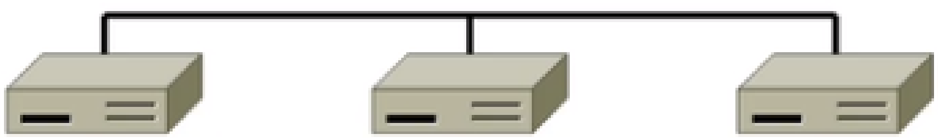
Random Multiple Access

Overview

- How do nodes share a single link?

  Who sends when, e.g., in WiFi?
  - Explore with a simple model



- Assume no one is in charge; this is a distributed system

Overview (2)

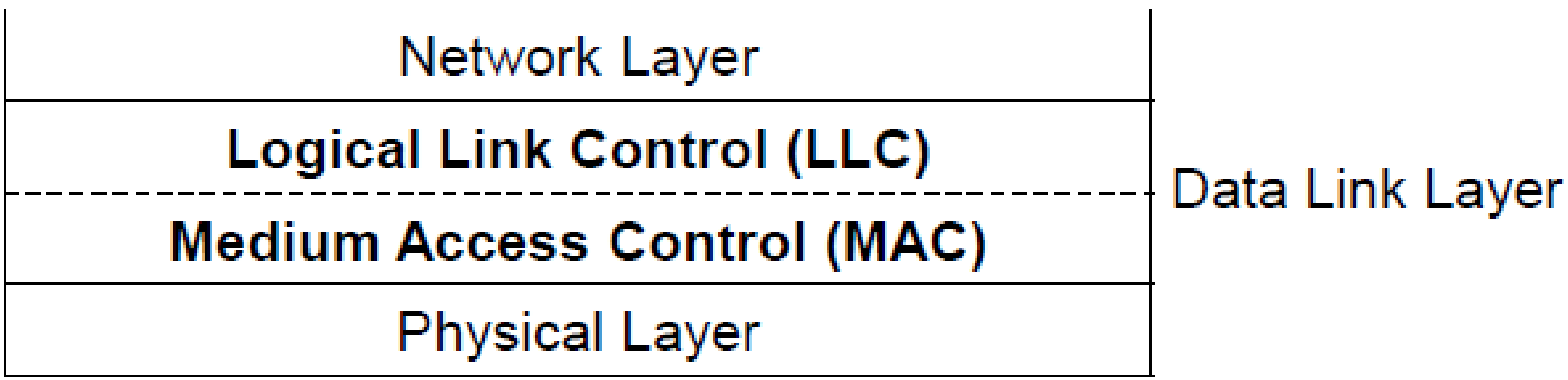
- We will explore multiple access control (MAC) protocols
  - This is the basis for Classic Ethernet
  - Remember: Data traffic is bursty

Medium Access Control Sublayer

- In broadcast networks, several stations share a single communication channel.
- The major issue in these networks is, which station should transmit data at a given time.
- This process of deciding the turn of different stations is known as **Channel Allocation**.
- To coordinate the access to the channel, multiple access protocols are required.
- All these protocols belong to the **MAC sublayer**.

Medium Access Control Sublayer

- Data Link layer is divided into two sublayers:
  - Logical Link Control (LLC)
  - Medium Access Control (MAC)
- LCC is responsible for error control & flow control.
- MAC is responsible for multiple access resolutions.



Channel Allocation Problem

- In broadcast networks, single channel is shared by several stations.
- This channel can be allocated to only one transmitting user at a time.
- There are two different methods of channel allocations:
  - Static Channel Allocation
  - Dynamic Channel Allocation

Static Channel Allocations

- In this method, a single channel is divided among various users either on the basis of frequency or on the basis of time.
- It either uses FDM (Frequency Division Multiplexing) or TDM (Time Division Multiplexing).
- In FDM, fixed frequency is assigned to each user, whereas, in TDM, fixed time slot is assigned to each user.

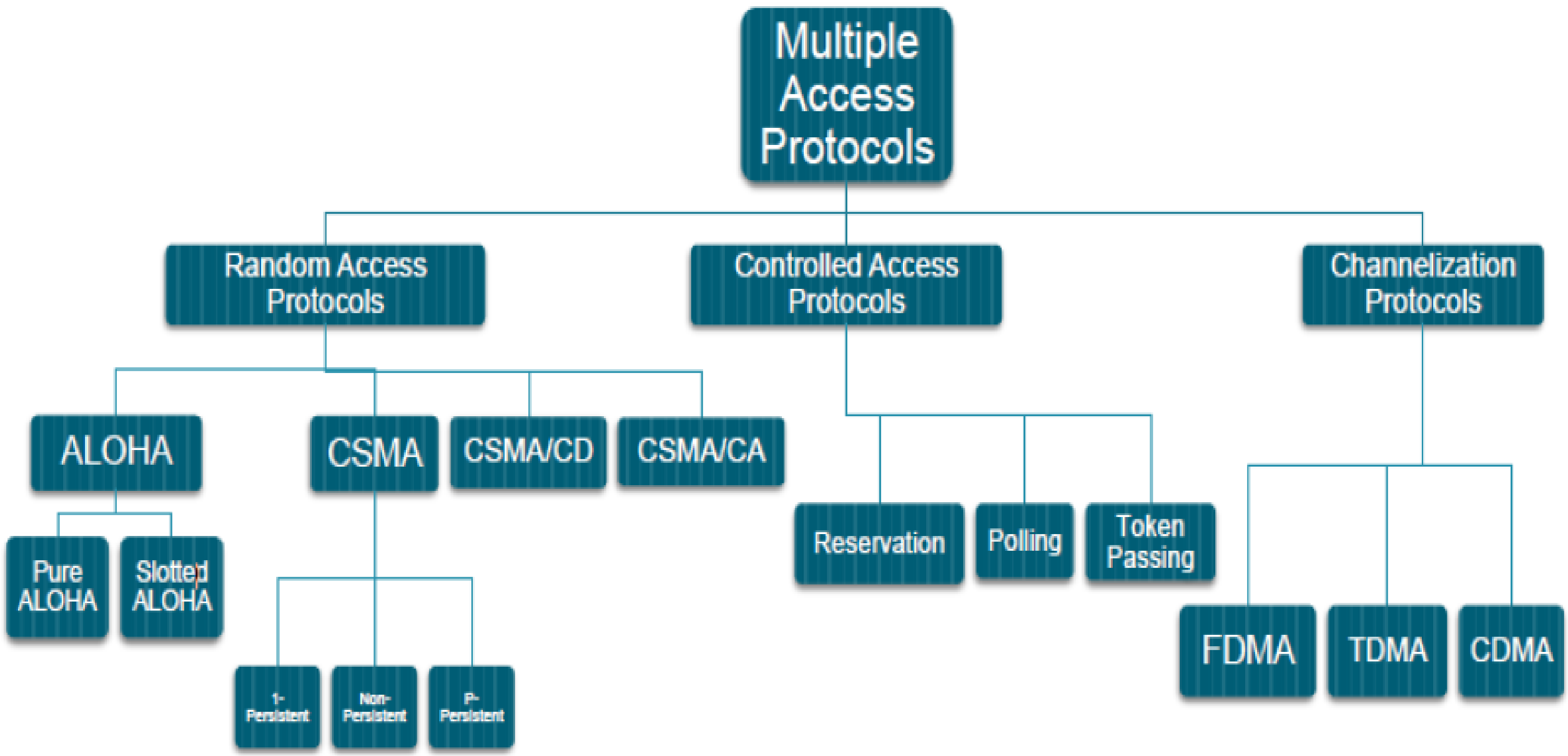
Dynamic Channel Allocation

- In this method, no user is assigned fixed frequency or fixed time slot.
- All users are dynamically assigned frequency or time slot, depending upon the requirements of the user.

Multiple Access Protocols

- Many protocols have been defined to handle the access to shared link.
- These protocols are organized in three different groups.:
  - Random Access Protocols
  - Controlled Access Protocols
  - Channelization Protocols

Multiple Access Protocols



Random Access Protocols

- It is also called **Contention Method**.
- In this method, there is no control station.
- Any station can send the data.
- The station can make a decision on whether or not to send data. This decision depends on the state of the channel, i.e. channel is busy or idle.
- There is no scheduled time for a station to transmit. They can transmit in random order.
- There is no rule that decides which station should send next.
- If two stations transmit at the same time, there is collision and the frames are lost.

