: the basic idea is to ensure that each packet has been received correctly before initiating the transmission of the next packet



τ_f – time to transmit 1 packet
d – propagation delay
τ_a – time to transmit an ACK/NCK packet
$S = \tau_d + \tau_f + 2d$ – round trip delay
T>=S bound of time out

S: the time between sending packet and receiving acknowledgement)

Stop-and-Wait Protocol

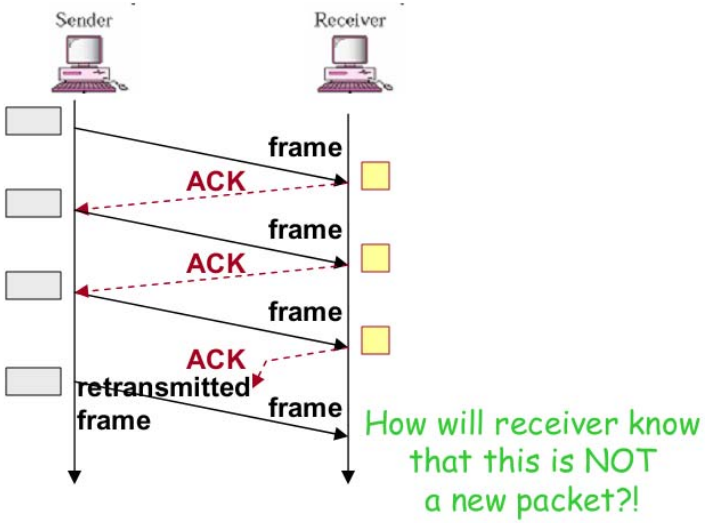
The sender appends a sequence number to each packet. The sequence number can be alternating "0" and "1". The sender transmits a packet and waits for up to T seconds for acknowledgement.

- ACK is received by sender, send frame "1"
- NAK is received by sender, send frame "0"
- No ACK/NAK after T seconds, resend frame "0"

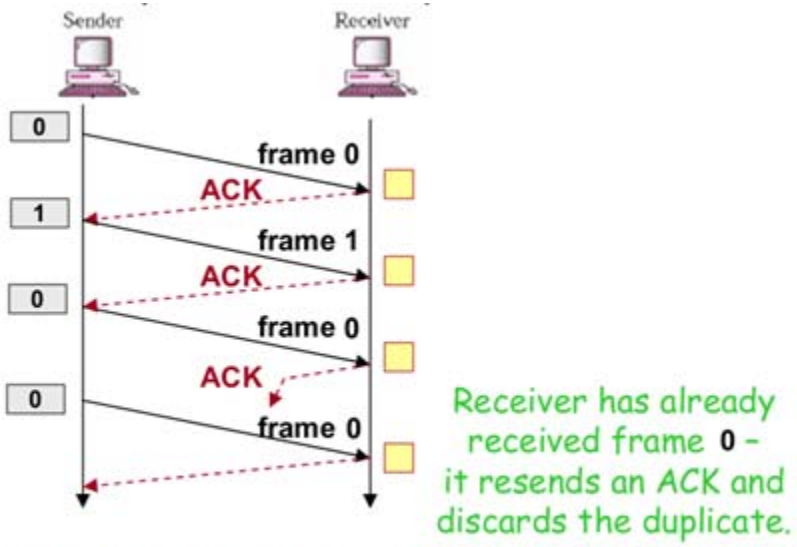
The receiver also appends a corresponding sequence number to each ACK/NAK packet.

Stop-and-Wait Protocol: Packet Numbering

- Frame received correctly, but ACK undergoes loss
  - after time-out period, sender resends frame
  - receiver receives the same frame twice
- Frames must be numbered so that receiver can recognize and discard duplicate frames
  - sequence # are included in packet header
  - only 1 bit required ("0" and "1")



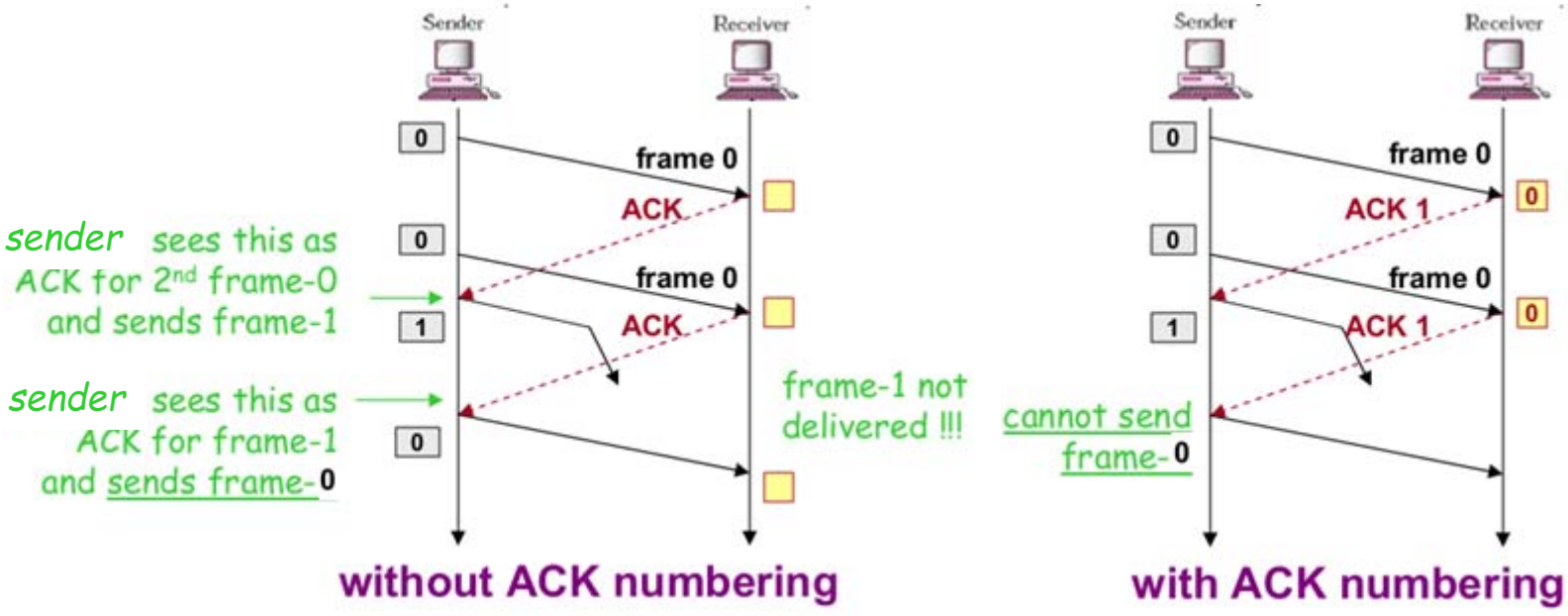
without packet numbering



with packet numbering

Stop-and-Wait Protocol: ACK Numbering

- ACKs can be delayed due to problems with links or network congestion, then sender resends frame.
  When delayed ACK arrives, sender assumes that given ACK is for the last frame sent
- ACKs must be numbered to prevent gaps in delivered packet sequence (1 bit required)



Stop-and-Wait Protocol

- **Efficiency**: the average rate at which the protocol sends correct packets divided by the channel transmission rate. It depends on the actual delay of the acknowledgement and the packet error rate.
- Let X be the time between transmission of packet n and that of packet (n+1). X is a random variable.

$$\text{Efficiency} = \frac{\tau_f}{E[X]}$$

No error: efficiency $\eta_{SWP} = \frac{\tau_f}{S}$

Let (1- ρ) represent the probability of packet error.
Let ρ represent the probability of no packet error.

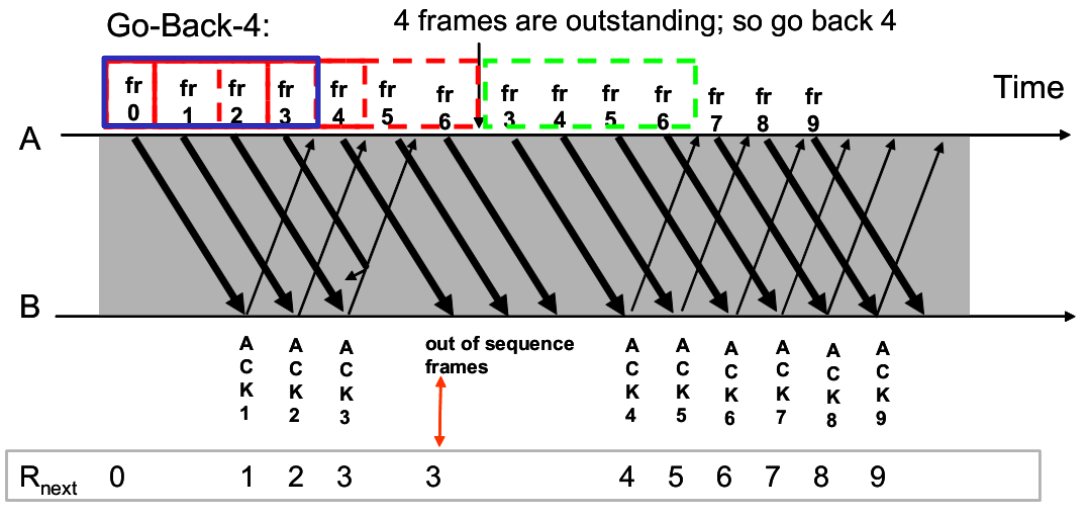
$$E[X] = \rho S + (1-\rho)[T + E[X]] \qquad \Rightarrow E[X] = S + \frac{1-\rho}{\rho} T$$

$$\eta_{SWP} = \frac{\tau_f}{E[X]} = \frac{\tau_f}{S + (1-\rho)/\rho \cdot T}$$

Advantages: Simple
Disadvantages: Low efficiency

Go Back N (GBN) Protocol

- N is called the sliding window size. $T \geq N \cdot \tau_f$
- The sender first transmits packets 0, 1, 2, …, N-1 and waits for each of their acknowledgements. As soon as the sender gets the ACK with sequence number 0, it transmits packet N and so on. If packet P undergoes transmission errors, the receiver ignores packet P and all subsequent packets (P+1, P+2,…). The sender is then forced to 'go back N' packets and begin re-transmitting all packets from P onwards.



