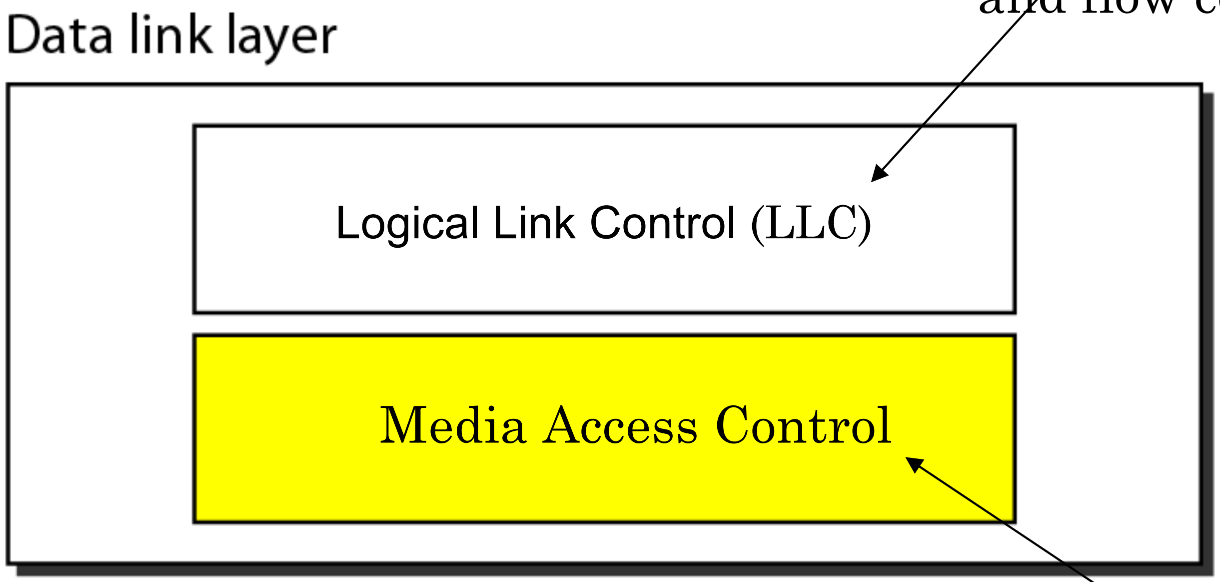
Data Link Layer

Mac sub layer

DATA LINK LAYER- SUB LAYERS

Data link layer divided into two functionality-oriented sub layers



Responsible for error and flow control

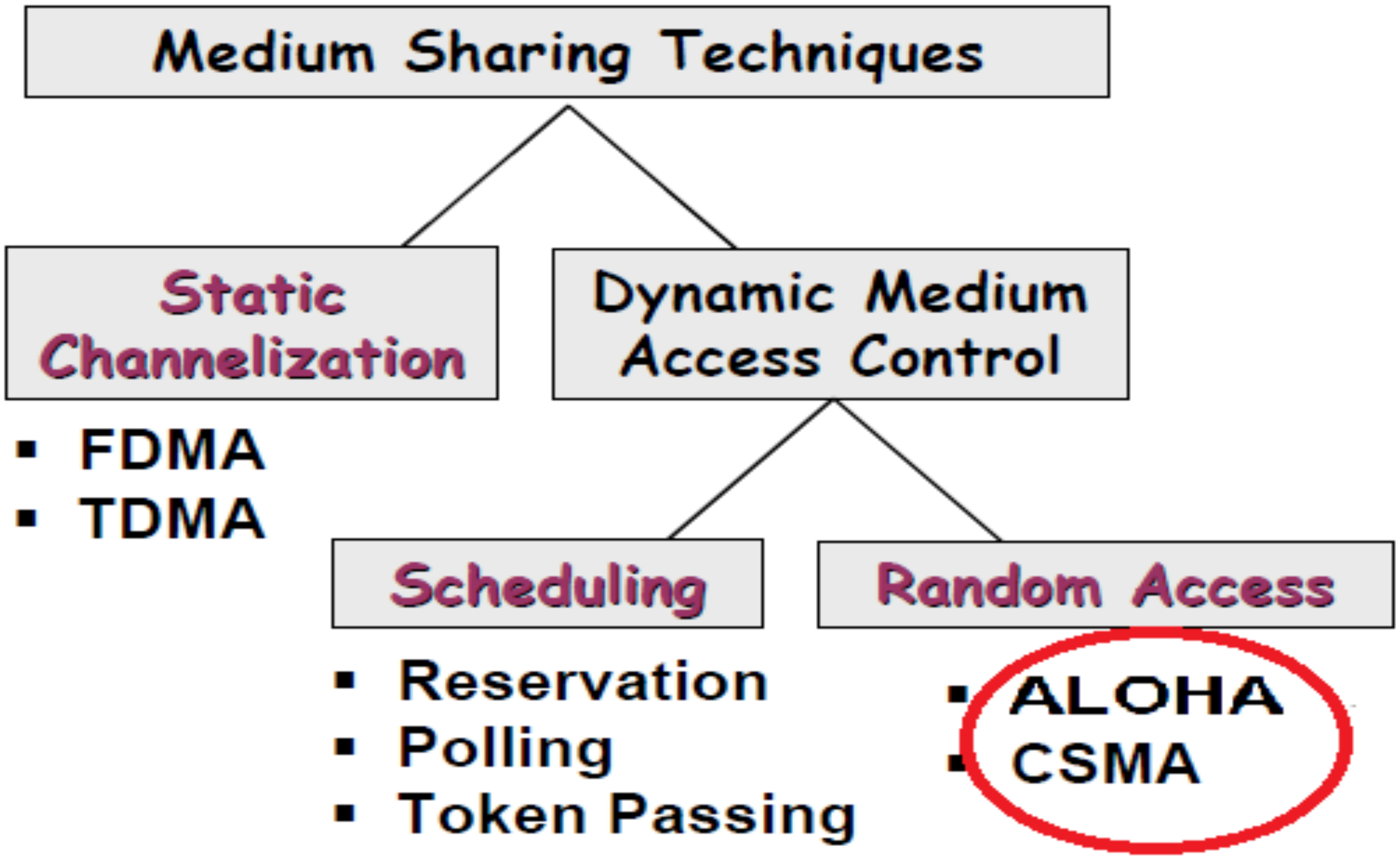
Responsible framing and MAC address and Multiple Access Control

Media Access Control Sub layer

MAC LAYER

- Data link Layer Deals with transmitting bits from one end to other end of a point-to-point Link
- But how we do this in broadcast networks (More than two stations share a common communication link)
- Key issue is who is going to use the channel when there is a competition for it
- The protocol which determine who is going to transmit next, on multi access channel belong to a sub layer of the DLL
- It is called MAC layer
- It is the bottom part of DLL

MEDIUM SHEARING TECHNIQUE



Random Access Technique: Pure ALOHA

Pure ALOHA Protocol Description

- All frames from any station are of fixed length (L bits)

- Stations transmit at equal transmission time (all stations produce frames with equal frame lengths).

- A station that has data can transmit at any time

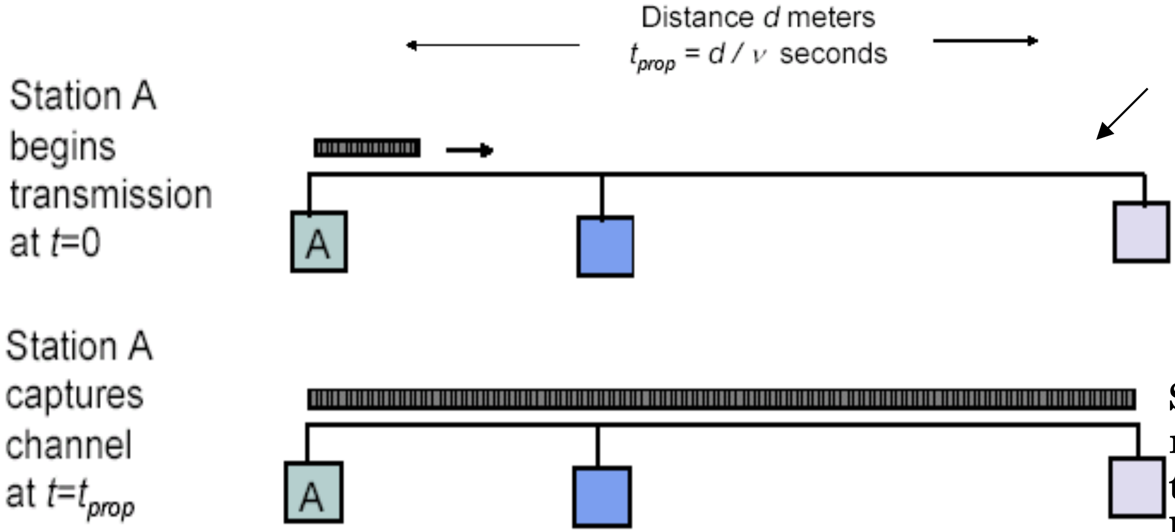
- After transmitting a frame, the sender waits for an acknowledgment for an amount of time (time out) equal to the maximum round-trip propagation delay = 2* t_{prop}(see next slide)

- If no ACK was received, sender assumes that the frame or ACK has been destroyed and resends that frame after it waits for a random amount of time

- If station fails to receive an ACK after repeated transmissions, it gives up

MAXIMUM PROPAGATION DELAY

- **Maximum propagation delay**(t_{prop}): time it takes for a bit of a frame to travel between the **two most widely** separated stations.

