Direct Link Networks: Media Access Control and Ethernet

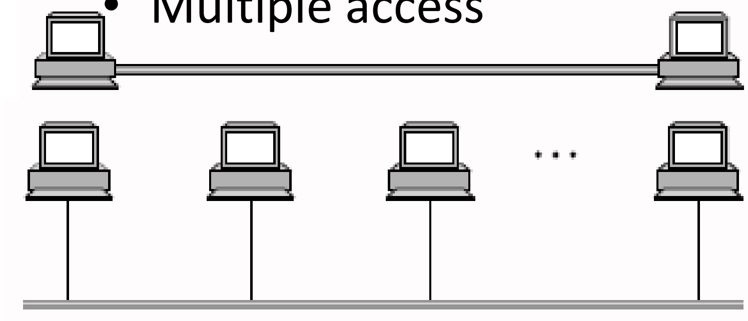
Hui Chen

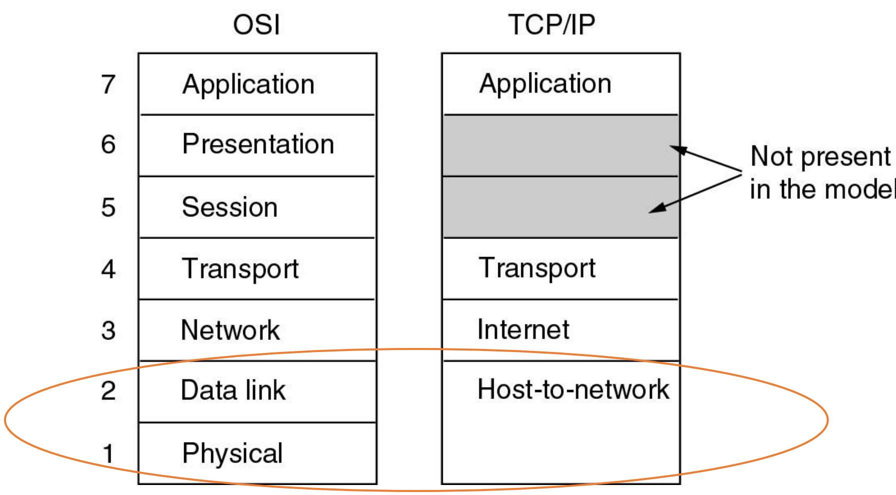
Department of Computer & Information Science

CUNY Brooklyn College

Direct Link Networks

- Types of Networks
  - Point-to-point
  - Multiple access

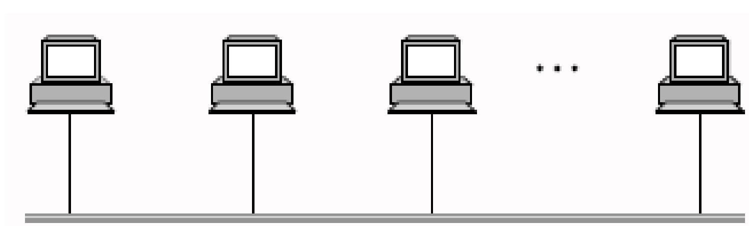




- Encoding
  - Encoding bits onto transmission medium
- Framing
  - Delineating sequence of bits into messages
- Error detection
  - Detecting errors and acting on them
- Reliable delivery
  - Making links appear reliable despite errors
- **Media access control**
  - **Mediating access to shared link**

Direct Link Networks

- Types of Networks
  - Point-to-point
  - Multiple access



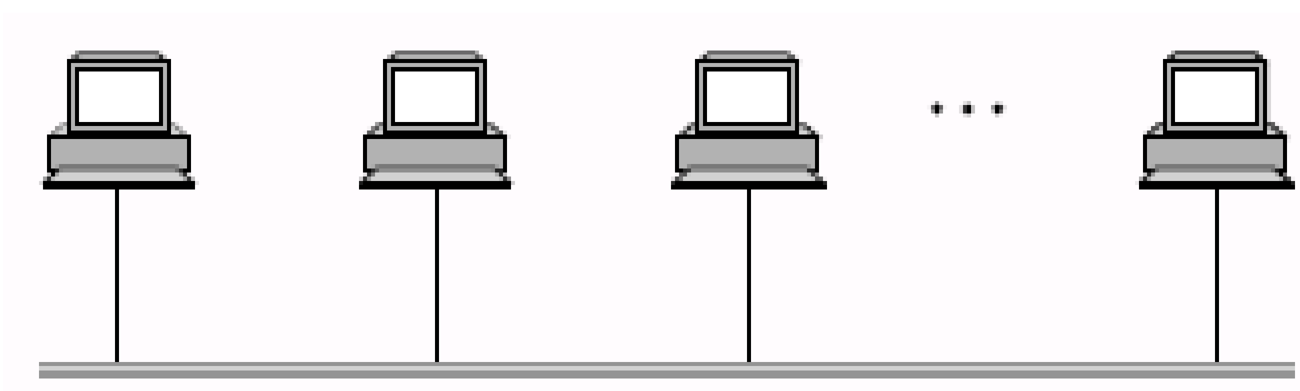
- Encoding
  - Encoding bits onto transmission medium
- Framing
  - Delineating sequence of bits into messages
- Error detection
  - Detecting errors and acting on them
- Reliable delivery
  - Making links appear reliable despite errors
- **Media access control**
  - **Mediating access to shared link**

Outlines

- Media Access Control
- Contention Resolution Approaches
  - Performance analysis
- Example networks
  - Ethernet
  - Wireless LAN (WiFi)

Multiple Access Network

- More than two nodes share a single physical link
  - Bus (Ethernet/802.3)
  - Ring (Token-ring/802.5)
  - Wireless (Wireless LAN/802.11)



Multiple Access Networks

- Characteristics
  - A transmitter can be heard by multiple receivers
  - A receiver can hear multiple transmitters
- Problems
  - How to identify nodes?
    - Cannot identify node by stating “the sender” and “the receiver”
    - Addressing
  - How to mediate nodes’ access to the link?
    - Interference and collision of transmission
    - Media access control

Media Access Control

- How to allocate a multi-access channel among multiple competing users
  - Rules that each node must follow to communicate and avoid interference and collision

Media Access Control Approaches

- Can be classified into two categories
  - Static
    - Channel’s capacity is divided into fixed portions
    - Each node is allocated a portion for all time
    - Better suited when traffic is predictable
    - Examples: TDMA, FDMA, and CDMA
  - Dynamic
    - Allocate channel capacity based on the traffic generated by the users
    - Try to obtain better channel utilization and delay when traffic is unpredictable
    - Examples: ALOHA, Slotted ALOHA, and MACA

Dynamic Channel Allocation

- Perfectly scheduled approaches
- Contention resolution approaches
- Approaches that combined both scheduling and contention resolution

Perfectly Scheduled Approaches

- A schedule is dynamically formed based on which users have data to send
- Users transmit contention free according to the schedule
- Schedule can be formed by polling, reservation, etc.

Contention Resolution Approaches

- Contention
  - A node transmits a packet when it has data to send
  - A collision occurs if multiple nodes transmit at the same time
  - Packets/Frames must be retransmitted based on some rule
- Examples
  - Pure ALOHA, Slotted ALOHA
  - MACA, MACAW
  - CSMA, CSMA/CD and CSMA/CA
  - D-MAC

Performance Metrics

- Latency (delay)
  - In particular, when traffic load is low
- Throughput (channel efficiency)
  - In particular, when traffic load is high
- Jitter

Performance Analysis

- Multiple-access model
- Pure ALOHA
- Slotted ALOHA
- CSMA

Performance Analysis

- References and Further Readings

  - Kleinrock, L.; Tobagi, F.A, "Packet Switching in Radio Channels: Part I--Carrier Sense Multiple-Access Modes and Their Throughput-Delay Characteristics," Communications, IEEE Transactions on , vol.23, no.12, pp.1400,1416, Dec 1975. doi: 10.1109/TCOM.1975.1092768.
  - Abramson, Norman, "Development of the ALOHANET," Information Theory, IEEE Transactions on , vol.31, no.2, pp.119,123, Mar 1985. doi: 10.1109/TIT.1985.1057021.

Multiple-Access Model

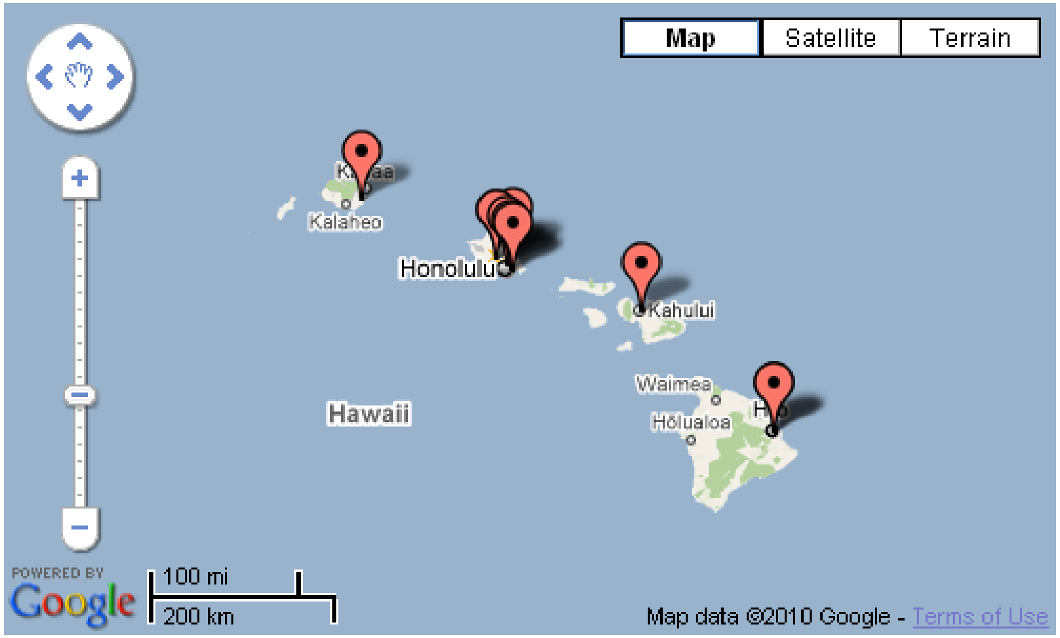
- User Model
  - N users (nodes, or stations).
  - At each station, frames to be transmitted randomly arrive
  - The arrivals are independent of each other
- Channel model
  - All communications of the N users rely on one single shared channel
- Transmission model
  - Frames are garbled and cannot be received, whenever the frames overlap in time (called a *collision*)
  - Only errors allowed are introduced by collisions. If no collisions, a frame is successfully received
- Feedback model
  - All stations are able to detect if a frame is collided with another or successfully sent after a complete frame is sent

Approaches in Feedback Model

- Listen while transmitting
  - Typically, collisions can be detected in a delay of ~RTT
    - Ethernet (link length, 4 segments, 2500 meter): 51.2 μs
    - Satellite: it may take as much as 270 ms delay
- If not possible, acknowledgements are used
  - *Not* until recently is it considered *possible to listen while transmitting* on wireless networks
  - Dinesh Bharadia, Emily McMilin, and Sachin Katti. 2013. Full duplex radios. In *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM* (SIGCOMM '13). ACM, New York, NY, USA, 375-386. DOI=10.1145/2486001.2486033. http://doi.acm.org/10.1145/2486001.2486033

Pure ALOHA

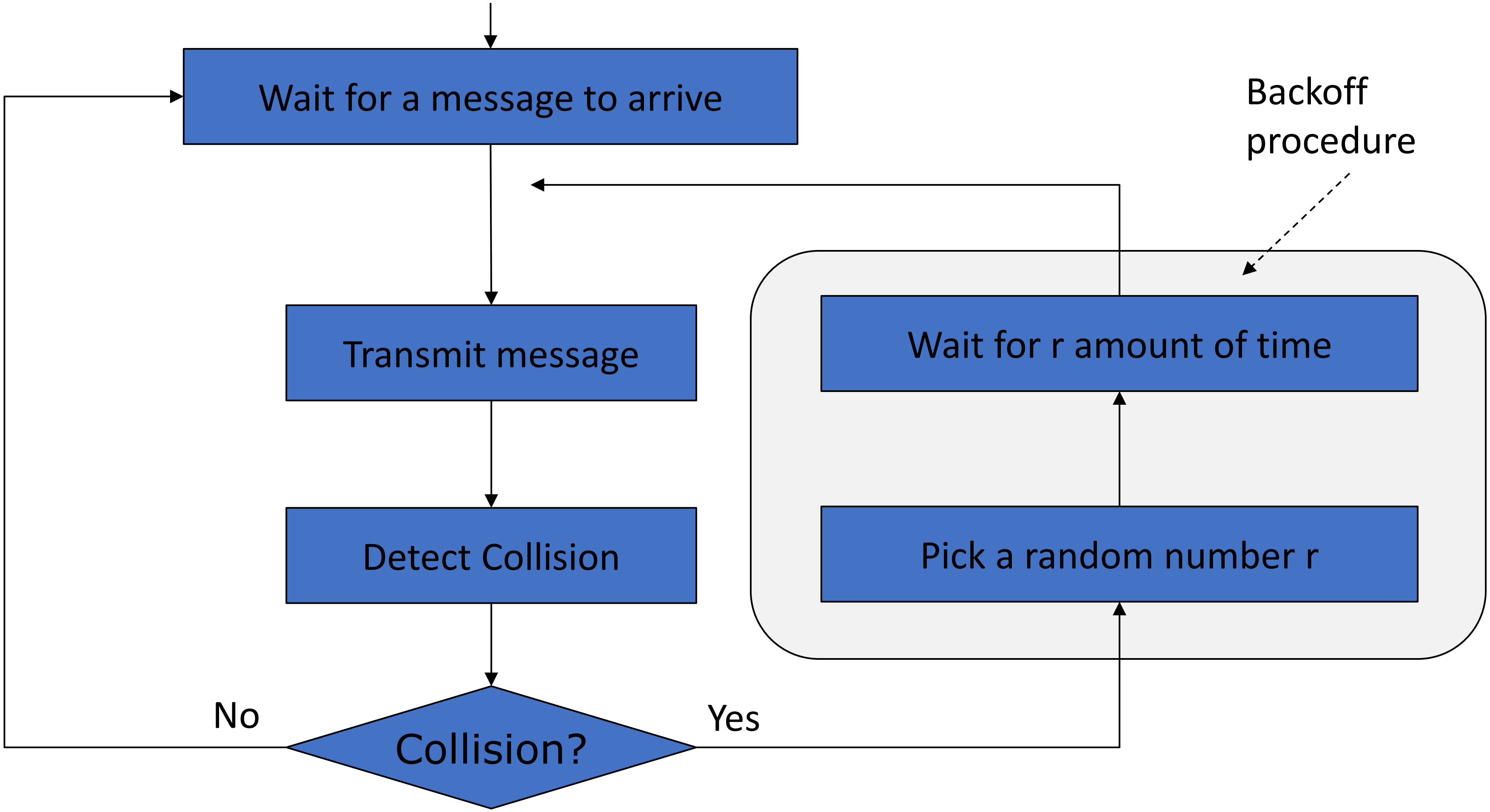
- Initially developed by Norman Abramson, University of Hawaii in 1970's
- Served as a basis for many contention resolution protocols



