AMRITA
VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY

15CSE312
COMPUTER NETWORKS
3-0-0 3

Chapter 5: Data Link Layer

- **Multiple Access Protocols**
  - Channel partitioning
    - ✓ FDMA
    - ✓ TDMA
    - ✓ CDMA
  - random access
    - ✓ Pure ALOHA
    - ✓ Slotted ALOHA
    - ✓ CSMA (carrier sense multiple access)
  - taking turns”

Data Link Layer- Introduction

This layer is the protocol layer that **transfers data between nodes** on a network segment across the physical layer. The data link layer provides the functional and procedural means to transfer data between network entities and may also provide the means to detect and possibly correct errors that can occur in the physical layer.

Any device that runs a link-layer protocol as a node. **Nodes** include hosts, routers, switches, and WiFi access points We will also refer to the communication channels that connect adjacent nodes along the communication path as **links**. In order for a datagram to be transferred from source host to destination host, it must be moved over each of the individual links in the end-to-end path.

The basic service of any link layer is to move a datagram from one node to an adjacent node over a single communication link

Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs
- addressing, ARP
- Ethernet
- switches
- VLANS

5.5 link virtualization: MPLS

5.6 data center networking

5.7 a day in the life of a web request

Multiple access links, protocols

two types of "links":

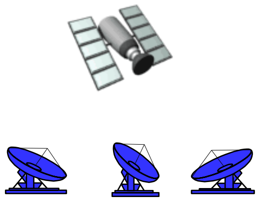
- point-to-point
  - PPP for dial-up access
  - point-to-point link between Ethernet switch, host
- *broadcast (shared wire or medium)*
  - old-fashioned Ethernet
  - upstream HFC
  - 802.11 wireless LAN

A **point-to-point link** consists of a single sender at one end of the link and a single receiver at the other end of the link

The second type of link, a **broadcast link**, can have multiple sending and receiving nodes all connected to the same, single, shared broadcast channel. Ethernet and wireless LANs are examples of broadcast link-layer technologies.
Analogy : CockTail party, teacher and class

shared wire (e.g., cabled Ethernet)

shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

Collision

When all nodes are capable of transmitting frames, **more than two nodes transmit frames at the same time**, all of the nodes receive multiple frames at the same time; that is, the transmitted frames **collide** at all of the receivers. Typically, when there is a collision, none of the receiving nodes can make any sense of any of the frames that were transmitted; in a sense, the signals of the colliding frames become inextricably tangled together. Thus, all the frames involved in the collision are lost, and the broadcast channel is wasted during the collision interval. Clearly, if many nodes want to transmit frames frequently, many transmissions will result in collisions, and much of the bandwidth of the broadcast channel will be wasted

Multiple access protocols- hundreds or even thousands of nodes can directly communicate over a broadcast channel.

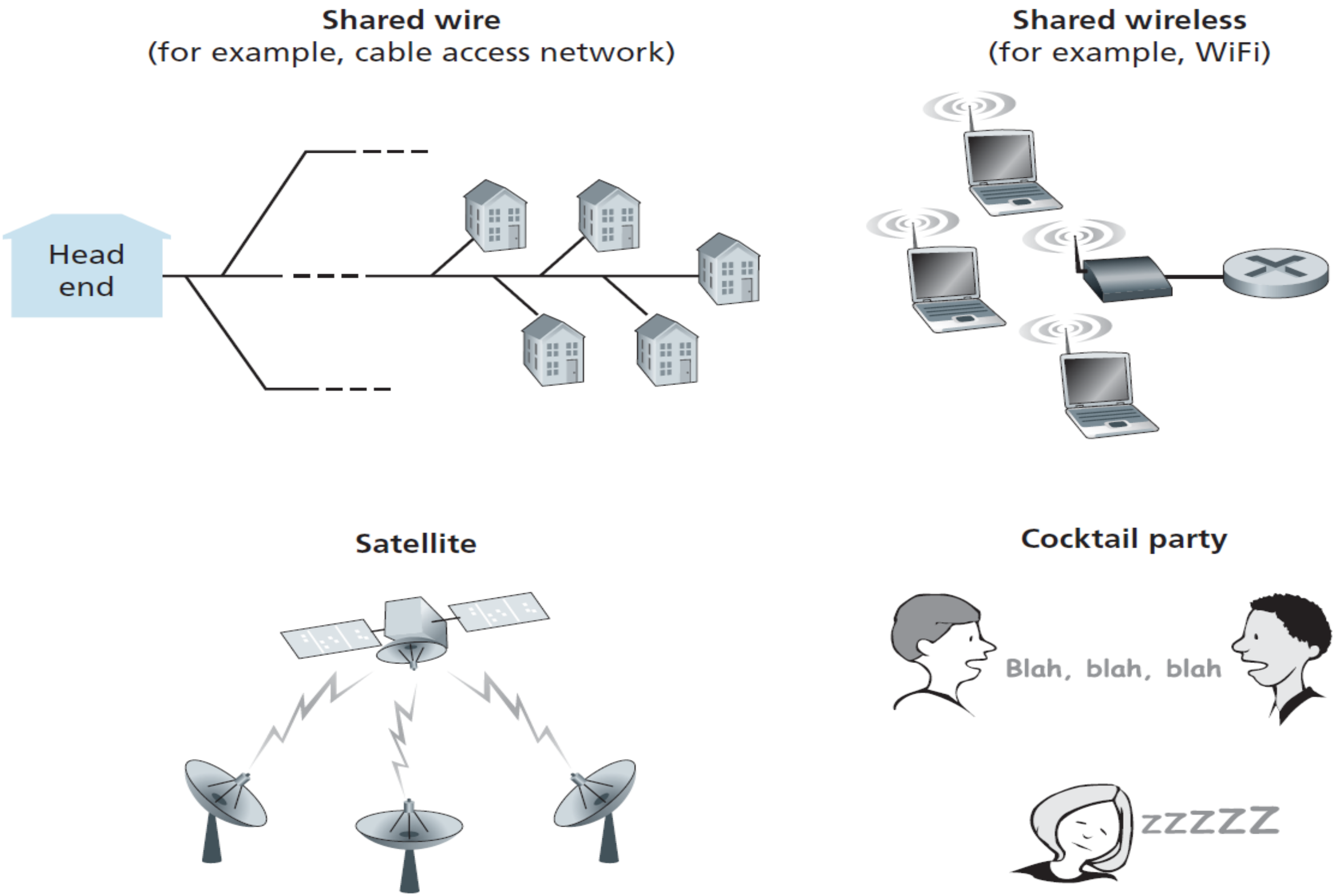


Figure 5.8 ♦ Various multiple access channels

Multiple access protocols- hundreds or even thousands of nodes can directly communicate over a broadcast channel.

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

multiple access protocol

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

An ideal multiple access protocol

given: broadcast channel of rate R bps

desiderata:

1. when one node wants to transmit, it can send at rate R.
2. When M nodes have data to send, each of these nodes has a throughput of R/M bps. This need not necessarily imply that each of the M nodes always. has an instantaneous rate of R/M, but rather that each node should have an average transmission rate of R/M over some suitably defined interval of time

- 3. fully decentralized:
  - no special node to coordinate transmissions
  - no synchronization of clocks, slots

4. Simple

MAC: Media Access Control

MAC protocols: taxonomy

three broad classes:

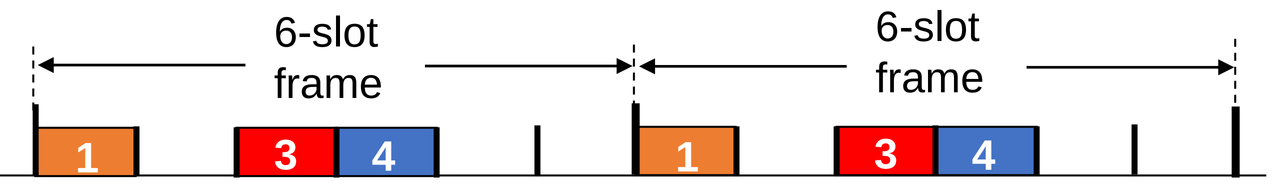
- *channel partitioning*
  - divide channel into smaller "pieces" (time slots, frequency, code)
  - allocate piece to node for exclusive use
- *random access*
  - channel not divided, allow collisions
  - "recover" from collisions
- *"taking turns"*
  - nodes take turns, but nodes with more to send can take longer turns