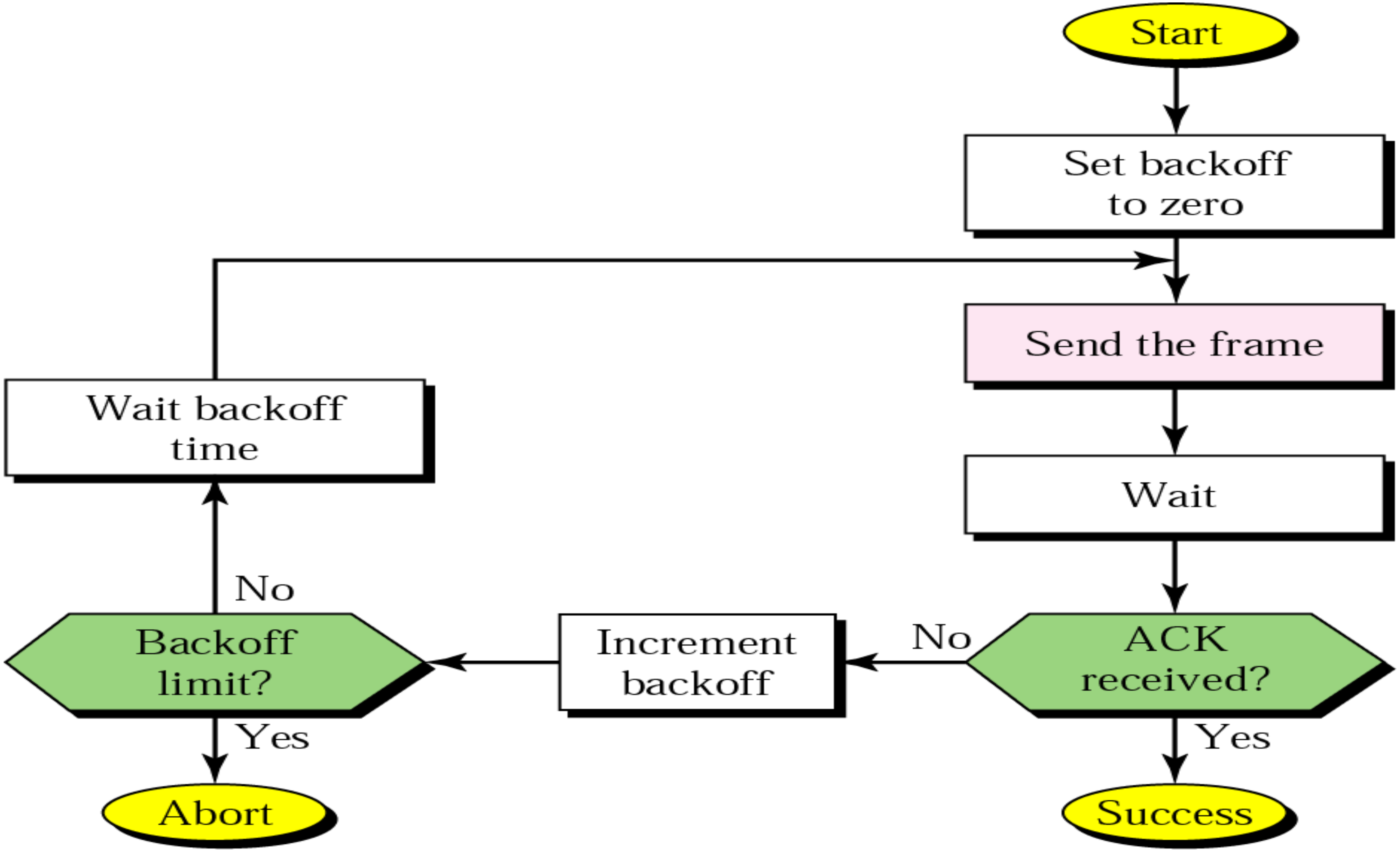


The farthest station

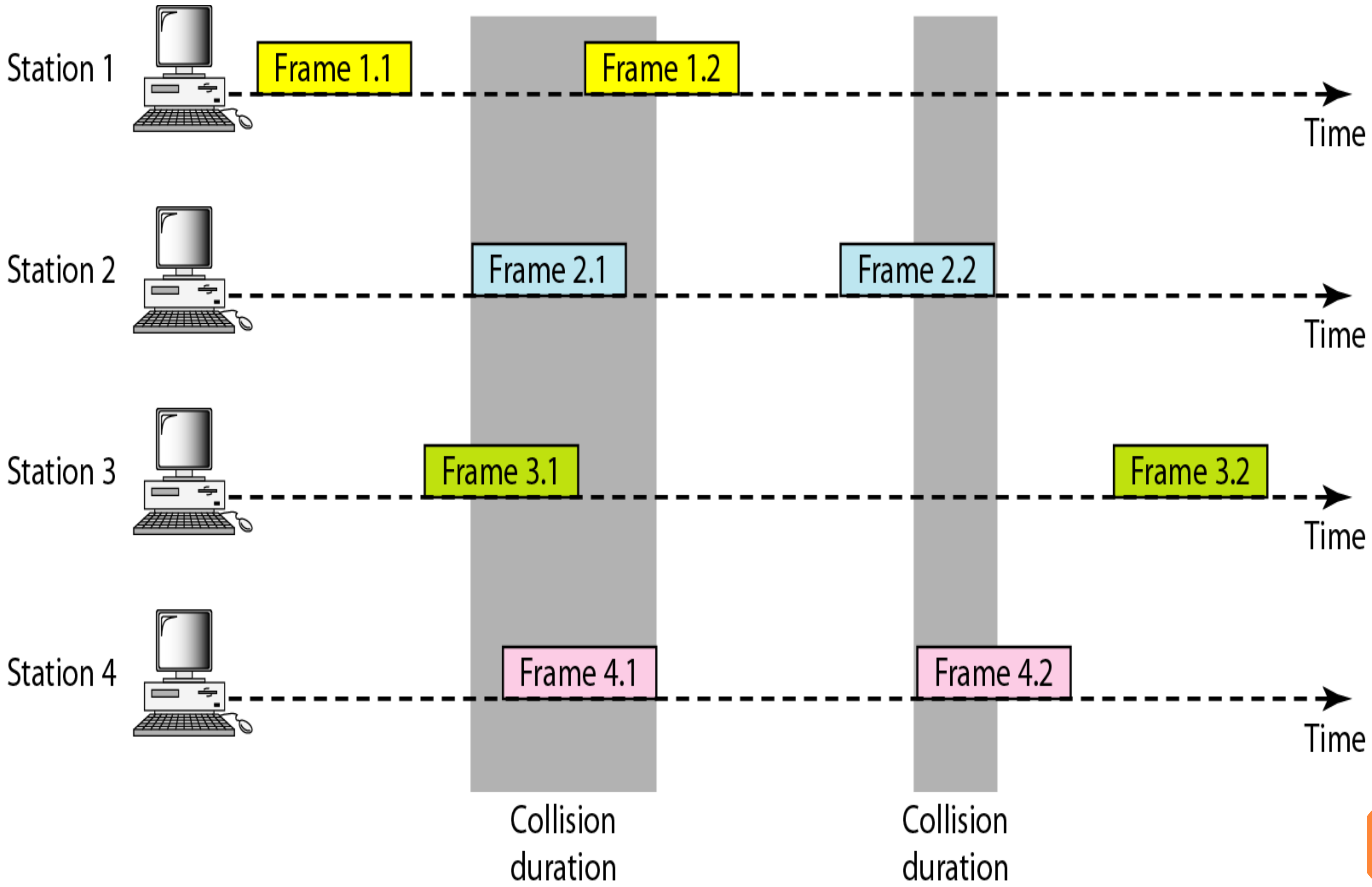
Station B receives the first bit of the frame at time t= t_{prop}