Go Back N (GBN) Protocol

No errors: if $S \geq N \cdot \tau_f$, for every S seconds, the sender transmits N packets,

$$\eta_{GBN} = \frac{N\tau_f}{S}$$

If $S < N \cdot \tau_f$, then the sender keep on sending packets and the efficiency is 100%.

Therefore, $$\eta_{GBN} = \min\left\{1, \frac{N\tau_f}{S}\right\}$$

If there are packet errors: $$\eta_{GBN} = \frac{\tau_f}{E[X]}$$

Where X is the time between transmission of packet n and that of packet n+1. Let ρ be the probability of no packet error,

Go Back N (GBN) Protocol

The average transmission time E[X] per packet is:

$$E[X] = \rho\tau_f + (S + E[X])(1-\rho)$$

$$\Rightarrow E[X] = \tau_f + \frac{(1-\rho)S}{\rho}$$

Assume that S = T = $N\tau_f$, i.e., the duration of the timeout T equals to the round-trip delay S.

In this way, the sender gets the ACK or NAK or Time-out of the first packet when it has transmitted N packets.

$$\eta_{GBN} = \frac{\tau_f}{E[X]} = \frac{\rho}{\rho + (1-\rho)N}$$

Local Area Networks and Medium Access Control Protocols

- A local area network (LAN) is a data communication system that allows a number of independent devices to communicate directly with each other in a limited geographic area.

  Area diameter · few km

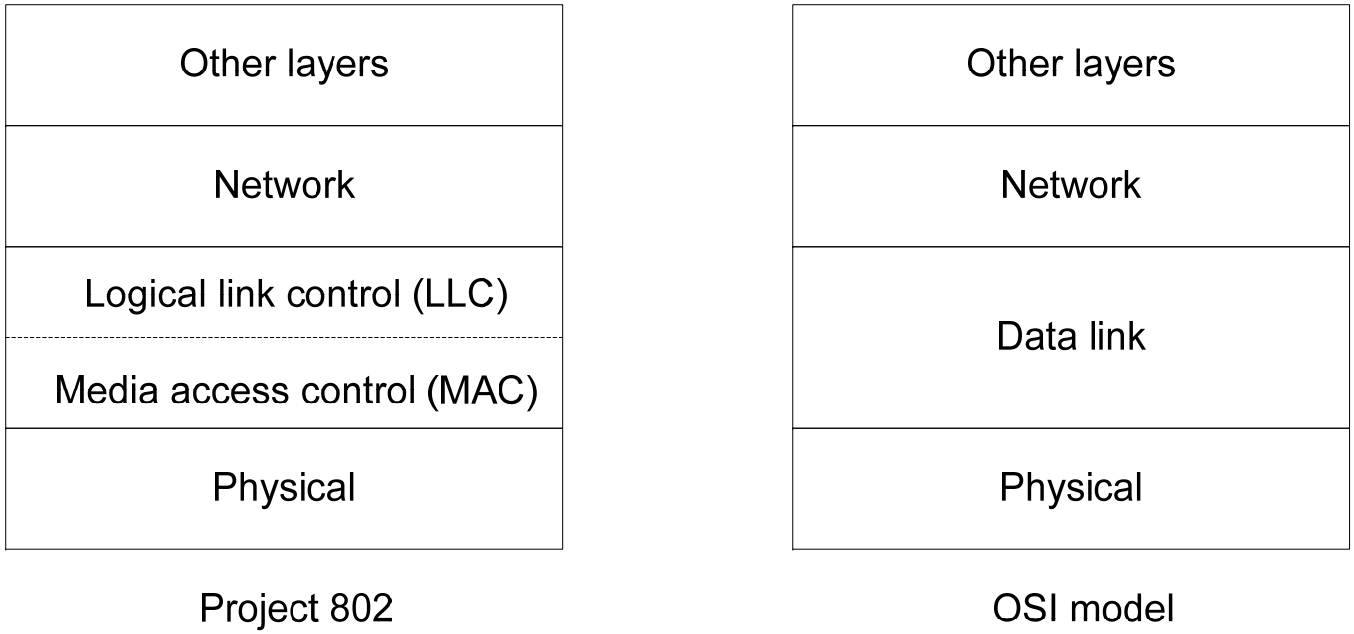
  Data rate = several Mbps

  Owned by a single organization

- LANs are broadcast networks. A single transmission medium is shared by a community of users. All information is received by all users. LANs are also called multiple access networks (with low-cost and simplicity).
- The medium access control (MAC) protocols are to coordinate the access to the channel so that information gets through from a source to a destination in the same broadcast network.

Medium Access Control

- In 1985, the Computer Society of the IEEE started a project, called Project 802, to set up standards so that LAN equipments manufactured by different companies is compatible.
- Project 802 divides the data link layer into sub-layers:
  - Logical link control (LLC)
  - Media access Control
- LAN compared with the OSI model



Project 802 OSI model

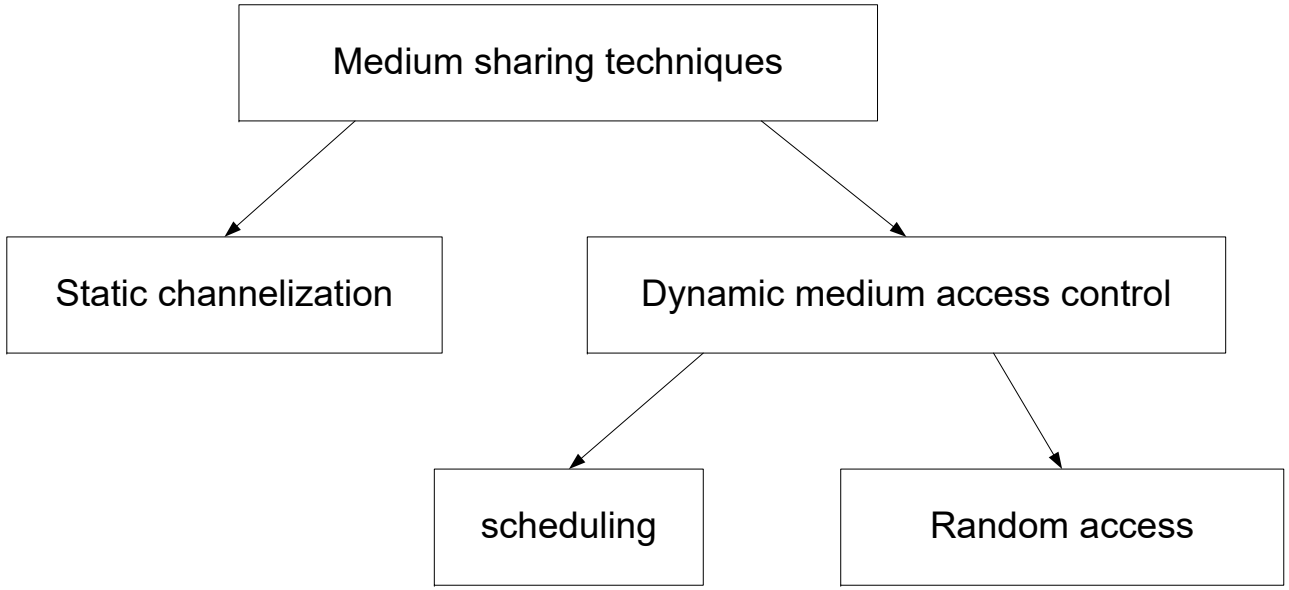
Medium Access Control

- The LLC is the upper sublayer and is same for all LANs. Its functions include error detection and retransmission.
- The MAC sublayer coordinates the data link tasks within a specific LAN.
- The MAC sublayer is manufacturer-specific and dependent on the LAN type.
- For LANs specified by project 802 are following:
  - Ethernet (802.3)
  - Token bus (802.4)
  - Token ring (802.5)
  - Wireless LANs (802.11)

Medium Access Control

Sharing a transmission medium

- Static sharing (channelization schemes) is a collision-free sharing
- Dynamic sharing (MAC schemes) minimizes the incidence of collision

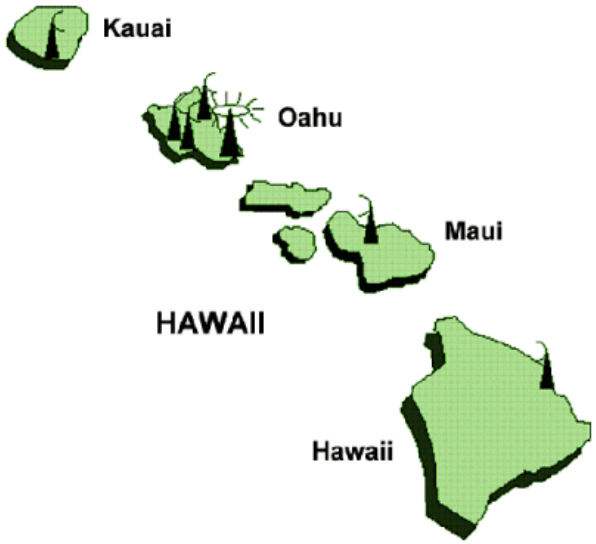


- Random access methods constitute the first major class of MAC procedures

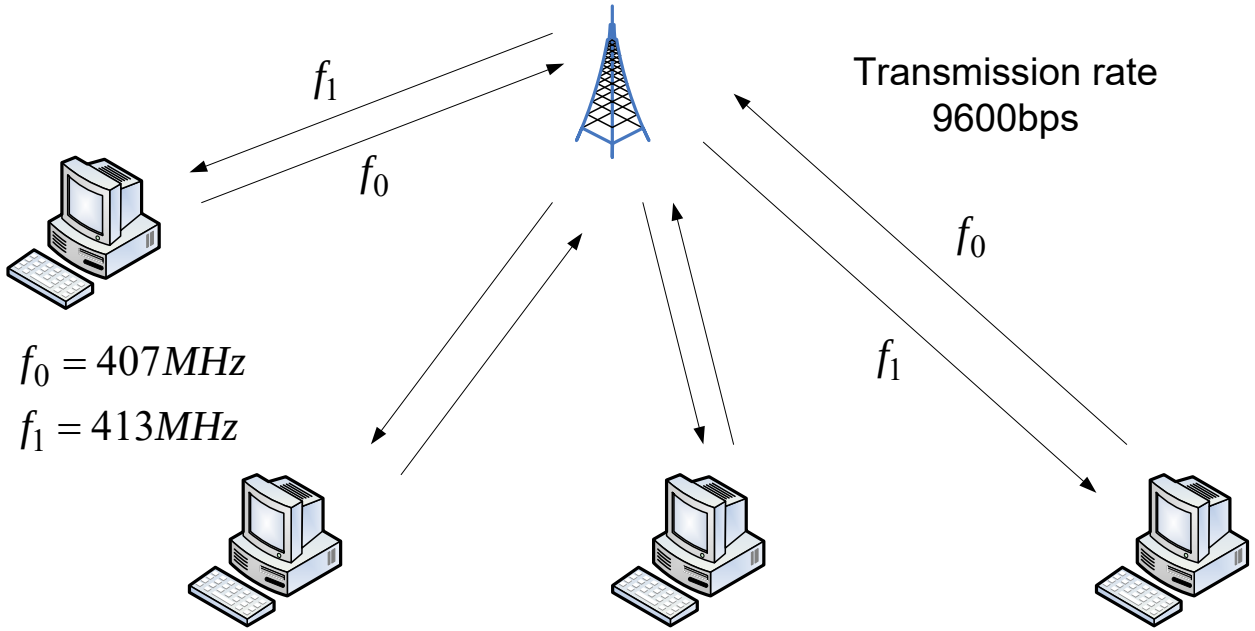
ALOHA random access scheme

- ALOHA is the first multiple access protocol.