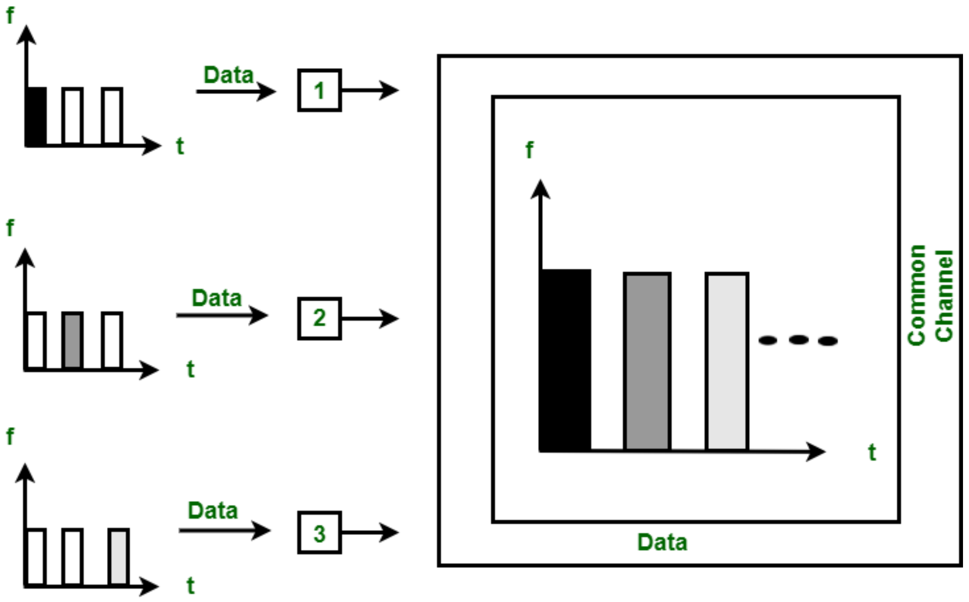
Channel partitioning MAC protocols: TDMA

TDMA: time division multiple access

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

Advantage: TDM is appealing because it eliminates collisions and is perfectly fair:
Disadvantage: Disadvantage using TDMA technology is that the users has a predefined time slot. When moving from one cell site to other, if all the time slots in this cell are full the user might be disconnected. Another problem in TDMA is that it is subjected to multipath distortion., wait for its slot





Channel partitioning MAC protocols: FDMA

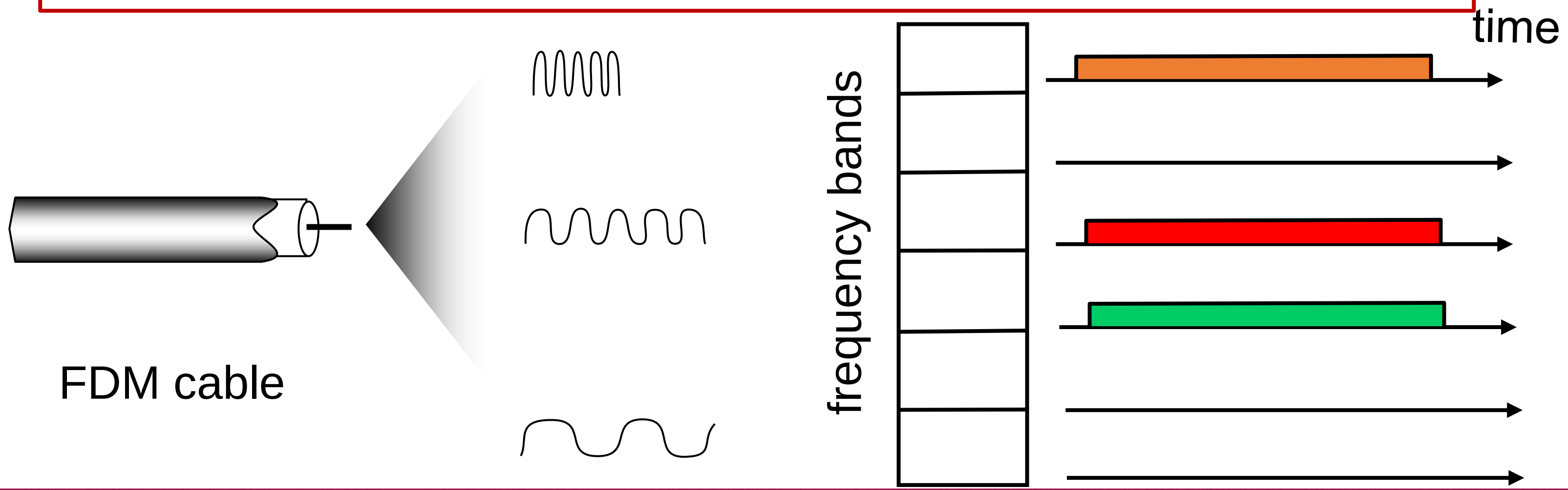
FDMA: frequency division multiple access

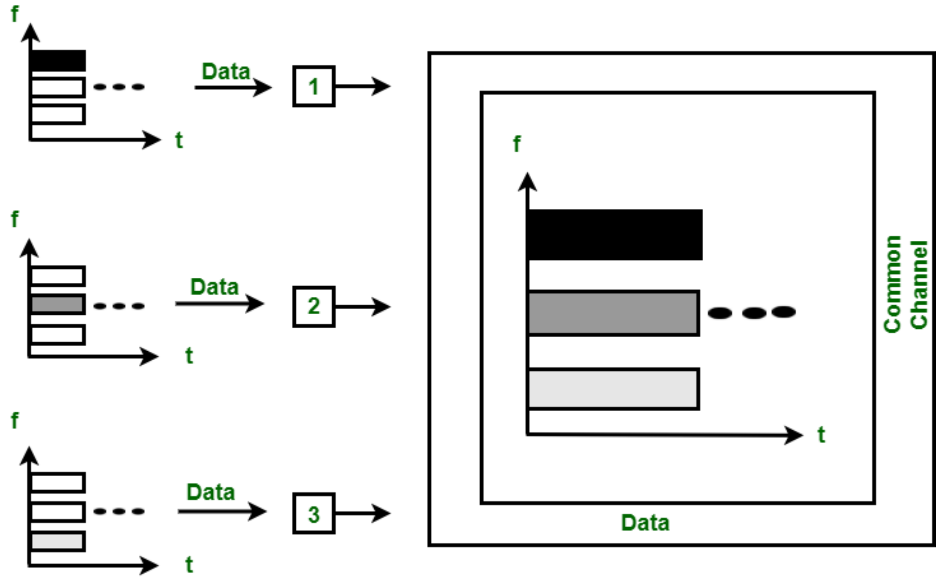
- ❖channel spectrum divided into frequency bands
- ❖each station assigned fixed frequency band
- ❖unused transmission time in frequency bands go idle
- ❖example: 6-station LAN, 1,3,4 have pkt, frequency bands 2,5,6 idle

In this bandwidth is divided into various frequency bands. Each station is allocated with band to send data and that band is reserved for particular station for all the time

The frequency bands of different stations are separated by small band of unused frequency and that unused frequency bands are called as **guard bands** that prevents the interference of stations.

It avoids collisions and divides the bandwidth fairly among the *N* nodes. However,
But—a node is limited to a bandwidth of *R*/*N*, even when it is the only node with packets to send





FDMA ,TDMA

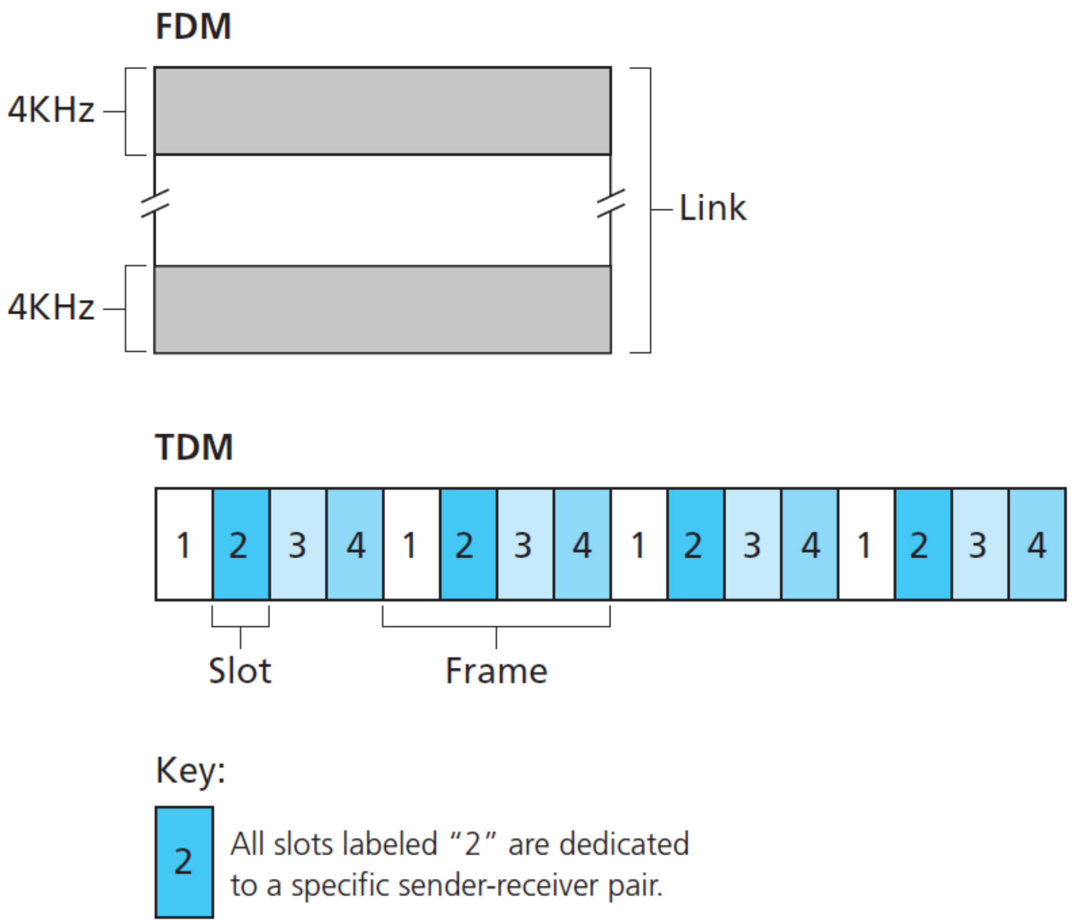


Figure 5.9 ♦ A four-node TDM and FDM example

Code Division Multiplexing

A third channel partitioning protocol is **code division multiple access (CDMA)**.