RANDOM ACCESS TECHNIQUE: PURE ALOHA

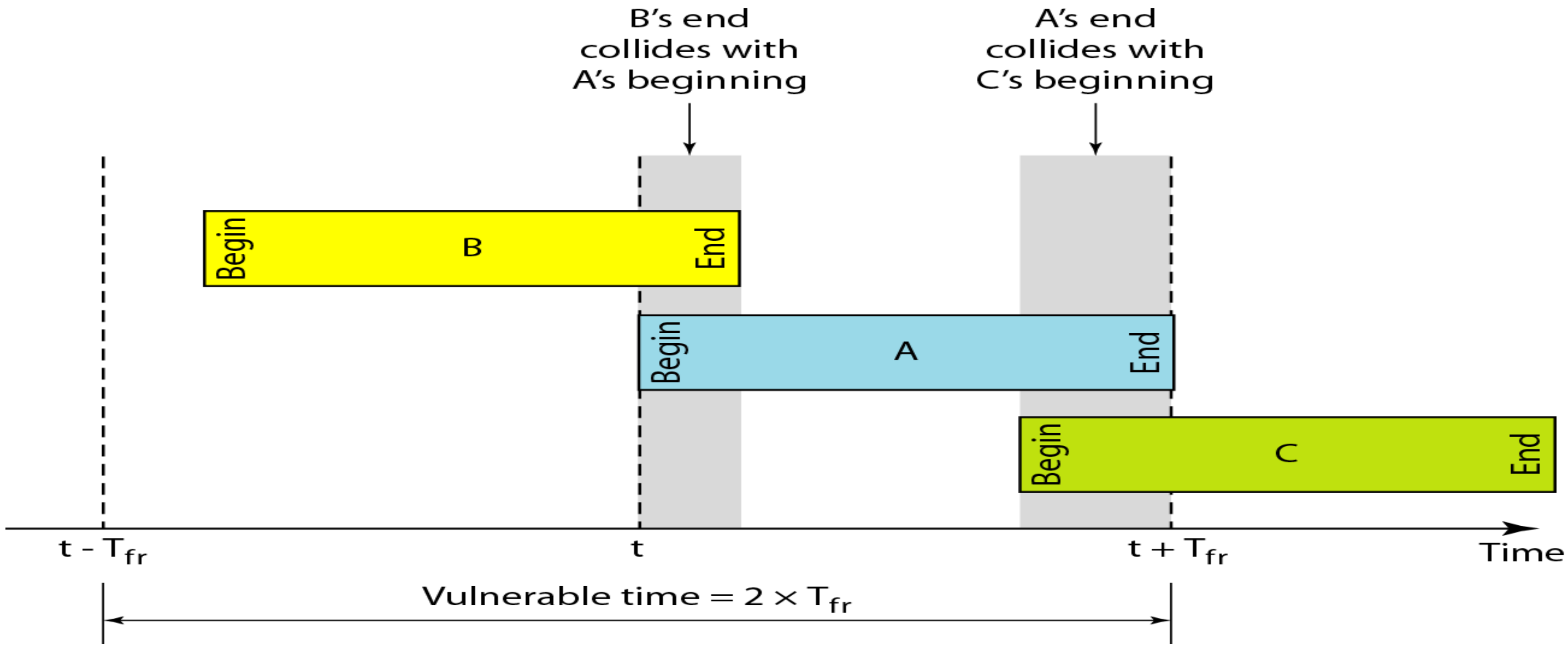
Procedure for ALOHA protocol



Frames in a pure ALOHA network



PURE ALOHA PROTOCOL



- If the frame transmission time is T_{fr} sec, then the vulnerable time is $= 2\ T_{fr}$ sec.

- This means no station should send during the T_{fr} -sec before this station starts transmission and no station should start sending during the Tfr -sec period that the current station is sending.

Pure ALOHA Protocol

- The throughput for pure ALOHA is $S = G \times e^{-2G}$.
- The maximum throughput $S_{max} = 0.184$ when G= (0.5).

Where

- T_{fr}= Average transmission time for a frame
- G= Average number of frames generated by the system (all stations) during one frame transmission time(T_{fr})
- Maximum throughput of pure aloha ($S_{max} = 0.184$) occurs at G=0.5 (which correspond to total arrival rate of "one frame per vulnerable period")
- $S_{max} = 0.184$ =>max pure aloha throughput =18% of channel capacity

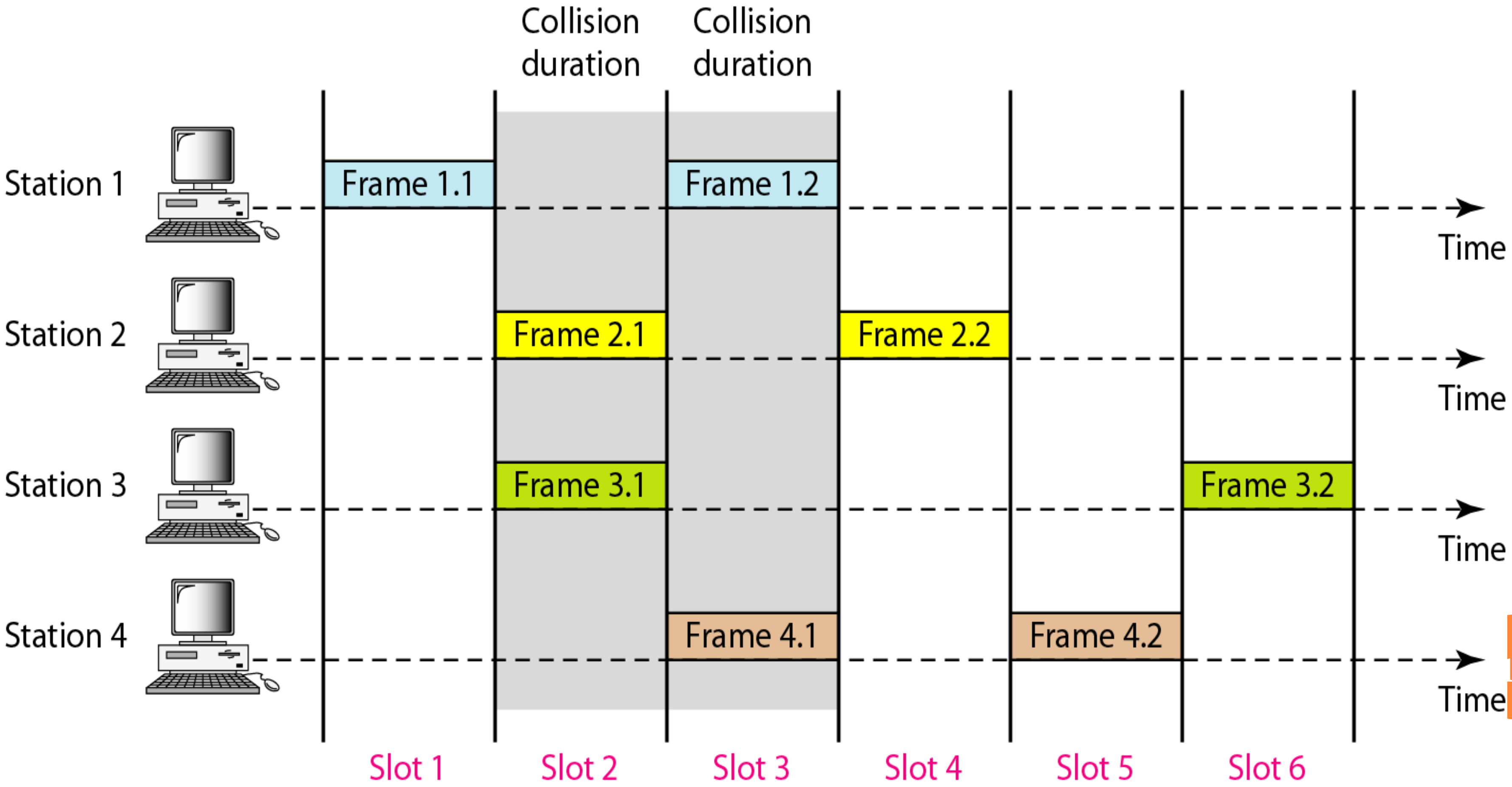
Note

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

Slotted ALOHA

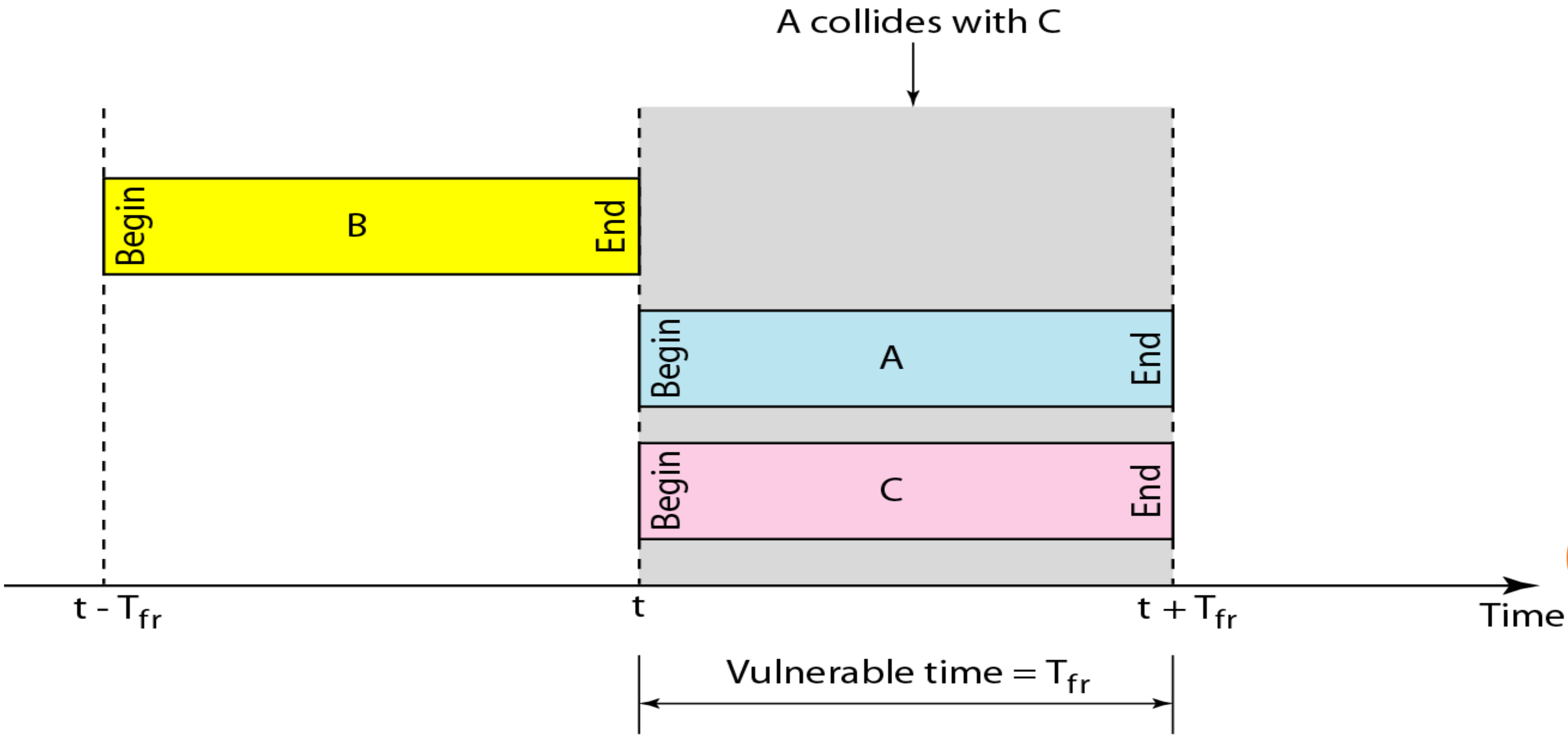
- Pure ALOHA vulnerable time = 2 x T_{fr} because there is no rule that defines when the station can send