Random Access Protocols

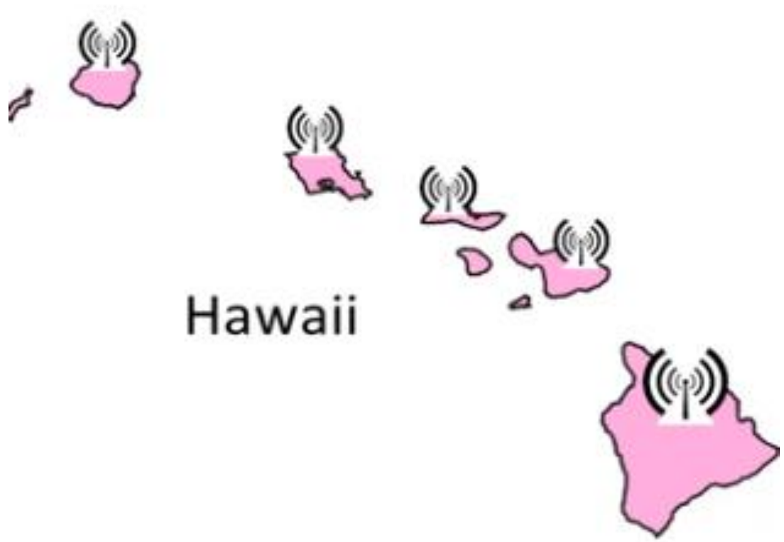
- The various random access methods are:
  - ALOHA
  - CSMA (Carrier Sense Multiple Access)
  - CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
  - CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)

ALOHA

- ALOHA was developed at University of Hawaii in early 1970s by Norman Abramson.
- It was used for ground-based radio broadcasting.
- In this method, stations share a common channel.
- When two stations transmit simultaneously, collision occurs and frames are lost.
- There are two different versions of ALOHA:
  - Pure ALOHA
  - Slotted ALOHA

ALOHA Network

- Seminal computer network connecting the Hawaiian Islands in the late 1960s
  - When should nodes send?
  - A new protocol was devised by Norm Abramson…



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-

Pure ALOHA

Case-01:

- Transmitting station receives an acknowledgement from the receiving station.
- In this case, transmitting station assumes that the transmission is successful.

Case-02:

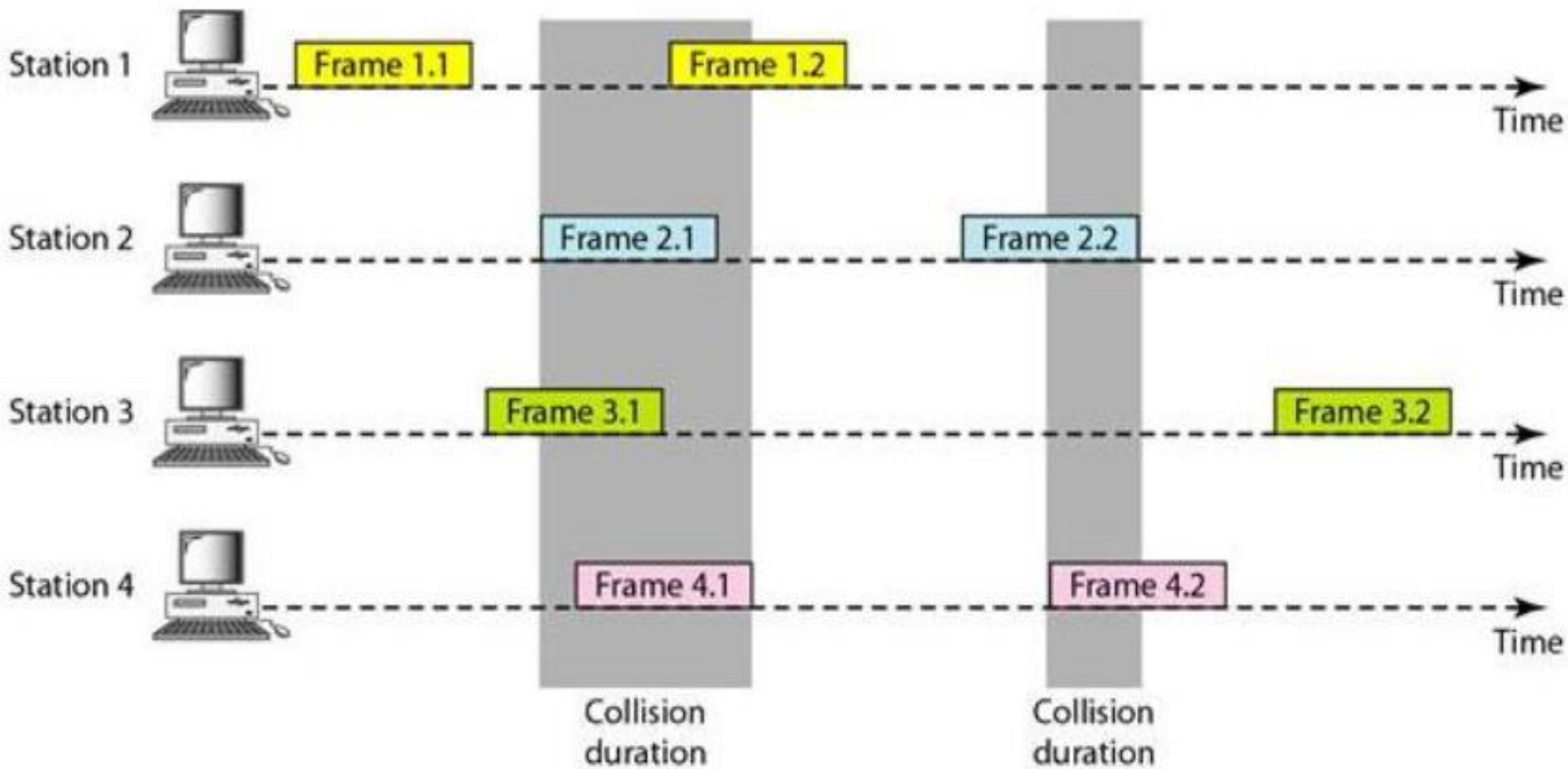
- Transmitting station does not receive any acknowledgement within specified time from the receiving station.
- In this case, transmitting station assumes that the transmission is unsuccessful.

Pure ALOHA

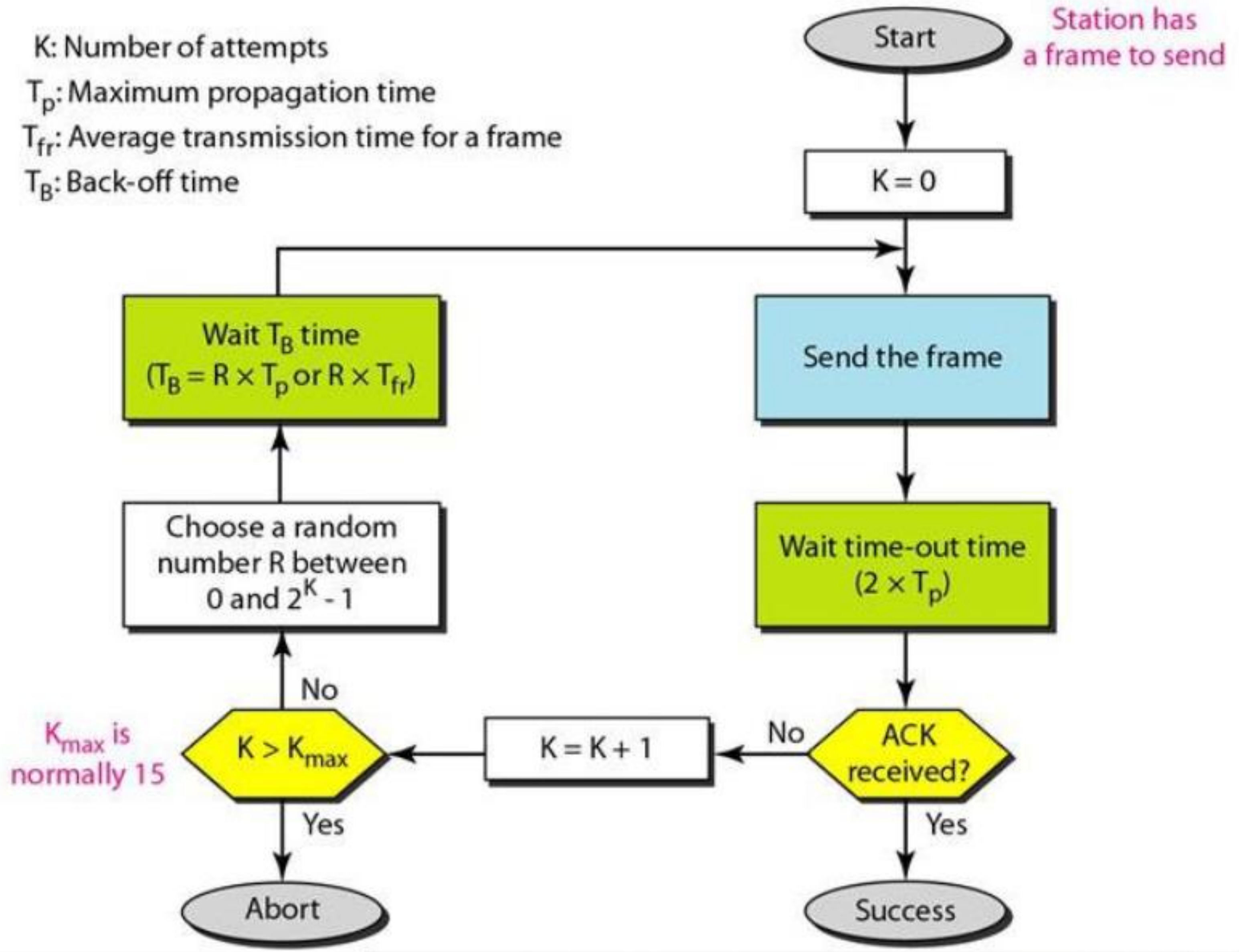
Then,

- 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.