While TDM and FDM assign time slots and frequencies, respectively, to the nodes, CDMA assigns a different *code* to each node. Each node then uses its unique code to encode the data bits it sends. If the codes are chosen carefully, CDMA networks have the wonderful property that different nodes can transmit *simultaneously* and yet have their respective receivers correctly receive a sender's encoded data bits (assuming the receiver knows the sender's code) in spite of interfering transmissions by other nodes.

CDMA has been used in military systems for some time (due to its anti-jamming properties) and now has widespread civilian use, particularly in cellular telephony. CDMA's use is tightly tied to wireless channels,

Differences

FDMA	TDMA	CDMA
FDMA stands for Frequency Division Multiple Access.	TDMA stands for Time Division Multiple Access.	CDMA stands for Code Division Multiple Access.
In this, sharing of bandwidth among different stations takes place.	In this, only the sharing of time of satellite transponder takes place.	In this, there is sharing of both i.e. bandwidth and time among different stations takes place.
There is no need of any codeword.	There is no need of any codeword.	Codeword is necessary.
In this, there is only need of guard bands between the adjacent channels are necessary.	In this, guard time of the adjacent slots are necessary.	In this, both guard bands and guard time are necessary.
Synchronization is not required.	Synchronization is required.	Synchronization is not required.
The rate of data is low.	The rate of data is medium.	The rate of data is high.
Mode of data transfer is continuous signal.	Mode of data transfer is signal in burts.	Mode of data transfer is digital signal.
It is little flexible.	It is moderate flexible.	It is highly flexible.

Random Access protocols

00110010
01001100

In a random access protocol, a transmitting node always transmits at the full rate of the channel, namely, *R* bps. When there is a collision, each node involved in the collision repeatedly retransmits its frame (that is, packet) until its frame gets through without a collision.

But when a node experiences a collision, it doesn't necessarily retransmit the frame right away. Instead it waits a random delay before retransmitting the frame. Each node involved in a collision chooses independent random delays. Because the random delays are independently chosen, it is possible that one of the nodes will pick a delay that is sufficiently less than the delays of the other colliding nodes and will therefore be able to sneak its frame into the channel without a collision.

collison


A
B


A
A-Frame
B-Frame
B


COLLISION
Frames lost /
Corrupted
A
BOOM!
B

Random access protocols

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

Pure (unslotted) ALOHA

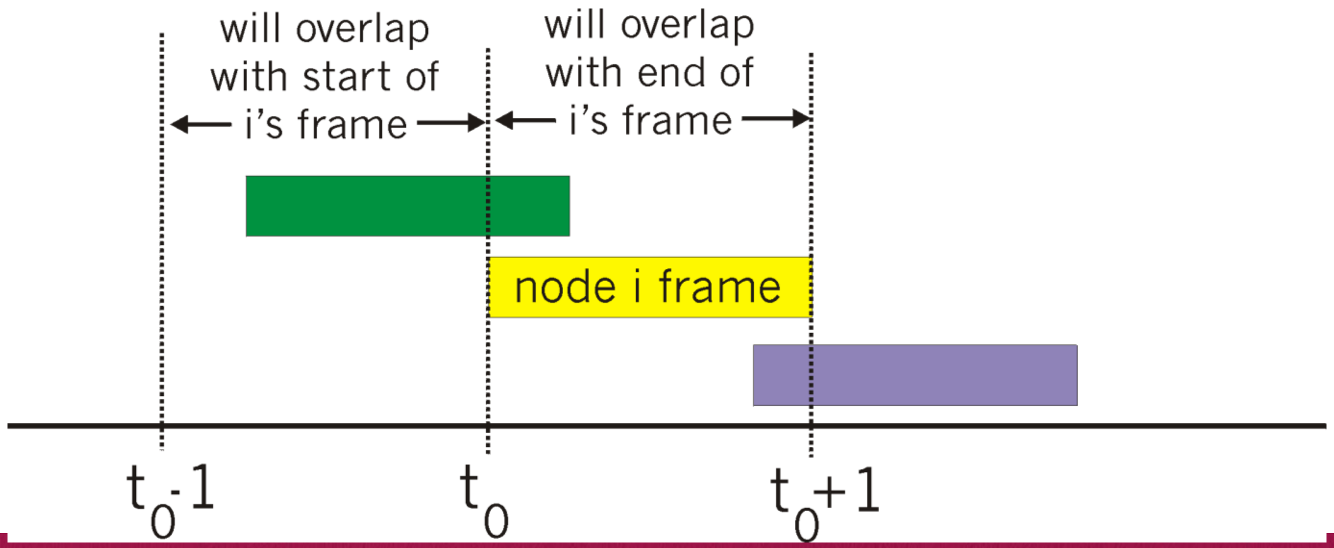
In pure ALOHA,

when a frame first arrives (that is, a network-layer datagram is passed down from the network layer at the sending node), the node immediately transmits the frame in its entirety into the broadcast channel. If a transmitted frame experiences a collision with one or more other transmissions, the node will then immediately (after completely transmitting its collided frame) retransmit the frame with probability p.

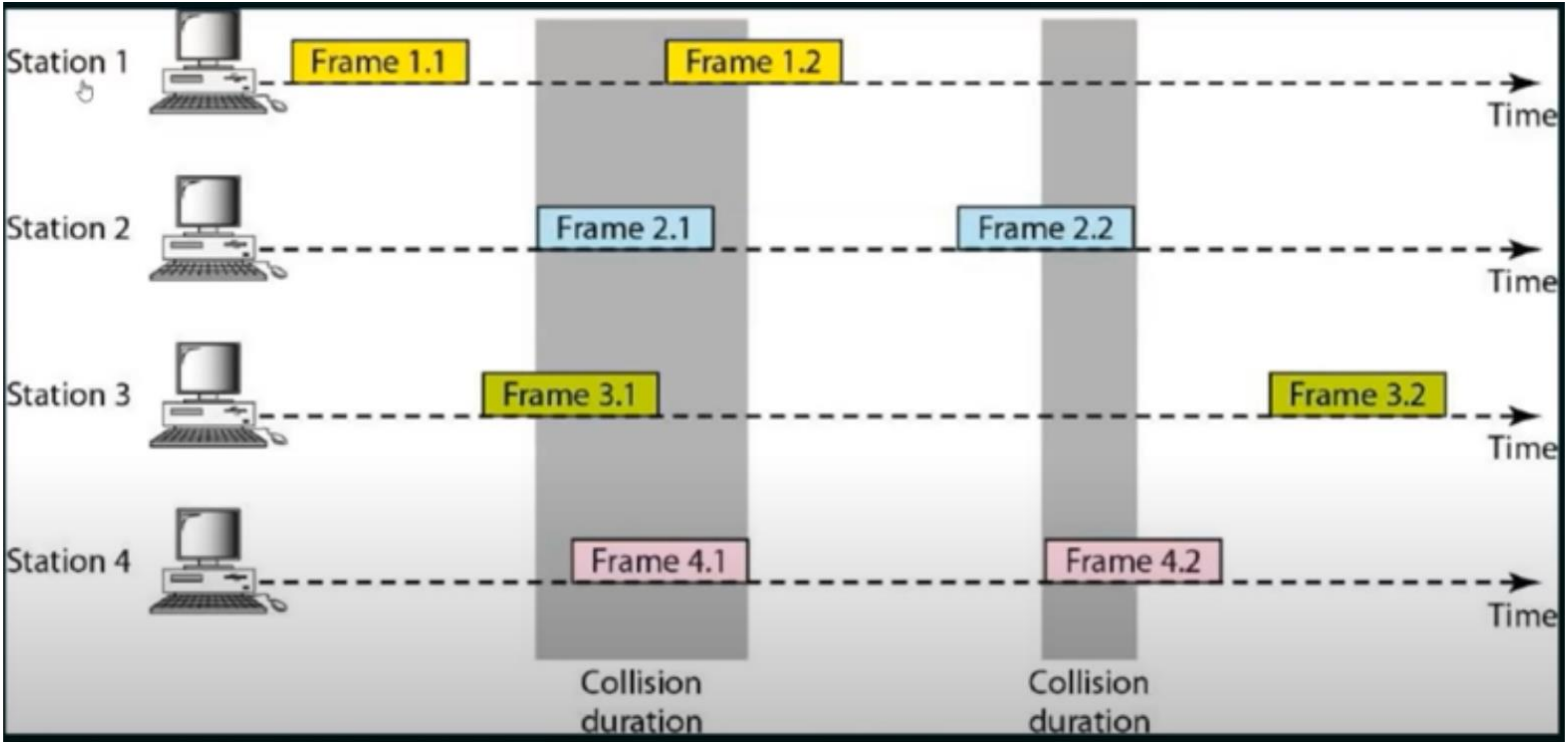
Otherwise, the node waits for a frame transmission time. After this wait, it then transmits the frame with probability p, or waits (remaining idle) for another frame time with probability $1 - p$.