- Slotted ALOHA was invented to improve the efficiency of pure ALOHA



Slotted ALOHA

- Throughput for slotted ALOHA is $S = G \times e^{-G}$.
- The maximum throughput $S_{max} = 0.368$ when $G = 1$(which correspond to total arrival rate of "one frame per vulnerable period")
- Slotted ALOHA vulnerable time = T_{fr}



Note

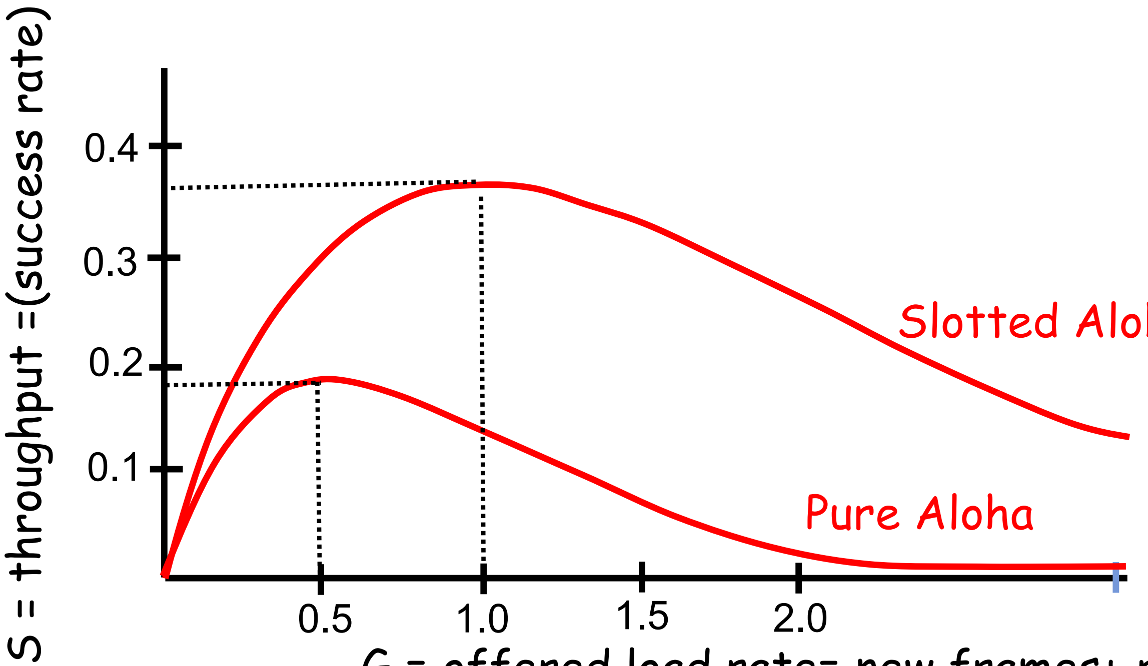
The throughput for slotted ALOHA is

$$S = G \times e^{-G}.$$

The maximum throughput

$S_{max} = 0.368$ when $G = 1$.

Efficiency of Aloha

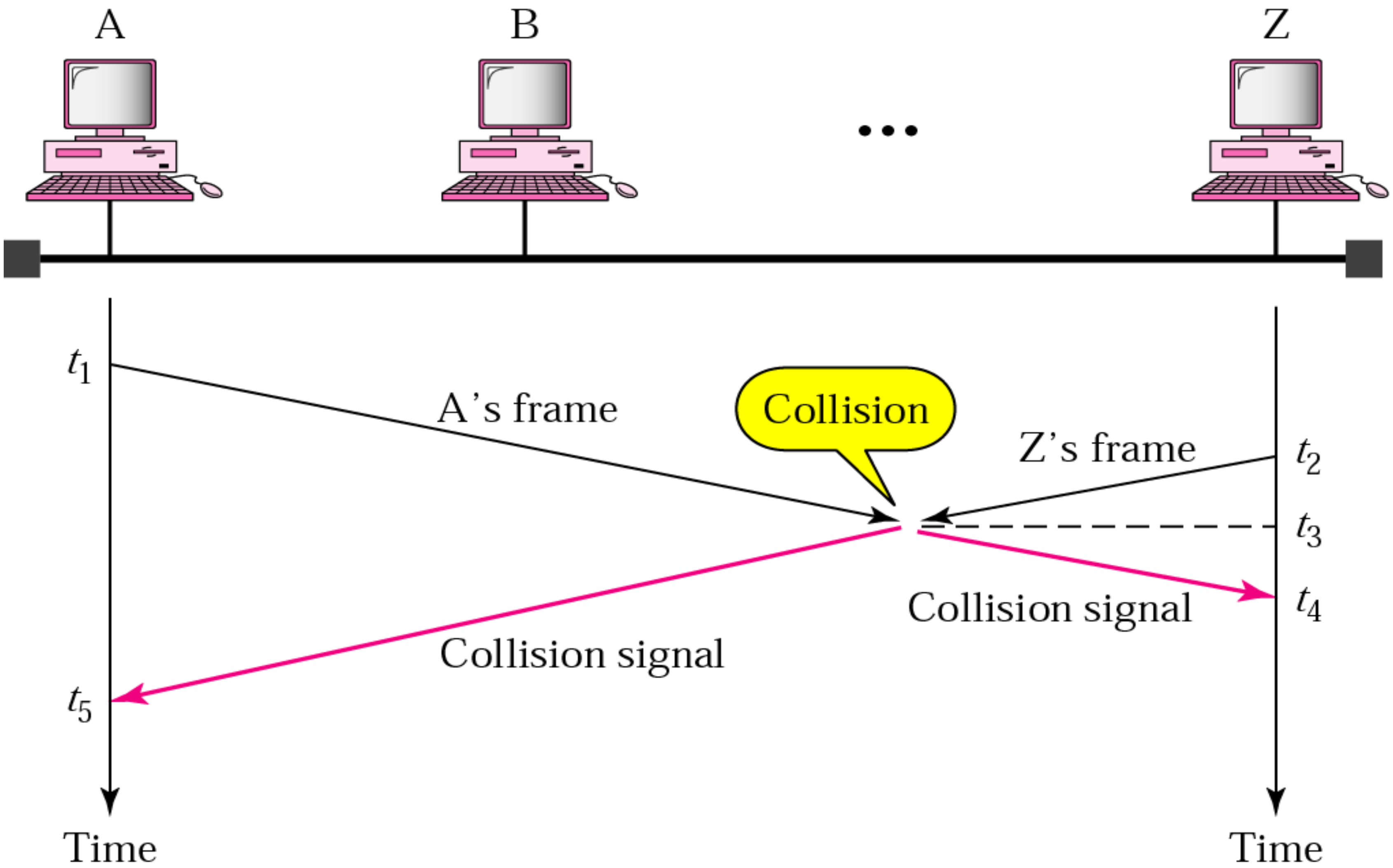


G = offered load rate= new frames+ retransmitted
= Total frames presented to the link per the transmission time of a single frame

Carrier Sense Multiple Access (CSMA)

- To improve performance, we should avoid transmissions that are definite to cause collisions
- Based on the fact that in LAN propagation time is **very small**
- If a frame was sent by a station, All stations knows immediately, so they can **wait before start sending**
- A station with frames to be sent, should sense the medium for the presence of another transmission (carrier) before it starts its own transmission
- This can **reduce** the possibility of collision but it ***cannot eliminate*** it.
- Collision can only happen when more than one station begin transmitting within a short time (the propagation time period)