Example of frame collisions in Pure ALOHA



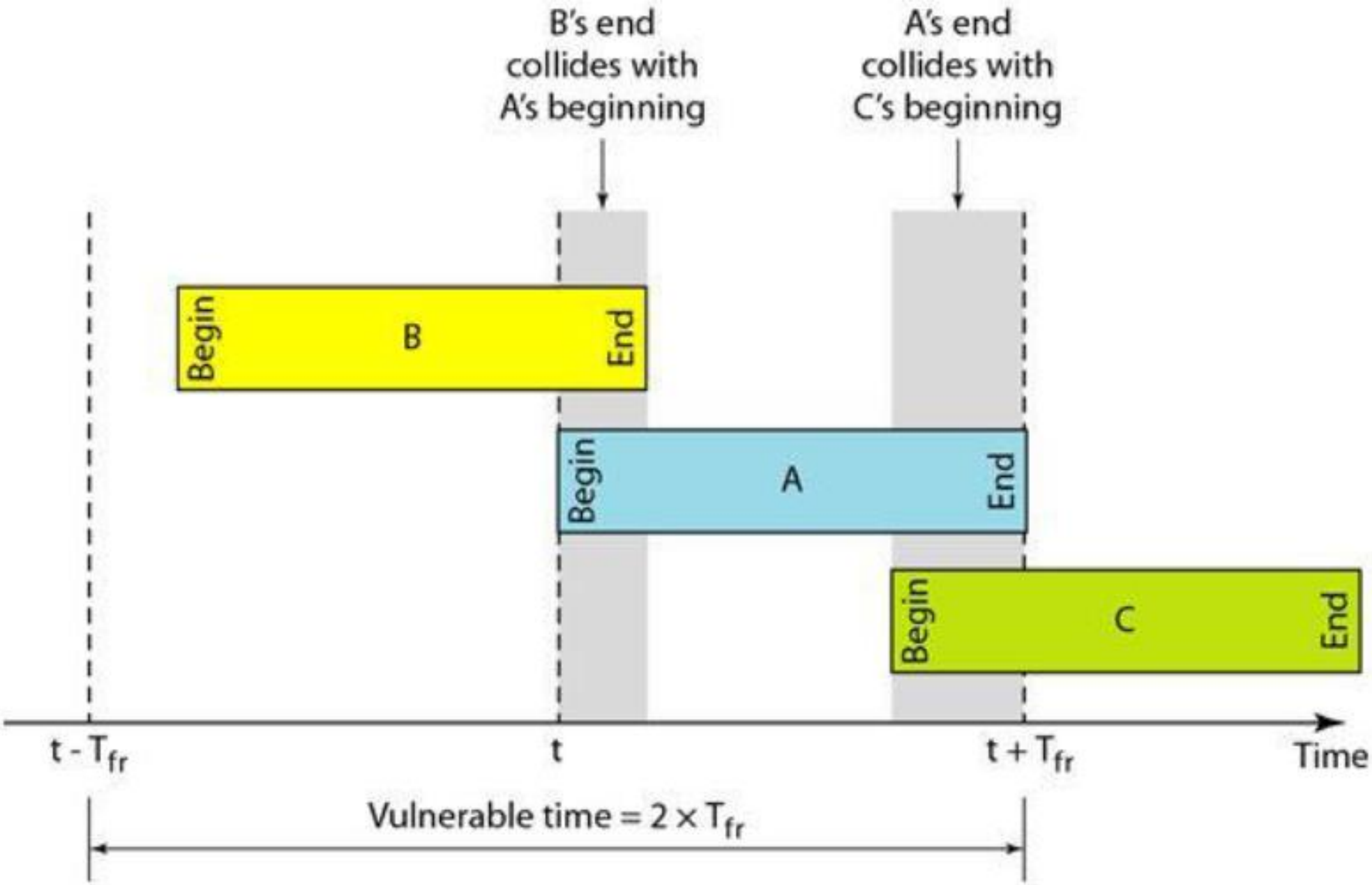
Pure ALOHA


K: Number of attempts
T_p: Maximum propagation time
T_fr: Average transmission time for a frame
T_B: Back-off time
Start
Station has a frame to send
K = 0
Send the frame
Wait time-out time (2 × T_p)
ACK received?
No
Yes
Success
K = K + 1
K > K_max
K_max is normally 15
Yes
Abort
No
Choose a random number R between 0 and 2^K - 1
Wait T_B time (T_B = R × T_p or R × T_fr)

Vulnerable time

- The vulnerable time is in which there is a possibility of collision.
- We assume that the stations send fixed-length frames with each frame taking Tfr S to send. The following figure shows the vulnerable time for station A.

Vulnerable time



Efficiency

- Efficiency of Pure Aloha (η) = $G \times e^{-2G}$

where G = Number of stations willing to transmit data

Maximum Efficiency

- For maximum efficiency,
- We put $d\eta / dG = 0$

  Maximum value of η occurs at $G = 1/2$

  Substituting $G = 1/2$ in the above expression, we get-

  Maximum efficiency of Pure Aloha

  $= 1/2 \times e^{-2 \times 1/2} = 1 / 2e = 0.184 = 18.4\%$

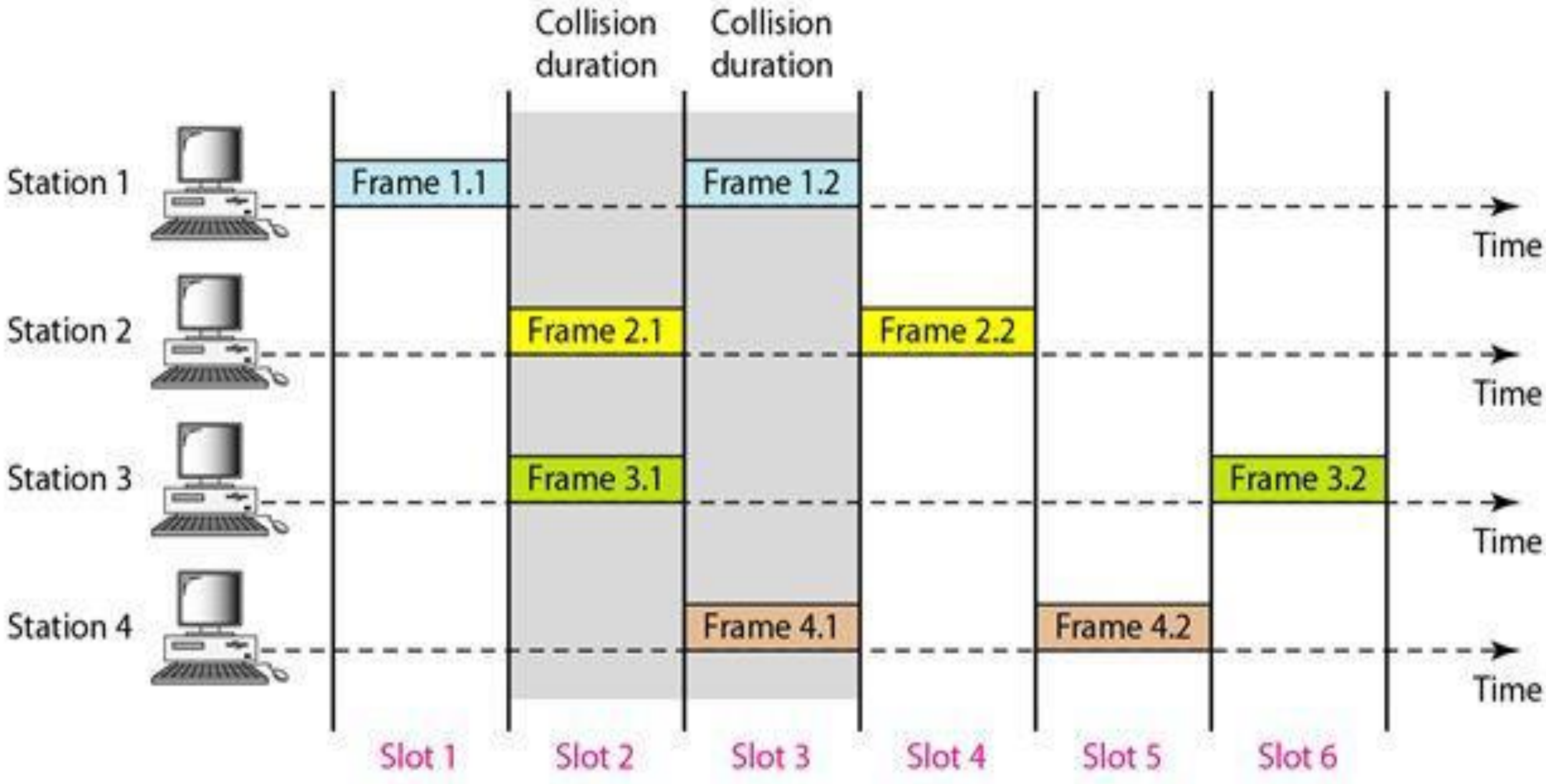
Thus,

- **Maximum Efficiency of Pure Aloha (η) = 18.4%**
- **The maximum efficiency of Pure Aloha is very less due to large number of collisions.**

Slotted ALOHA

- Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- In slotted ALOHA we divide the time into slots of Tfr s and force the station to send only at the beginning of the time slot.