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 $[t_0-1, t_0+1]$

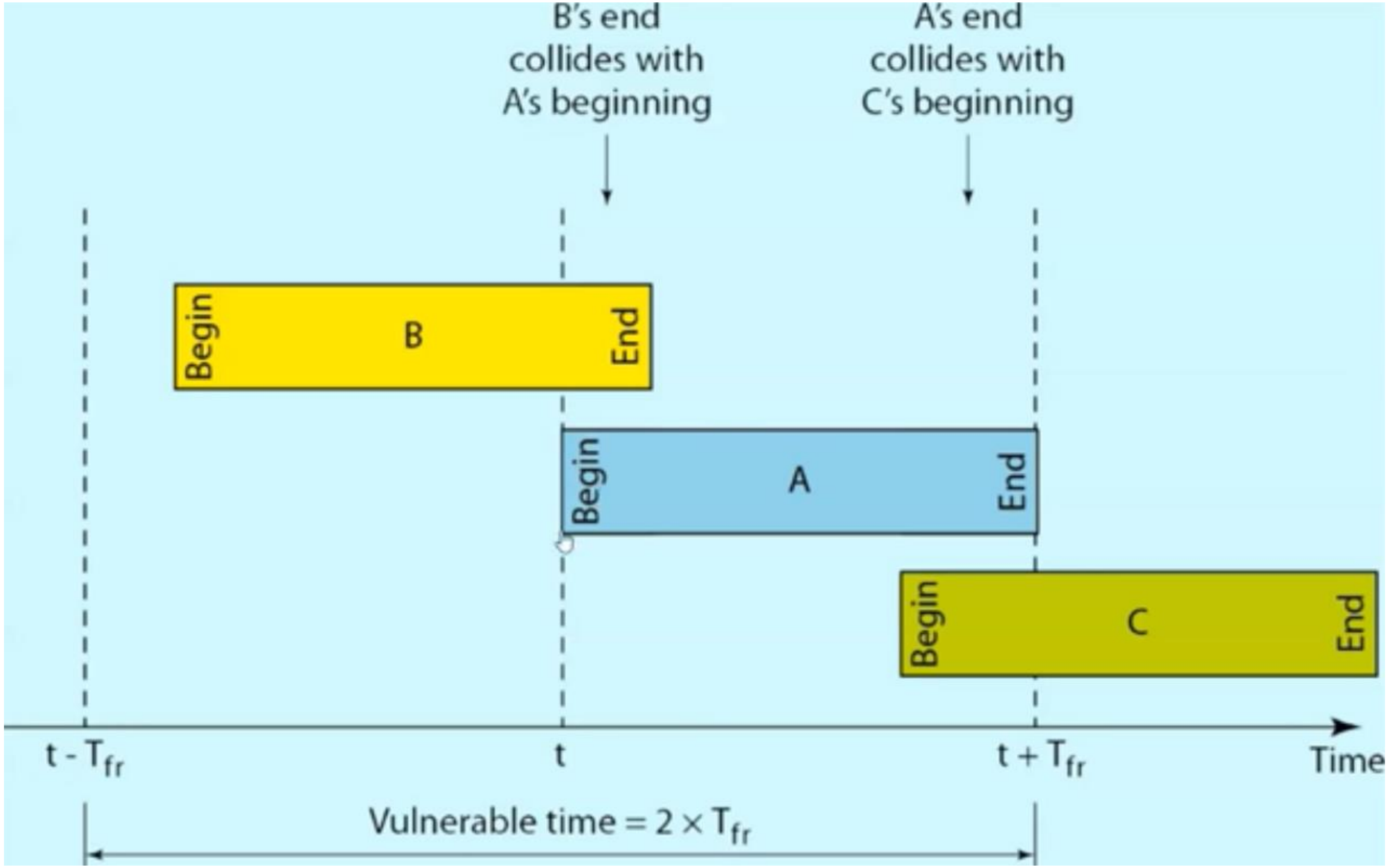


Pure ALOHA



Station 1
Frame 1.1
Frame 1.2
Time
Station 2
Frame 2.1
Frame 2.2
Time
Station 3
Frame 3.1
Frame 3.2
Time
Station 4
Frame 4.1
Frame 4.2
Time
Collision duration
Collision duration

Pure ALOHA



B's end collides with A's beginning
A's end collides with C's beginning
Begin
B
End
Begin
A
End
Begin
C
End
t - T_fr
t
t + T_fr
Time
Vulnerable time = 2 × T_fr

1. Pure Aloha-

- It allows the stations to transmit data at any time whenever they want.
- After transmitting the data packet, station waits for some time.

Then, following 2 cases are possible-

Case-01:

Transmitting station receives an acknowledgement from the receiving station.

- In this case, transmitting station assumes that the transmission is successful.

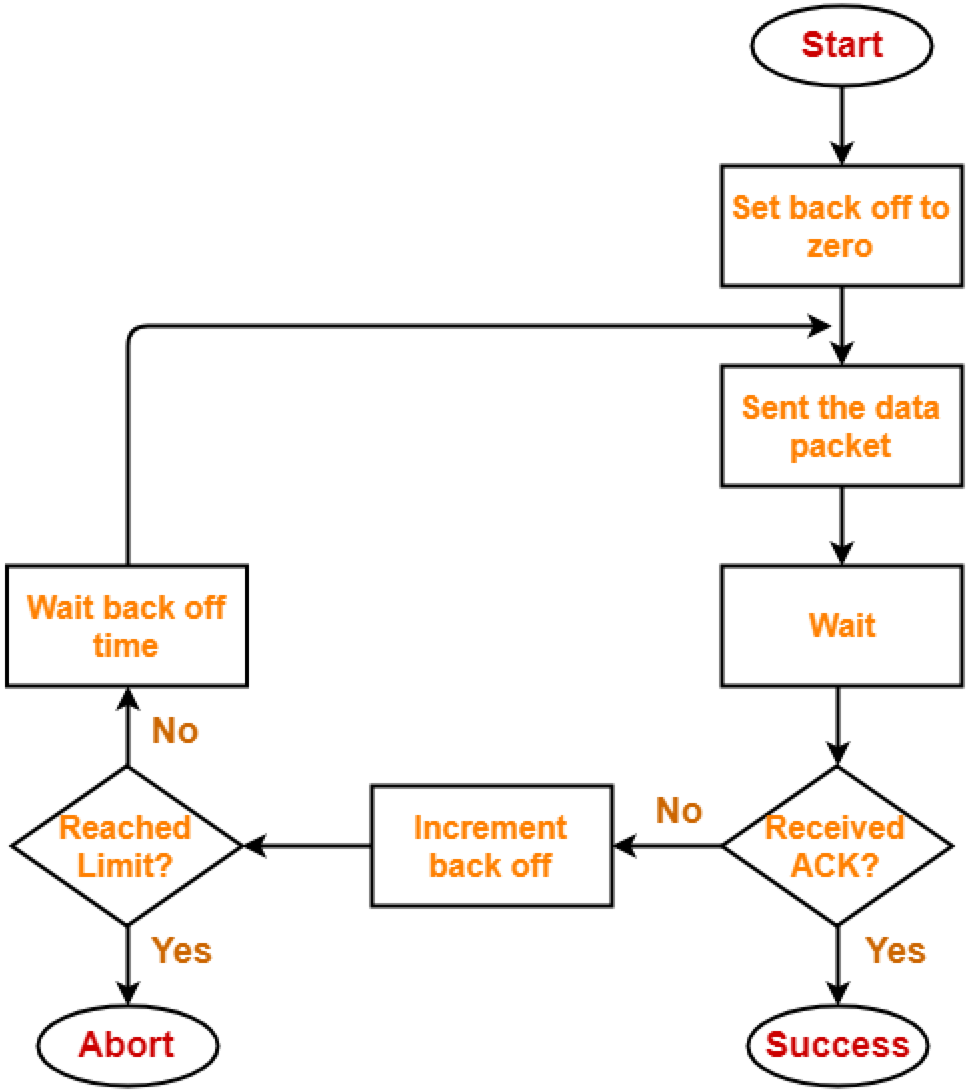
Case-02:

Transmitting station does not receive any acknowledgement within specified time from the receiving station.

- In this case, transmitting station assumes that the transmission is unsuccessful.

Then, Transmitting station uses a **Back Off Strategy** and waits for some random amount of time.

- After back off time, it transmits the data packet again.
- It keeps trying until the back off limit is reached after which it aborts the transmission.