- It was developed at the University of Hawaii in the early 1970s to connect computers situated on different Hawaiian islands. The computers of the ALOHA network transmit on the same radio channel whenever they have a packet to transmit. From time-to-time packet transmission will collide, but these can be treated as transmission errors, and recovery can take place by retransmission. When traffic is very light, the probability of collision is very small, and so retransmissions need to be carried out infrequently.



ALOHA network

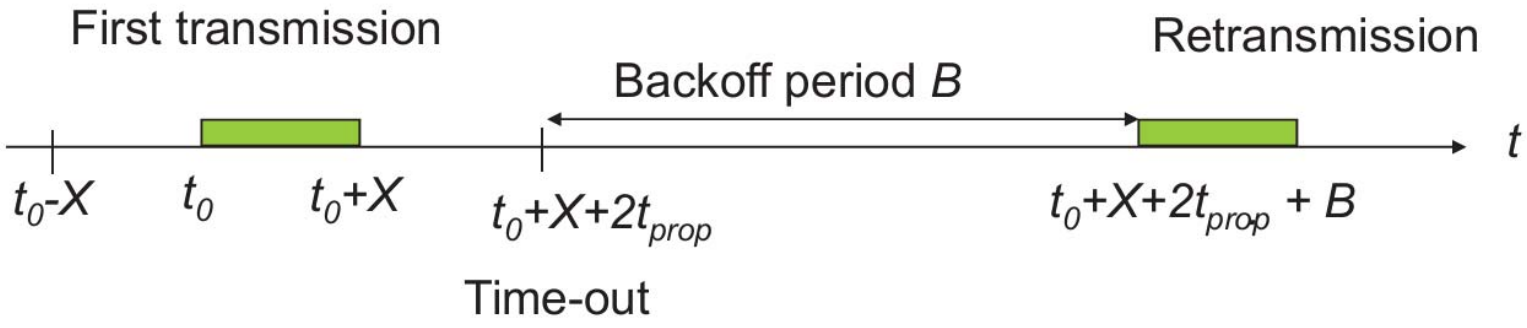


ALOHA is a packet-switched radio communication network.

Ethernet is its direct descendant.

Pure ALOHA (unslotted ALOHA)

1. Protocol
   - A user transmits whenever it has packets to transmit
   - When two or more packet transmissions overlap in time, a collision occurs and all the packets involved in the collision are destroyed.
   - If ACK not received within timeout, then a user picks random backoff time (to avoid repeated collision)
   - User retransmits packet after backoff time

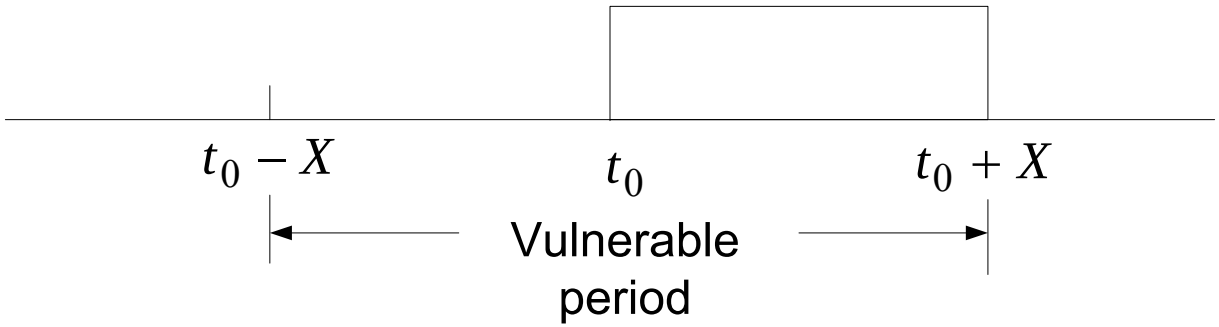


Throughput Analysis

1. Definitions and assumptions
    1. X: packet transmission time (assume constant)
    2. S: throughput (average # successful packet transmissions per X seconds)
    3. G: load (average # transmission attempts per X sec)

$$L = \text{packet length}$$
$$R = \text{transmission rate}$$
$$\Rightarrow X = \frac{L}{R} = \text{transmission time}$$

2. The probability of a successful transmission is the probability that they are no additional packet transmission in the vulnerable period.

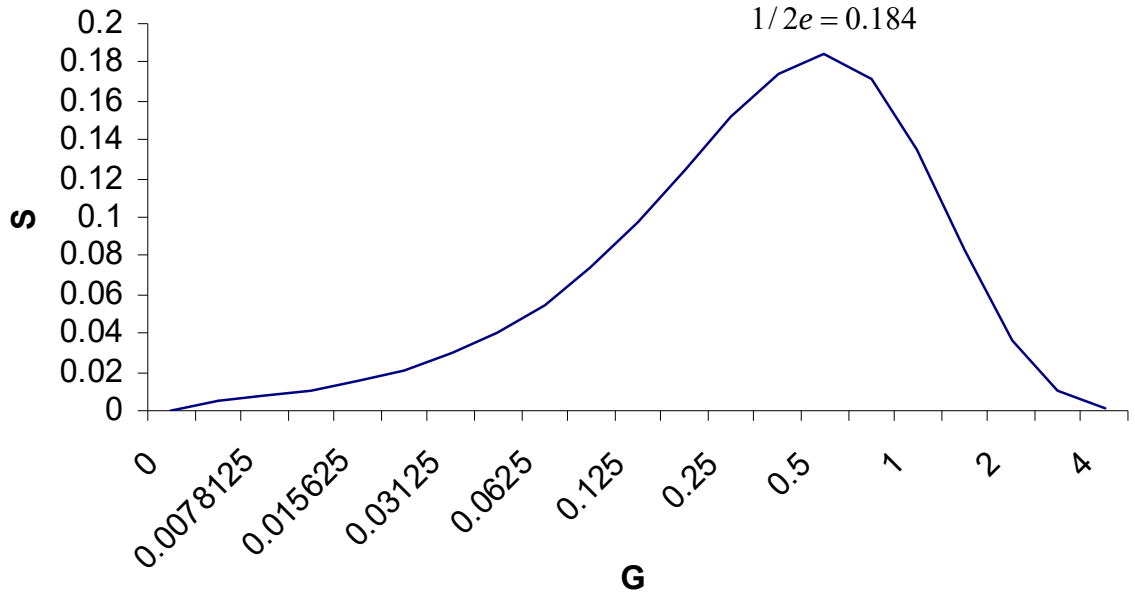


Throughput Analysis

The throughput $S = GP$(no collision)

$= GP$[0 transmission in 2X seconds]

The throughput for pure ALOHA is
$S = G \times e^{-2G}$.
The maximum throughput
S_{max} =1/2e=0.184 when G= (1/2).



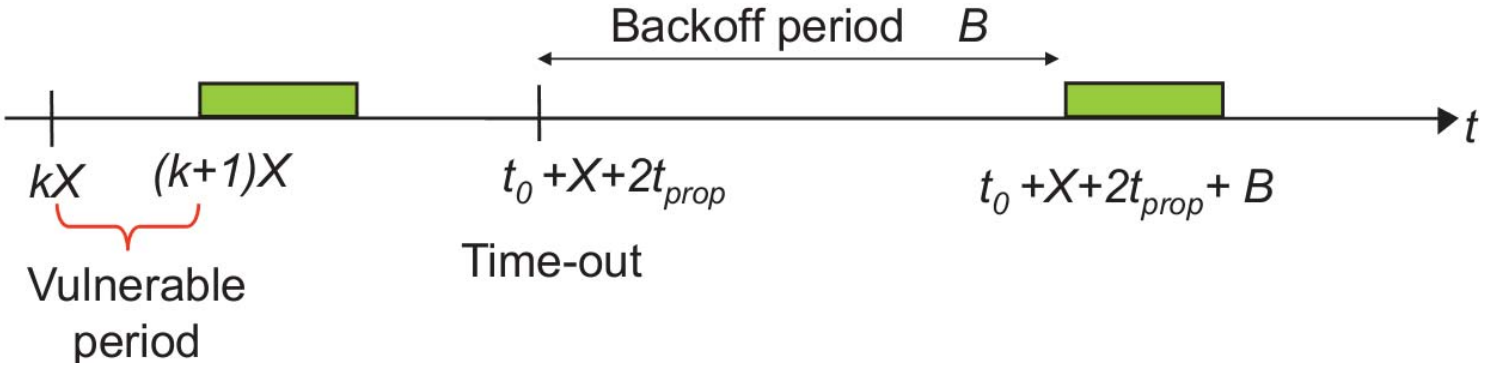
For small G, S ≈ G.

For large G, there are many backlogged users.

ALOHA system cannot achieve throughput higher than 18.4 percent (1/2e).