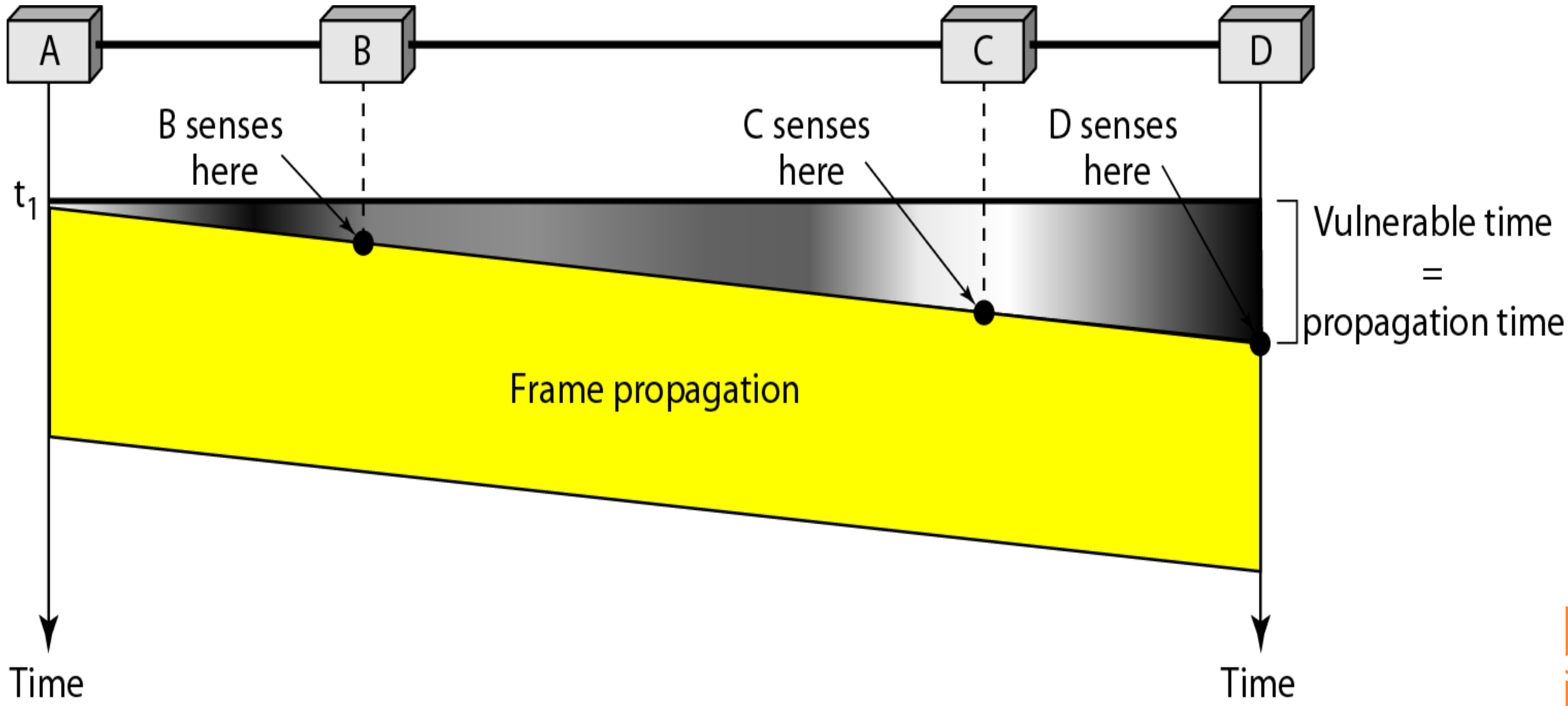
Carrier Sense Multiple Access (CSMA)



Carrier Sense Multiple Access (CSMA)

- Vulnerable time for CSMA is the **maximum propagation time**

- The longer the propagation delay, the worse the performance of the protocol because of the above case.



Types of CSMA Protocols

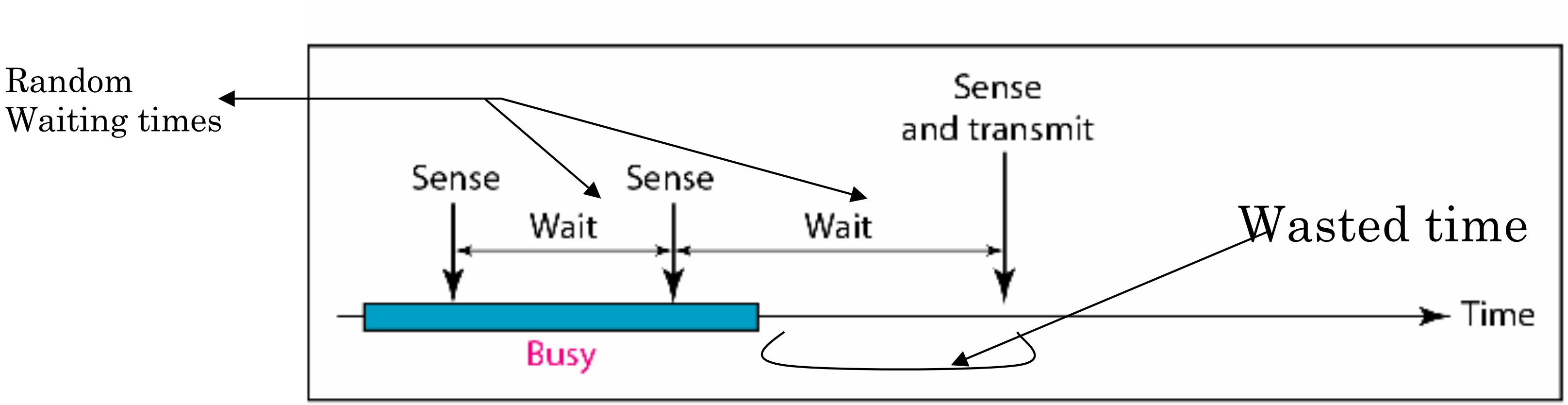
Different CSMA protocols that determine:

- What a station should do when the medium is **idle**?
- What a station should do when the medium is **busy**?

1. Non-Persistent CSMA
2. 1-Persistent CSMA
3. p-Persistent CSMA

NONPERSISTENT CSMA

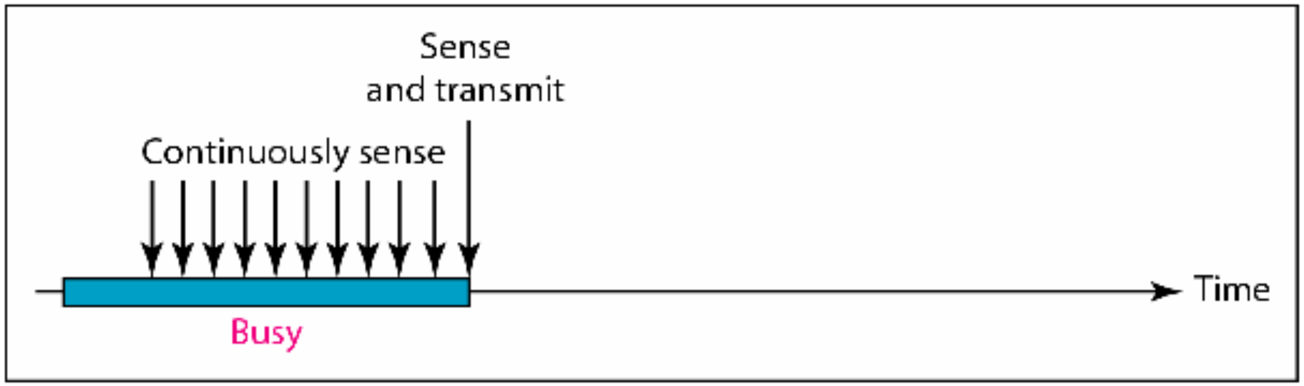
- A station with frames to be sent, should sense the medium
  1. If medium is idle, transmit; otherwise, go to 2
  2. If medium is busy, (back off) wait a random amount of time and repeat 1
- Non-persistent Stations are deferential (respect others)
- Performance:
  - Random delays reduces probability of collisions because two stations with data to be transmitted will wait for different amount of times.
  - Bandwidth is wasted if waiting time (back off) is large because medium will remain idle following end of transmission even if one or more stations have frames to send



b. Nonpersistent

1-PERSISTENT CSMA

- To avoid idle channel time, 1-persistent protocol used
- Station wishing to transmit listens to the medium:
  1. If medium idle, **transmit** immediately;
  2. If medium busy, **continuously listen** until medium becomes idle; then transmit immediately with probability 1
- Performance
  - 1-persistent stations are **selfish**
  - If two or more stations becomes ready at the same time, **collision guaranteed**