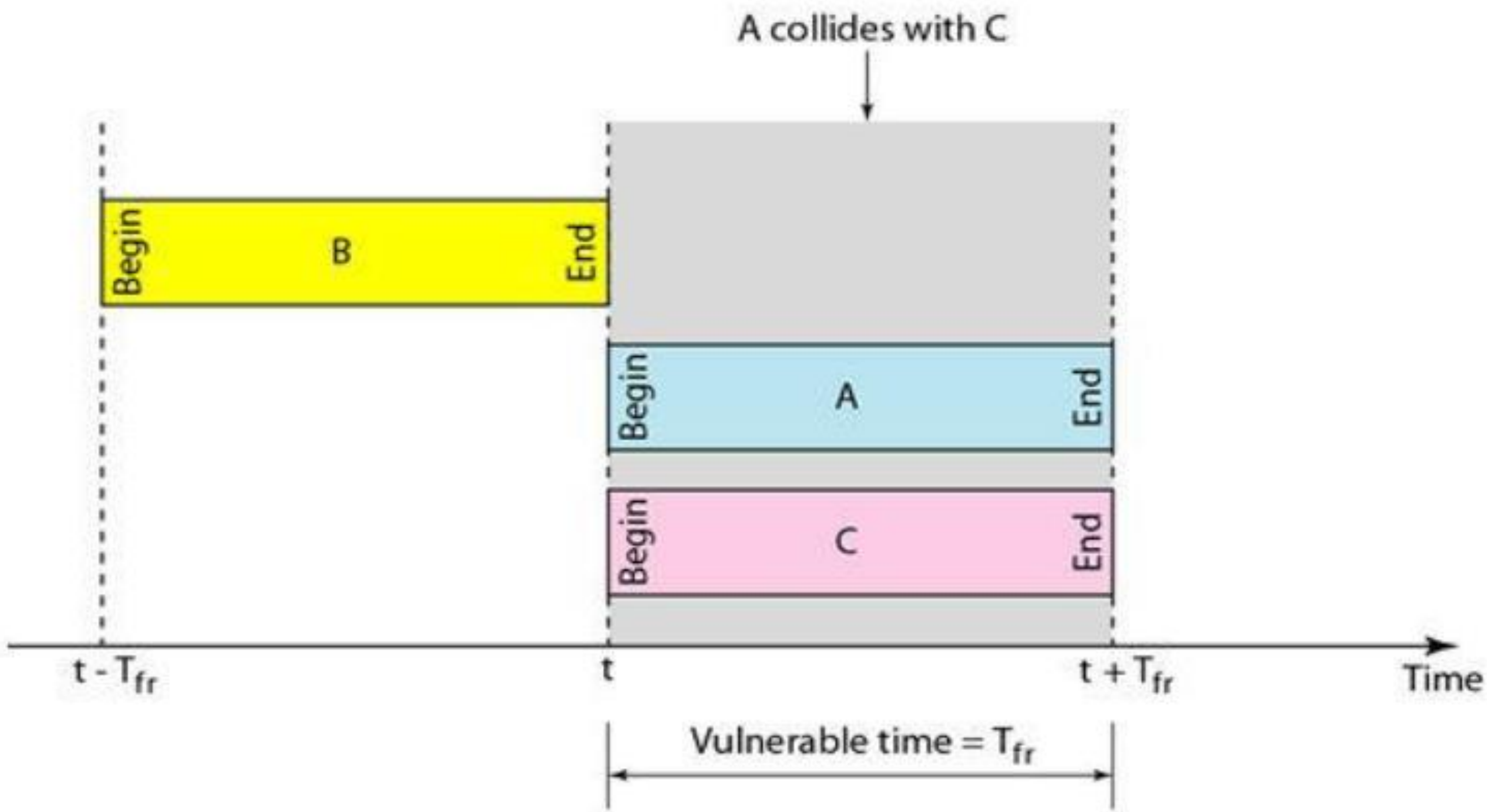
Slotted ALOHA



Vulnerable time

- But, still there is the possibility of collision if two stations try to send at the beginning of the same time slot. However, the vulnerable time is now reduced to one-half, equal to Tfr.

Vulnerable time



Efficiency

- Efficiency of Slotted Aloha (η) = $G \times e^{-G}$
- where G = Number of stations willing to transmit data at the beginning of the same time slot

Maximum Efficiency

- For maximum efficiency,
- We put $d\eta / dG = 0$
- Maximum value of η occurs at $G = 1$
- Substituting $G = 1$ in the above expression, we get-

  Maximum efficiency of Slotted Aloha

  $= 1 \times e^{-1} = 1 / e = 0.368 = 36.8\%$

Thus,

- **Maximum Efficiency of Slotted Aloha (η) = 36.8%**
- **The maximum efficiency of Slotted Aloha is high due to less number of collisions.**

ALOHA Protocol

- Simple, decentralized protocol that works well under low load!
- Not efficient under high load
  - Analysis shows at most 18% efficiency
  - Improvement: divide time into slots and efficiency goes up to 36%
- We'll look at other improvements

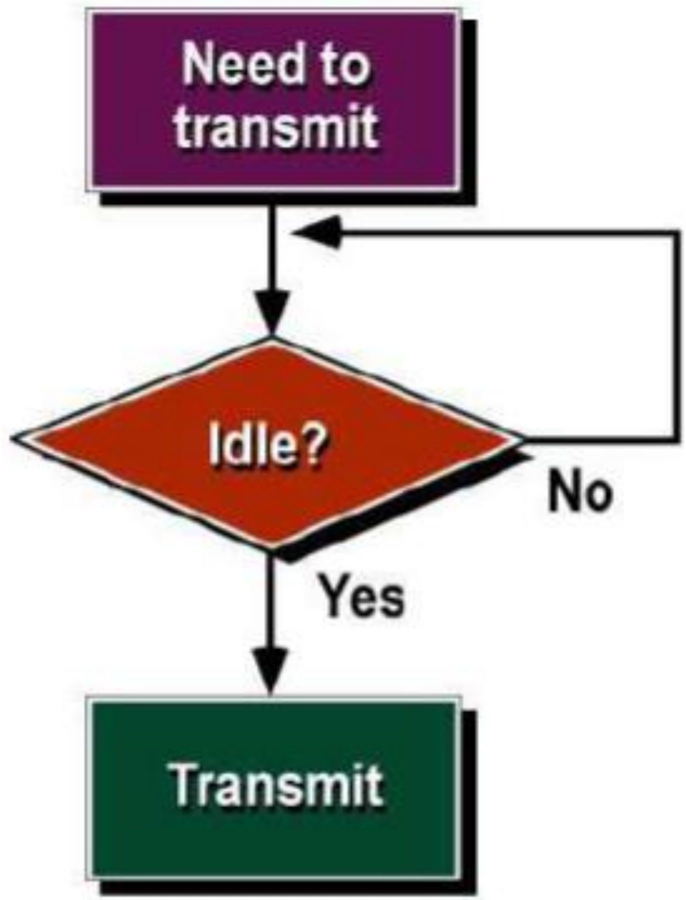
Pure Aloha	Slotted Aloha
Any station can transmit the data at any time.	Any station can transmit the data at the beginning of any time slot.
The time is continuous and not globally synchronized.	The time is discrete and globally synchronized.
Vulnerable time in which collision may occur $= 2 \times T_t$	Vulnerable time in which collision may occur $= T_t$
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)
The main advantage of pure aloha is its simplicity in implementation.	The main advantage of slotted aloha is that it reduces the number of collisions to half and doubles the efficiency of pure aloha.

Carrier Sense Multiple Access (CSMA) Protocol

- CSMA was developed to overcome the problems of ALOHA i.e. to minimize the chances of collision.