Slotted ALOHA

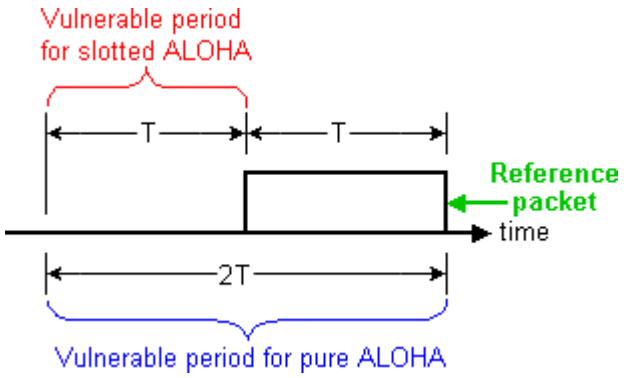
Slotted ALOHA is to constrain the user to transmit in synchronized fashion. All users keep track of transmission slots and are allowed to initiate transmission only at the beginning of a time slot (the time axis is divided into time slots with durations equal to the time to transmit a packet)

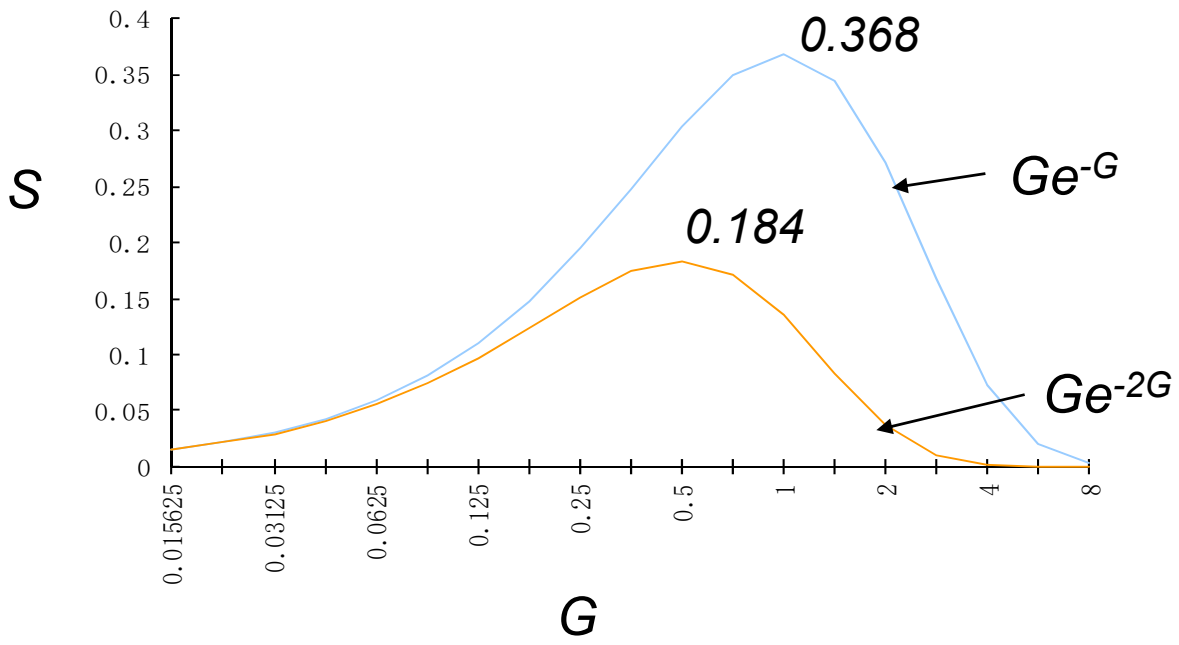


Only packets that arrive during prior X seconds collide.

Throughput of Slotted ALOHA

$$S = GP[\text{no collision}] = GP[\text{no arrivals in X seconds}]$$
$$= G \cdot e^{-G}$$





Example – ALOHA and Slotted ALOHA

Suppose that a radio system uses a 9600 bps channels for sending call setup request messages to a base station. Suppose that packets are 120 bits long. What is the maximum throughput possible with ALOHA and slotted ALOHA?

The system transmits packets at a rate of

9600 bits/second x 1 packet/120bits = 80 packets/second.

The maximum throughput for ALOHA is then

80 x 0.184≈15 packets/second.

The maximum throughput for slotted ALOHA is then

30 packets/second.

Carrier Sense Multiple Access with Collision Detection

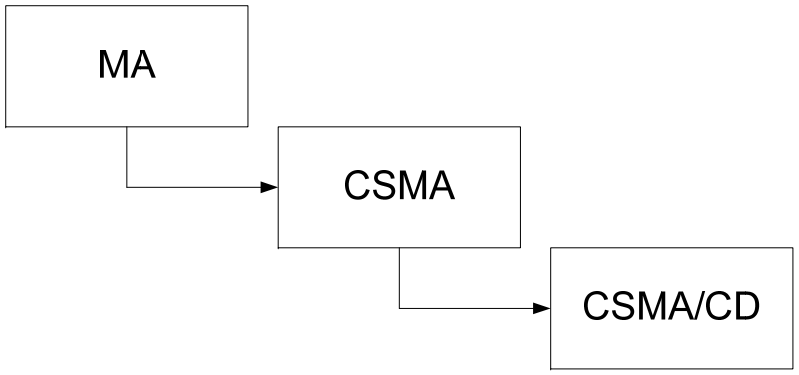
The access mechanism used in an Ethernet is called CSMA/CD, standardized in IEEE 802.3

CSMA/CD is the result of an evolution from multiple access (MA) to carrier sense multiple access (CSMA), and finally, to CSMA/CD.

In MA, there was no provision for traffic coordination.

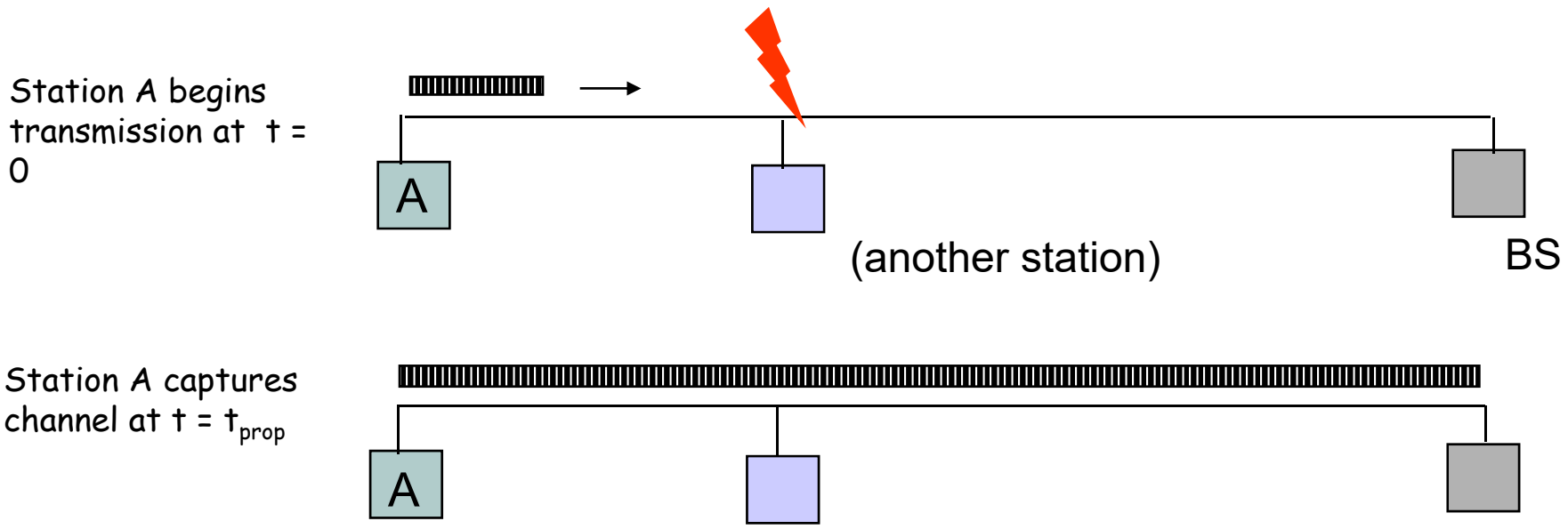
In a CSMA system, any user (work station) wishing to transmit must listen to the existing traffic on the line. A device listens by checking for a voltage. No voltage means the line is idle. CSMA cuts down on the number of collisions but does not eliminate them.

In CSMA/CD system, the station listens again after each packet transmission. The extremely high voltages indicate a collision.



Carrier Sensing Multiple Access (CSMA)

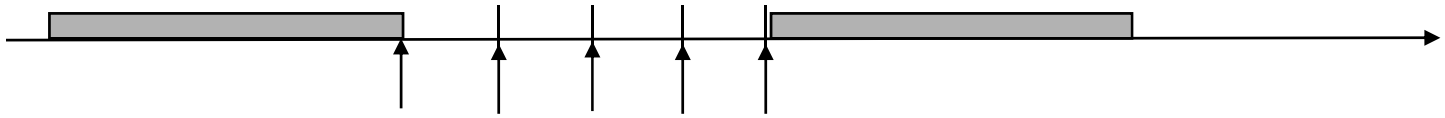
- A station senses the channel before it starts transmission
  - If busy, either wait or schedule backoff (different options)
  - If idle, start transmission
  - Vulnerable period is reduced to t_{prop} (due to channel capture effect)
  - When collisions occur they involve entire frame transmission times
  - If t_{prop} >X, no gain compared to ALOHA or slotted ALOHA



CSMA Options

- Transmitter behavior when busy channel is sensed
  - 1-persistent CSMA (most greedy)
    - Start transmission as soon as the channel becomes idle
    - Low delay and low efficiency
  - Non-persistent CSMA (least greedy)
    - Wait a backoff period, then sense carrier again
    - High delay and high efficiency
  - p-persistent CSMA (adjustable greedy)
    - Wait till channel becomes idle, transmit with prob. p; or wait one t_{prop} time & re-sense with probability 1-*p*
    - Delay and efficiency can be balanced