Flowchart for Pure Aloha

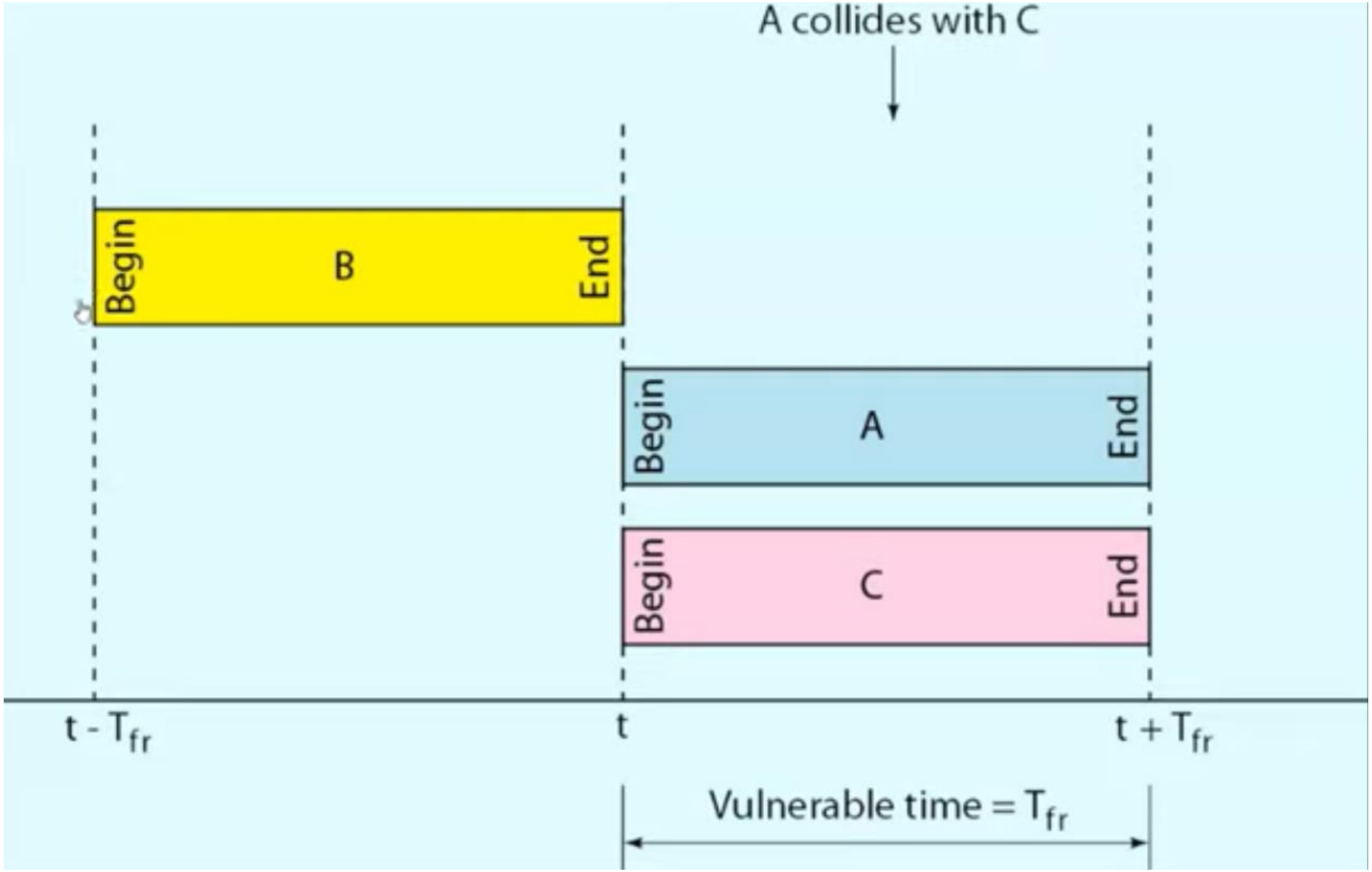
Slotted ALOHA

Assumptions

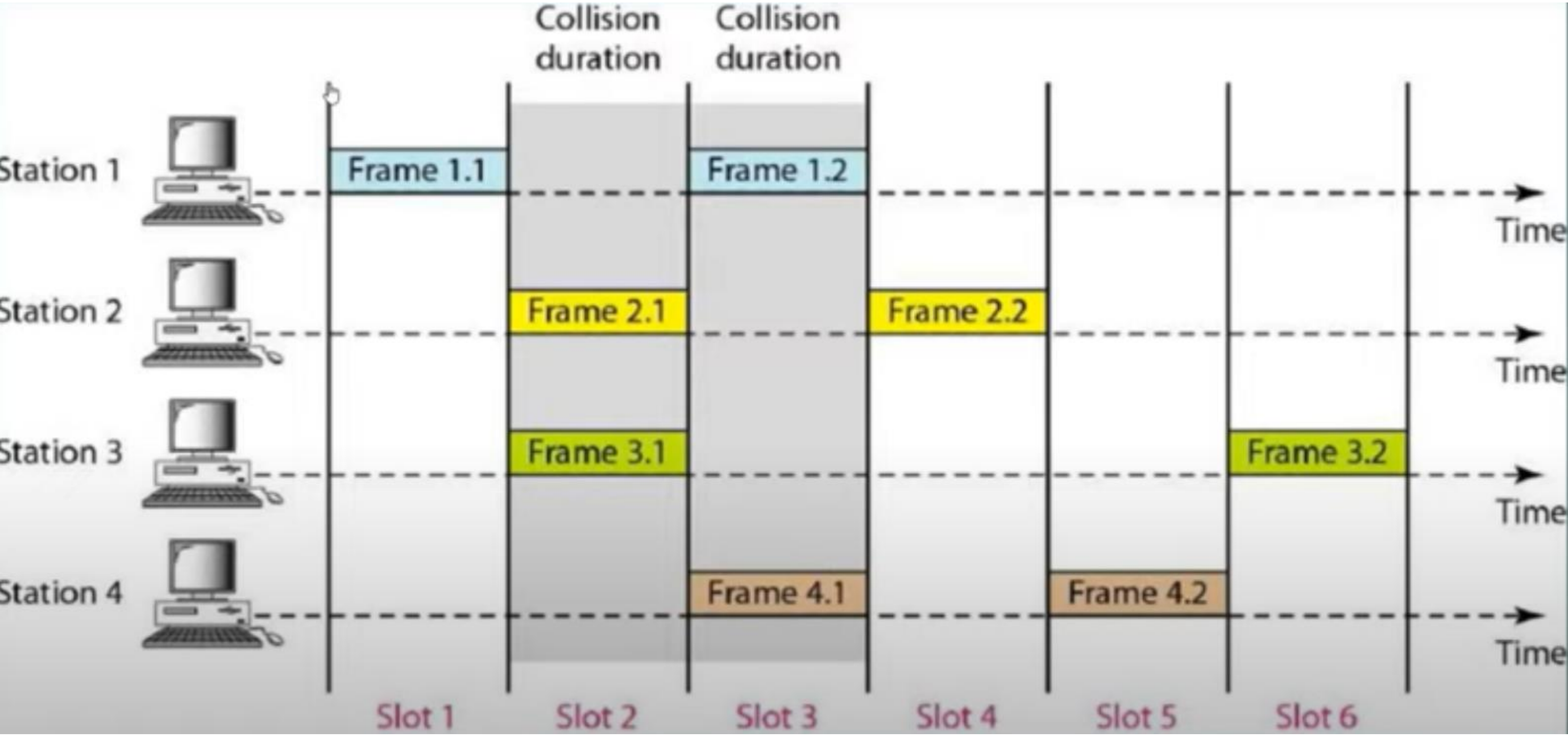
- All frames consist of exactly L bits.
- Time is divided into slots of size L/R seconds (that is, a slot equals the time to transmit one frame).
- Nodes start to transmit frames only at the beginnings of slots.
- The nodes are synchronized so that each node knows when the slots begin.
- If two or more frames collide in a slot, then all the nodes detect the collision event before the slot ends.

operation:

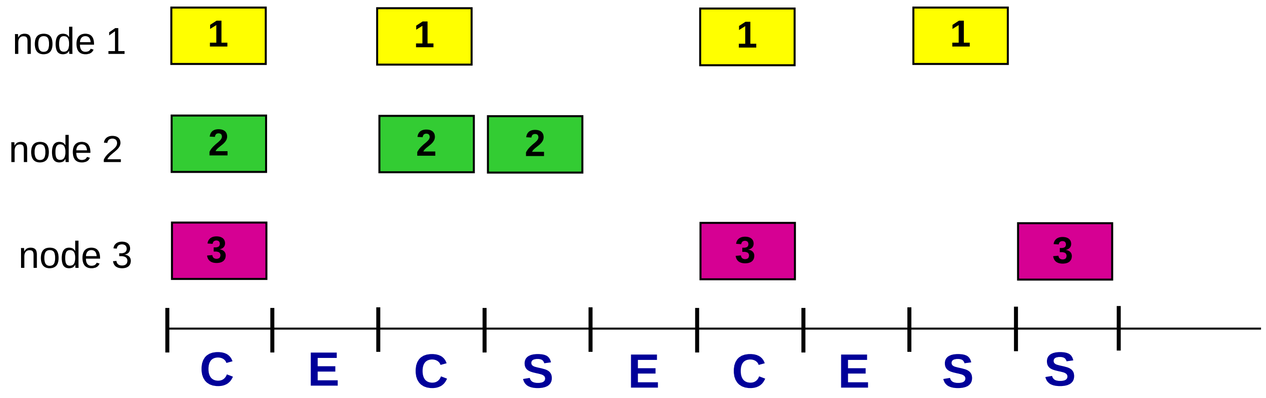
- when node obtains fresh frame, transmits in next slot
  - *if no collision:* node can send new frame in next slot
  - *if collision:* node retransmits frame in each subsequent slot with prob. p until success