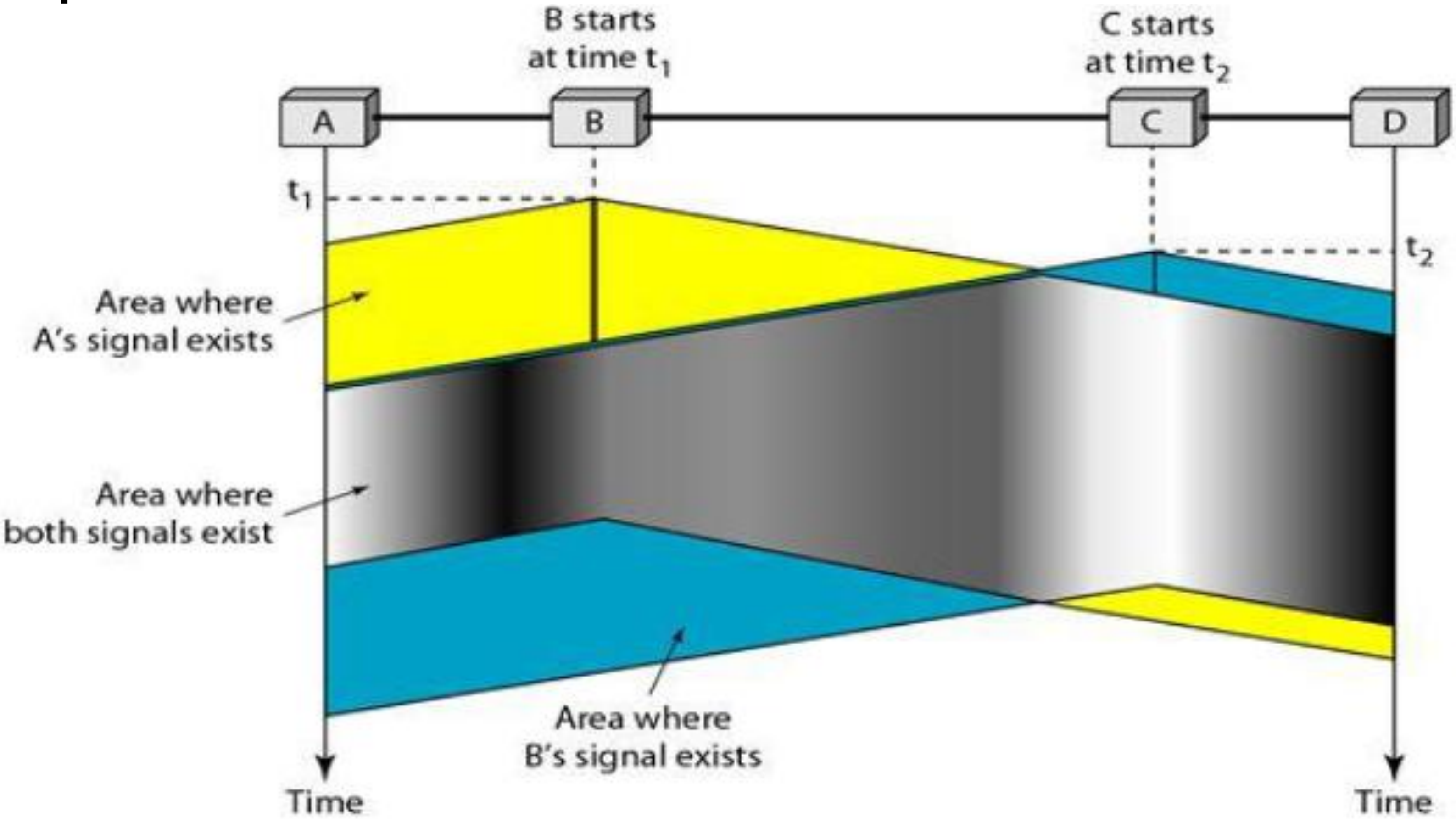
- CSMA is based on the principle of "carrier sense".
- Basic concept: "sense before transmit" or "listen before talk"
- If channel is sensed idle -> transmit the frame
- If channel is sensed busy -> defer the transmission
- Reduces the collisions and increases the efficiency

CSMA



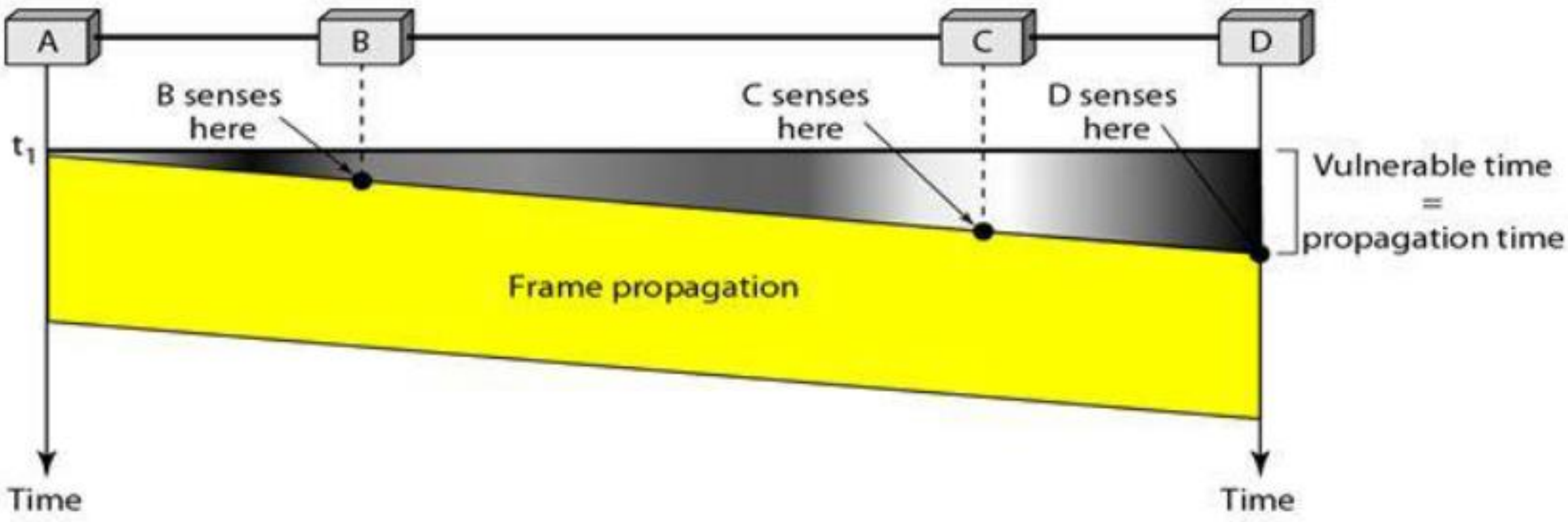
CSMA

- Space and time model of a CSMA network.



The possibility of collision still exists because of propagation delay

Vulnerable Time



The vulnerable time for CSMA is the propagation time T_p

CSMA Protocols

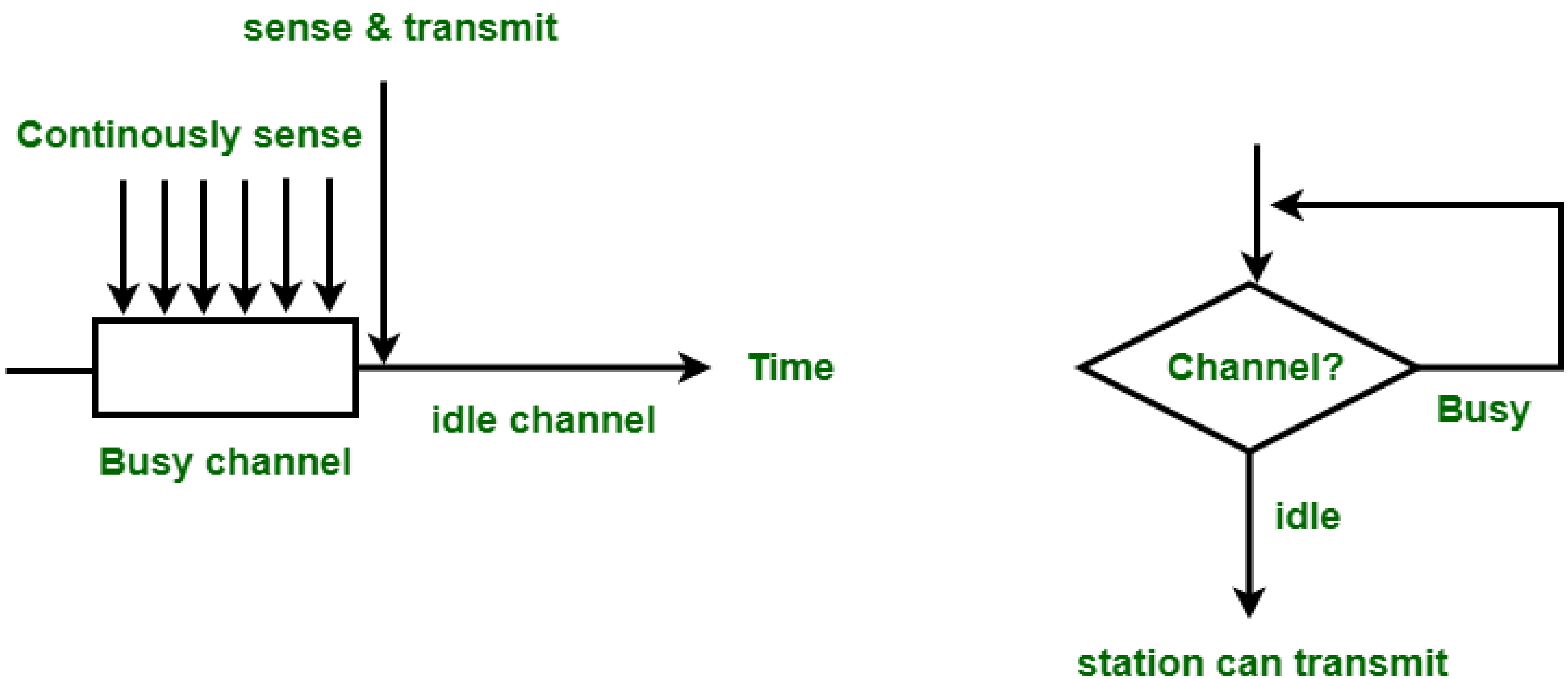
- Answer the following queries:
  - What to do if the channel is busy?
  - What to do if the channel is idle?