Pure ALOHA: Protocol

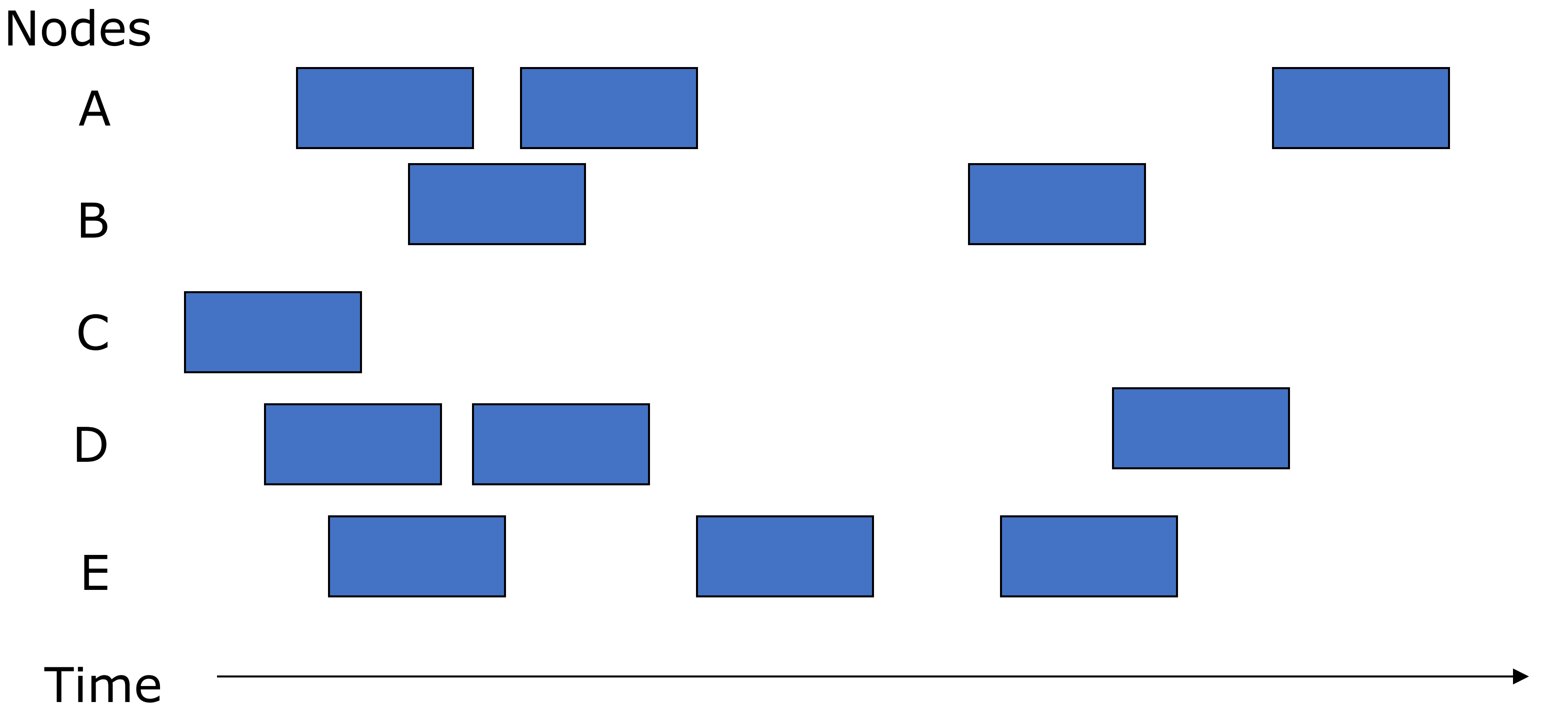
- Transmit message : A node transmits whenever it has data to send
- Detect collision: The sender wait to see if a collision occurred after the complete frame is sent
  - Note: a collision may occur if multiple nodes transmit at the same time
- Random backoff: If collision occurs, all the stations involved in collision wait a random amount of time, then try again
- Questions
  - Is it a good protocol? (how much can the throughput be?)
  - How would we choose the random amount of waiting time?

Pure ALOHA: Protocol



Pure ALOHA: Throughput Analysis

- Frames are transmitted and retransmitted at completely arbitrary times

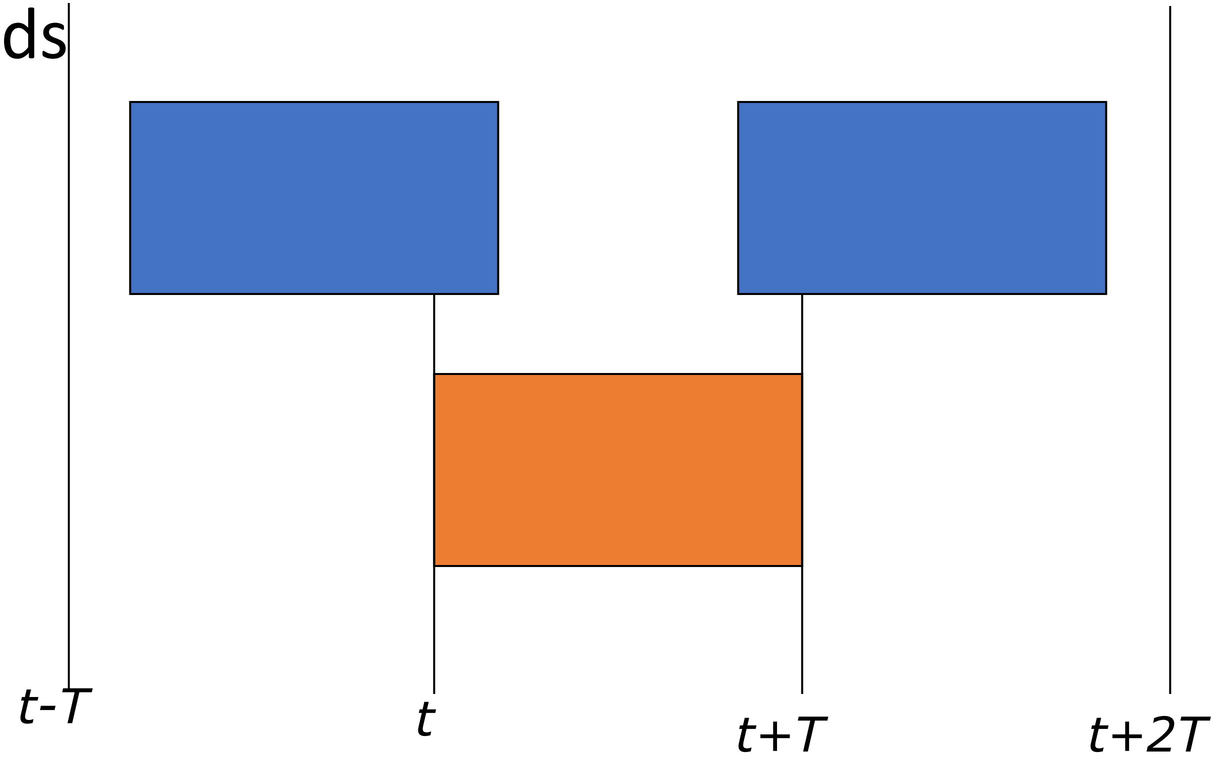


Pure ALOHA: Throughput Analysis

- Assume
  - Infinite number of nodes
  - Fixed length frames. Denote length as T
  - Overall arrival of frames is a Poisson process with rate λ frames/second
- Then, denote S as the number of frames arriving in T seconds
  - $S = \lambda T$
- In case of a collision, retransmission happens
  - New transmission and retransmission combined (all transmissions) is a Poisson process
  - Let the rate be G attempts per T seconds
- Note that
  - $S \leq G$
  - Equality only if there are no collisions.
- Assume the system is in a stable state and denote the probability of a successful transmission by P_0
  - $S = GP_0$

Vulnerable Period/Contention Window

- A frame is successfully transmitted, if there are no frames transmitted in the contention window of 2T seconds



Frames Generated in Vulnerable Period

- Vulnerable Period: 2T seconds
- The rate of all transmissions in 2T seconds: 2G
- The probability that k frames are generated during 2T seconds is given by a Poisson distribution

$$\Pr[k] = \frac{(2G)^k e^{-2G}}{k!}$$

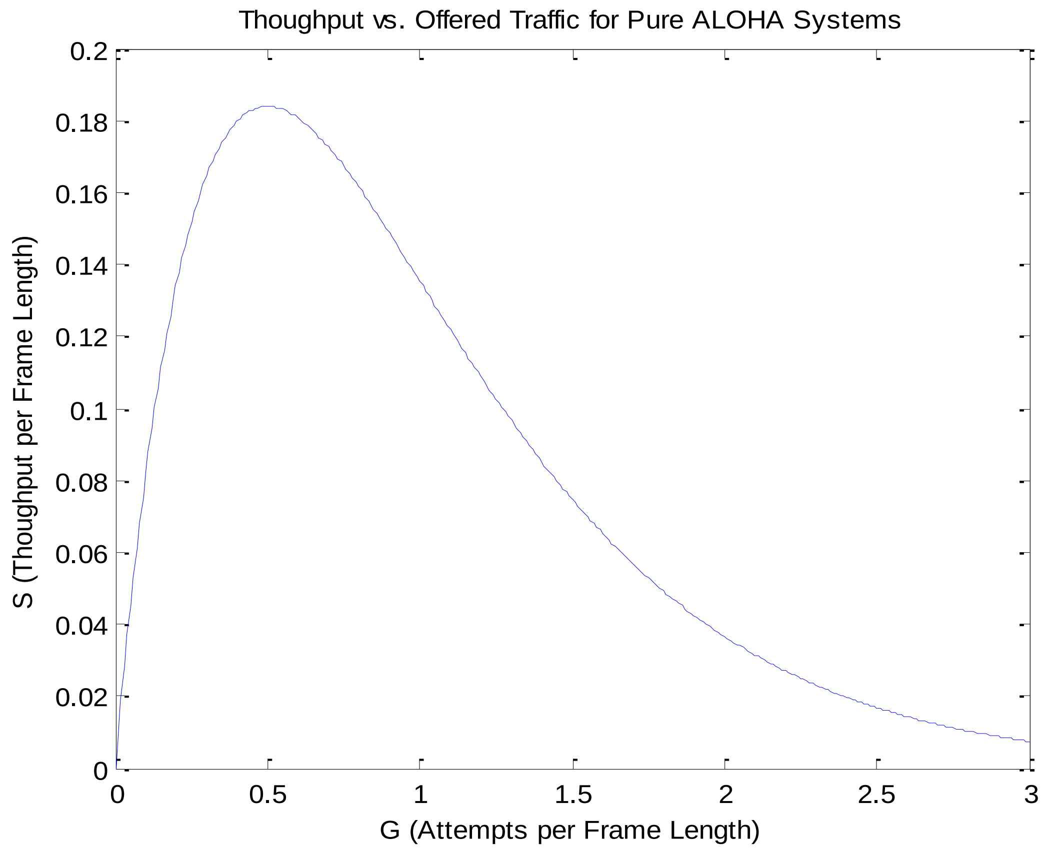
- The probability of no other frames being initiated (new transmission and retransmission) during the entire vulnerable period is

$$S = GP_0 = G\frac{(2G)^0 e^{-2G}}{0!} = Ge^{-2G}$$

Throughput of Pure ALOHA

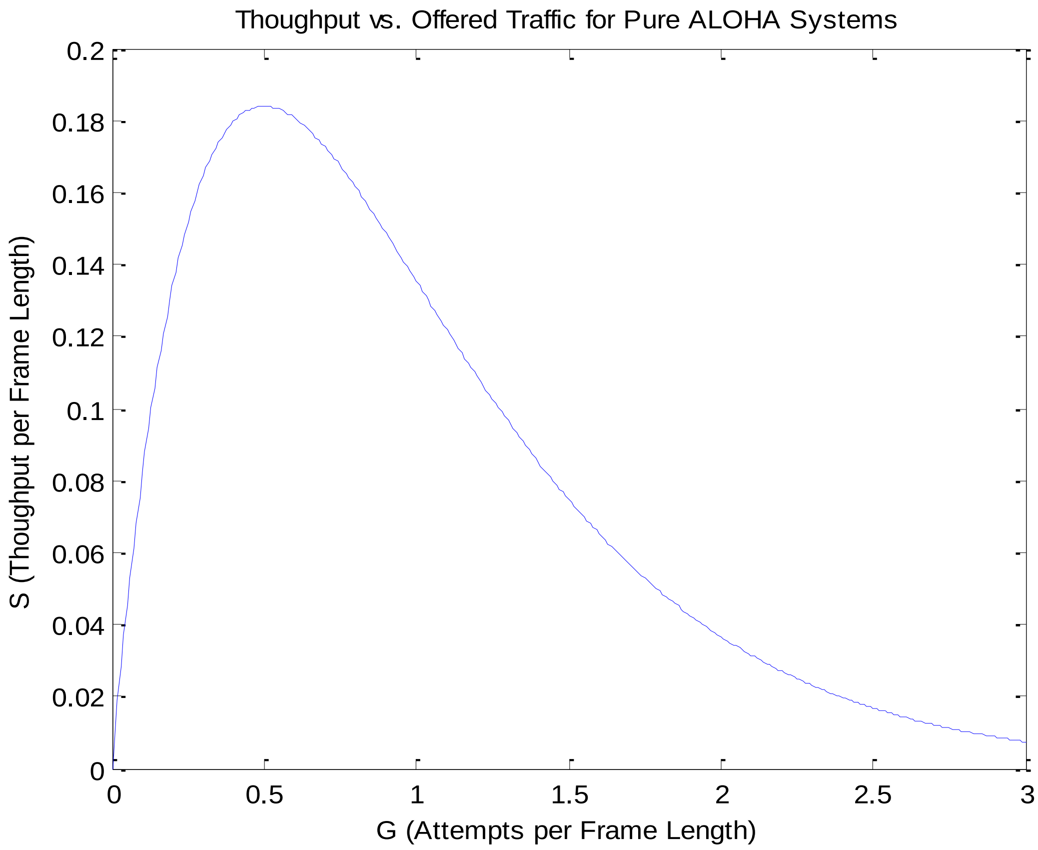
- Let us graph it

$$S = Ge^{-2G}$$

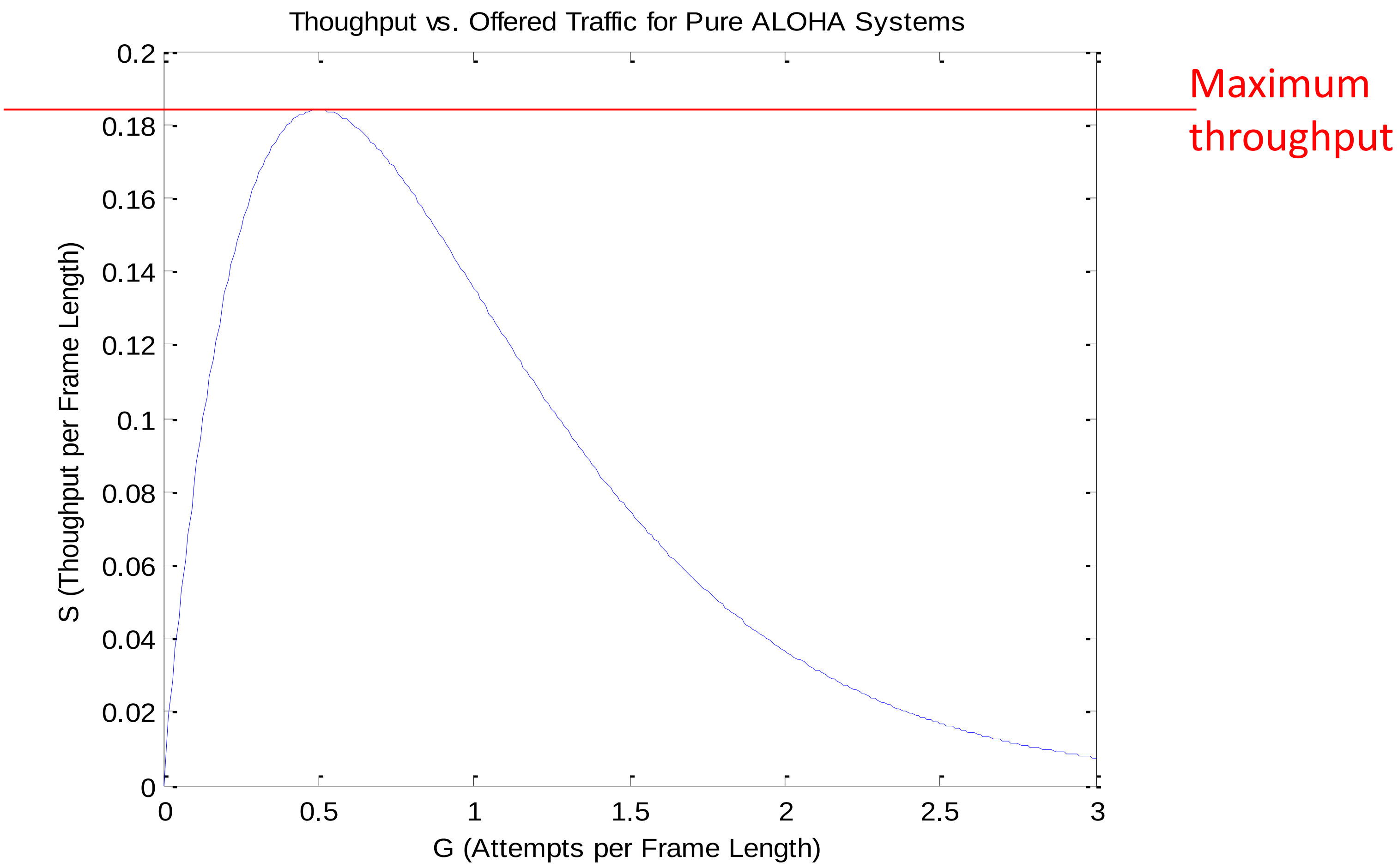


Throughput of Pure ALOHA

- What is the implication?