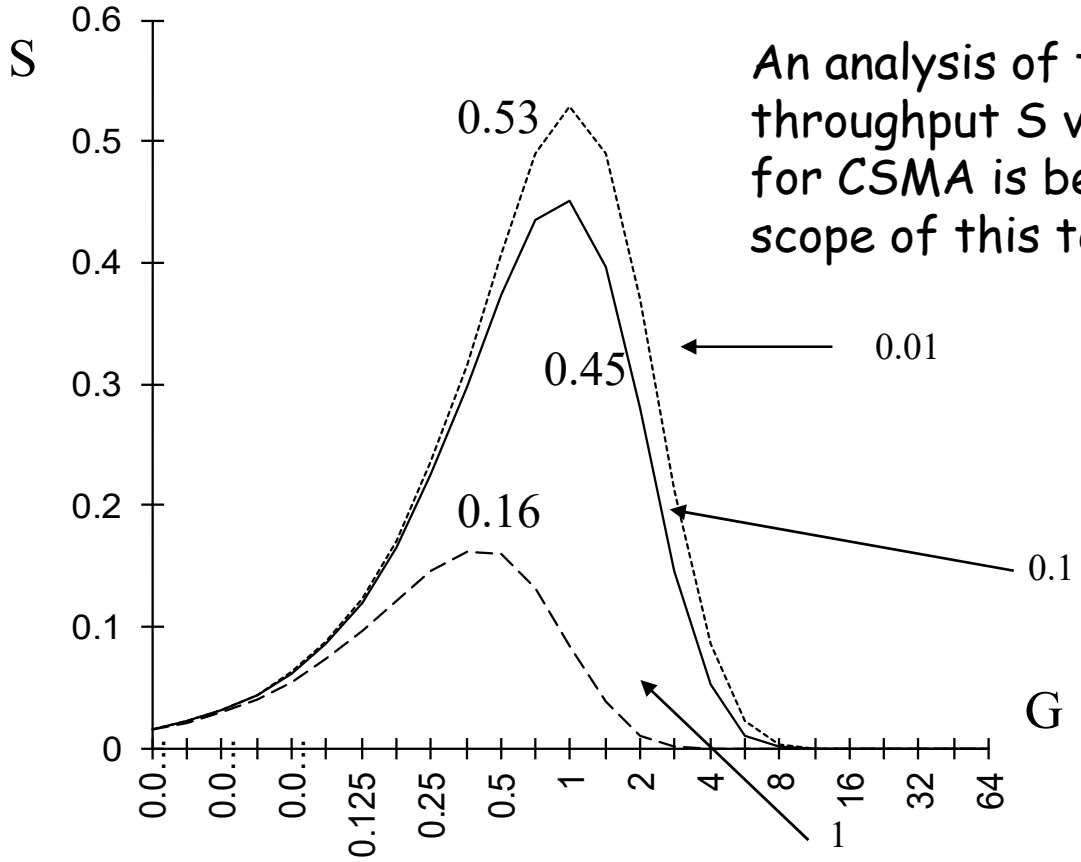
(spread out the transmission attempts)



Sensing

Throughput versus load G for 1-persistent (three different $a=t_{prop}/X$)

1-Persistent CSMA

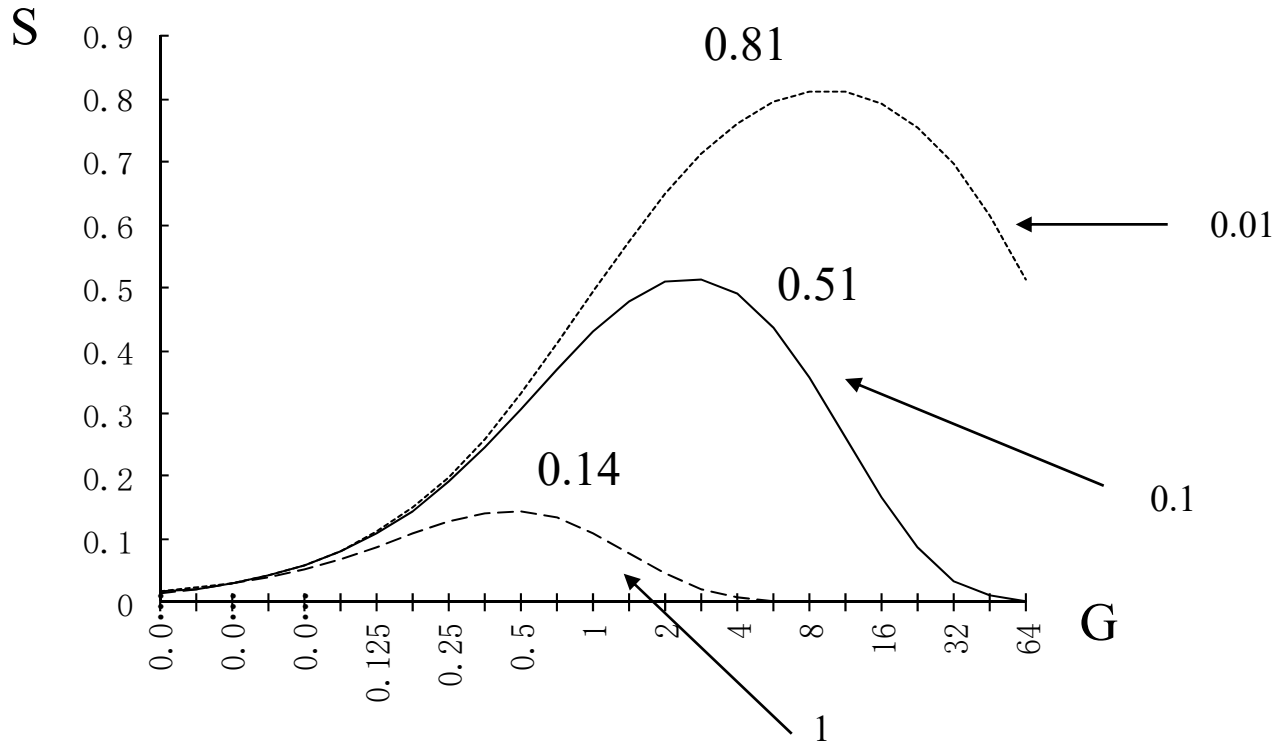


An analysis of the throughput S versus load G for CSMA is beyond the scope of this text.

The normalized propagation delay $a = t_{prop}/X$ has a significant impact on the maximum achievable throughput since t_{prop} constitutes the vulnerable period.

Throughput versus load G for non-persistent (three different $a=t_{prop}/X$)

Non-Persistent CSMA

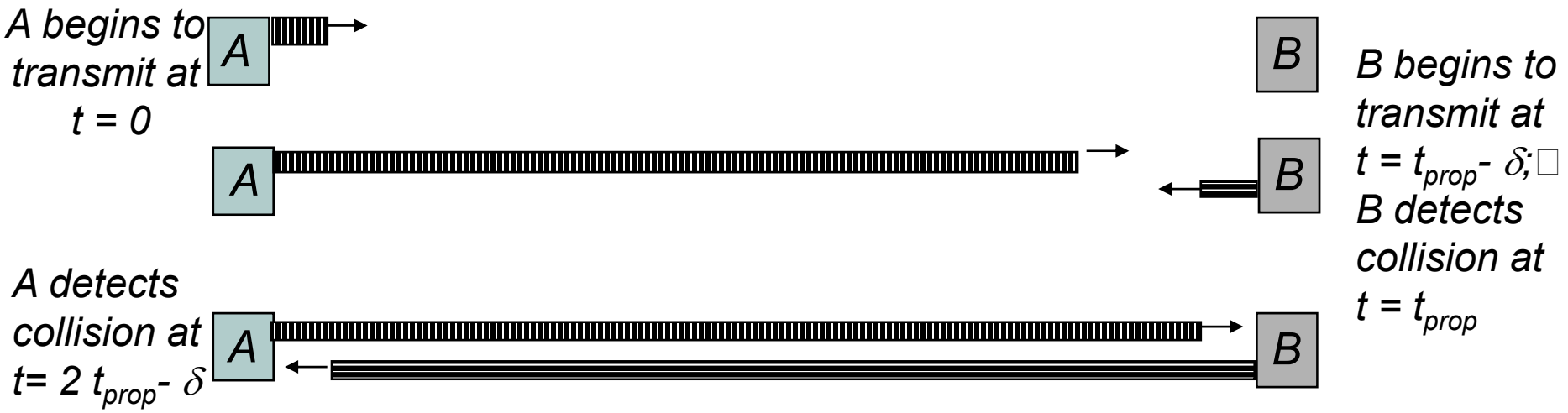


1-persistent is sharper than non-persistent.
$a=t_{prop}/X$ has import impact on the throughput.
When a approaches 1, both 1-persistent and non-persistent is worse than ALOHAs.

Analysis of CSMA/CD protocol

- In CSMA/CD protocol, a node with a packet to transmit must proceed as follows:
  1. Wait until the channel is idle;
  2. When the channel is idle, transmit and listen while transmitting
  3. In case of a collision, stop the packet transmission, and then wait for a random delay and go to (1).
- Note: when a node "goes back to (1)" after its waiting time, it senses the signal from the other nodes and must then wait until the end of that transmission before transmitting.
- In CSMA, collisions result in wastage of X seconds spent transmitting an entire frame
- CSMA-CD reduces wastage of time (or bandwidth) to detect collision and abort transmission

CSMA/CD reaction time



It takes 2 t_{prop} to find out if channel has been captured

(The reaction time in CSMA-CD is 2t_{prop})

Efficiency (throughput):

The nodes attempt to transmit at discrete times, in time slots with a duration of $2t_{prop}$ each. The slot duration of $2t_{prop}$ is used to guarantee that, if nodes select to transmit at the beginning of two different slots, then they can't collide.

Let α be the probability that during a given time slot there is no collision and that one node starts transmitting. Let N ($N \geq 2$) nodes compete for a time-slotted channel by transmitting packets with probability p, independently of one another, in any given time slot.