- There are three different types of CSMA protocols:
  - 1-Persistent CSMA
  - Non-Persistent CSMA
  - P-Persistent CSMA

1-Persistent CSMA

- Senses the carrier:
  - If it is idle, sends the frame immediately (with the probability 1)
  - If it is busy, station will continuously sense (listen) the carrier until it becomes idle and will send the frame immediately.
- Ethernet use this method.

1-Persistent CSMA



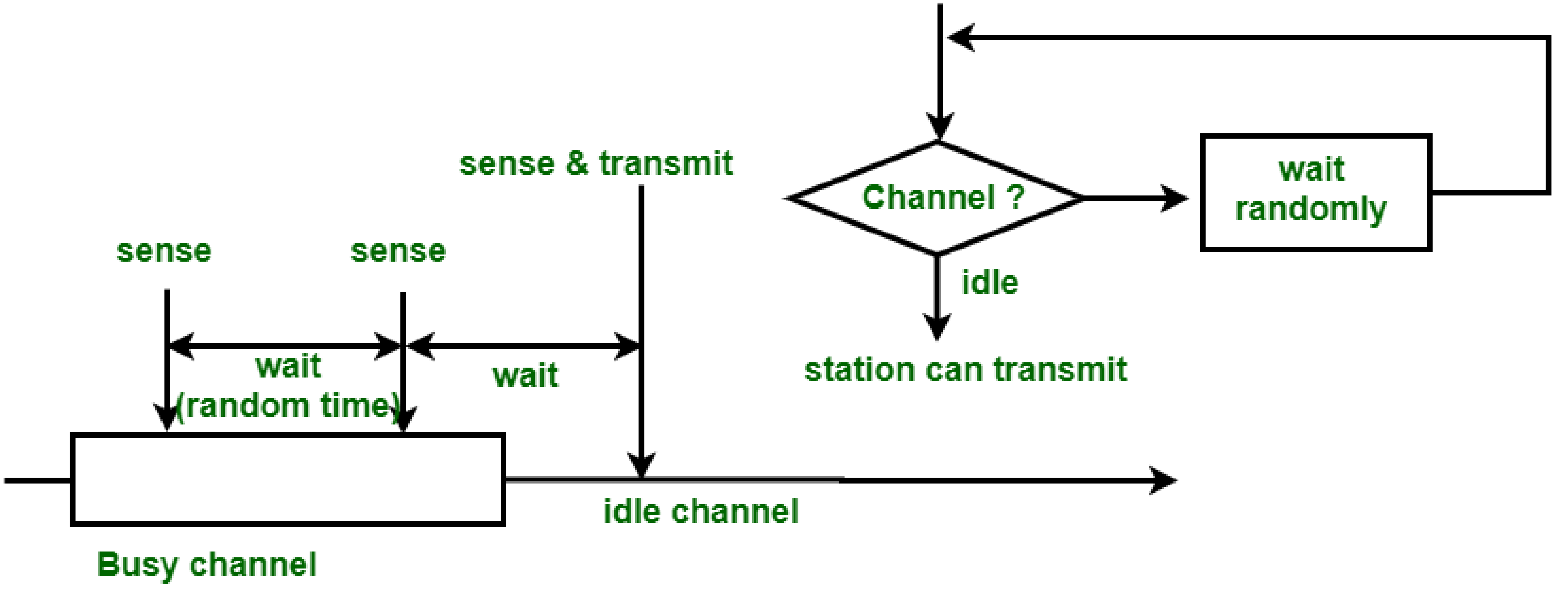
This method has the highest chance of collision because two or more stations may find channel to be idle at the same time and transmit their frames.

Non-Persistent CSMA

- Senses the channel
  - If the channel is idle, it sends the frame immediately.
  - If the channel is busy, it waits a random amount of time and then senses the channel again.

- Reduces the chance of collision because the stations wait for a random amount of time.

Non-Persistent CSMA

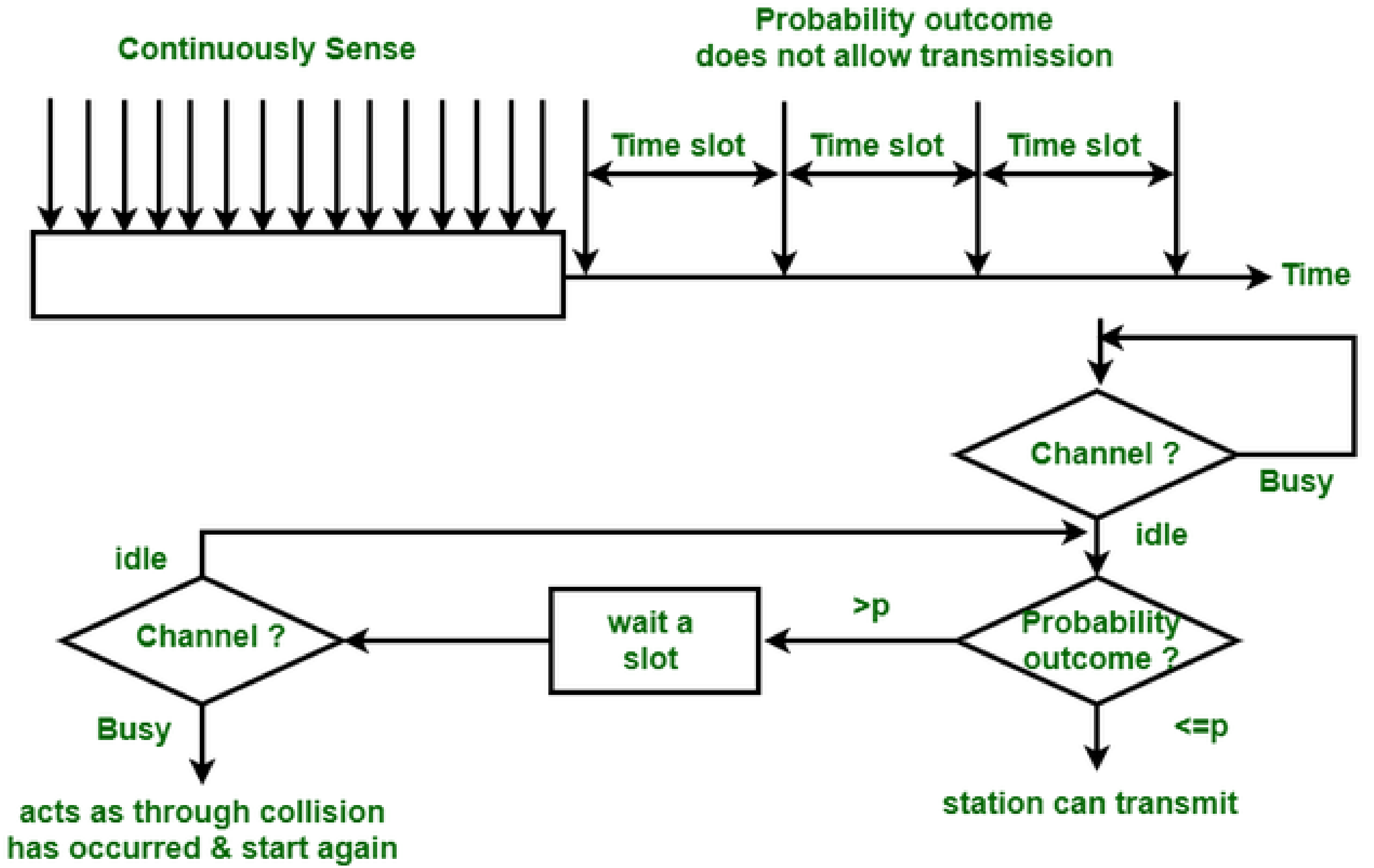


Reduces the efficiency of the network

P-Persistent CSMA

- The p-persistent method is used if the channel has time slots with a slot duration equal to or greater than the maximum propagation time.
- In this method, after the station finds the line idle it follows these steps:
  1. With probability p, the station sends its frame.
  2. With probability q = 1 - p, the station waits for the beginning of the next time slot and checks the line again.
     a) If the line is idle, it goes to step 1.
     b) If the line is busy, it acts as though a collision has occurred and uses the **back off** procedure.