A collides with C
Begin
B
End
Begin
A
End
Begin
C
End
t - Tfr
t
t + Tfr
Vulnerable time = Tfr


Collision duration
Collision duration
Station 1
Station 2
Station 3
Station 4
Frame 1.1
Frame 1.2
Frame 2.1
Frame 2.2
Frame 3.1
Frame 3.2
Frame 4.1
Frame 4.2
Time
Time
Time
Time
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Slot 6

Slotted ALOHA



Cons:

- collisions, wasting slots
- idle slots
- nodes may be able to detect collision in less than time to transmit packet
- clock synchronization

Pros:

- single active node can continuously transmit at full rate of channel
- highly decentralized: only slots in nodes need to be in sync
- simple

operation of slotted ALOHA

Let *p* be a probability, that is, a number between 0 and 1.

The operation of slotted ALOHA in each node is simple:

• When the node has a fresh frame to send, it waits until the beginning of the next slot and transmits the entire frame in the slot.

• If there isn't a collision, the node has successfully transmitted its frame and thus need not consider retransmitting the frame. (The node can prepare a new frame for transmission, if it has one.)

• If there is a collision, the node detects the collision before the end of the slot. The node retransmits its frame in each subsequent slot with probability *p* until the frame is transmitted without a collision.

Advantages

Slotted ALOHA would appear to have many advantages. Unlike channel partitioning, slotted ALOHA allows a node to transmit continuously at the full rate, *R*, when that node is the only active node. (A node is said to be active if it has frames to send.) Slotted ALOHA is also highly decentralized, because each node detects collisions and independently decides when to retransmit

Disadvantages

Slotted ALOHA works well when there is only one active node, but how efficient is it when there are multiple active nodes?

There are two possible efficiency concerns here.

- First, when there are multiple active nodes, a certain fraction of the slots will have collisions and will therefore be “wasted.”
- The second concern is that another fraction of the slots will be empty because all active nodes refrain from transmitting as a result of the probabilistic transmission policy. The only “unwasted” slots will be those in which exactly one node transmits. A slot in which exactly one node transmits is said to be a **successful slot**.

The **efficiency** of a slotted multiple access protocol is defined to be the long-run fraction of successful slots in the case when there are a large number of active nodes, each always having a large number of frames to send

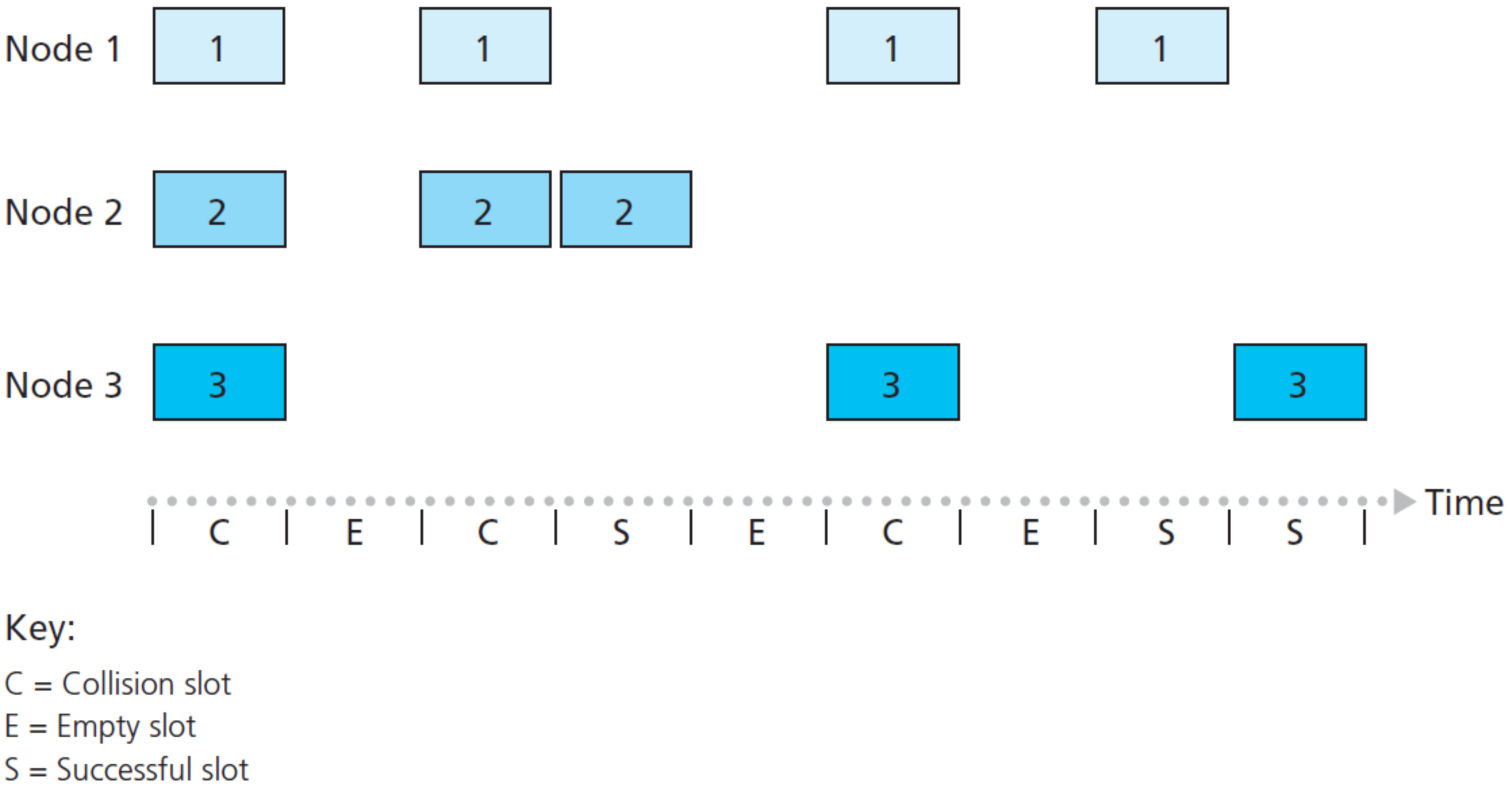


Figure 5.10 ♦ Nodes 1, 2, and 3 collide in the first slot. Node 2 finally succeeds in the fourth slot, node 1 in the eighth slot, and node 3 in the ninth slot

Difference

Pure Aloha	Slotted Aloha
Any station can transmit the data at any time.	Any station can transmit the data at the beginning of any time slot.
The time is continuous and not globally synchronized.	The time is discrete and globally synchronized.
Vulnerable time in which collision may occur $= 2 \times T_{Fr}$	Vulnerable time in which collision may occur $= T_{Fr}$
Probability of successful transmission of data packet= $G \times e^{-2G}$	Probability of successful transmission of data packet= $G \times e^{-G}$
Maximum efficiency = 18.4% (Occurs at G = 1/2)	Maximum efficiency = 36.8% (Occurs at G = 1)

Throughput of Pure ALOHA

Let T be the frame time, i.e. the time required for 1 frame to be transmitted.

Let G be the number of transmission attempts per frame time.

The probability that k frames are generated during the frame time is given by the Poisson distribution−

$$P(k) = \frac{G^k e^{-G}}{k!}$$

From this we can say that the probability that 0 frames are generated (kk = 0) during the frame time is e^{-G}.

In case of pure ALOHA, the vulnerable time period so that collision does not occur between two frames is equal to two frame times, i.e. 2TT. In 2T time,average number of transmission attempts is 2G.

The probability that 0 frames are initiated in the vulnerable time period will be −

$$P(0) = e^{-2G}$$

.

The throughput, S, is calculated as the number of transmission attempts per frame time, G, multiplied by the probability of success, $P(0)$.

$$S = G.P(0)$$

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

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 $[t_0-1,t_0]$

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

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

… choosing optimum p and then letting $n \rightarrow \infty$

$$= 1/(2e) = .18$$

even *worse* than slotted Aloha!