Maximum Throughput of Pure ALOHA



Maximum Throughput of Pure ALOHA

- The derivative is 0

$$\frac{dS}{dG} = \frac{dGe^{-2G}}{dG}$$

$$= e^{-2G} - 2Ge^{-2G}$$

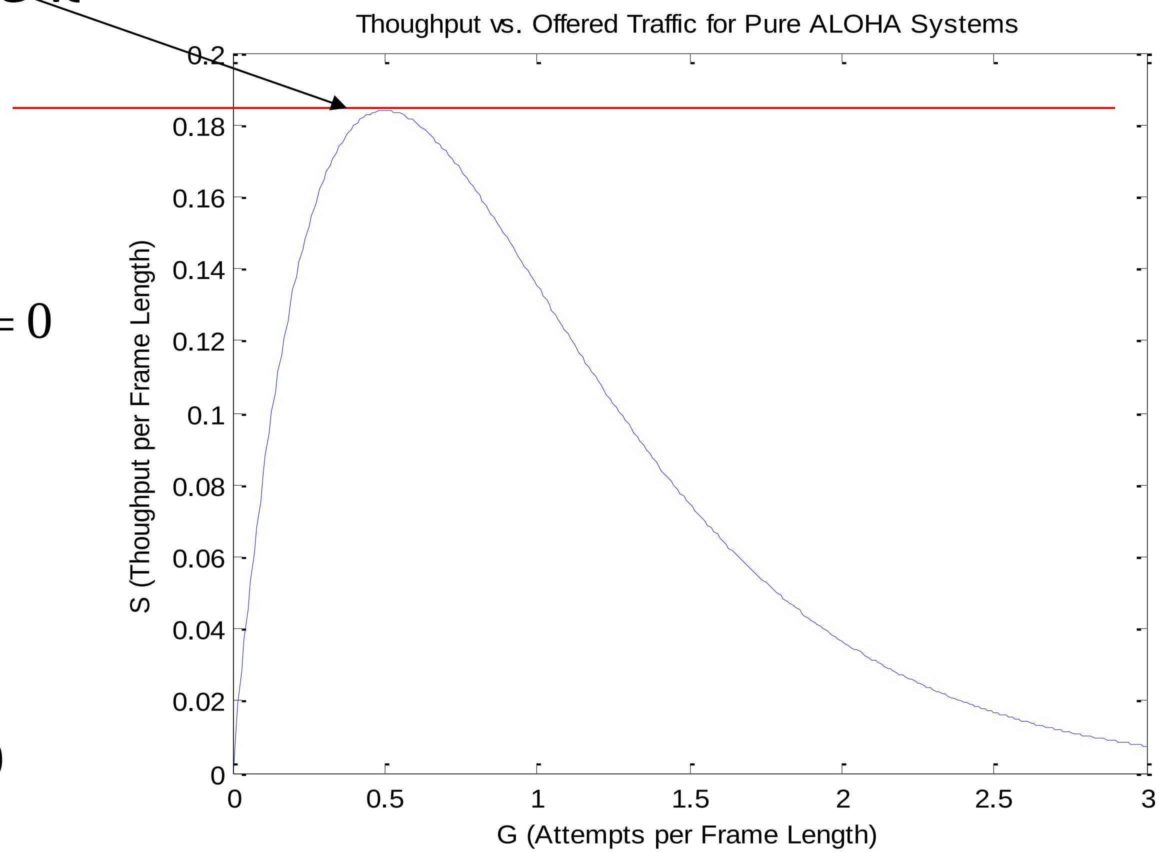
$$\frac{dS}{dG} = e^{-2G} - 2Ge^{-2G} = 0$$

$$G^* = \frac{1}{2}$$

$$S = G^* e^{-2G^*}$$

$$= \frac{1}{2} e^{-2\frac{1}{2}} \approx 0.1839$$

$$S = Ge^{-2G}$$



Pure ALOHA: Remark

- Considered a simplified analysis of a pure Aloha
  - Found that the maximum throughput is limited to be at most 1/(2*e*).
  - Not taken into account
    - How the offered load changes with time
    - How the retransmission time may be adjusted.
- Channel utilization of a *busy* Pure ALOHA system is 18%
- What improvement can we make?

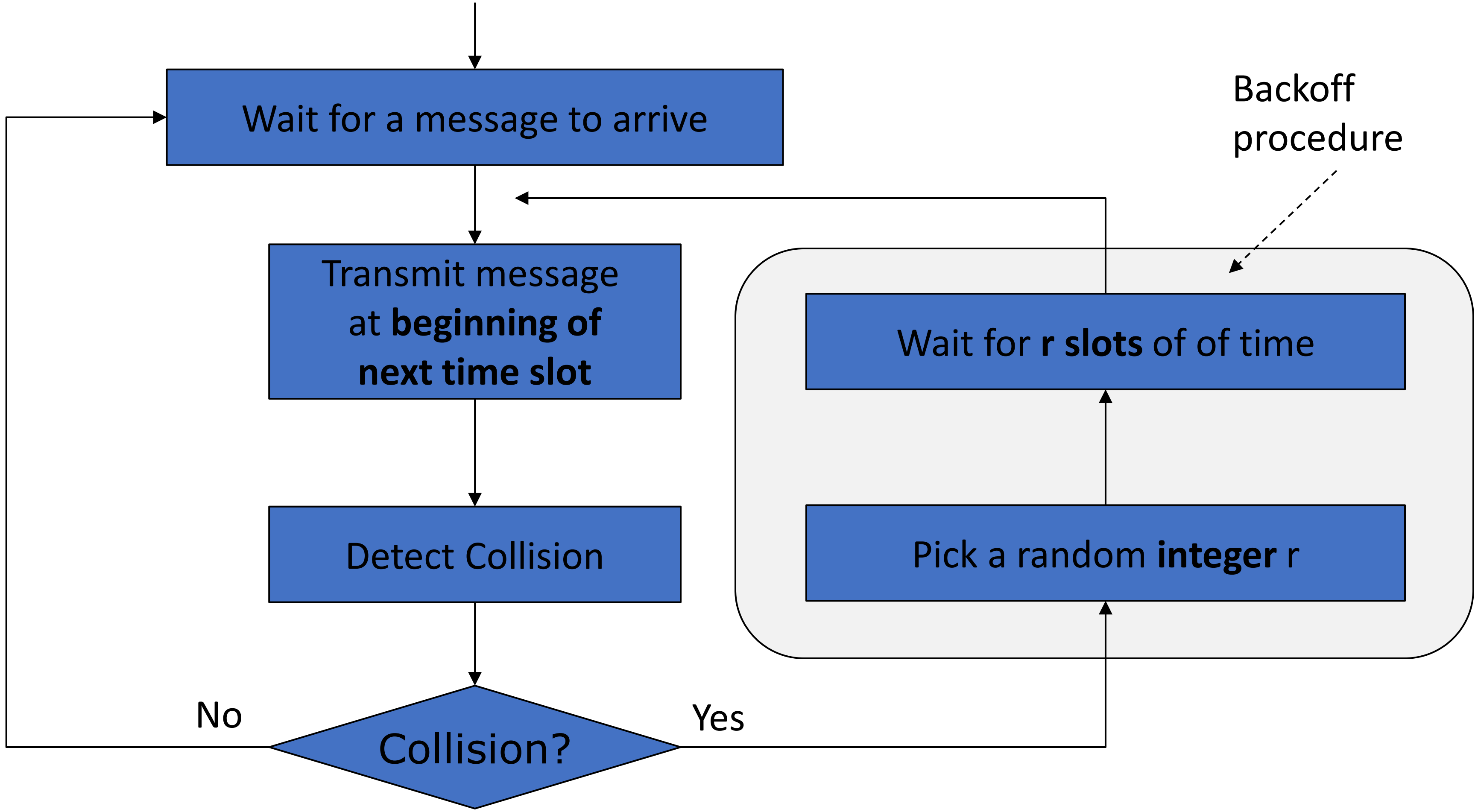
Pure ALOHA: Remark

- What improvement can we make?
  - Collision causes retransmission and reduces throughput
  - Can we reduce chance of collisions?
    - Collisions happen within the Vulnerable Period/Contention Window.
    - Can we shorten the Vulnerable Period/Contention Window?
    - Slotted ALOHA

Slotted ALOHA

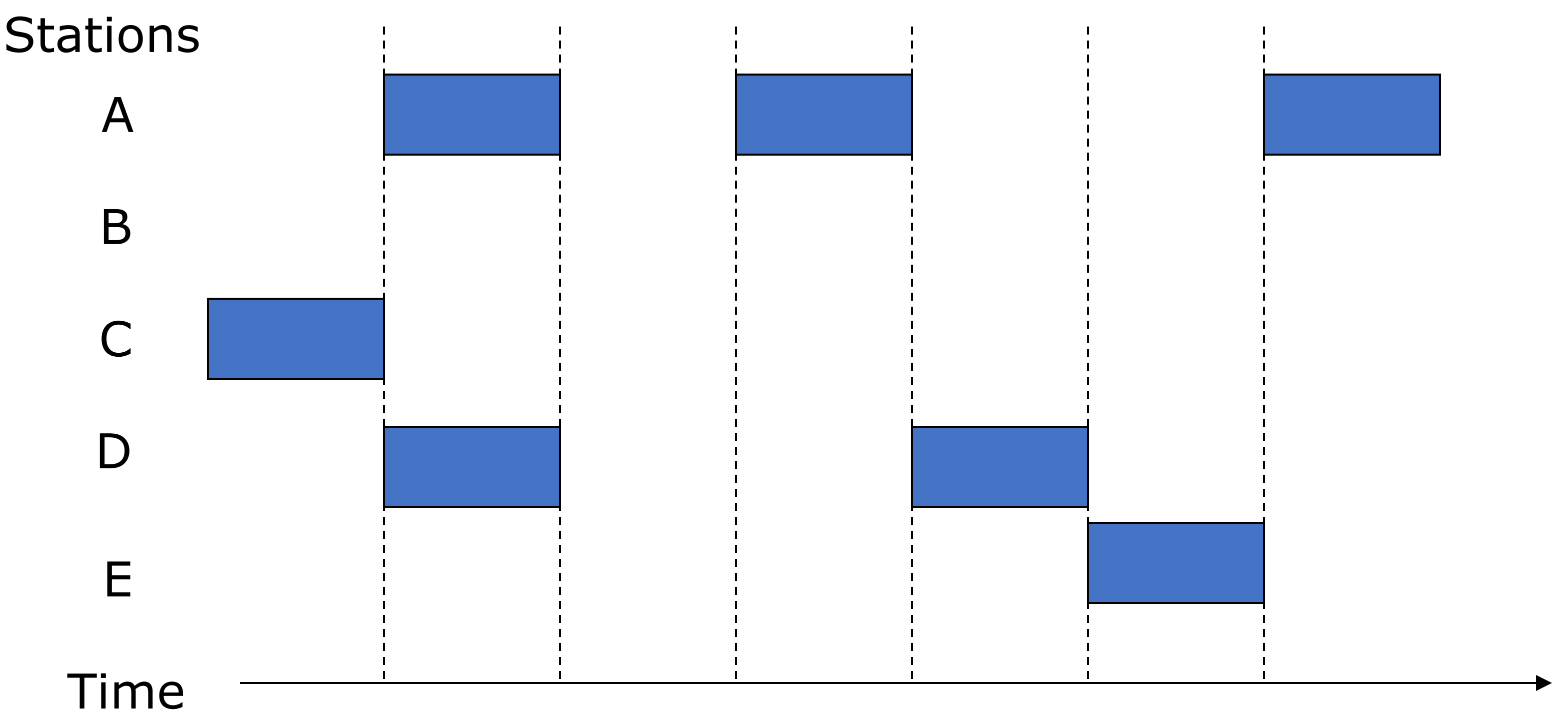
- Improvement to Pure ALOHA
  - Divided time into discrete intervals
  - Each interval corresponds to a frame
  - Require stations agree on slot boundaries

Slotted ALOHA: Protocol



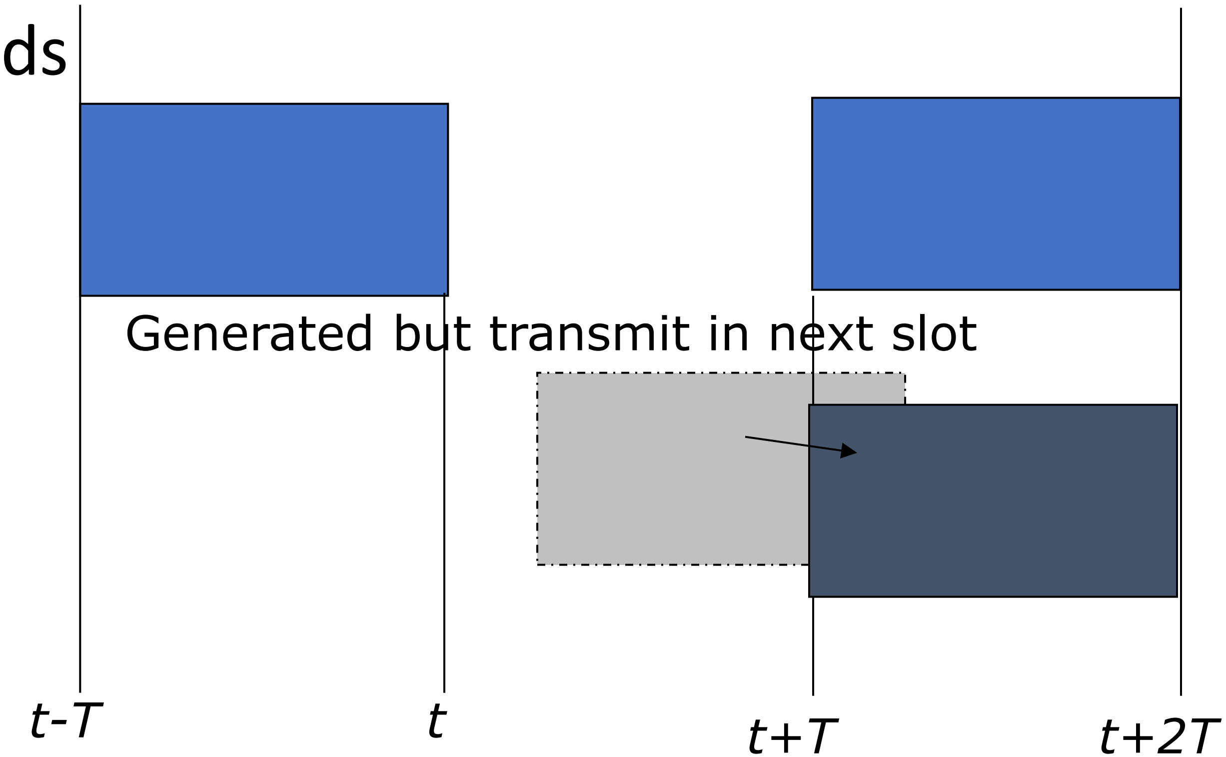
Slotted ALOHA: Throughput Analysis

- Time is slotted



Vulnerable Period/Contention Window

- A frame is successfully transmitted, if there are no frames transmitted in the contention window of T seconds