P-Persistent CSMA



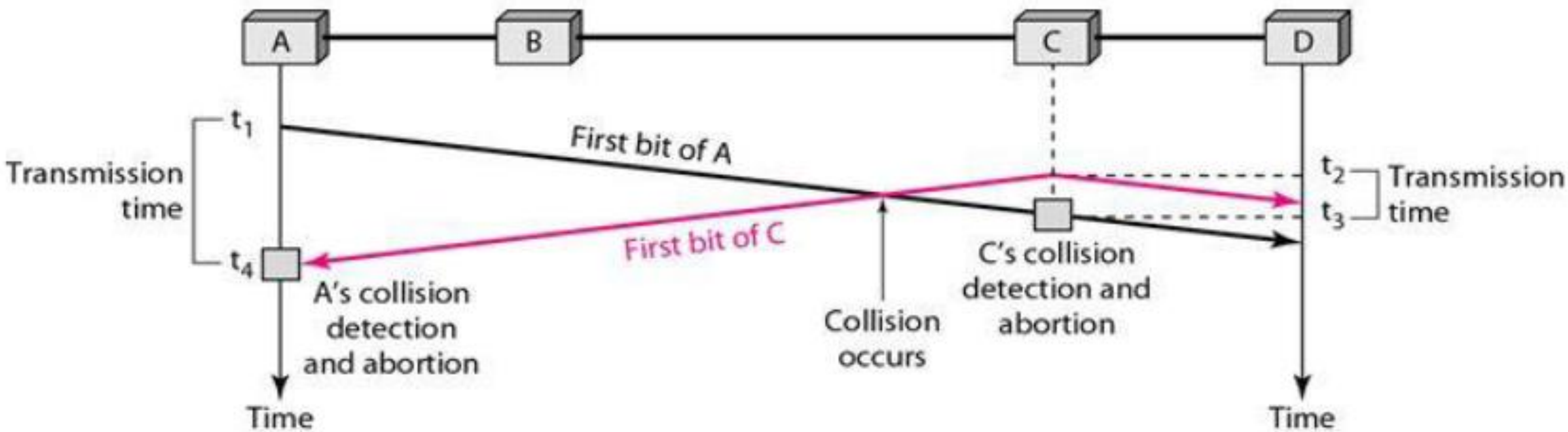
Carrier Sense Multiple Access with Collision Detection (CSMA/CD)

- The CSMA method does not specify the procedure following a collision.
- CSMA/CD augments the algorithm to handle the collision.

CSMA/CD

- Station monitors the medium after it sends a frame to see if the transmission was successful.
- If, however, there is a collision, the frame is sent again.
- This scheme is known as Carrier Sense Multiple Access with Collision Detection (CSMA/CD) or Listen-while-talk

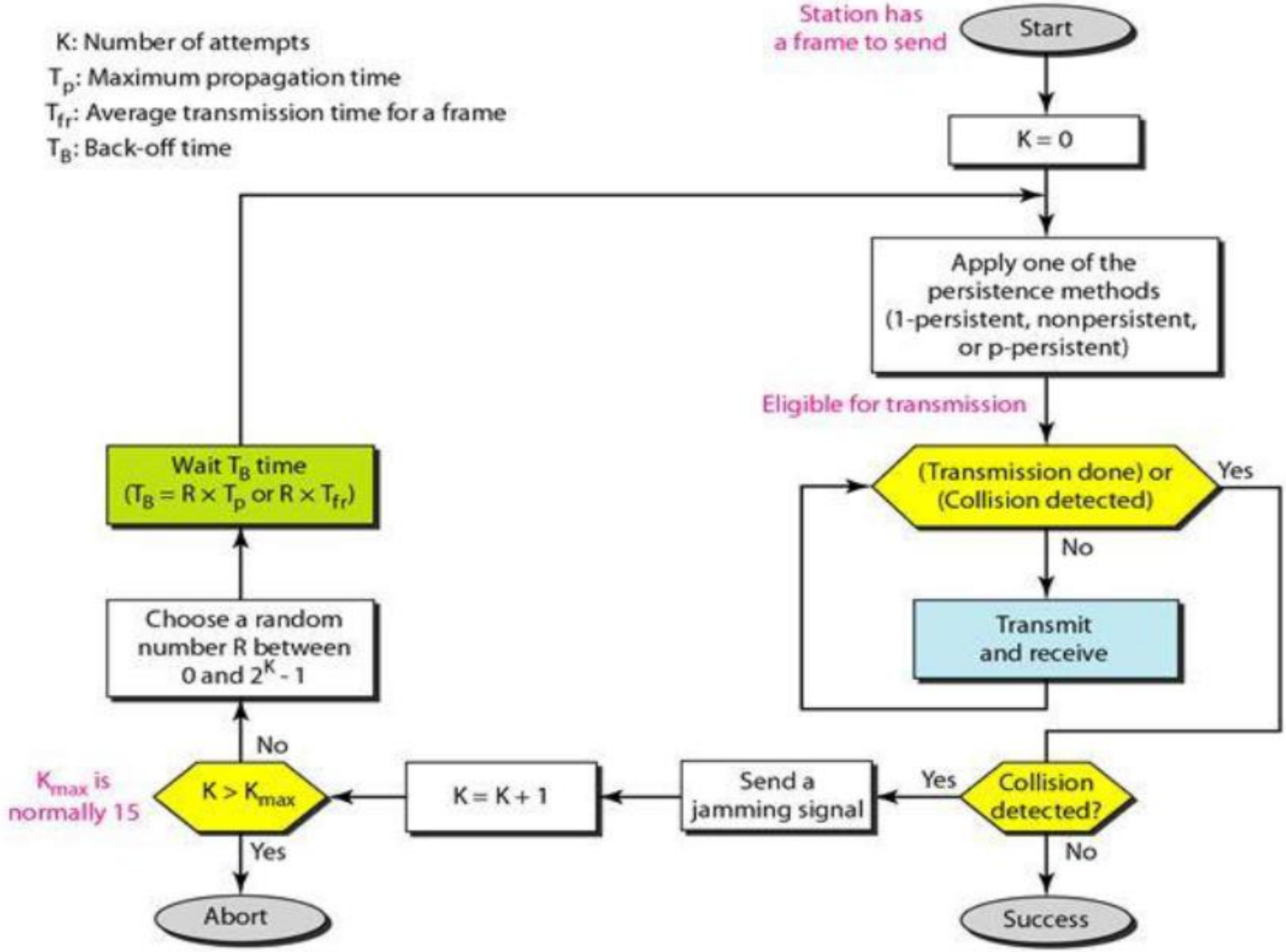
CSMA/CD



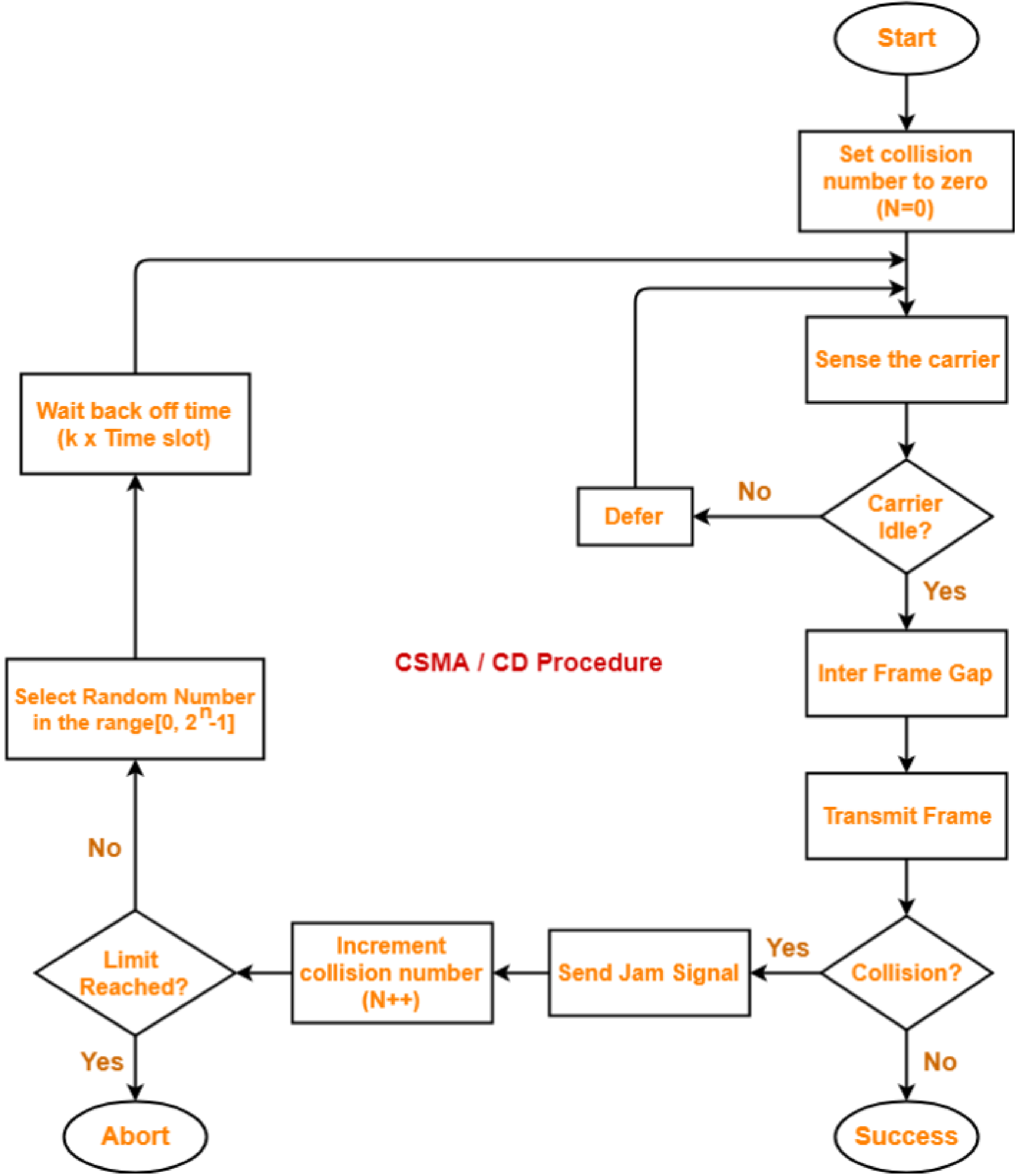
A
B
C
D
Transmission time
t1
First bit of A
t2
Transmission time
t3
First bit of C
t4
A's collision detection and abortion
Collision occurs
C's collision detection and abortion
Time
Time