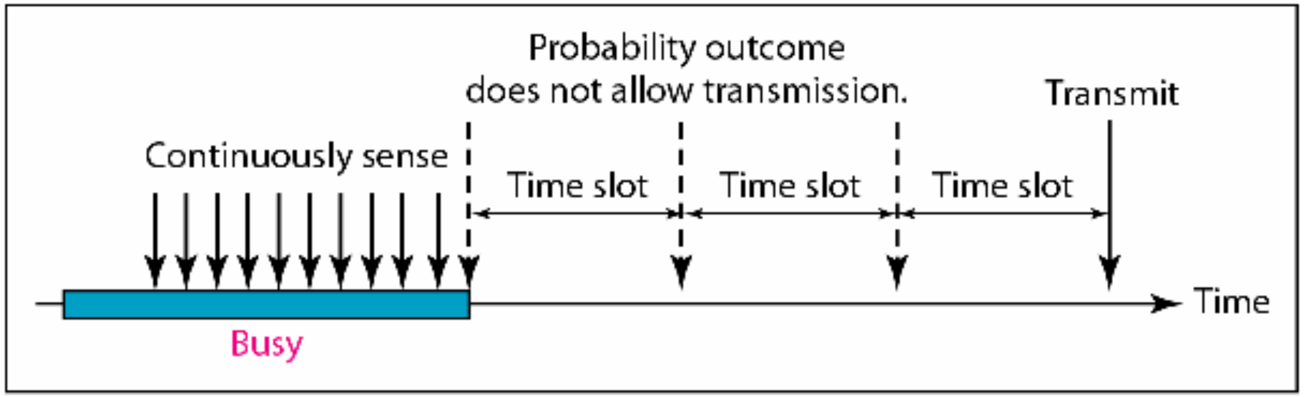


a. 1-persistent

P-PERSISTENT CSMA

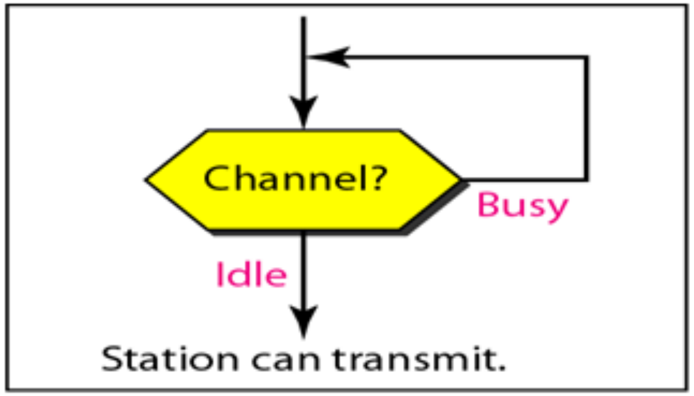
- Time is divided to slots where each Time unit (slot) typically equals **maximum propagation delay**
- Station wishing to transmit listens to the medium:

1. If medium idle,
   - transmit with probability (**p**), OR
   - wait **one time unit (slot)** with probability (**1 – p**), then repeat 1.
2. If medium busy, **continuously listen until idle** and repeat step **1**
3. Performance
   - Reduces the possibility of collisions like **non persistent**
   - Reduces channel idle time like **1-persistent**

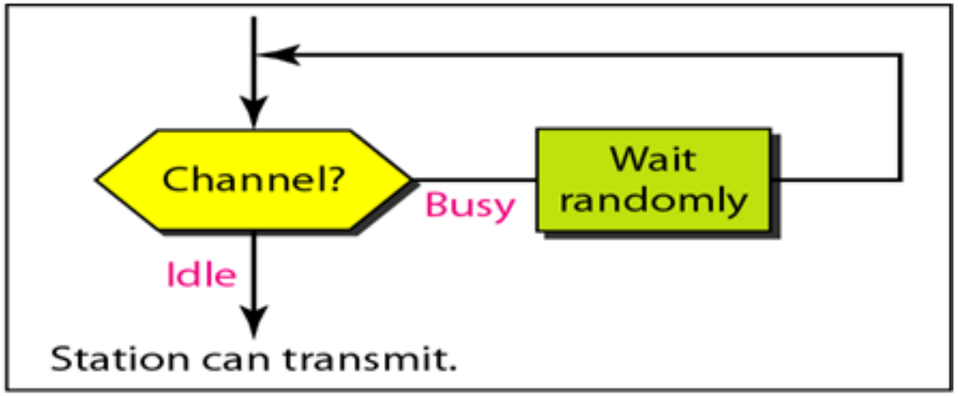


c. p-persistent

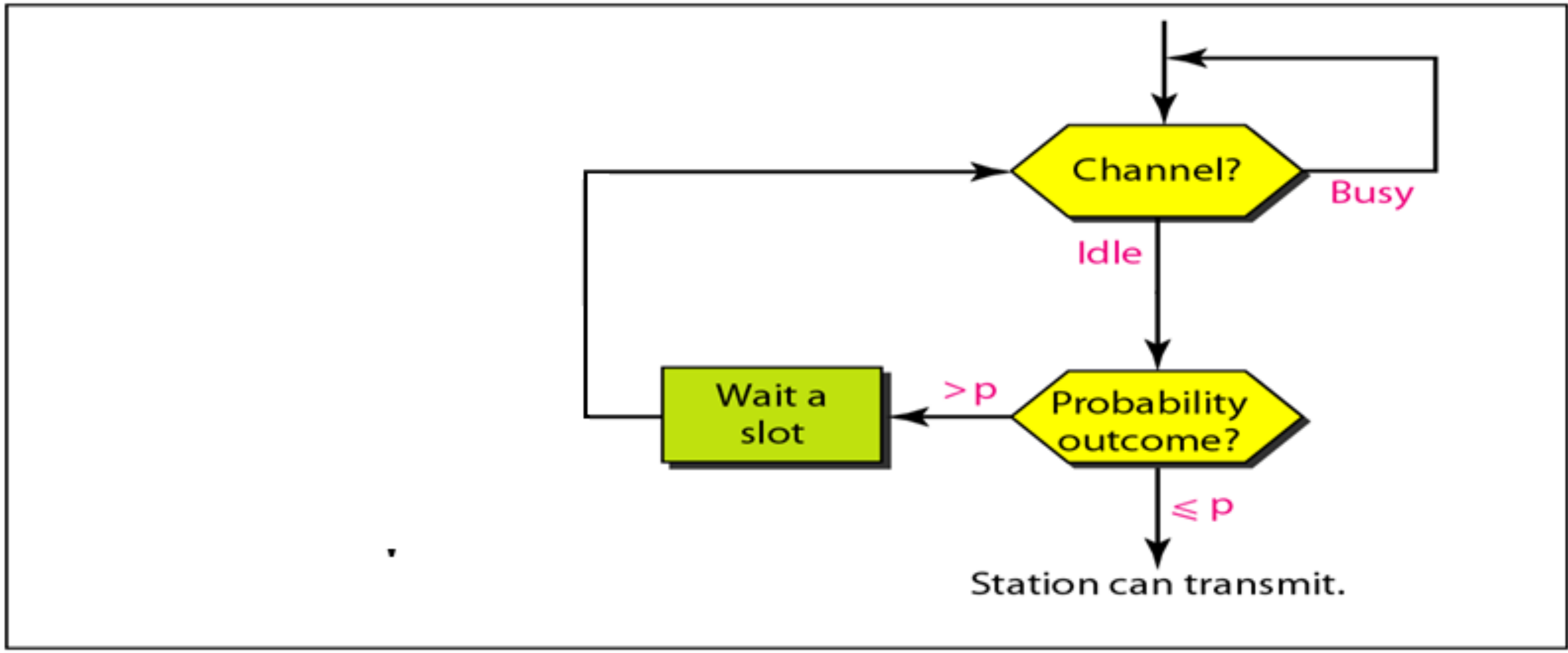
Flow diagram for three persistence methods



a. 1-persistent



b. Nonpersistent



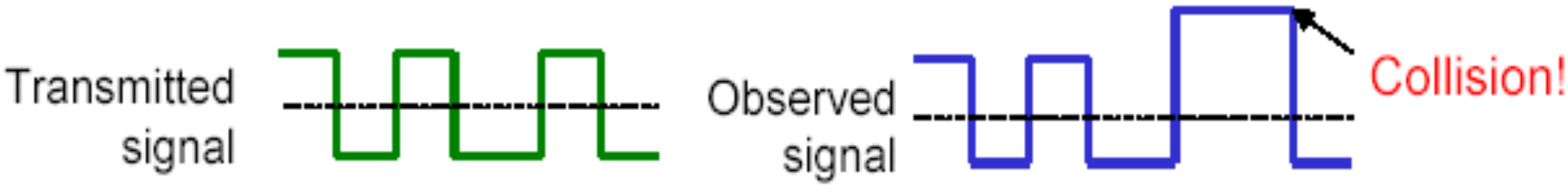
c. p-persistent

CSMA/CD (COLLISION DETECTION)

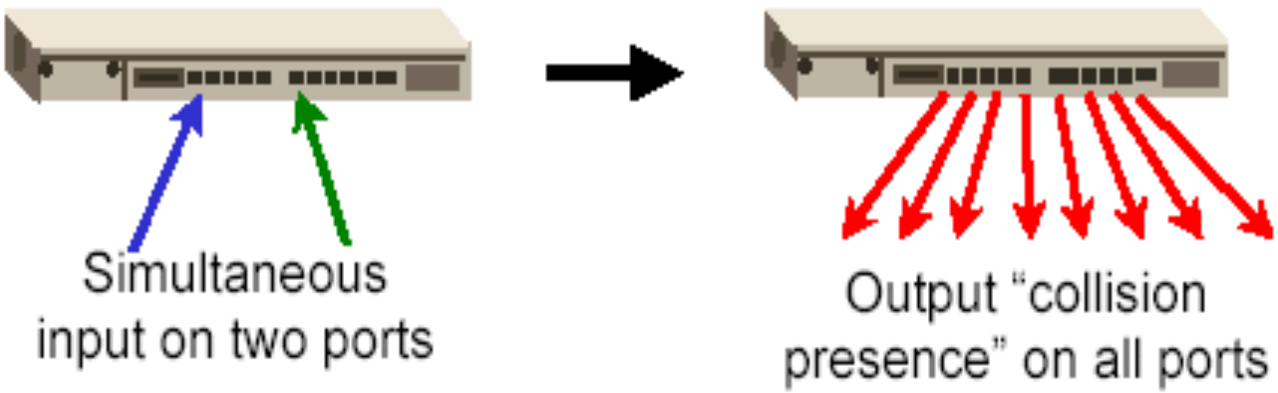
- ***CSMA (all previous methods) has an inefficiency:***
  - If a collision has occurred, the channel is **unstable** until colliding packets have **<u>been fully transmitted</u>**
- ***CSMA/CD*** *(Carrier Sense Multiple Access with Collision Detection)* ***overcomes this as follows:***
  - While transmitting, the sender is **listening to medium** for collisions.
  - Sender **stops transmission** if collision has occurred **reducing channel wastage** .