Frames Generated in Vulnerable Period

- Vulnerable Period: T seconds
- The rate of all transmissions in T seconds: G
- The probability that k frames are generated during T seconds is given by a Poisson distribution

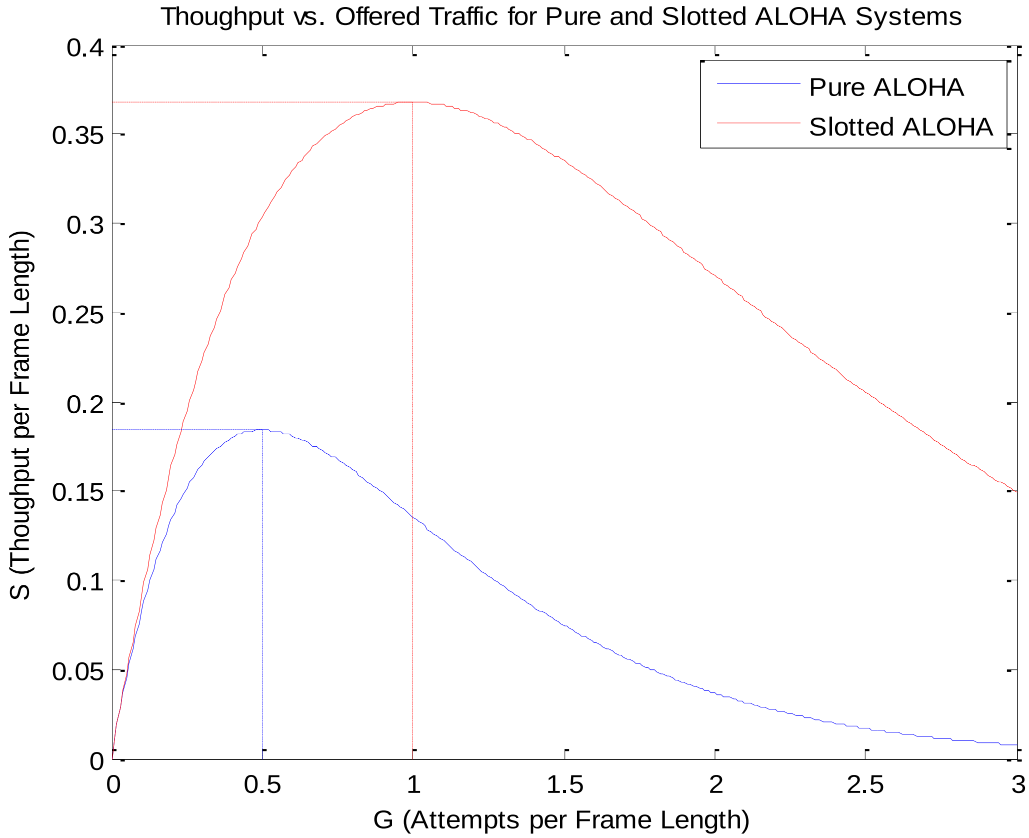
$$\Pr[k] = \frac{G^k e^{-G}}{k!}$$

- The probability of no other frames being initiated (new transmission and retransmission) during the entire vulnerable period is

$$S = GP_0 = G\frac{G^0 e^{-G}}{0!} = Ge^{-G}$$

Throughput of Slotted ALOHA

$$S = Ge^{-G}$$



Exercise

- Derive the maximum throughput of the Slotted ALOHA protocol
- How much is the maximum throughput?
- Note

$$S = Ge^{-G}$$

Implications of Performance Analysis (1)

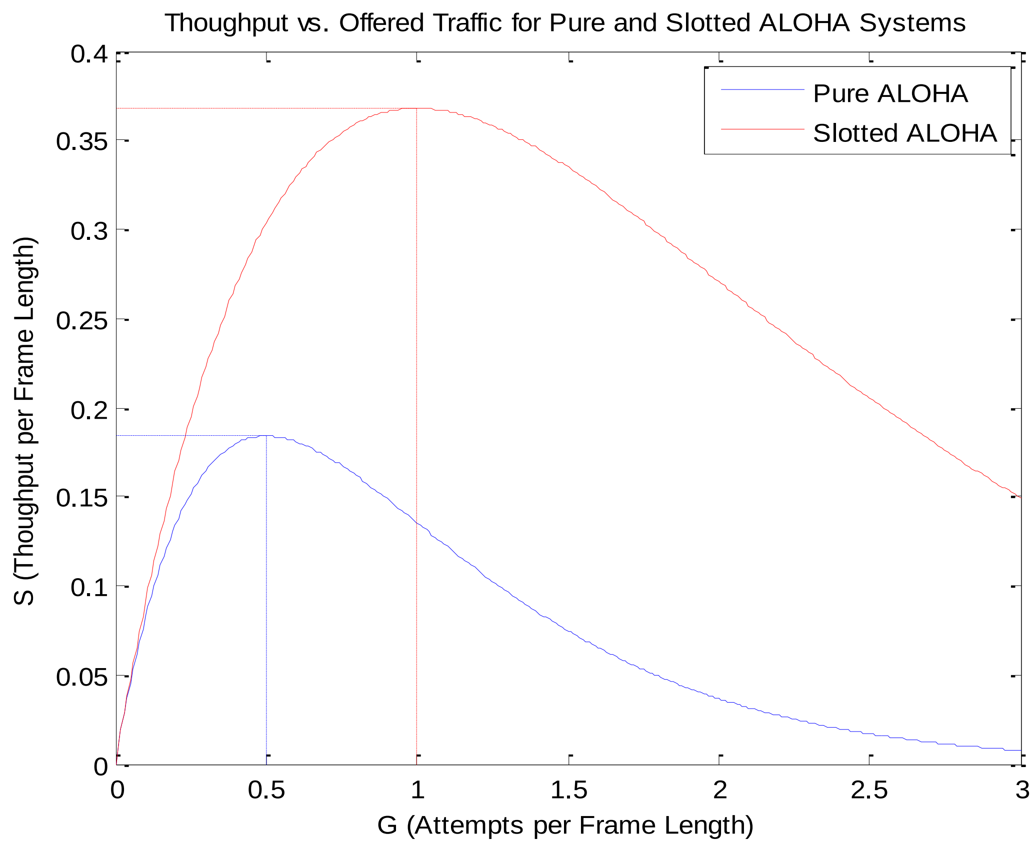
- In original ALOHA system, packets are of fixed size of 34 ms. Assume each active user sending a message packet at an average rate of once every 60 seconds. Estimate maximum number of users does the system can concurrently support?
- Answer:
    - Maximum throughput = maximum channel utilization = 1/(2e) → channel can only be *1/(2e)* full.
    - packet rate: $\lambda = 1/60$
    - Packet length: $\tau = 34$ ms
    - Maximum # of concurrent users: k_{max}
    - $k_{max}\lambda\tau = 1/(2e)$
    - $k = 1/(2e\lambda\tau) \approx 1/(2\times 2.7183\times 1/60\times 0.034) \approx 324$

Application of Performance Analysis (2)

- In an ALOHA system, packets are 816 bits and link bandwidth is 24 kbps. Assume each active user sending a message packet at an average rate of once every 60 seconds. Estimate maximum number of users does the system can concurrently support?
- Answer:
  - Maximum throughput = maximum channel utilization = 1/(2e) → channel can only be *1/(2e)* full.
  - packet rate: $\lambda = 1/60$
  - Packet length: $\tau = 816/24\ kbps = 816/24000 = 0.034\ sec = 34\ ms$
  - Maximum # of concurrent users: k_{max}
  - $k_{max}\lambda\tau = 1/(2e)$
  - $k = 1/(2e\lambda\tau) \approx 1/(2\times2.7183\times1/60\times0.034) \approx 324$

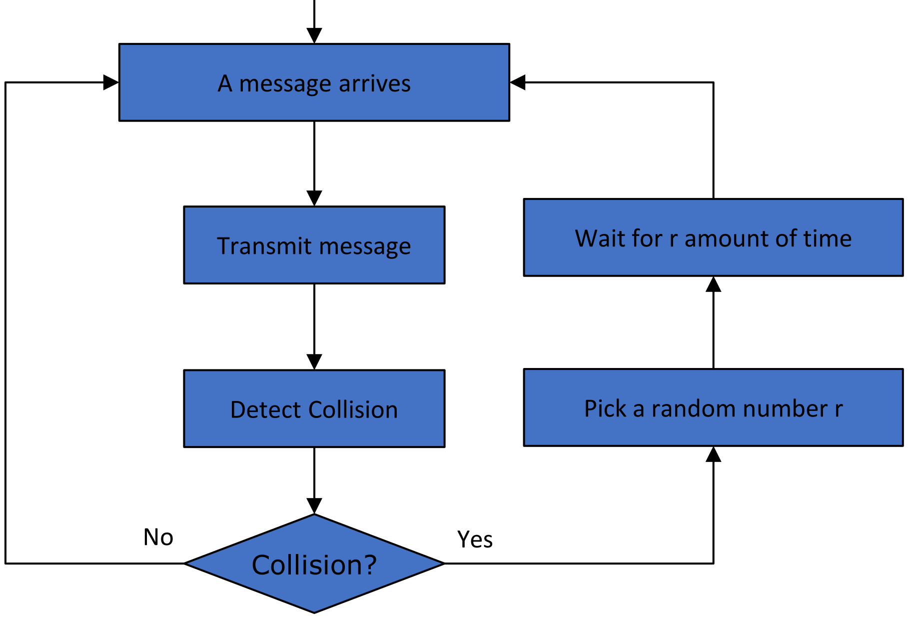
Making Further Improvements?

- Maximum throughputs are small

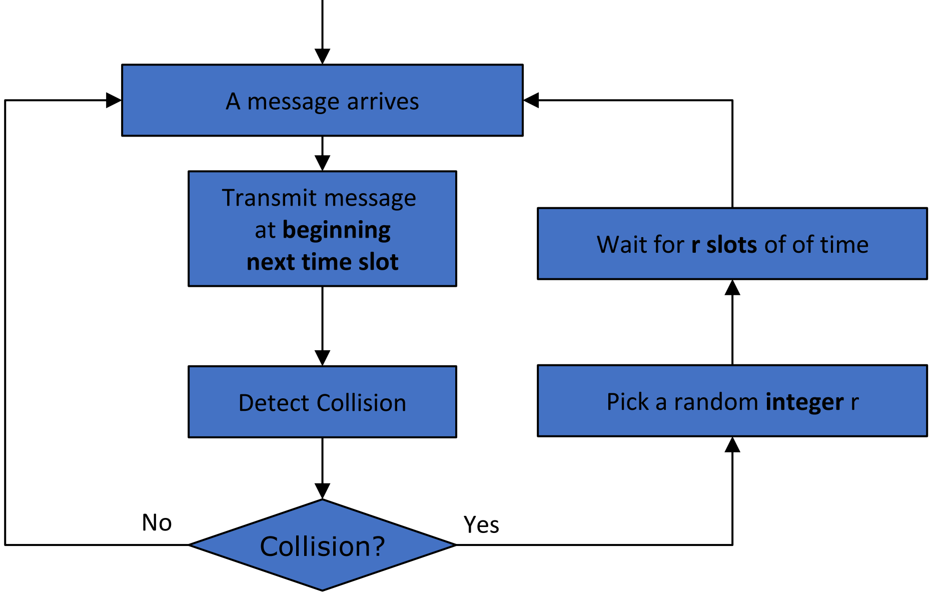


Making Further Improvements?

Pure ALOHA

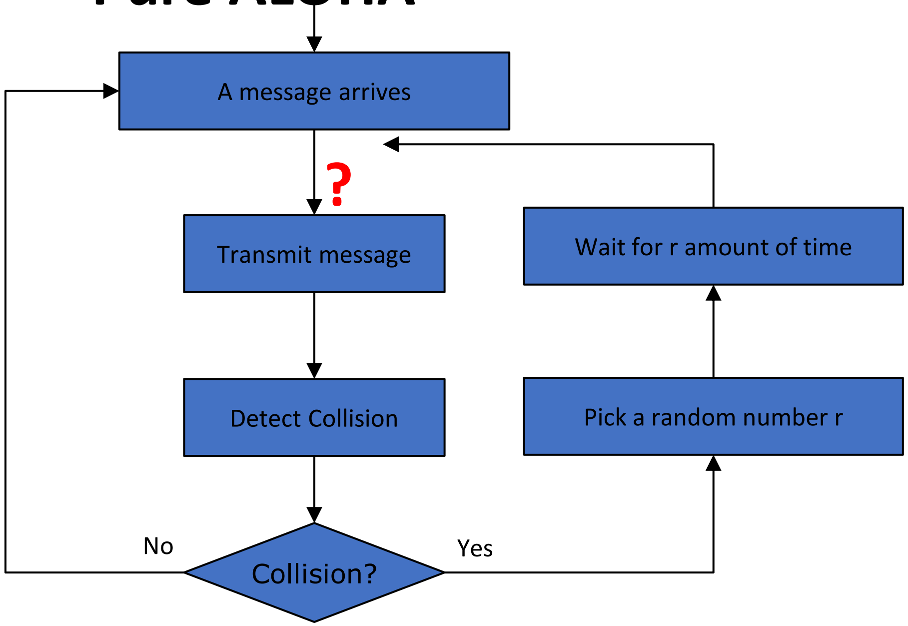


Slotted ALOHA

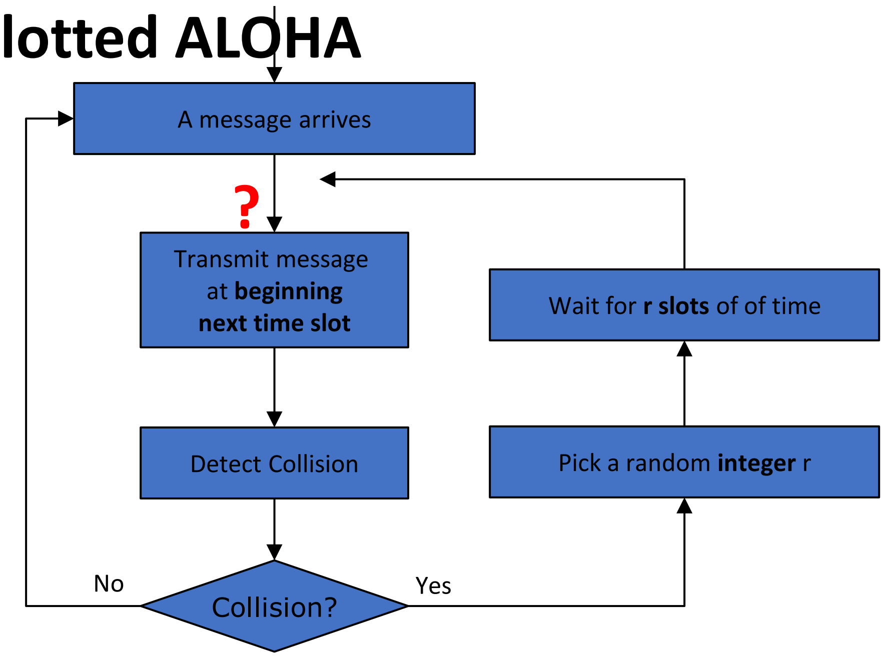


Making Further Improvements?