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

$$\frac{d\alpha(p)}{dp} = N(1-p)^{N-1} - N(N-1)p(1-p)^{N-2} = 0$$

$$N(1-p) - N(N-1)p = 0 \Rightarrow p = \frac{1}{N}$$

Efficiency (throughput):

$$\alpha(\frac{1}{N}) = (1 - \frac{1}{N})^{N-1} \approx 40\%$$

$$\alpha(1/4) = 42\%, \alpha(1/10) = 39\%, \alpha(1/20) = 38\%$$

Let A be the average number of time slots that are wasted before a successful transmission

$$A = \alpha \cdot 0 + (1-\alpha)(1+A)$$

First time slot is successful

First time slot is wasted

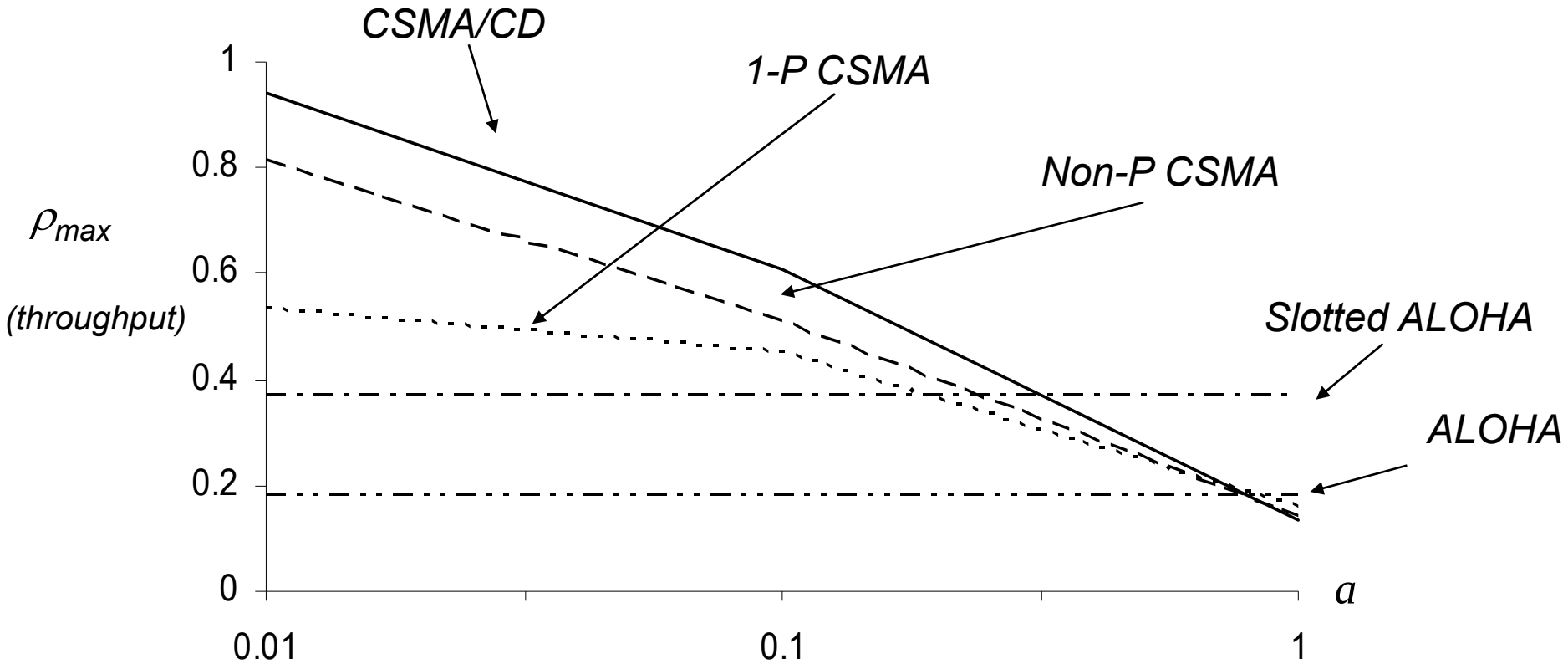
) $A = \alpha^{-1} - 1$

With α=0.4 A=1.5 slots

$$\eta_{CSMA/CD} \approx \frac{\tau}{\tau + 1.5 \times 2t_{prop}} = \frac{\tau}{\tau + 3 \times t_{prop}}$$

where τ is the time to transmit a packet.

Throughput for Random Access MACs



- For small a: CSMA-CD has best throughput
- For larger a: Aloha & slotted Aloha better throughput

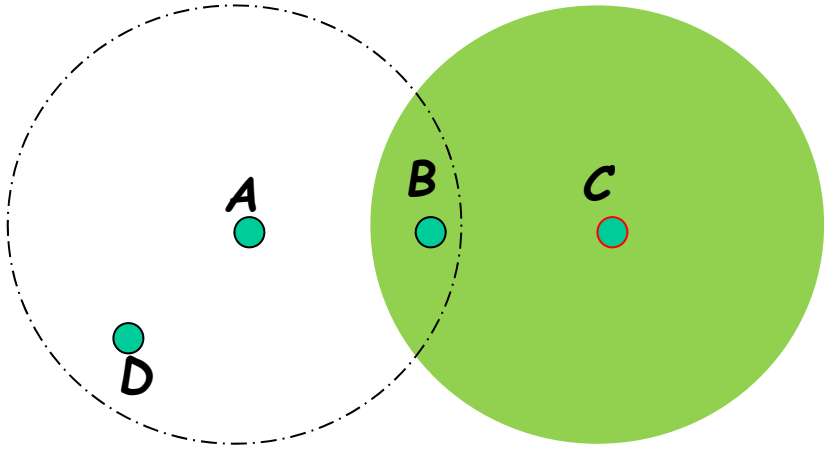
Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA)

- For WiFi, transmit signal is MUCH stronger than received signal
- High path loss in the wireless environment (up to 100dB)
- Impossible to "listen" while transmitting because you will drown out anything you hear
- Also transmitter may not even have much of a signal to detect due to geometry

Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA)

- Procedure
  - Similar to CSMA but instead of sending packets, control frames are exchanged
  - Sender asks receiver whether it is able to receive a transmission - ***Request to Send (RTS)***
  - Receiver agrees, sends out a ***Clear to Send (CTS)***
  - Sender sends, receiver ***Acknowledgements (ACKs)***

Hidden Terminal Problem

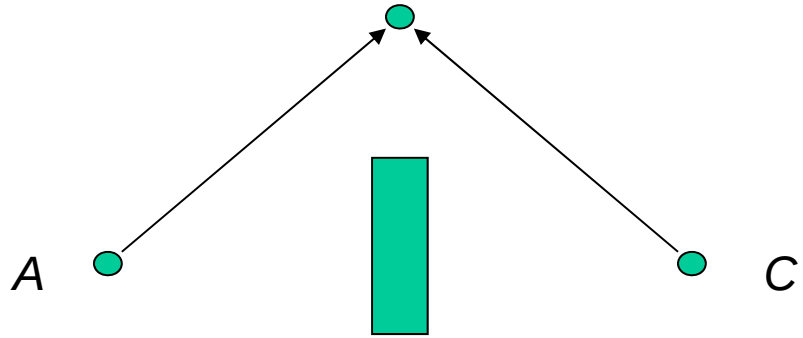


- **Problem**
  - C is transmitting a frame to B.
  - A is unaware of C's Tx.
  - Now, if A transmits, A's Tx will collide with C's at B

- The above problem is due to C being **hidden** from A.
  - Hidden means being "far away" …

CSMA-CA solving Hidden Terminal Problem

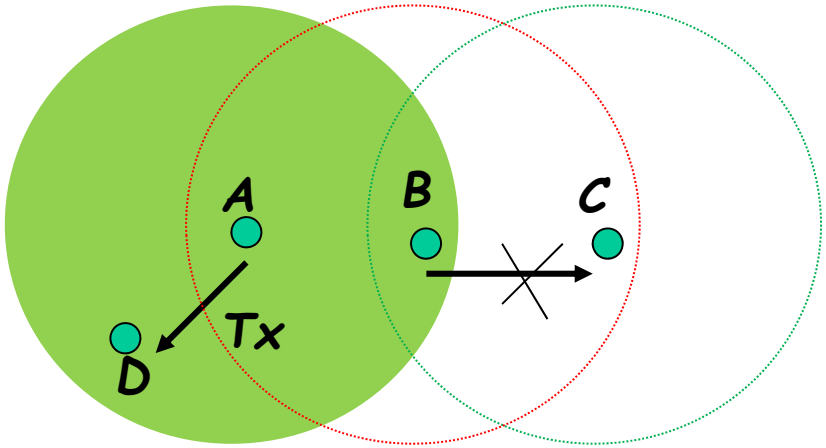
- Advantages
  - Small control frames lessen the cost of collisions (when data is large)
  - RTS + CTS provide "virtual" carrier sense which protects against hidden terminal collisions (where A can't hear C)



- *Disadvantages*

  Not as efficient as CSMA-CD

Exposed Terminal Problem



- ***Problem***
  - *A is transmitting a frame to D.*
  - *B knows that someone is transmitting.*
  - *If B transmits a frame to **C**, no problem. However, B does not want because it is unaware of D's location.*

- *The above problem is due to B being **exposed** to A's Tx.*

CSMA-CA solving Exposed Terminal Problem

When node B hears an RTS from a neighboring node A, but not the corresponding CTS, node B can deduce that it is an *exposed terminal* and is permitted to transmit to other neighboring nodes.

Random Contention Access

- Slotted contention period
  - Used by all carrier sense variants
  - Provides random access to the channel
- Operation
  - Each node selects a random back off number
  - Waits that number of slots monitoring the channel
  - If channel stays idle and reaches zero then transmit
  - If channel becomes active wait until transmission is over then start counting again

802.11 Contention Window

- Random number selected from [0,cw]
- Small value for cw
  - Less wasted idle slots time
  - Large number of collisions with multiple senders (two or more stations reach zero at once)
- Optimal cw for known number of contenders & know packet size
  - Computed by minimizing expected time wastage (by both collisions and empty slots)
  - Tricky to implement because number of contenders is difficult to estimate and can be VERY dynamic

802.11 Adaptive Contention Window

- 802.11 adaptively sets cw
  - Starts with cw = 31
  - If no CTS or ACK then increase to 2*cw+1 (63, 127, 255)
  - Reset to 31 on successful transmission
- 802.11 adaptive scheme is unfair
  - Under contention, unlucky nodes will use larger cw than lucky nodes (due to straight reset after a success)
  - Lucky nodes may be able to transmit several packets while unlucky nodes are counting down for access
- Fair schemes should use same cw for all contending nodes (better for high congestion too)

Summary

- A Perfect MAC Protocol
  - Collision avoidance to reduce wasted transmissions
    - Cope with hidden terminal problems
    - Allow exposed terminals to talk
  - Reasonable fairness

- No MAC protocol does all this!