HOW DOES A NODE DETECT COLLISION?

Transceiver: **A node monitors the media while transmitting. If the observed power is more than transmitted power** of its own signal, it means collision occurred



Hub: **if input occurs simultaneously on two ports, it indicates a collision. Hub sends a collision presence signal on all ports.**



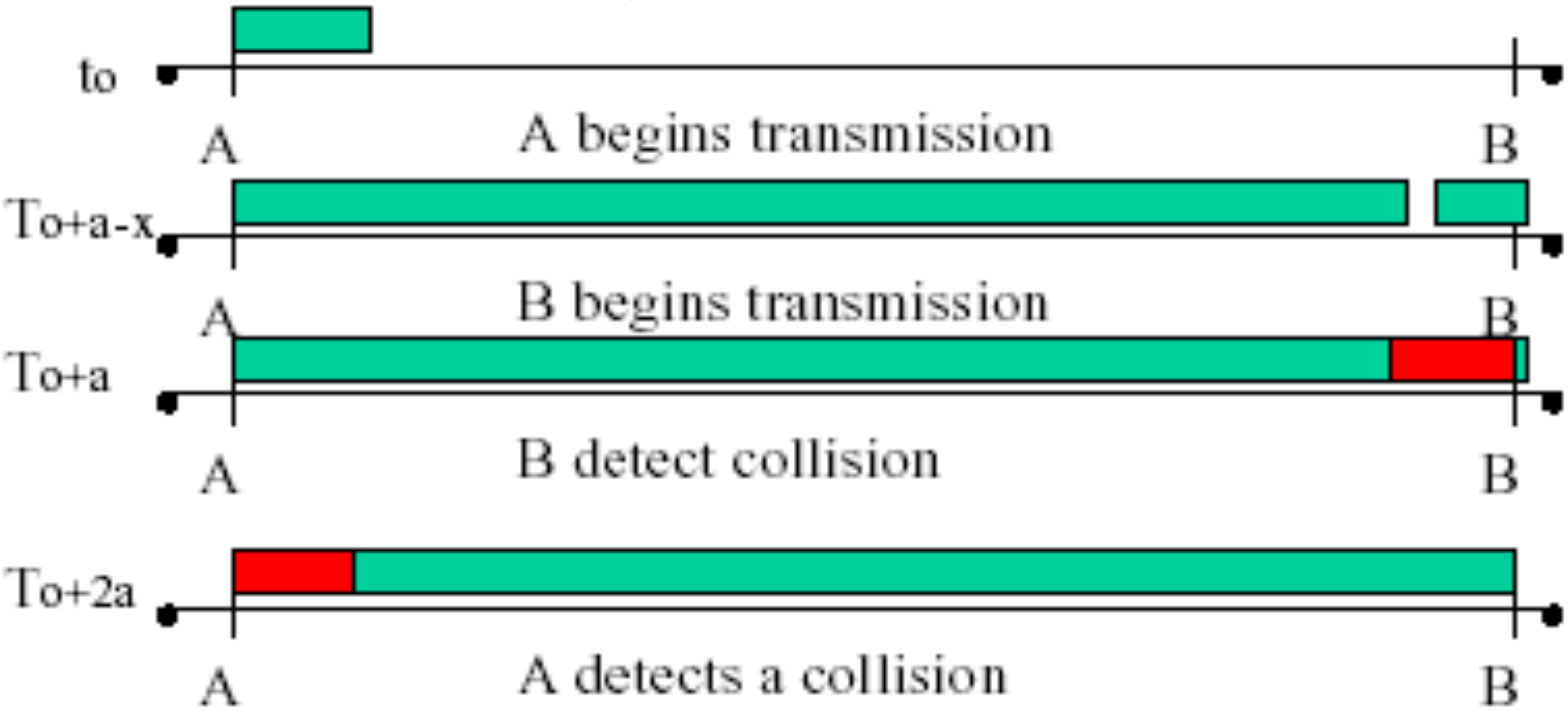
CSMA/CD PROTOCOL

- Use one of the CSMA persistence algorithm **(non-persistent, 1-persistent, p-persistent)** for transmission
- If a collision is detected by a station during its transmission then it should do the following:
  - **Abort transmission** and
  - **Transmit** a ***jam signal*** (48 bit) to notify other stations of collision so that they will **discard the transmitted fra**me
  - After sending the ***jam signal***, **back off (wait) for a *random*** amount of time, then Transmit the frame again

CSMA/CD

- ***Question:*** How long does it take to detect a collision?
- ***Answer:*** *In the* ***worst case***, **twice the maximum propagation delay of the medium**

Note: **a = maximum propagation delay**



CSMA/CD

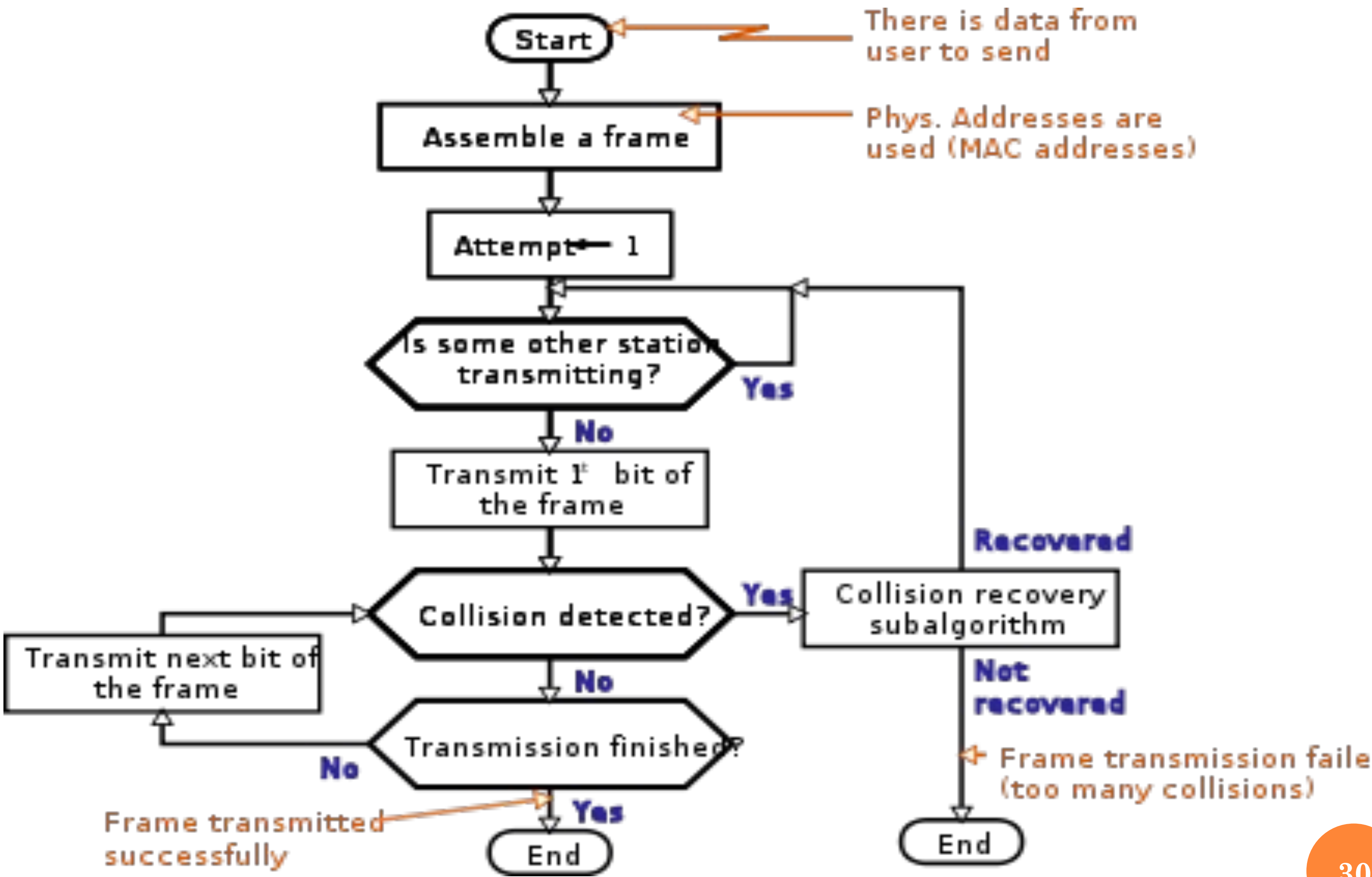
- Restrictions of CSMA / CD:
  - Packet **transmission time** should be **at least** as long as the time needed to detect a collision (2 * maximum propagation delay + *jam sequence* transmission time)