Pure ALOHA



Slotted ALOHA



- ALOHA transmits even if another node is transmitting → collision

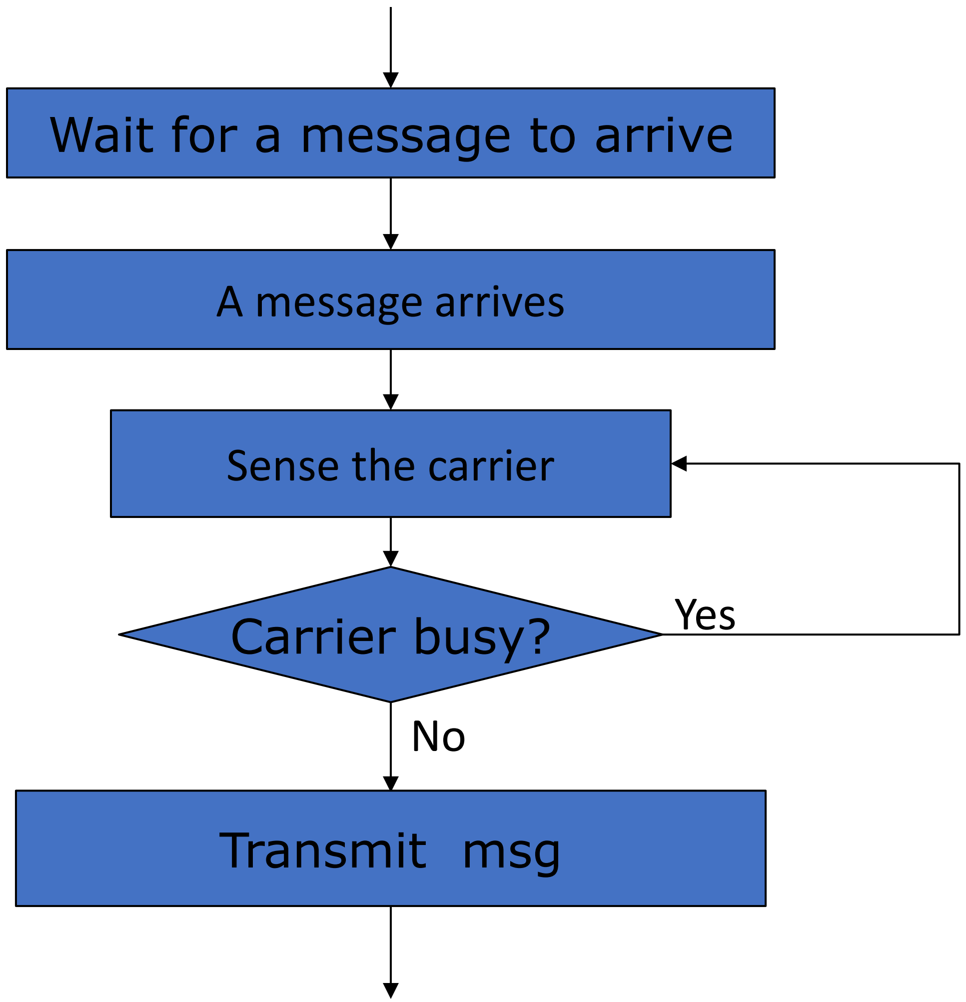
Carrier Sense

- Listen first, transmit when the channel is idle → reduce chance of collision

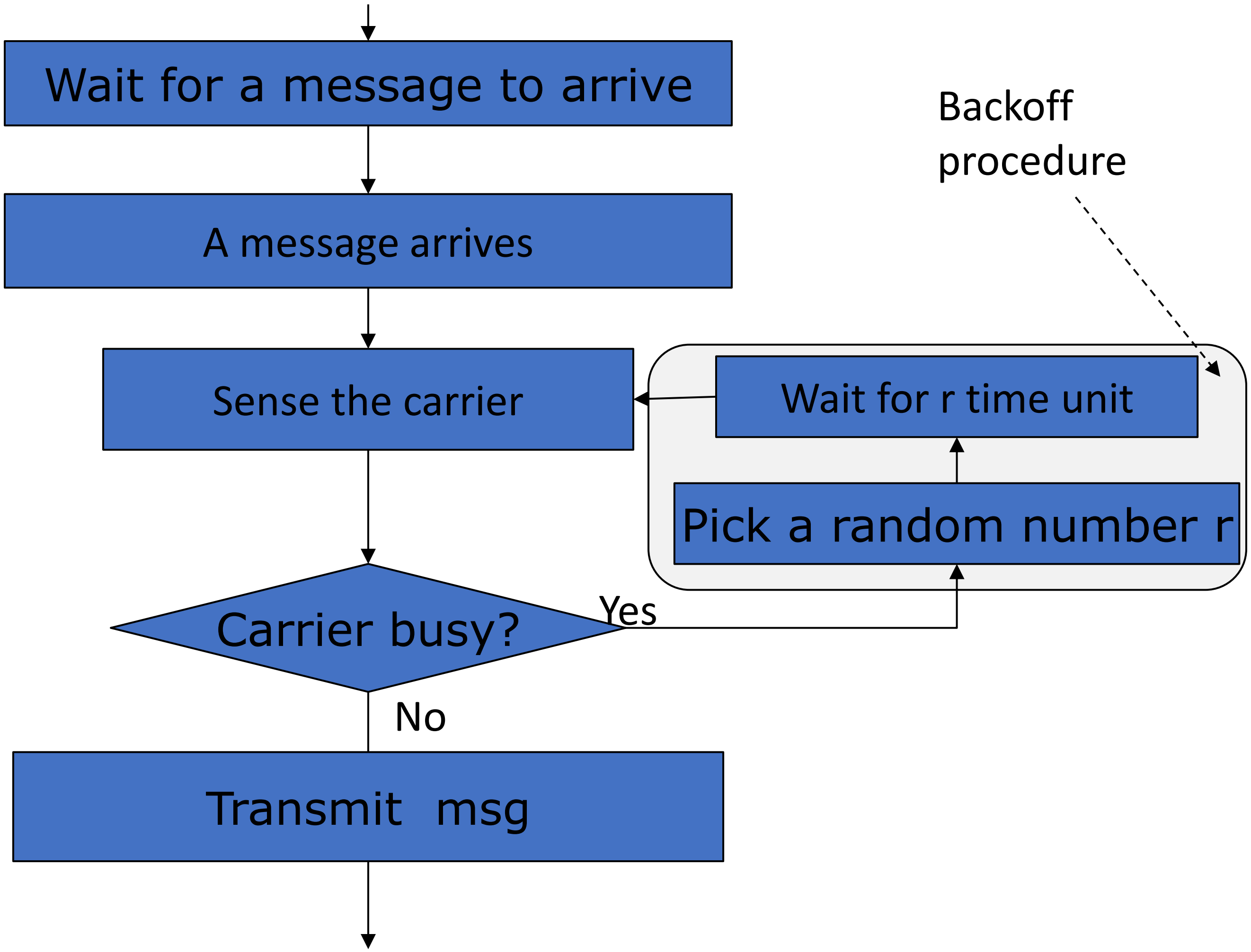
Carrier Sense (without Collision Detection)

- Non-persistent CSMA
  - Transmit after a random amount of waiting time regardless if channel is idle (from carrier sense)
  - Large delay when channel is idle
- 1-persistent CSMA
  - Transmit as soon as the channel becomes idle
  - Collision happens when two or more nodes all want to transmit
- p-persistent CSMA
  - If idle, transmit the frame with a probability p

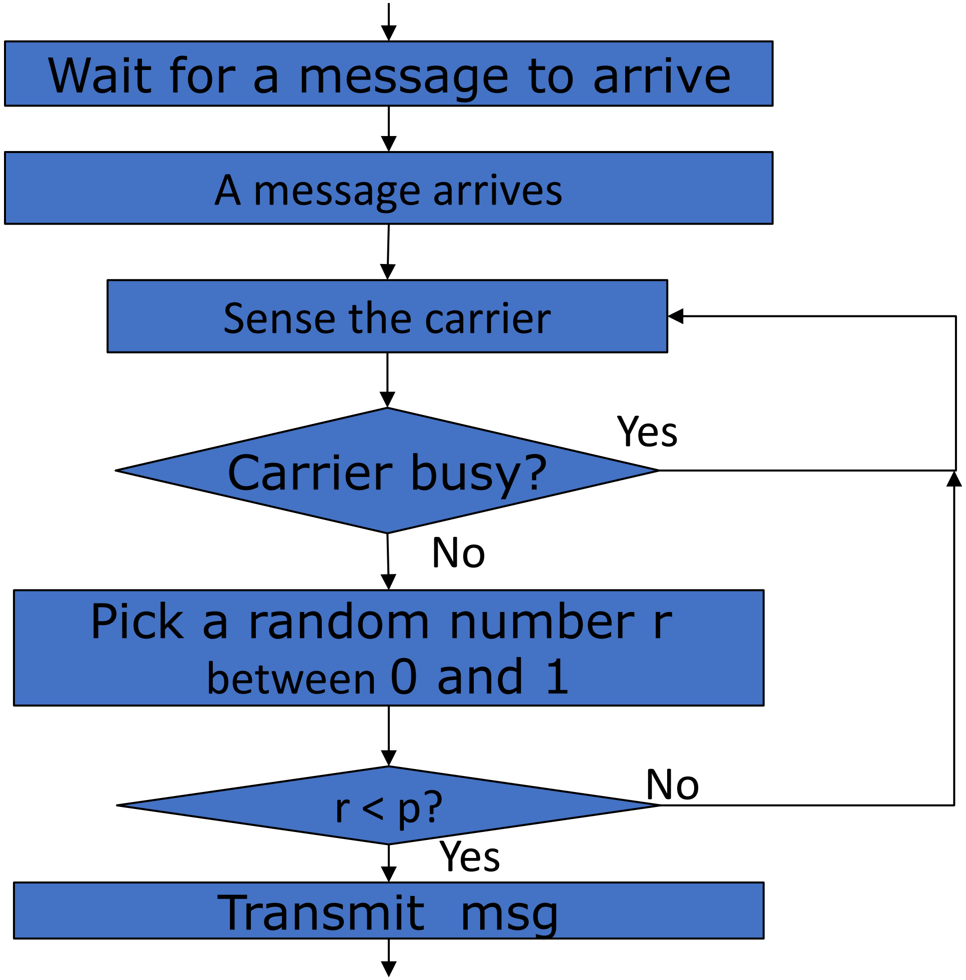
1-persistent CSMA



Non-persistent CSMA



p-persistent CSMA



Comparison of Throughput

- Pure ALOHA

$$S = Ge^{-2G}$$

- Slotted ALOHA

$$S = Ge^{-G}$$

- Nonpersistent CSMA

$$S = \frac{Ge^{-aG}}{G(1+2a)+e^{-aG}}$$

- 1-persistent CSMA
  - Unslotted

$$S = \frac{G\left[1+G+aG\left(1+G+aG/2\right)\right]e^{-G(1+2a)}}{G\left(1+2a\right)-\left(1-e^{-aG}\right)+\left(1+aG\right)e^{-G(1+a)}}$$

  - Slotted

$$S = \frac{Ge^{-G(1+a)}\left[1+a-e^{-aG}\right]}{\left(1+a\right)\left(1-e^{-aG}\right)+ae^{-G(1+a)}}$$

- p-persistent CSMA
  - skipped

Comparison of Throughput

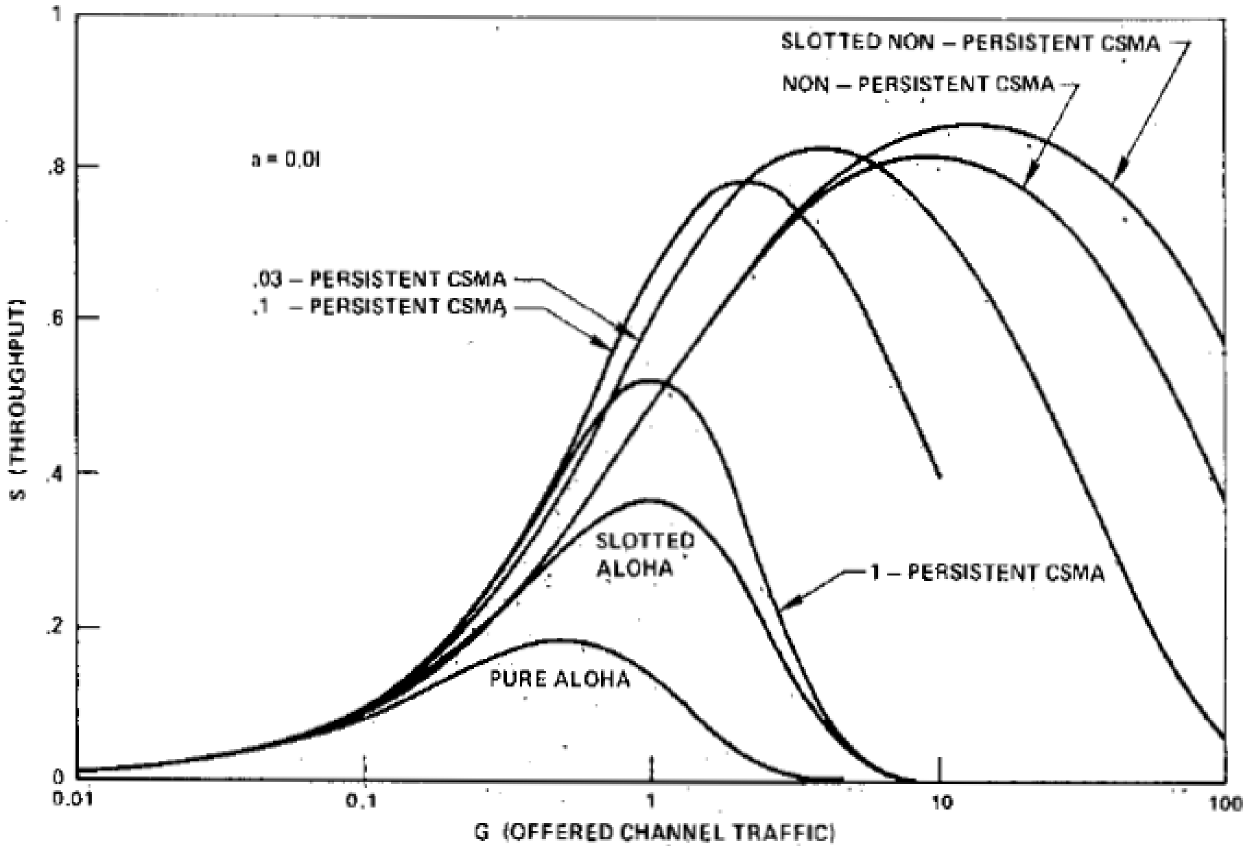


Fig. 9. Throughput for the various access modes ($a = 0.01$).

From LEONARD KLEINROCK, 1975

Carrier Sense

- Listen first, transmit when the channel is idle → reduce chance of collision
- Can collisions be **completely** mitigated?

Carrier Sense

- Listen first, transmit when the channel is idle → reduce chance of collision
- Can collisions be **completely** mitigated?
- Q: Under what condition can Carrier Sense be more beneficial to throughput?