Scheduling approaches to MAC

- About random access:
  - Simple and easy to implement
  - In low-traffic, packet transfer has low-delay
  - However, limited throughput and in heavier traffic, packet delay has no bound. A station of bad luck may never have a chance to transfer its packet.
- Scheduling approach:
  - provides orderly access to shared medium so that every station has chance to transfer

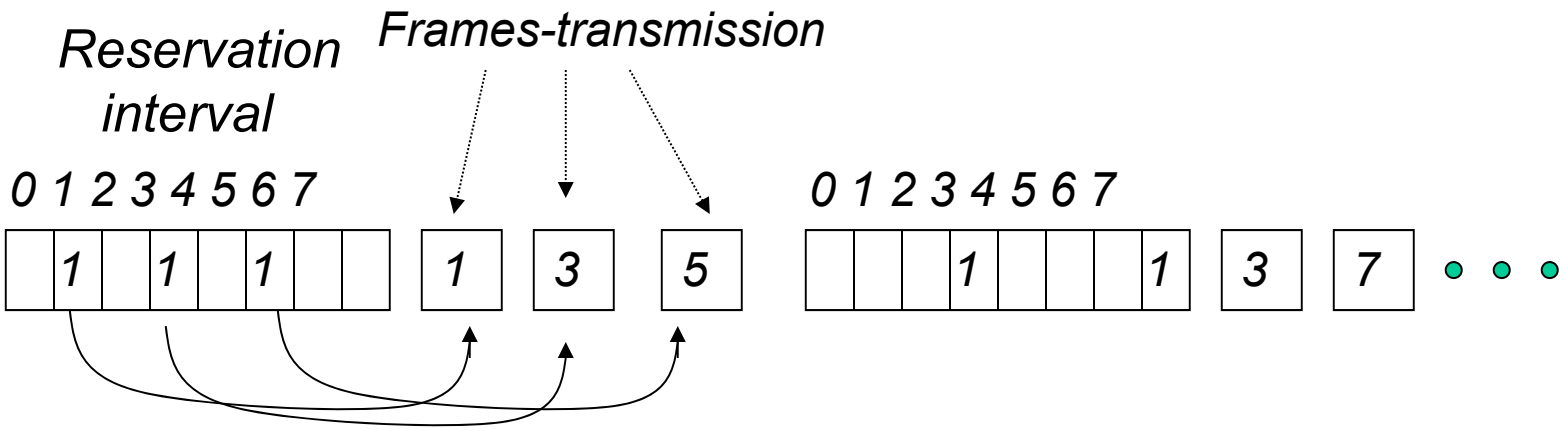
Scheduling approach—reservation protocol

- The time line has two kinds of periods:
  - Reservation interval of fixed time length
  - Data transmission period of variable frames.
- Suppose there are *M* stations, then the reservation interval has *M* minislots, and each station has one minislot.
- Whenever a station wants to transfer a frame, it waits for reservation interval and broadcasts reservation bit in its minislot.

Scheduling approach—reservation protocol (Cont.)

- By listening to the reservation interval, every station knows which stations will transfer frames, and in which order.
- The stations having reserved for their frame transfer their frames in that order
- After data transmission period, next reservation interval begins.

Reservation protocol



Scheduling approach—polling protocol

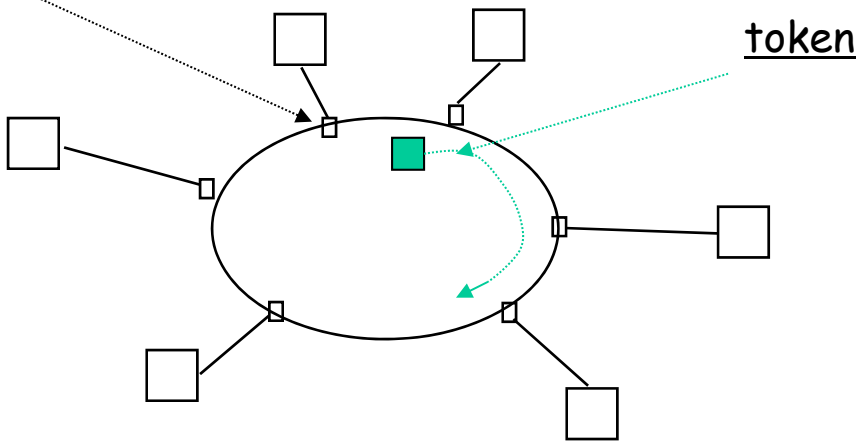
- Stations take turns accessing the medium:
  - At any time, only one station has access right to transfer into medium
  - After this station has done its transmission, the access right is handed over (by some mechanism) to the next station.
  - If the next station has frame to transfer, it transfers the frame, otherwise, the access right is handed over to the next next station.
  - After all stations are polled, next round polling from the station 1 begins.

Centralized polling vs. distributed polling

- Centralized polling: a center host which polls the stations one by one
- Distributed polling: station 1 will have the access right first, then station 1 passes the access right to the next station, which will passes the access right to the next next station, …

Station interfaces: are connected to form a ring by point-to-point lines

Stations: are attached to the ring by station interfaces.



Note: point-to-point lines, not a shared bus.

Token: a small frame, runs around the ring, whichever gets the token, it has the right to transmit data frames.

The information flows in one direction.

Station interfaces have important functions.

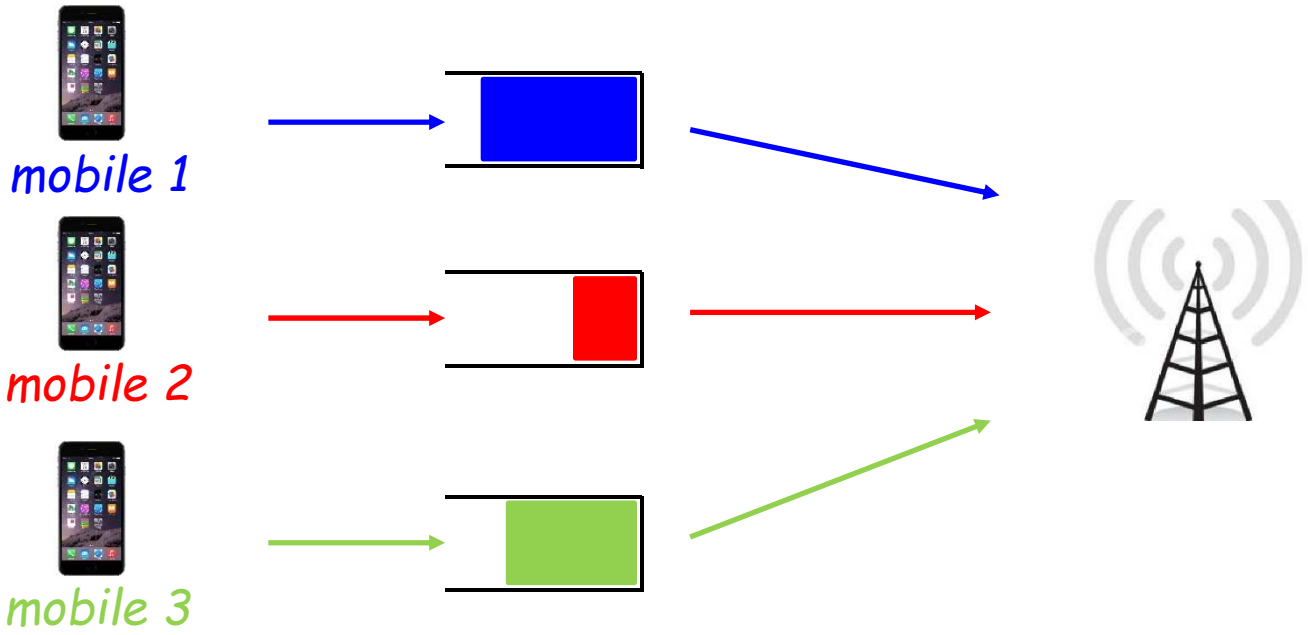
Token-passing rings – a distributed polling network

Round-Robin

Flows in the system are served in a cyclic, or round-robin order

One packet from each flow is served, during each service instant

Fairness

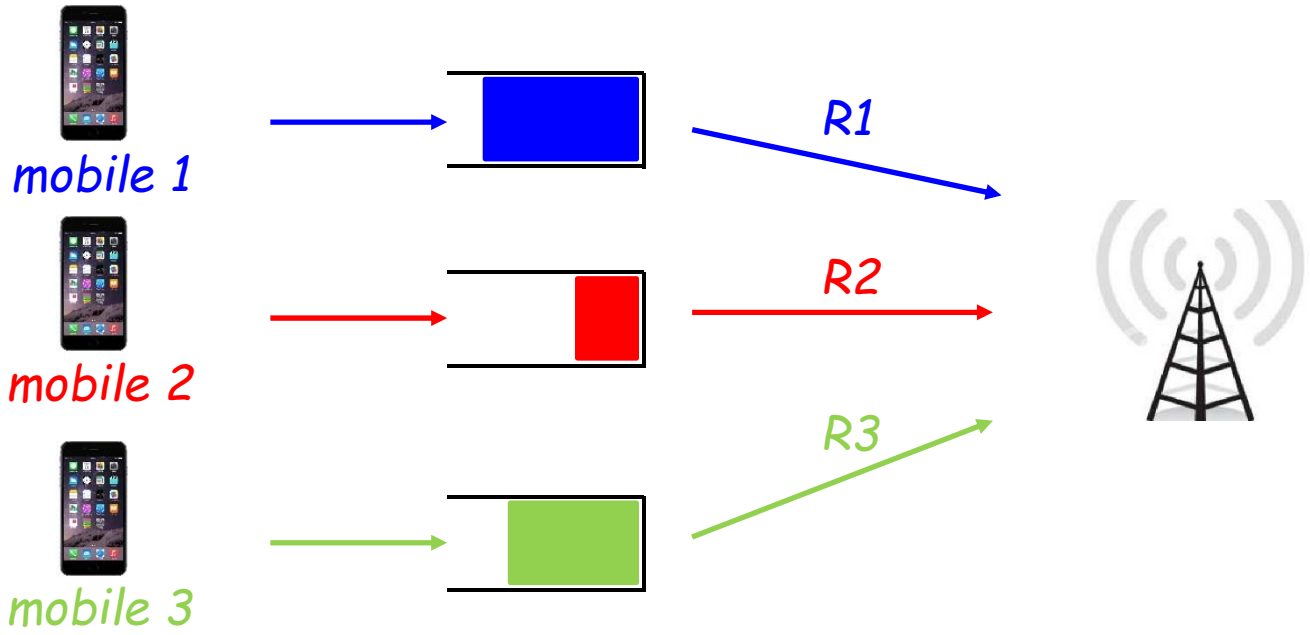


Maximum Rate

Guarantees that the Mac Scheduler will always assign resource to the mobile user with the best channel quality