CSMA (carrier sense multiple access)

CSMA: listen before transmit:

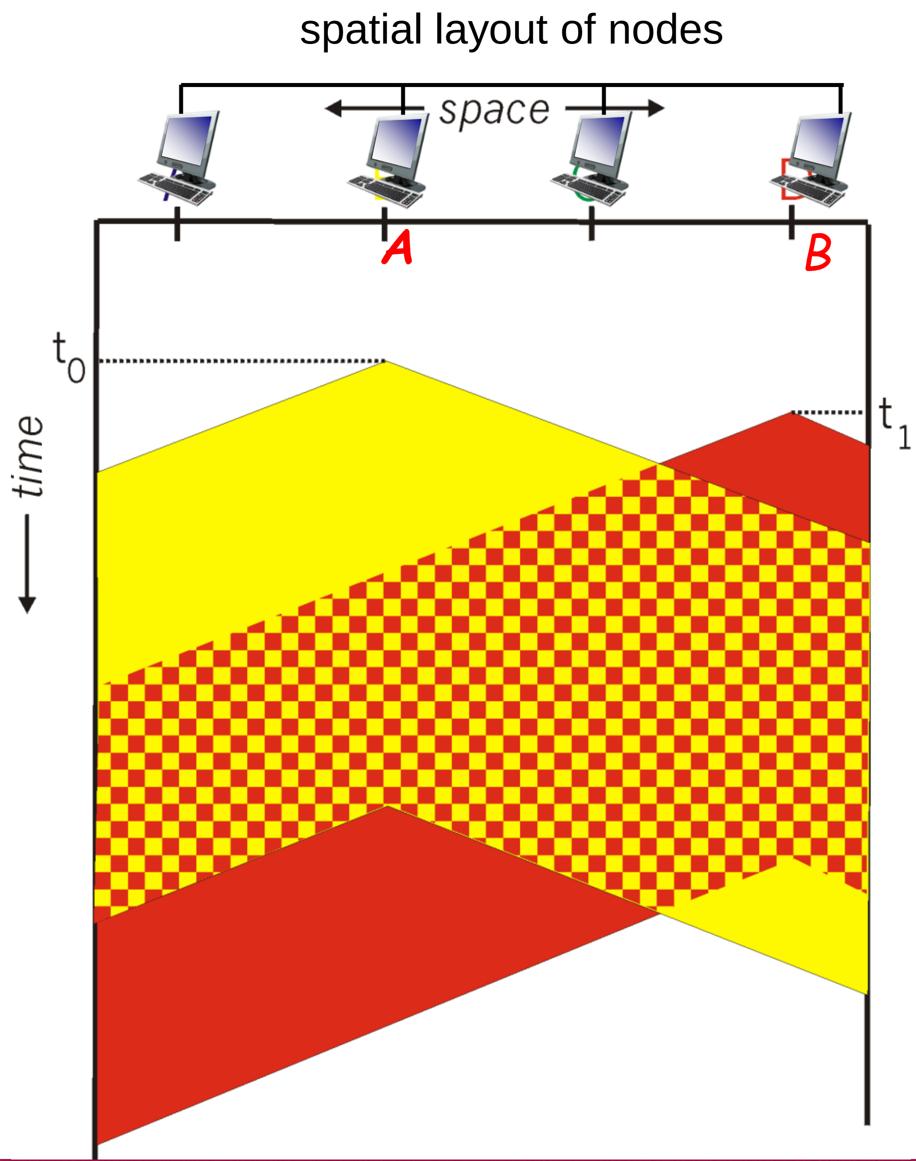
if channel sensed idle: transmit entire frame

- if channel sensed busy, defer transmission

- human analogy: don't interrupt others!

CSMA collisions

- collisions *can* still occur: propagation delay means two nodes may not hear each other's transmission
- collision: entire packet transmission time wasted
  - distance & propagation delay play role in in determining collision probability



CSMA/CD (collision detection)

CSMA/CD: carrier sensing, deferral as in CSMA

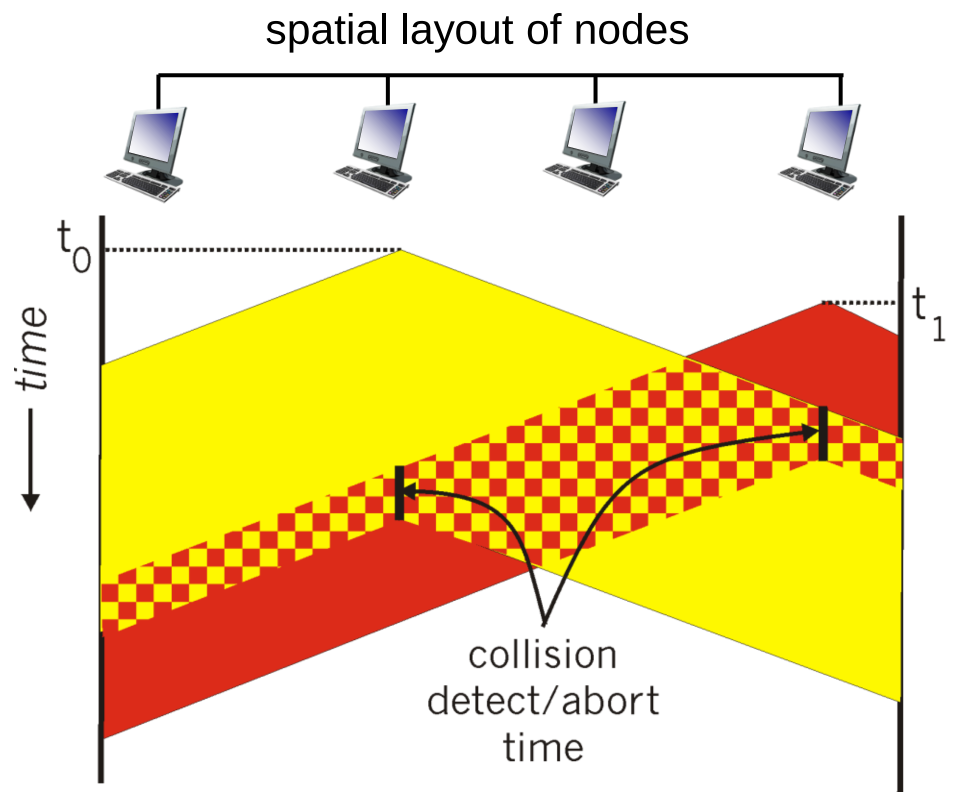
- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage

❖ collision detection:

- easy in wired LANs: measure signal strengths, compare transmitted, received signals
- difficult in wireless LANs: received signal strength overwhelmed by local transmission strength

❖ human analogy: the polite conversationalist

CSMA/CD (collision detection)



Ethernet CSMA/CD algorithm

1. NIC receives datagram from network layer, creates frame
2. If NIC senses channel idle, starts frame transmission. If NIC senses channel busy, waits until channel idle, then transmits.
3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame !
4. If NIC detects another transmission while transmitting, aborts and sends jam signal
5. After aborting, NIC enters *binary (exponential) backoff:*
   - after m^{th} collision, NIC chooses K at random from $\{0,1,2, \ldots, 2^m-1\}$. NIC waits $K \cdot 512$ bit times, returns to Step 2
   - longer backoff interval with more collisions

CSMA/CD efficiency

- T_{prop} = max prop delay between 2 nodes in LAN
- t_{trans} = time to transmit max-size frame

$$efficiency = \frac{1}{1 + 5t_{prop}/t_{trans}}$$

- efficiency goes to 1
  - as t_{prop} goes to 0
  - as t_{trans} goes to infinity
- better performance than ALOHA: and simple, cheap, decentralized!

"Taking turns" MAC protocols

channel partitioning MAC protocols:

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

random access MAC protocols

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

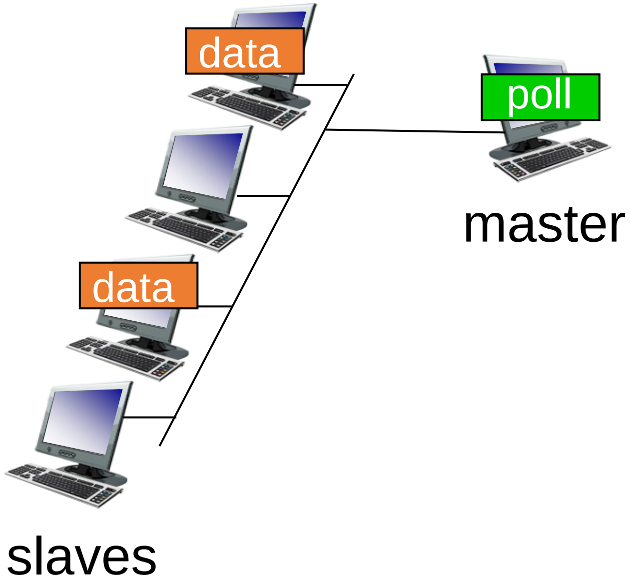
"taking turns" protocols

look for best of both worlds!