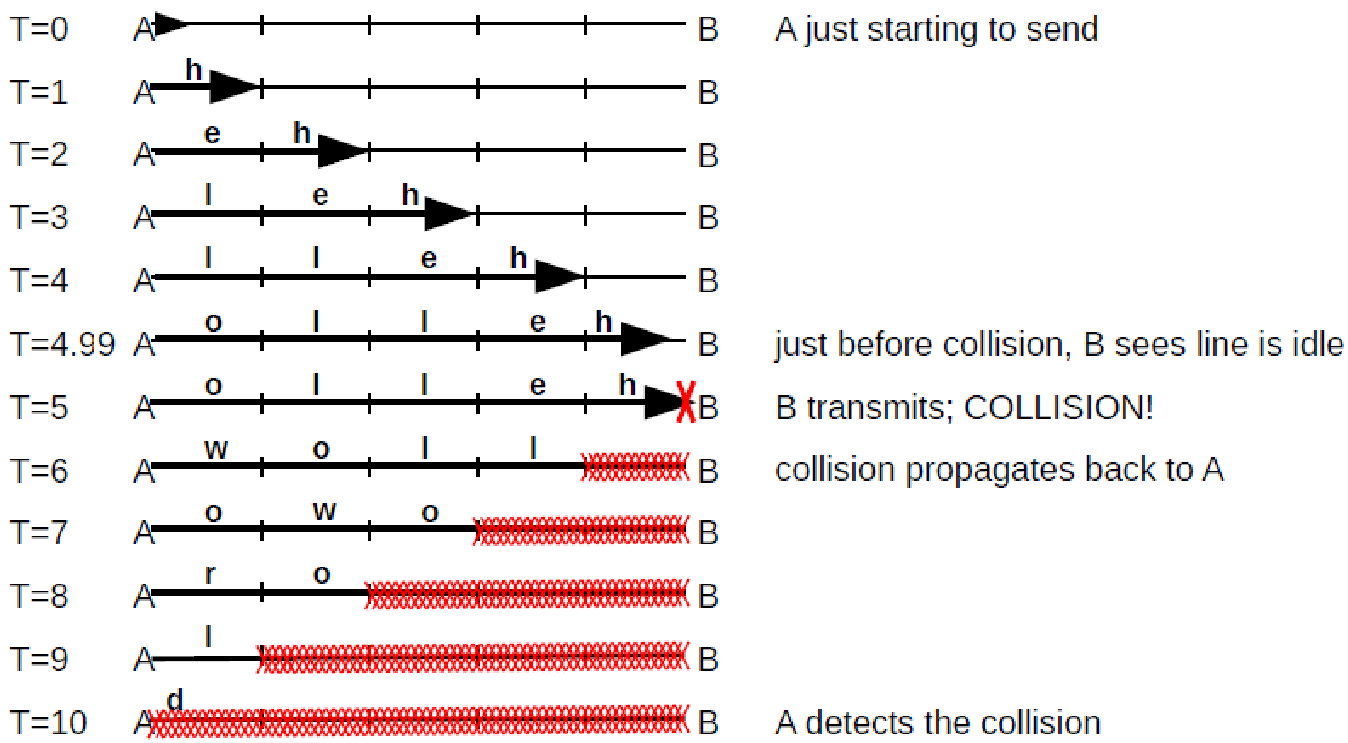
Carrier Sense and Collision

- Even with CSMA there can still be collisions.
- What do Pure ALOHA and Slotted ALOHA do?

Collision Detection

- If nodes can detect collisions, abort transmissions!
  - Requires a minimum frame size ("acquiring the medium")
  - Continues to transmit a jamming signal (called runt) until other nodes detects it
  - Requires a full duplex channel

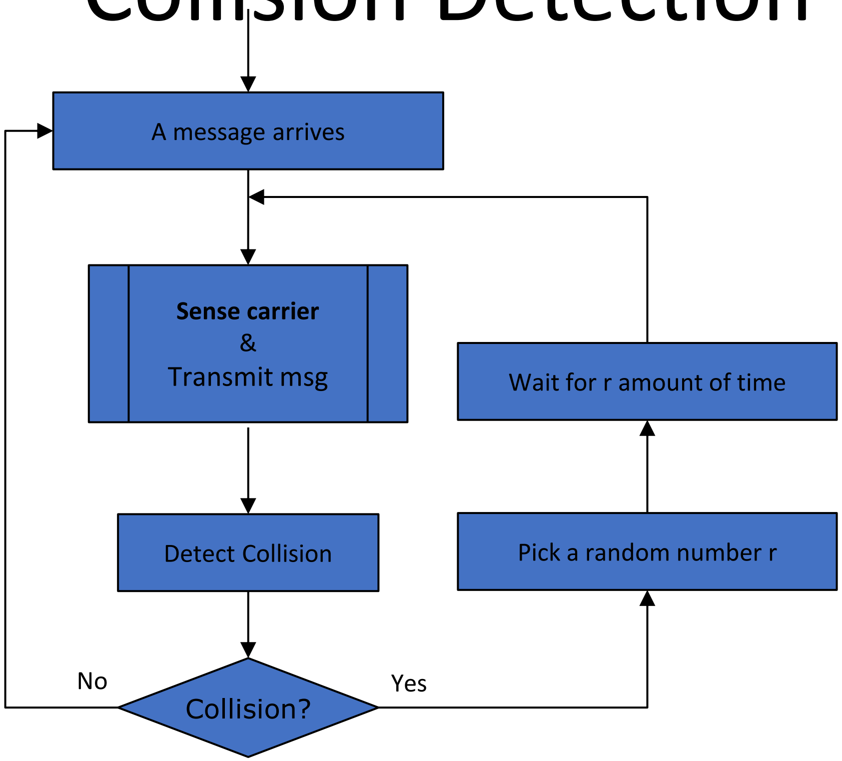
Collision Detection

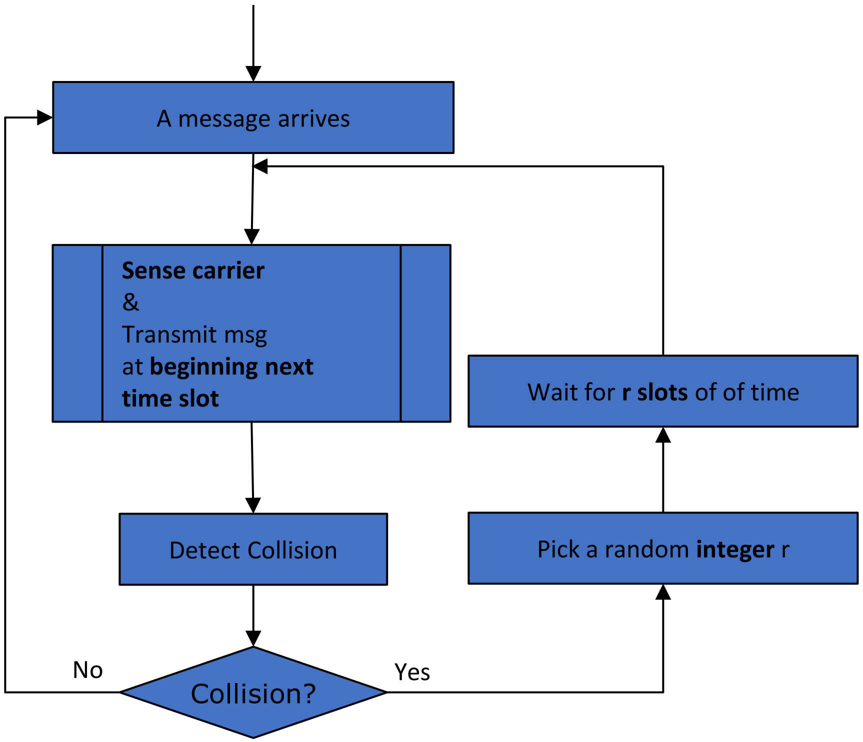


- https://intronetworks.cs.luc.edu/current2/html/_images/Ethernet_collision_rtt_scaled.svg

Complete the Picture

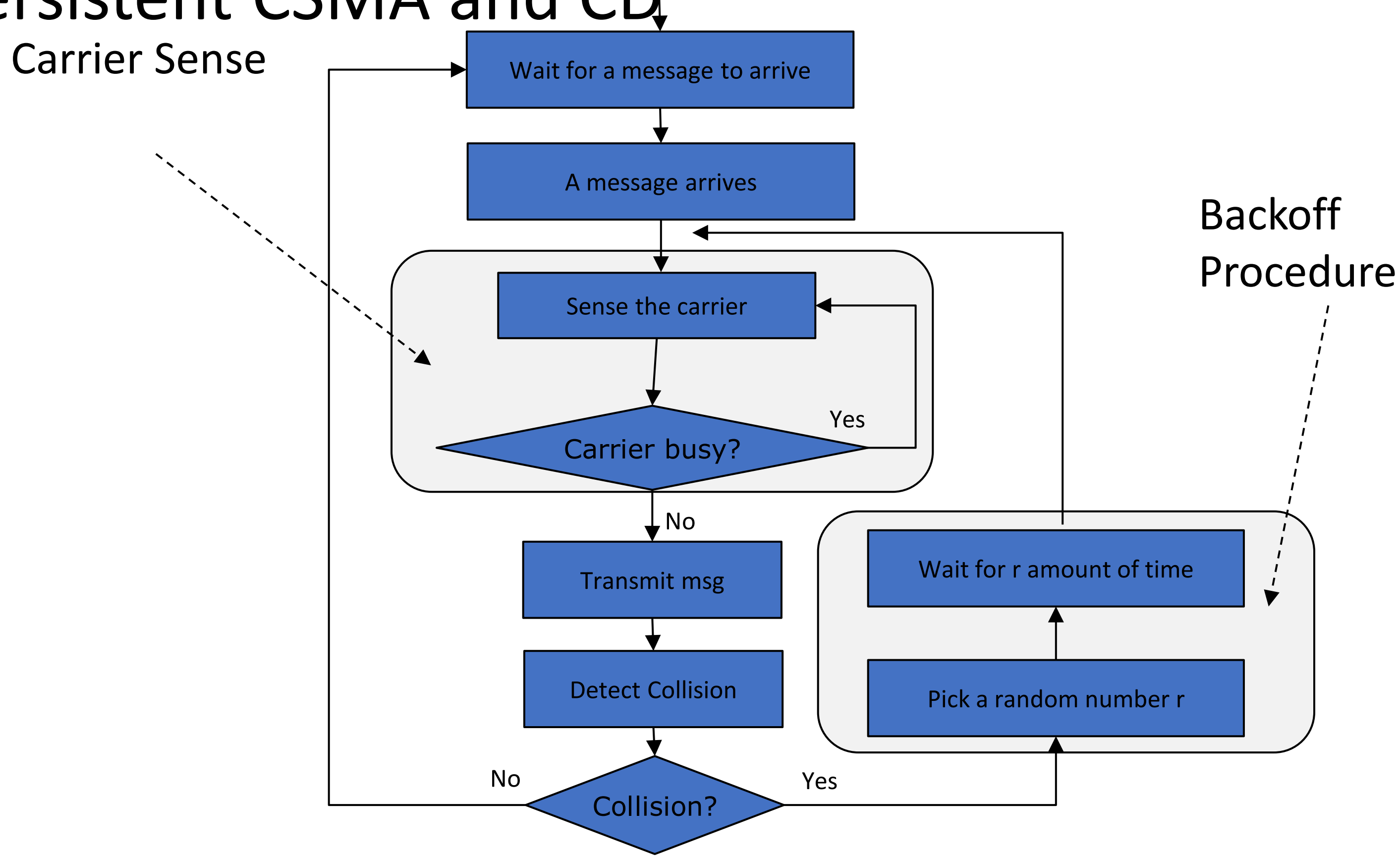
- Carrier Sense Multiple Access and Collision Detection





CSMA/CD

- 1-Persistent CSMA and CD



Summery: Dynamic Channel Allocation

- Concept of media access control
- Dynamic channel allocation
- Contention resolution protocols
  - ALOHA
  - CSMA
  - CD

Ethernet

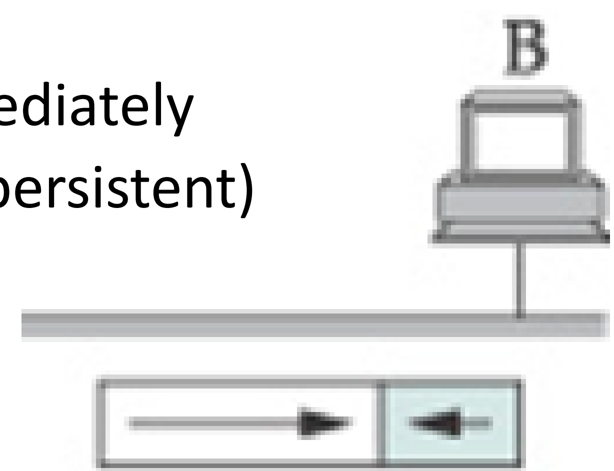
- Multiple Access Networks
- Inspired by the ALOHA network at the University of Hawaii
- Developed by Robert Metcalfe and Bob Boggs at Xerox PARC
- Standardized by IEEE as IEEE 802.3
  - http://standards.ieee.org

Ethernet: Media Access Control

- Carrier Sense Multiple Access and Collision Detection (CSMA/CD) with Exponential Backoff
  - Carrier Sense (CS)
  - Multiple Access (MA)
  - Collision Detection (CD)

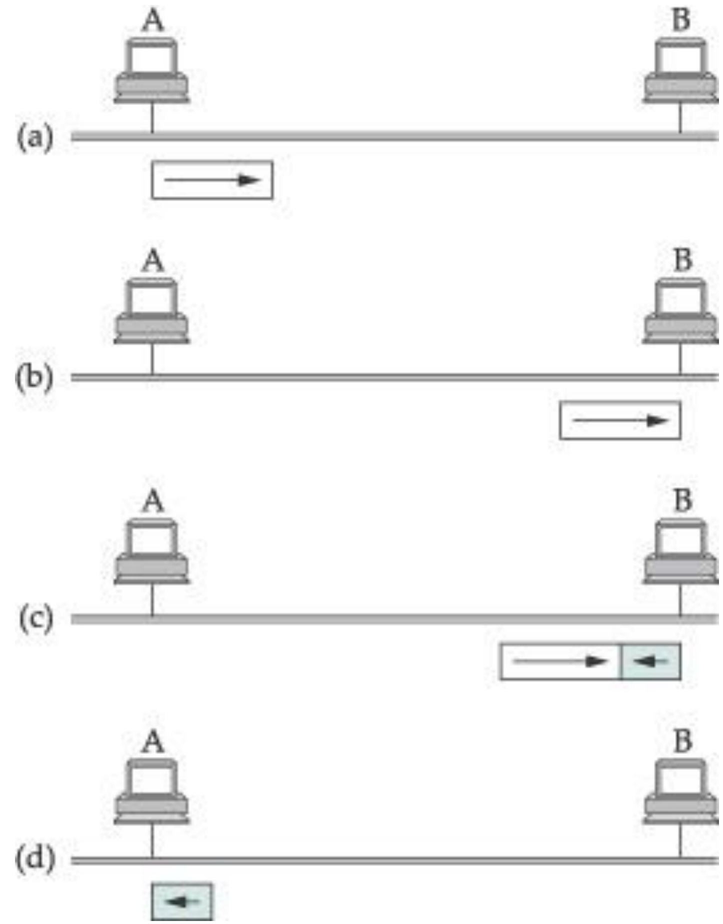
Ethernet: Carrier Sensing

- If line is idle
  - Send immediately
  - Upper bound message size = 1500 bytes
- If line is busy
  - Wait until idle and transmit immediately
  - 1-persistent (a special case of p-persistent)



Collision Detection on Ethernet

- No centralized control, distributed algorithm
- Two nodes may transmit almost at the same time → collision
- Worst case scenario
  - (a) A sends a frame at time t
  - (b) A's frame arrives at B at t + d
  - (c) B begins transmitting at time t + d and collides with A's frame. Upon detecting the collision, B sends a *runt* (32-bit frame) to A
  - (d) B's runt frame arrive at A at t + 2d
  - **Why does B need to send a runt to A?**
  - **How long does it take for A to detect the collision?**