Unfair: users with bad channel may starve

Maximize the total throughput of the network

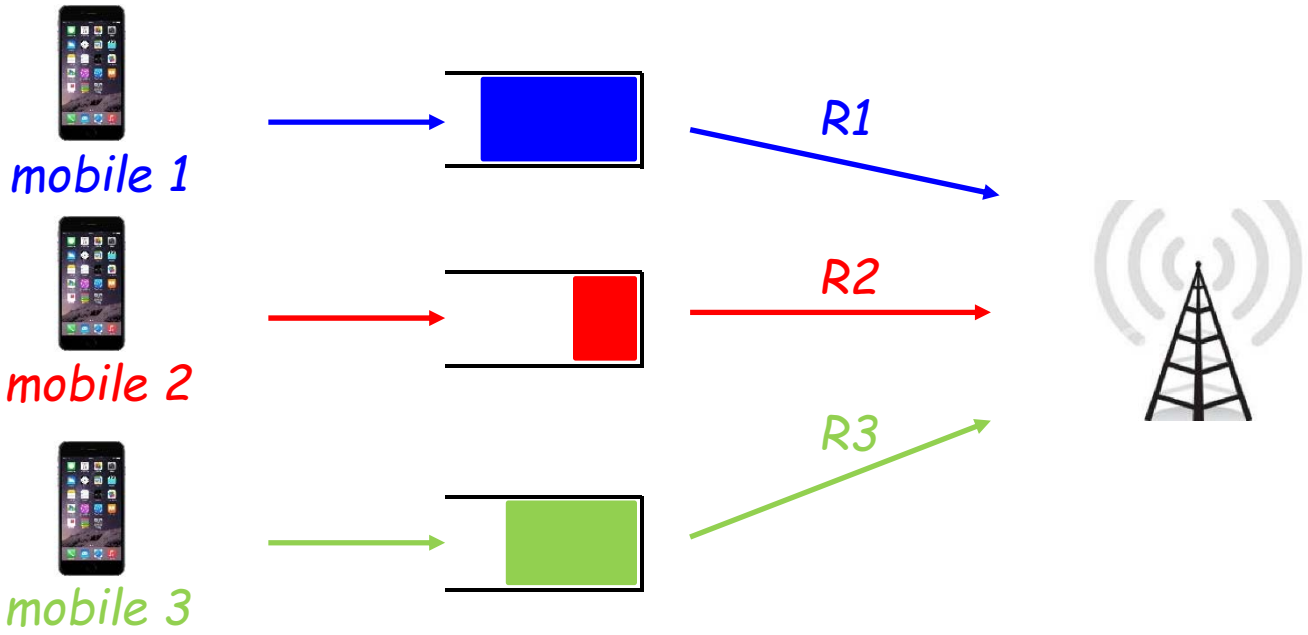


Proportional fair

Schedule user $k = arg\max_i \frac{R_i}{\overline{R}_i}$

Compute the average rate over the (recent) past for each mobile
Pick the mobile with the highest PF ratio

A good compromise between the maximum throughput and user fairness

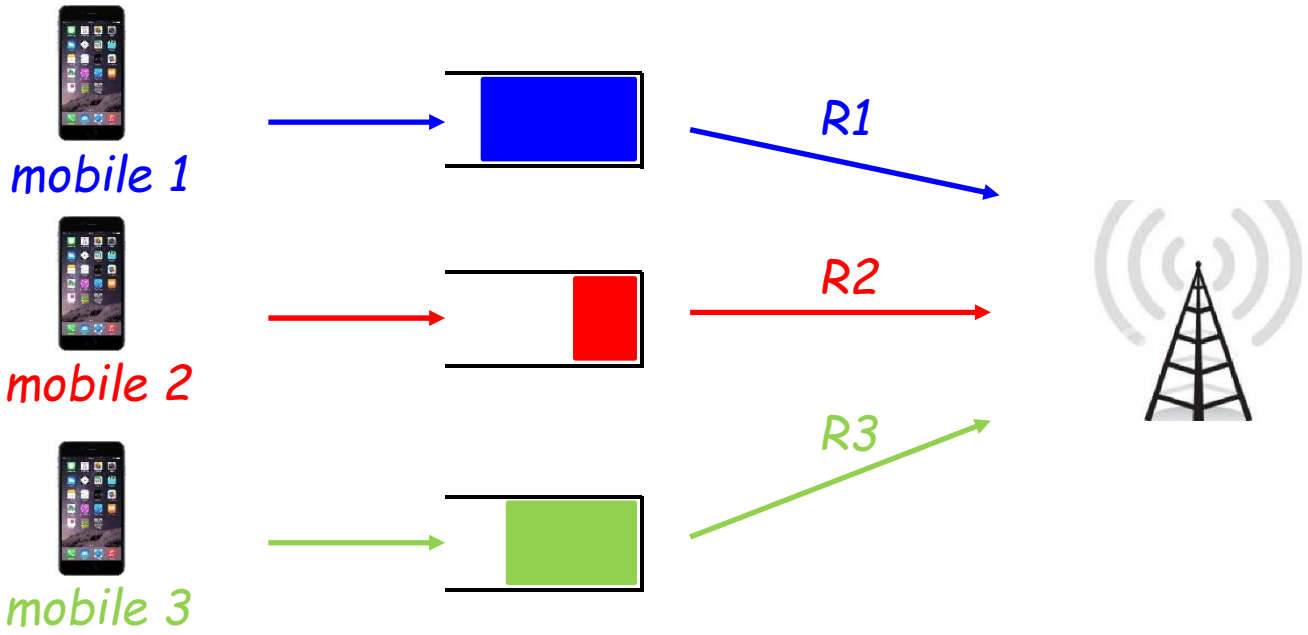


Why PF is actually fair

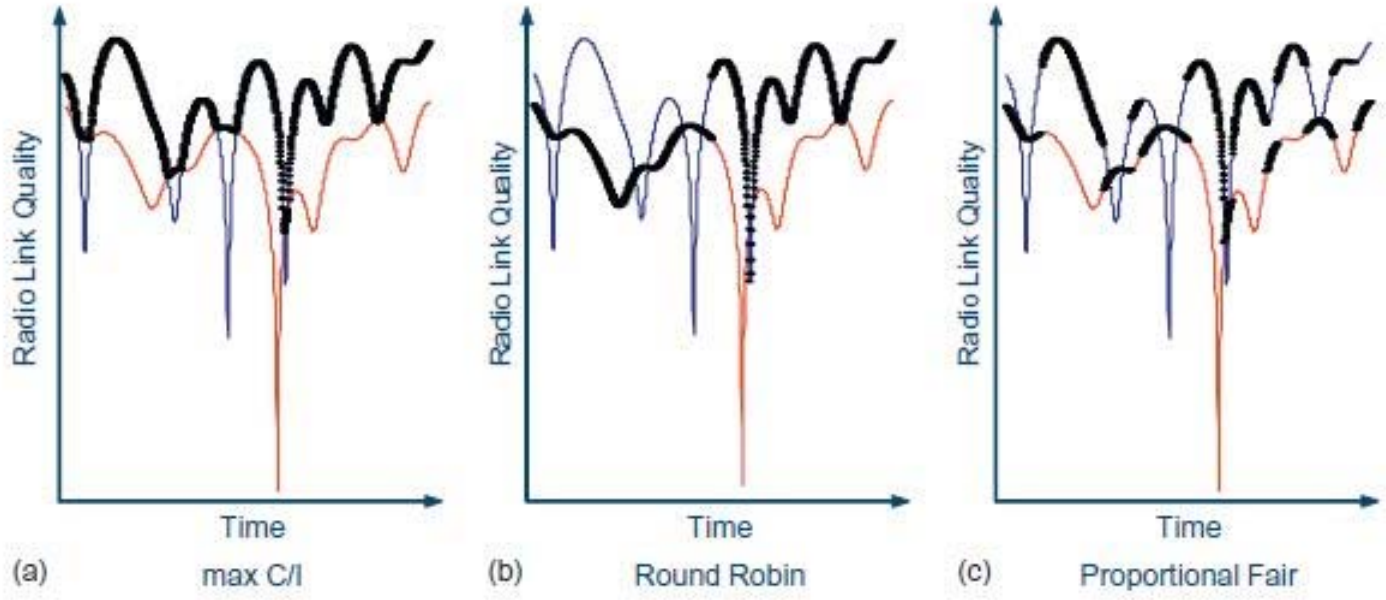
Assume all UEs have backlog

If you don't get served

- *your Avg goes to zero*
- *your priority goes to infinite*
- *you get to the front of the line*



MR, RR, PF


Radio Link Quality
Time
(a)
max C/I
Radio Link Quality
Time
(b)
Round Robin
Radio Link Quality
Time
(c)
Proportional Fair

Comparison of scheduling & random access

- Scheduling
  - Methodical orderly access: dynamic form of time division multiplexing, round-robin (only) when the stations have information to send.
  - Less variability in delay, supporting applications with stringent delay requirement. In high load, performance is good. E.g., token-ring may reach nearly 100 percent of performance when all stations have plenty of information to send.
  - Some channel bandwidth carries explicit scheduling information.

Comparison of scheduling & random access (Cont.)

- Random access
  - Chaotic, unordered access
  - If rich bandwidth and light load, random access has low delay, otherwise, delay is undeterminably large.
  - Quite a lot bandwidth is used in collision to alert stations of the presence of other transmissions.

Summary of T4

- We have studied the principles behind data link layer services:
  - error detection and correction (by the receiver)
  - reliable data transfer
  - link access by sharing a broadcast channel: MAC