“Taking turns” MAC protocols

polling:

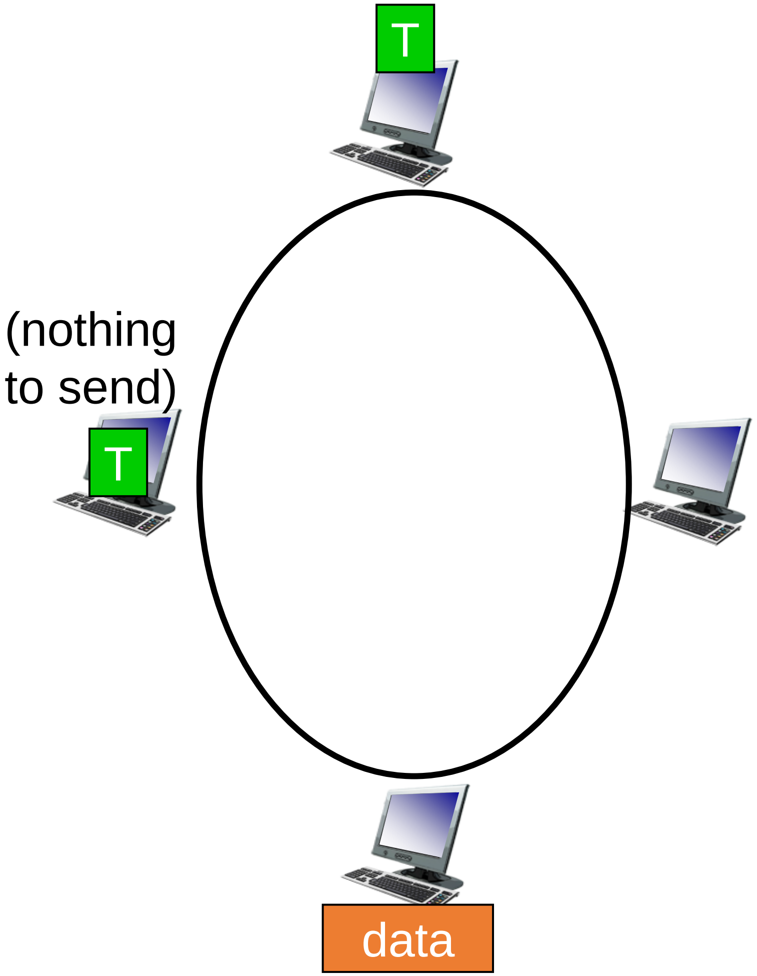
- master node “invites” slave nodes to transmit in turn
- typically used with “dumb” slave devices
- concerns:
  - polling overhead
  - latency
  - single point of failure (master)



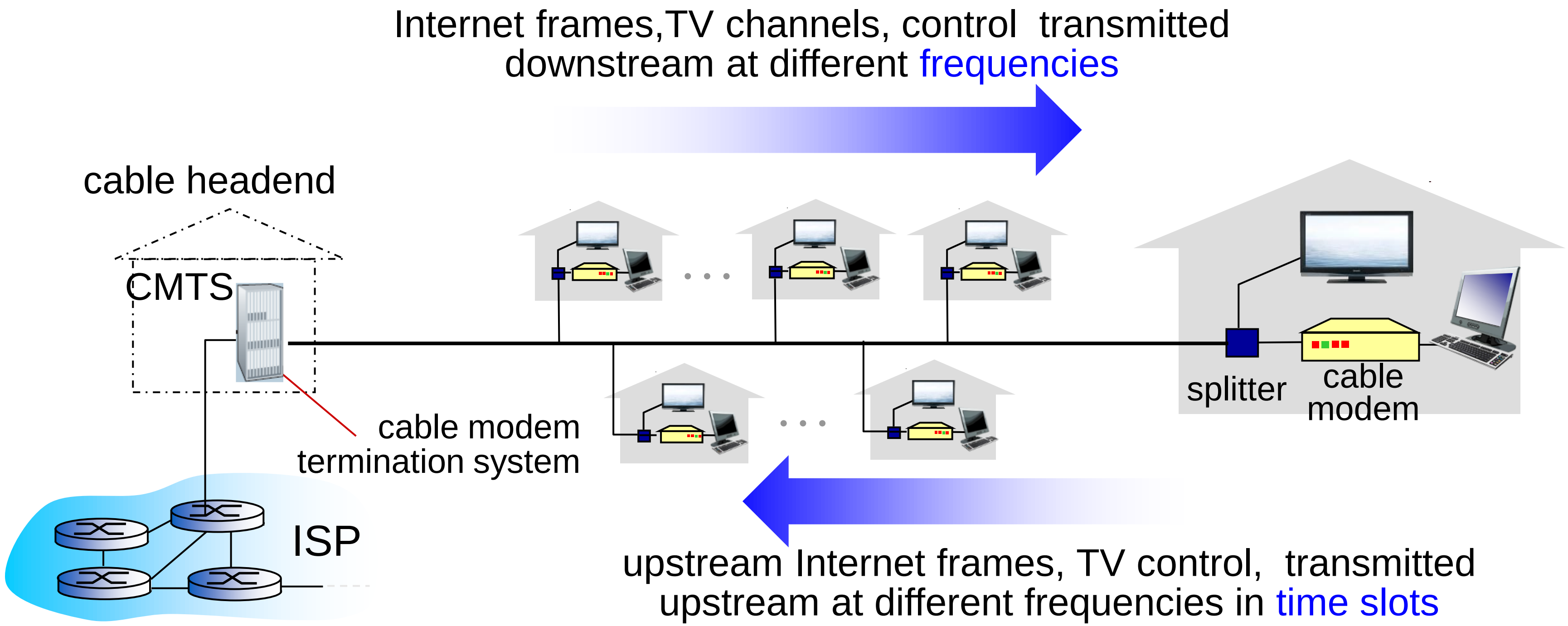
"Taking turns" MAC protocols

token passing:

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

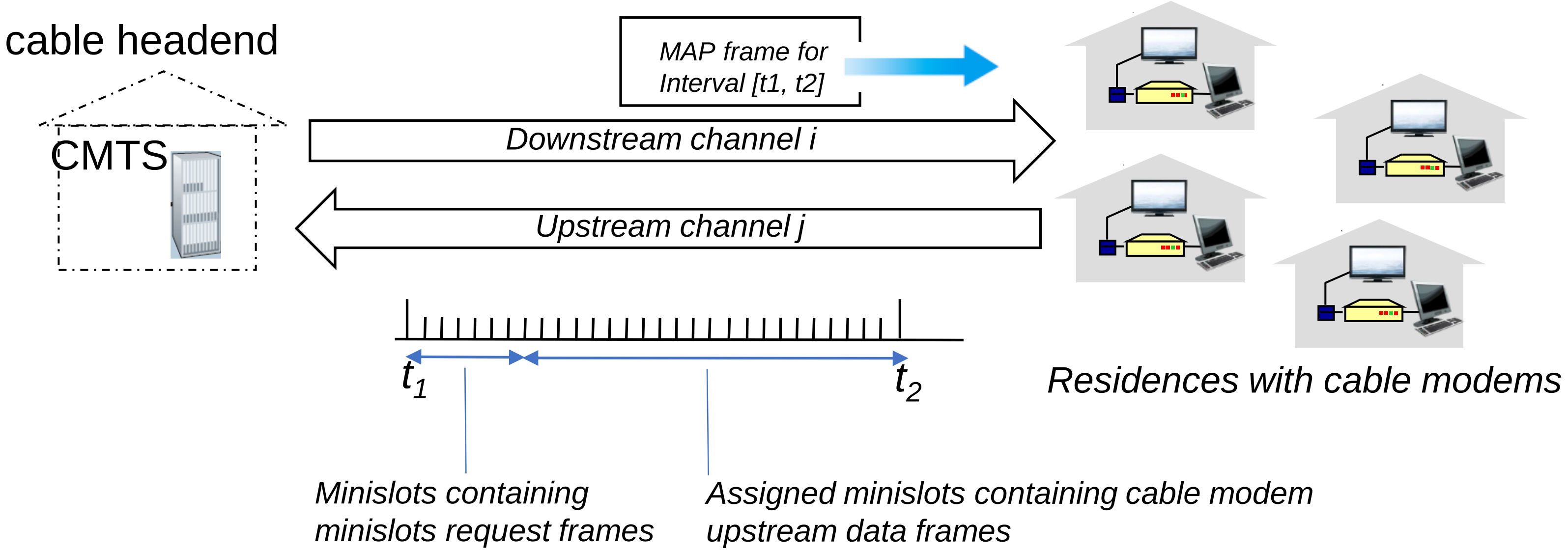


Cable access network



- *multiple* 40Mbps downstream (broadcast) channels
  - single CMTS transmits into channels
- *multiple* 30 Mbps upstream channels
  - *multiple access: all* users contend for certain upstream channel time slots (others assigned)

Cable access network



DOCSIS: data over cable service interface spec

- FDM over upstream, downstream frequency channels
- TDM upstream: some slots assigned, some have contention
  - downstream MAP (bandwidth allocation map) frame: assigns upstream slots
  - request for upstream slots (and data) transmitted random access (binary backoff) in selected slots

Summary of MAC protocols

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

Link layer, LANs: outline

5.1 introduction, services

5.2 error detection, correction

5.3 multiple access protocols

5.4 LANs

- addressing, ARP
- Ethernet
- switches
- VLANS

Namah Shivaya