Collision Detection on Ethernet

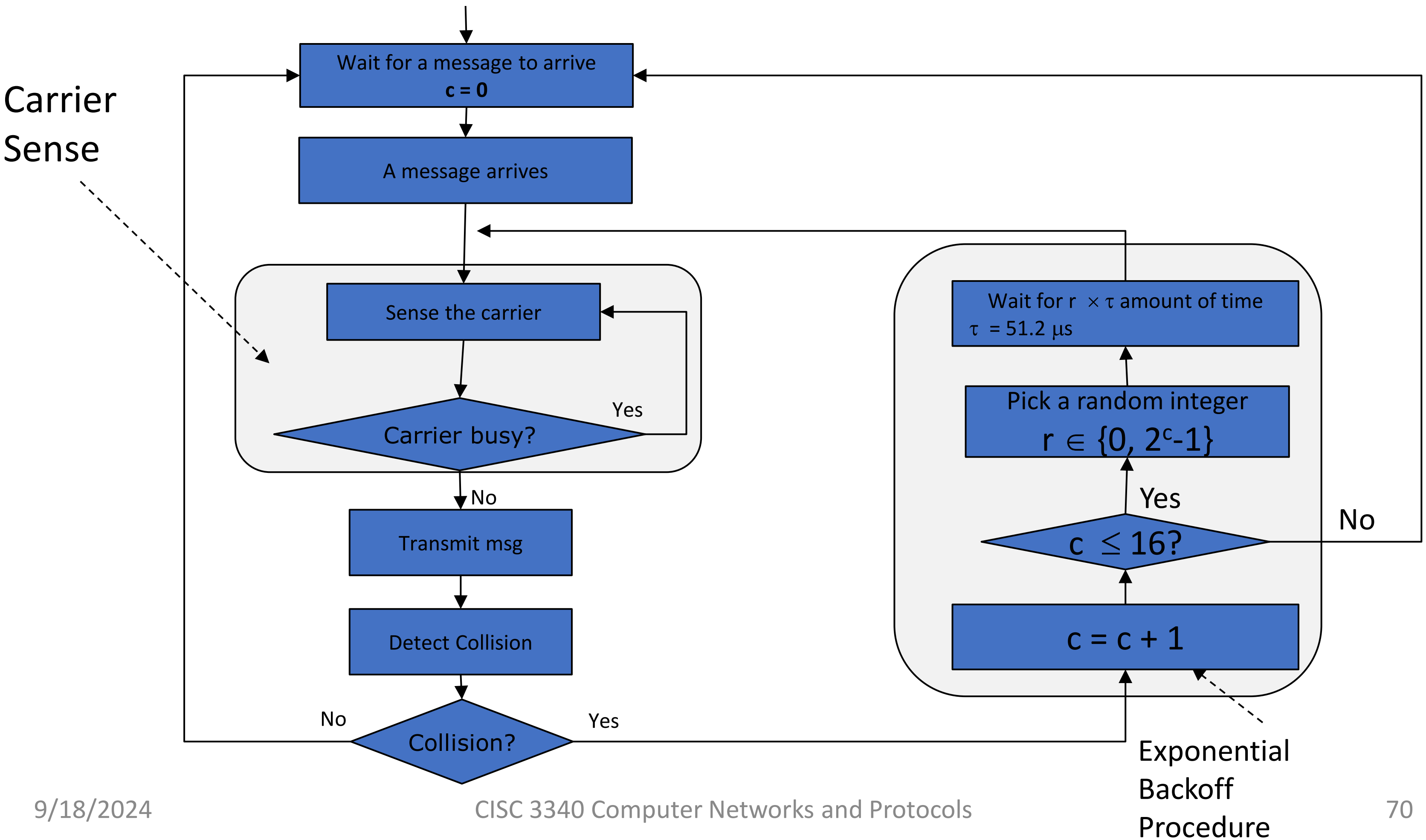
- Want the nodes that collide to know that a collision happened
  - Time during which a node (the transmitting node) may hear of a collision is $1 \times$ RTT
    - Recall: under what condition can a network be benefited most from “carrier sense”?
  - Impose a minimum frame size that lasts for $1 \times$ RTT
    - So the node can not finish transmitting before a collision takes place → carrier sense benefits the network the most
    - Consider an Ethernet: minimum frame is 64 bytes, longest link 2500 meters (4 repeaters, 500 meter segment), 10-Mbps bandwidth
      - $1 \times$ RTT = 51.2 μs and $1 \times$ RTT $\times$ Bandwidth = 512 bits = 64 bytes

Ethernet: Collision Detection with Binary Exponential Backoff

- If collision
  - Jam for 32 bits (by sending a runt), and stop transmitting frame
  - Minimum frame is 64 bytes (14 bytes header + 46 bytes of data + 4 bytes CRC) for 10 Mbps Ethernet
  - Exponential backoff
    - 1^{st} time: 0 or 51.2 μs
      - Randomly select one of these two: imagine throwing an evenly made coin, if it lands tail, choose 0; otherwise, 51.2 μs
    - 2^{nd} time: 0, 51.2, or 51.2 x 2 μs
      - Randomly select one of these two: imagine throwing a 3-sided die whose three faces are labeled as 0, 1, and 2. If it lands on side 0, choose 0; on side 1, 51.2 μs; on side 2, 51.2 x 2 μs
    - 3^{rd} time: 0, 51.2, 51.2 x 2, or 51.2 x 3 μs
      - Similar process with 4-sided die
    - n-th time: k x 51.2 μs, randomly select k from $0..2^n-1$
      - Similar as before, you die (very strange die) has 2^n sides labeled from 0 to 2^n-1
    - Give up after 16 times

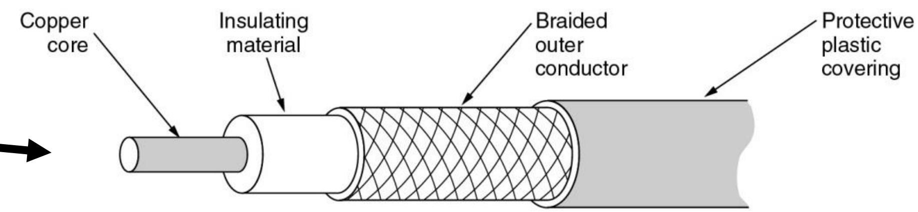
Ethernet: CSMA/CD with Exponential Backoff

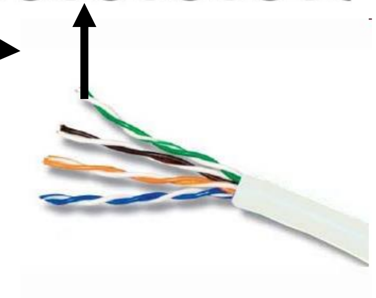
- 1-Persistent CSMA and CD

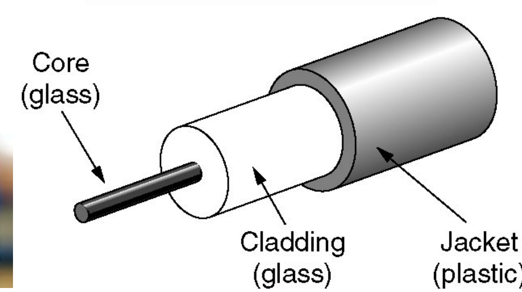


Ethernet (IEEE 802.3): Physical Media

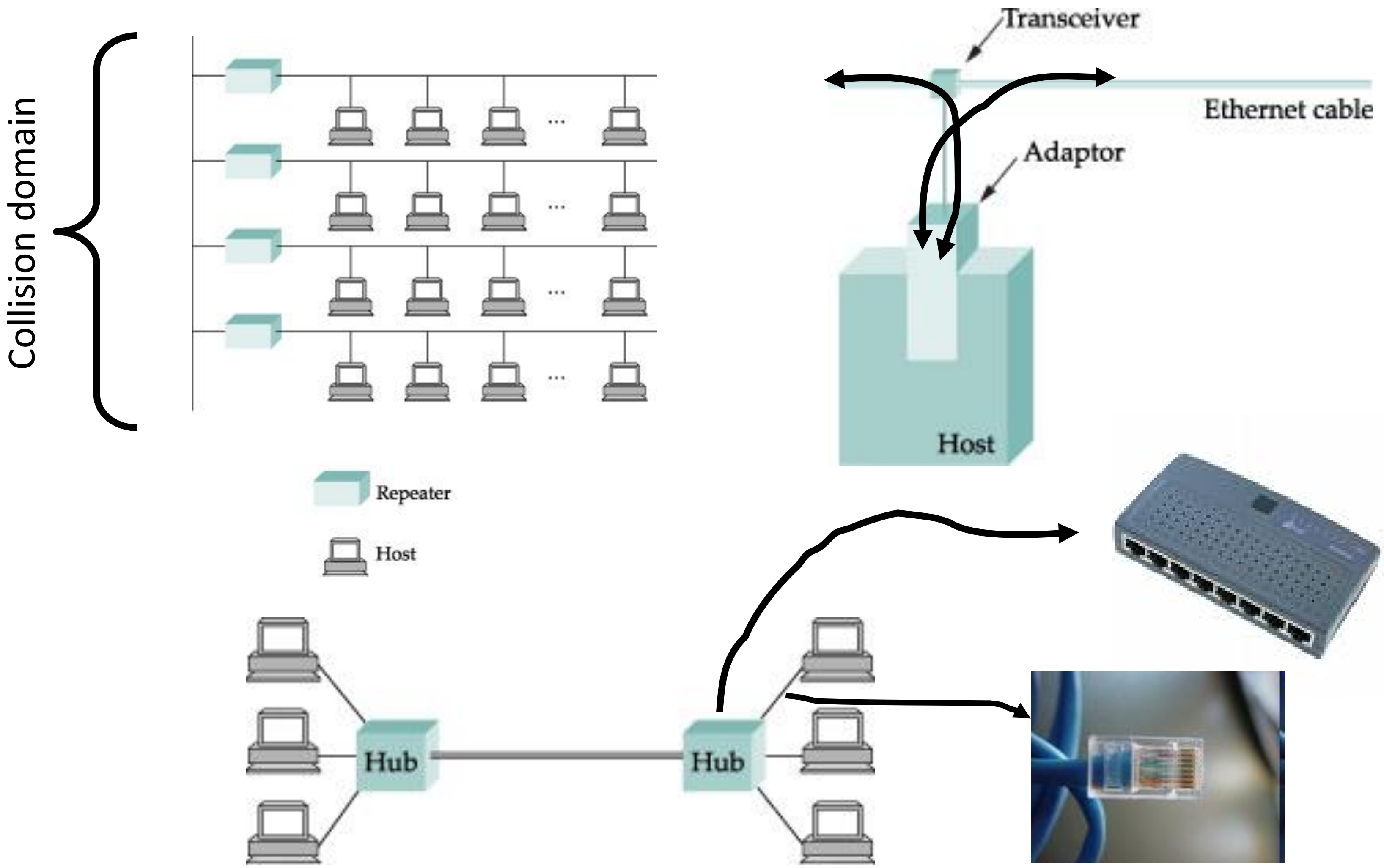
- Transmission Media: guided
  - Coaxial cable
    - 10Base5 (thick, 500m)
    - 10Base2 (thin, 200m)
  - Twisted pair cable
    - 10BaseT (100m)
    - 100BaseT (100m)
    - 1000BaseT (100m)
    - 10GBaseT (100m)
    - …
  - Optic fiber
    - 100BASE-FX
    - 10GBASE-R
    - ……