    Packet **transmission time > (2***Maximum propagation delay + *Jam sequence* transmission time)

  - To ensure that packet transmit with out collision, a host must be able to detect a collision before it finishes transmitting a packet

  - In other words, there is a minimum length packet for CSMA/CD networks

Simplified Algorithm of CSMA/CD



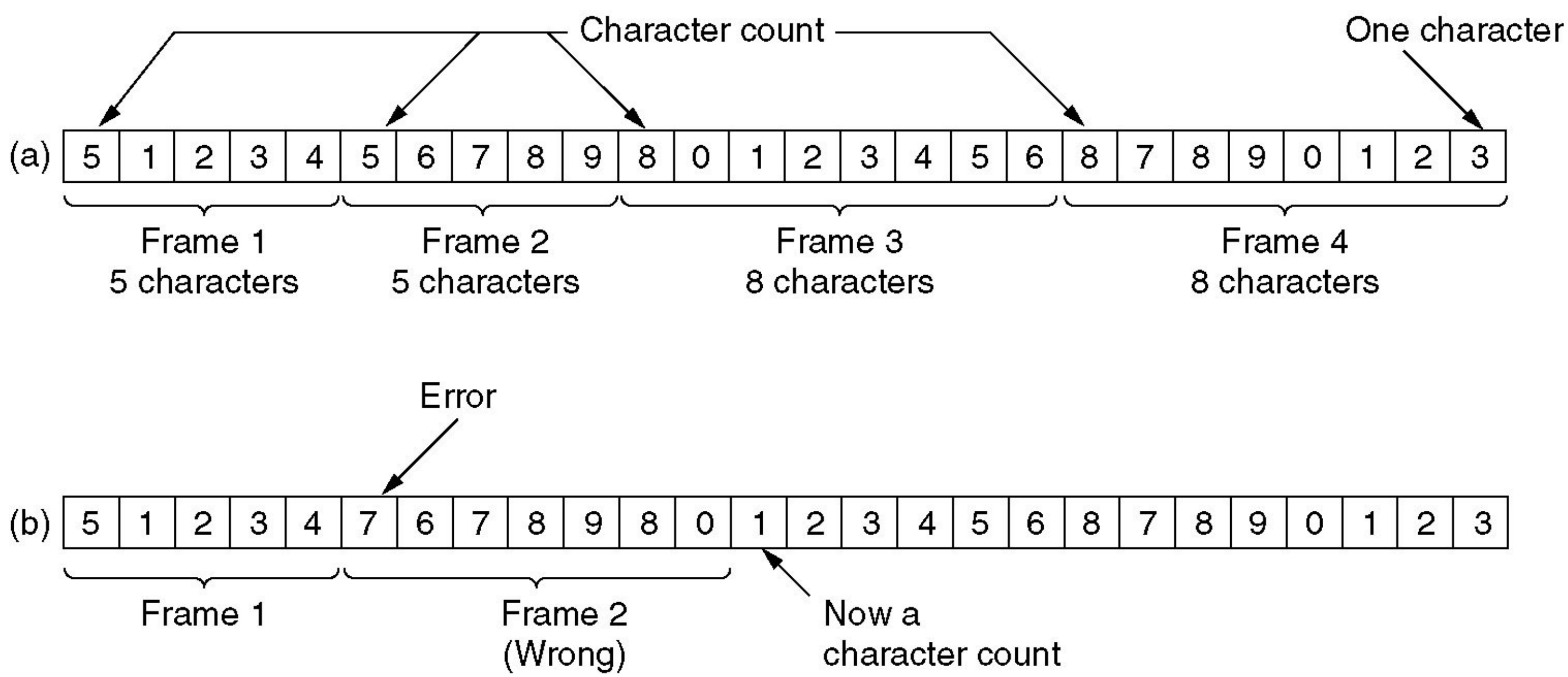
DOES SWITCHED NETWORK NEED CSMA/CD?

FRAMING

- Character Count
- Flag bytes with byte stuffing
- Flag bytes with bit stuffing

Framing with Character Count

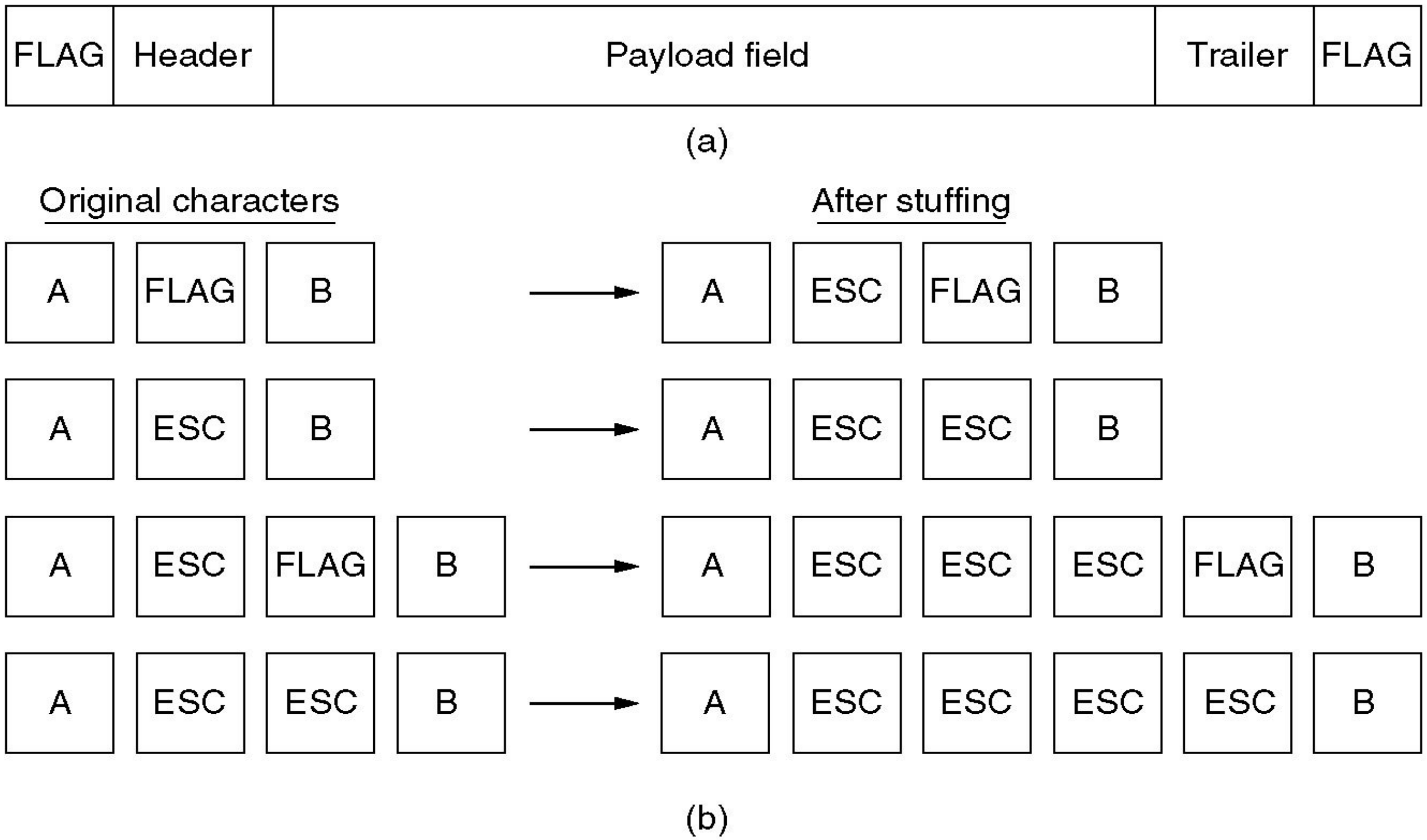
A character stream. (a) Without errors.
(b) With one error.



Problem with Framing with Character Count

- What if the count is garbled
- Even if with checksum, the receiver knows that the frame is bad there is no way to tell where the next frame starts.
- Asking for retransmission doesn't help either because the start of the retransmitted frame is not known
- No longer used independently

FRAMING WITH BYTE STUFFING



Problem in Framing with byte stuffing

- A major disadvantage of using this framing method is that it is closely tied to the use of 8-bit characters
- Not all character codes use 8-bit characters
- Example. UNICODE uses 16-bit characters
- Can't handle heterogeneous environment

FRAMING WITH BIT STUFFING

- This method allows character codes with an arbitrary number of bits per character

- Each frame begins and ends with a special bit pattern, 01111110 (a flag byte).

- Sender's data link layer encounters five consecutive 1s in the data, it automatically stuffs a 0 bit into the outgoing bit stream

- When the receiver sees five consecutive incoming 1 bits, followed by a 0 bit, it automatically destuffs (deletes) the 0 bit

(a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

(b) 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0

Stuffed bits

(c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

(a) The original data.

(b) The data as they appear on the line.

(c) The data as they are stored in receiver's memory after destuffing.

Problems with bit stuffing

- This method only applicable to networks in which the encoding on the physical medium contains some redundancy
- Example, some LANs encode 1 bit of data by using 2 physical voltages. Normally, a 1 bit is a high-low pair and a 0 bit is a low-high pair
- Transition in the middle, making it easy for the receiver to locate the bit boundaries.