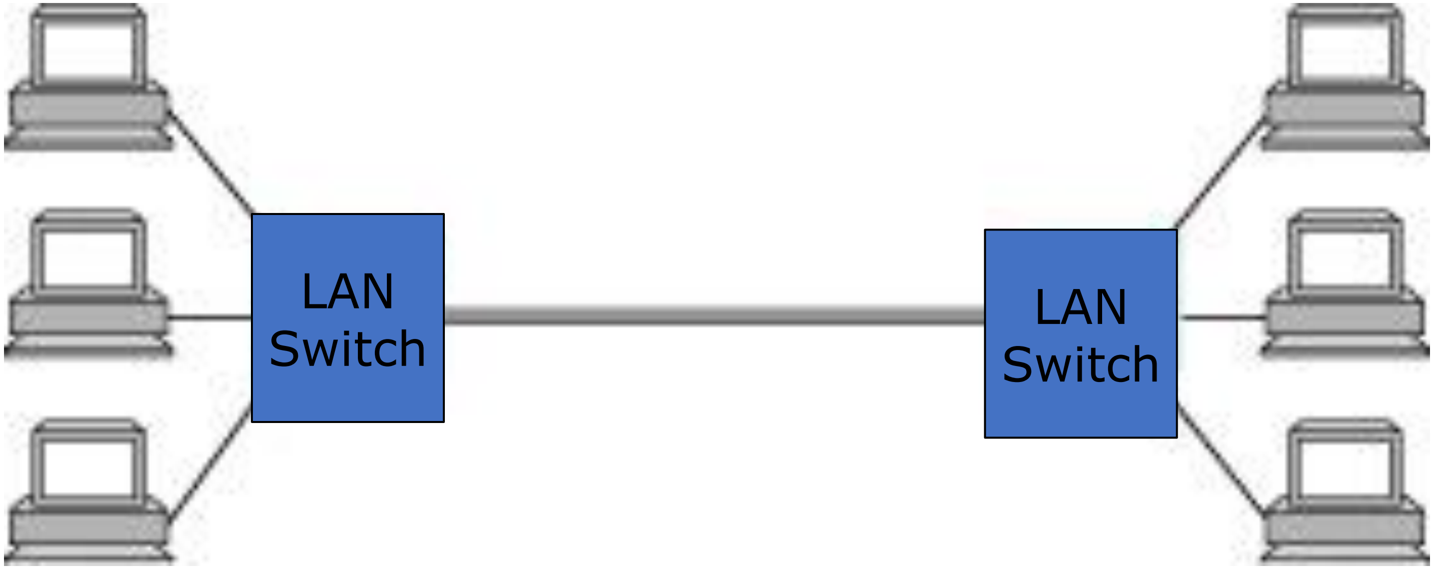


Ethernet (IEEE 802.3): Networking

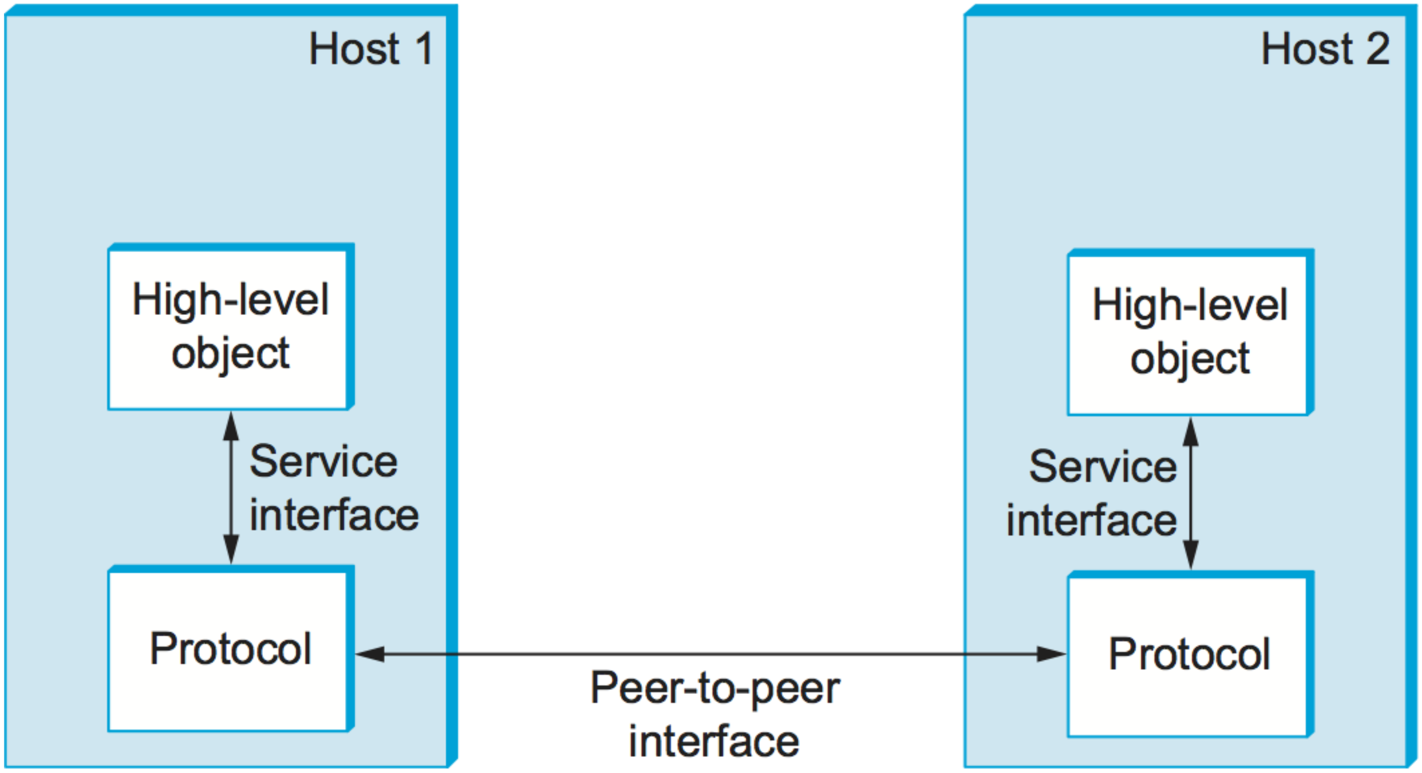


Ethernet (IEEE 802.3): Networking

- Today's deployment: point-to-point deployment

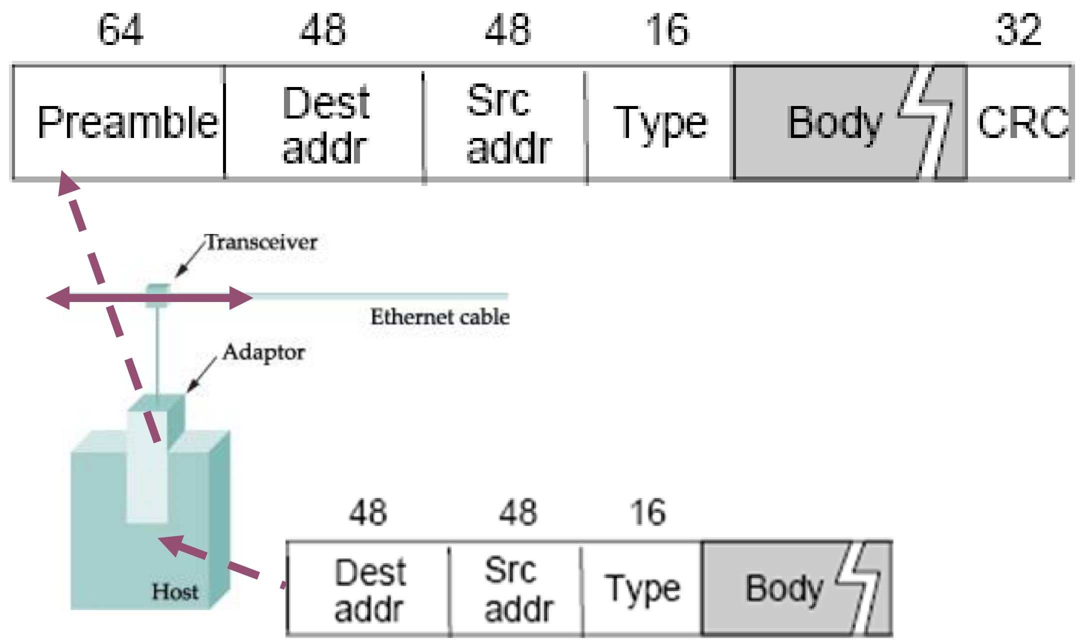


Recall Service and Peer-to-Peer Interfaces …

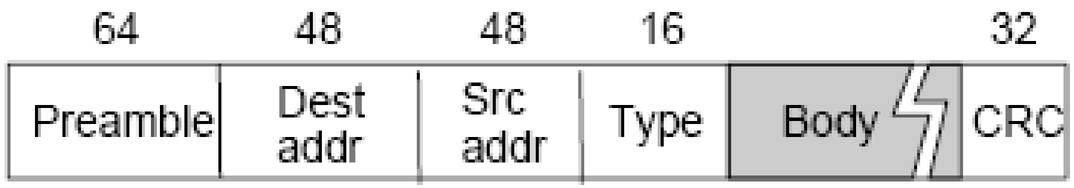


Ethernet: Frame Format

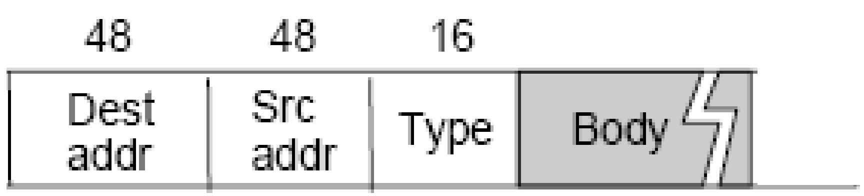
- Bit-oriented framing
  - Preamble (64 bits): 101010… for signal synchronization
  - Destination address (48 bits)
  - Source address (48 bits)
  - Type/length (16 bits)
  - Body (46 – 1500 bytes)
  - CRC (32 bits)



Frame Preamble and CRC



- Be aware that Ethernet network interface cards often do not pass *preamble* and *CRC* to hosts
- In the future, we do not include preamble and CRC when discussing Ethernet frames



Ethernet Address

- Unique in the world
- Assigned to adaptors
- 48-bit
  - 0000 1000 0000 0000 0010 1011 1110 0100 1011 0001 0000 0010
  - 08:00:2b:e4:b1:02 (human-friendly form)

24-bit Organization Unique Identifier (OUI), now called MA-L assignment

Look it up from https://standards.ieee.org/products-programs/regauth/mac/

Human-Friendly Notation

- Two common human-friendly notations
- Hex-digits-and-colons notation
    - Example
    - 08:00:2b:e4:b1:02
- Hex-digits-and-dash notation
    - Example
    - 08-00-2b-e4-b1-02

Ethernet Address Types

- Unicast address
  - For one to one communication
  - Each adapter is assigned a unicast address
- Broadcast address
  - For one to all communication
- Multicast address (group address)
  - For one to a group communication

Unicast address

- Address of an adaptor (e.g., my_addr)
- Each frame transmitted on an Ethernet is received by every adapter connected to that Ethernet
- Each adapter recognizes those frames addressed to its address and passes only those frames onto the host
- In pseudo code,

```
If dest_addr == my_addr
    pass the frame to the host
```

Broadcast Address

- Broadcast address
- One single broadcast address, i.e., all 1's in the address (ff:ff:ff:ff:ff:ff)
- All adapters pass frames addressed to the broadcast address up to their hosts.
- In pseudo code,

```
If dest_addr == 0xff ff ff ff ff ff
    Pass the frame to the host
```

Multicast address

- Multicast address (group address)
- A given host can program its adaptor to accept some set of multicast addresses (the group).
- An adapter in the group passes frames addressed to the group to the host

□ Complex and requires group management

- Multicast addresses are addresses has the first bit set to 1, but is not the broadcast address (Ethernet transmits bytes from low-order bit to high-order bit)
- In pseudo code,

```
If (dest addr & 0x01 00 00 00 00 00) && (it has been instructed to listen to that multicast address)
    deliver the frame to the host
```

Promiscuous Mode

- Not a normal mode
- An adaptor can also be programmed to run in *promiscuous* mode
- All frames will be delivered to the host

Experiment: Looking up Ethernet Adapters (1)

- On MS Windows (various version of NT systems, including 2000, XP, Vista, 7, 8, and 10 etc)
- Use the following tools
  - getmac
  - ipconfig

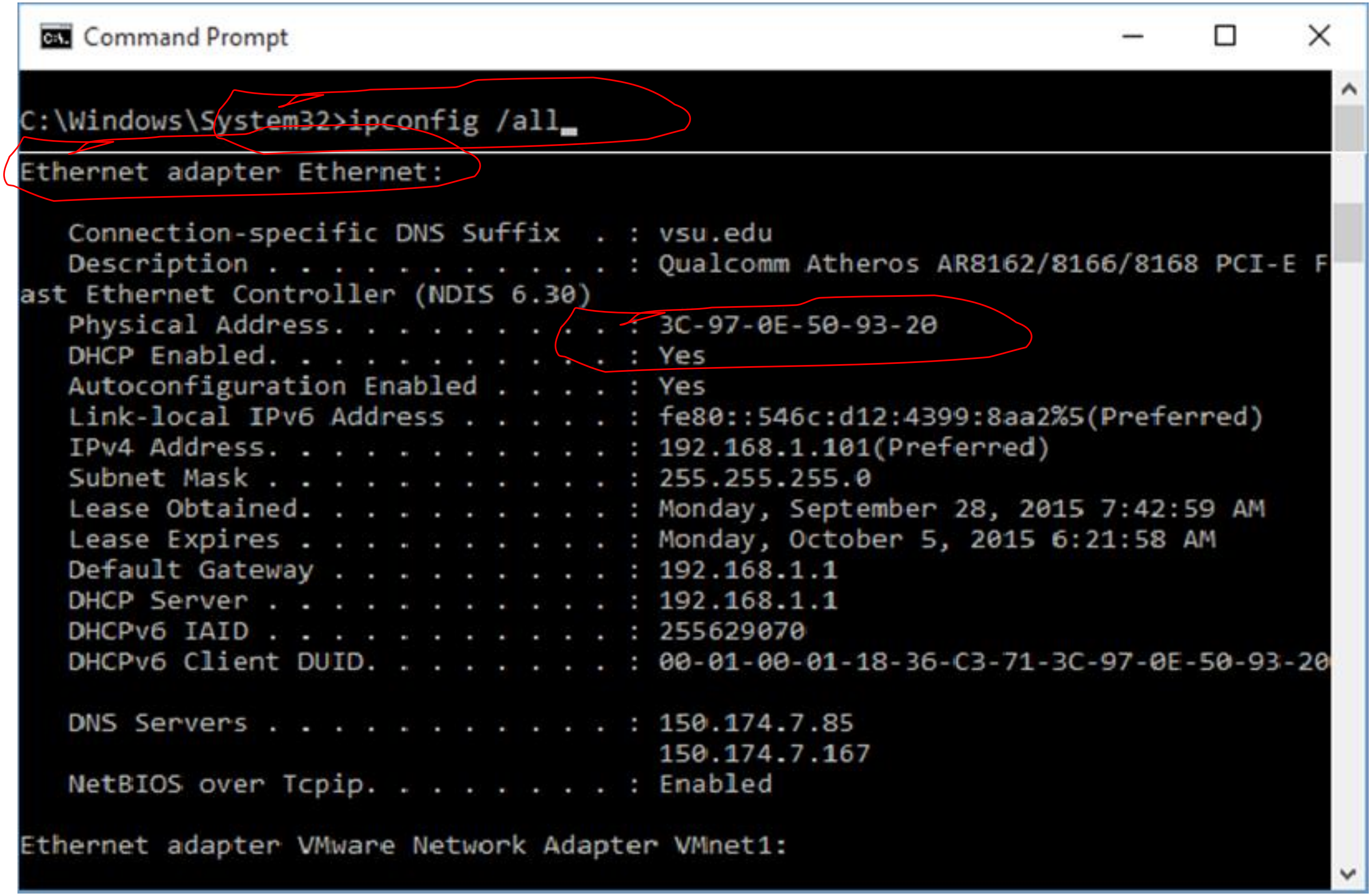
Command Prompt

```
C:\Windows\System32>getmac

Physical Address    Transport Name
=================== ==========================================================
00-FF-E6-92-DB-9D   Media disconnected
00-50-56-C0-00-01   \Device\Tcpip_{95A7F0DA-E226-4511-9EF2-469B24D732C5}
00-50-56-C0-00-08   \Device\Tcpip_{45DD0C71-670F-4D7C-9E16-B0508E1B9EA0}
3C-97-0E-50-93-20   \Device\Tcpip_{3A90E898-57F2-44D7-A67A-3A08F194E5D0}
68-94-23-19-24-7C   Media disconnected

C:\Windows\System32>_
```

Check out https://standards.ieee.org/products-programs/regauth/



Check out https://standards.ieee.org/products-programs/regauth/

Experiment: Looking up Ethernet Adapters (2)

- Similar query can be done on Unix/Linux systems
- Use following tools
  - ip (On Linux)
  - Ifconfig (On most Unix/Linux)

hchen@ubuntu: ~

```
hchen@ubuntu:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:89:7a:4d
          inet addr:192.168.101.127  Bcast:192.168.101.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe89:7a4d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1544943 errors:0 dropped:0 overruns:0 frame:0
          TX packets:727704 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:278511304 (278.5 MB)  TX bytes:126084223 (126.0 MB)
          Interrupt:18 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:13527 errors:0 dropped:0 overruns:0 frame:0
          TX packets:13527 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1026774 (1.0 MB)  TX bytes:1026774 (1.0 MB)

lxcbr0    Link encap:Ethernet  HWaddr d6:1f:0d:4c:d5:5e
          inet addr:10.0.3.1  Bcast:10.0.3.255  Mask:255.255.255.0
          inet6 addr: fe80::d41f:dff:fe4c:d55e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

Check out: https://standards.ieee.org/products-programs/regauth/

```
hchen@ubuntu: ~
hchen@ubuntu:~$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
 group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mo
de DEFAULT group default qlen 1000
    link/ether 00:0c:29:89:7a:4d brd ff:ff:ff:ff:ff:ff
3: lxcbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOW
N mode DEFAULT group default
    link/ether b6:aa:39:e5:30:7e brd ff:ff:ff:ff:ff:ff
hchen@ubuntu:~$
```

Check out https://standards.ieee.org/products-programs/regauth/

Exercise

- Q1: How many Ethernet adapters (NICs) does the system you are using have? What are their Ethernet addresses (i.e., physical addresses as reported by the system)?
- Q2: What is the vendors of the adapters you listed? Use the following to look up the vendors
  - Look up vendor prefix from https://standards.ieee.org/products-programs/regauth/

Ethernet: Experience

- Great success in pratice
  - In practice, observations
    - fewer than 200 hosts
    - Far shorter than 2,500 m (RTT ~ 5 μs)
    - Host implements end-to-end flow control (such as TCP/IP), hosts do not pumping frames to NIC when busy
    - *Extended LANs using Ethernet switches (2 nodes on an Ethernet)*
  - Easy to administer and maintain
    - no routing
    - no configuration
  - Simple: hardware such as adaptors are cheap

Summary

- Media access control
  - Contention-based solutions
    - ALOHA, CSMA, CSMA/CD
- Ethernet

Direct Link Networks: Summary

- Encoding
  - Encoding bits onto transmission medium
- Framing
  - Delineating sequence of bits into messages
- Error detection
  - Detecting errors and acting on them
- Reliable delivery
  - Making links appear reliable despite errors
- Media access control
  - Mediating access to shared link
- Q: how many hosts an Ethernet can have? What is the approximate perimeter of an Ethernet? What if we want to have a network that covers entire campus, a city, a nation, a continent, a planet, or the galaxy? → network of networks: Switched Networks