Minimum Frame Size

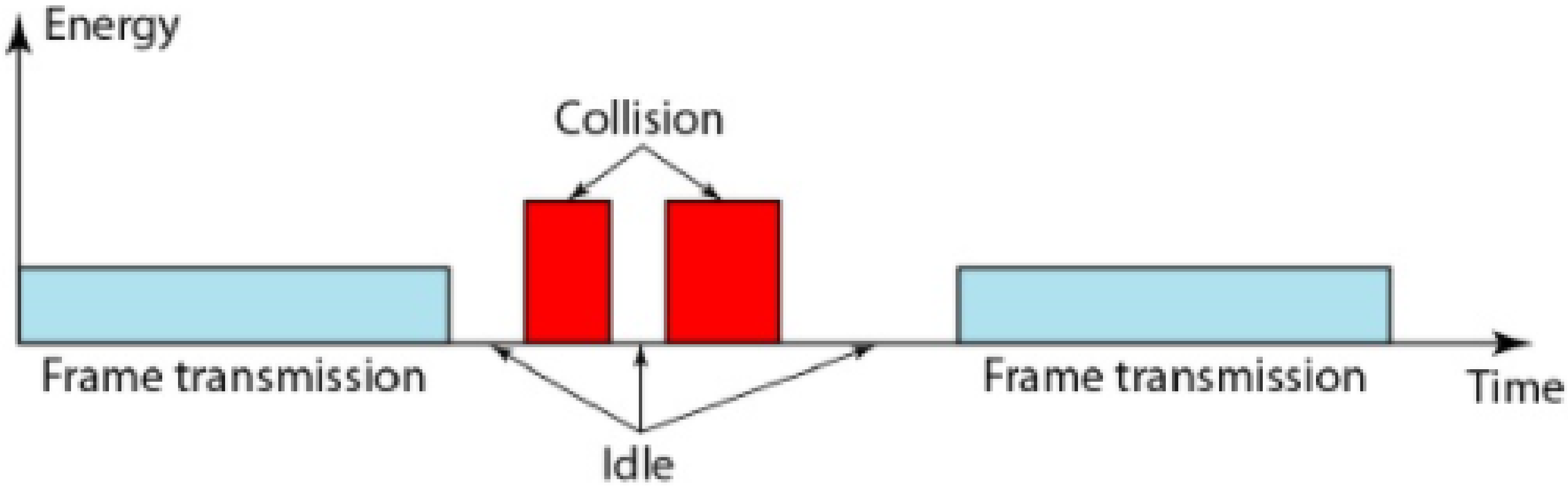
- Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission.
- This is so because the station, once the entire frame is sent, does not keep a copy of the frame and does not monitor the line for collision detection.
- The frame transmission time T_{fr} must be at least two times the maximum propagation time T_p.



Flow diagram for CSMA/CD


Start
Set collision number to zero (N=0)
Sense the carrier
Carrier Idle?
No
Defer
Yes
Inter Frame Gap
Transmit Frame
Collision?
Yes
Send Jam Signal
Increment collision number (N++)
Limit Reached?
No
Select Random Number in the range[0, 2^n-1]
Wait back off time (k x Time slot)
Yes
Abort
No
Success
CSMA / CD Procedure

Energy Level



Energy level during transmission, idleness, or collision

Throughput

- The throughput of CSMA/CD is greater than that of pure or slotted ALOHA.
- The maximum throughput occurs at a different value of G and is based on the persistence method and the value of p in the p-persistent approach.
- For 1-persistent method the maximum throughput is around 50 percent when G =1.
- For nonpersistent method, the maximum throughput can go up to 90 percent when G is between 3 and 8.

Binary Exponential BackOff Algorithm

- Back Off Time-

In CSMA / CD protocol,

- After the occurrence of collision, station waits for some random back off time and then retransmits.
- This waiting time for which the station waits before retransmitting the data is called as **back off time**.
- **Back Off Algorithm** is used for calculating the back off time.

Back Off Algorithm-

After undergoing the collision,

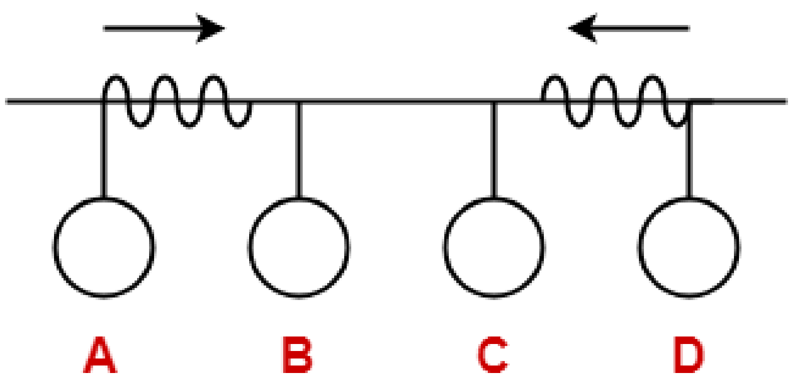
- Transmitting station chooses a random number in the range $[0, 2^n-1]$ if the packet is undergoing collision for the n^{th} time.
- If station chooses a number k, then-

Back off time = k x Time slot

where value of one time slot = 1 RTT

Example

- Consider the following scenario where stations A and D start transmitting their data simultaneously-



- For simplicity,
  - We consider the value of time slot = 1 unit.
  - Thus, back off time = K units.

Example

- Scene-01: For 1st Data Packet Of Both Stations-
  - Both the stations start transmitting their 1st data packet simultaneously.
  - This leads to a collision.
  - Clearly, the collision on both the packets is occurring for the 1st time.
  - So, collision number for the 1st data packet of both the stations = 1.

Example

- At Station A-

After detecting the collision,

- Station A randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station A chooses the number K_A, then back off time = K_A units.

- At Station D-

After detecting the collision,

- Station D randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station D chooses the number K_D, then back off time = K_D units.

Example

Following 4 cases are possible-

K_A	K_D	Remarks
0	0	• In this case, both the stations start retransmitting their data immediately. • This case leads to a collision again.
0	1	• In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. • This case leads to A successfully retransmitting its data after the 1st collision.
1	0	• In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. • This case leads to D successfully retransmitting its data after the 1st collision.
1	1	• In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. • This case leads to a collision again.

Example

From here,

- Probability of station A to successfully retransmit its data after the 1st collision = 1 / 4
- Probability of station D to successfully retransmit its data after the 1st collision = 1 / 4
- Probability of occurrence of collision again after the 1st collision = 2 / 4 = 1 / 2

Example

Consider case-02 occurs.

- This causes station A to successfully retransmit its 1st packet after the 1st collision.
- Scene-02: For 2nd Data Packet Of Station A And 1st Data Packet Of Station D-
- Consider after some time,
  - Station A starts transmitting its 2nd data packet and station D starts retransmitting its 1st data packet simultaneously.
  - This leads to a collision.

Example

At Station A-

- The 2nd data packet of station A undergoes collision for the 1st time.
- So, collision number for the 2nd data packet of station A = 1.
- Now, station A randomly chooses a number in the range $[0, 2^1-1] = [0,1]$.
- If station A chooses the number K_A, then back off time = K_A units.

Example

At Station D-

- The 1st data packet of station D undergoes collision for the 2nd time.
- So, collision number for the 1st data packet of station D = 2.
- Now, station D randomly chooses a number in the range $[0, 2^2-1] = [0,3]$.
- If station D chooses the number K_D, then back off time = K_D units.

Example

Following 8 cases are possible-

K_A	K_D	Remarks
0	0	• In this case, both the stations start retransmitting their data immediately. • This case leads to a collision again.
0	1	• In this case, station A starts retransmitting its data immediately while station D waits for 1 unit of time. • This case leads to A successfully retransmitting its data after the 2nd collision.
0	2	• In this case, station A starts retransmitting its data immediately while station D waits for 2 unit of time. • This case leads to A successfully retransmitting its data after the 2nd collision.
0	3	• In this case, station A starts retransmitting its data immediately while station D waits for 3 unit of time. • This case leads to A successfully retransmitting its data after the 2nd collision.

Example

1	0	• In this case, station A waits for 1 unit of time while station D starts retransmitting its data immediately. • This case leads to D successfully retransmitting its data after the 2^{nd} collision.
1	1	• In this case, both the stations wait for 1 unit of time and then starts retransmitting their data simultaneously. • This case leads to a collision again.
1	2	• In this case, station A waits for 1 unit of time while station D waits for 2 unit of time. • This case leads to A successfully retransmitting its data after the 2^{nd} collision.
1	3	• In this case, station A waits for 1 unit of time while station D waits for 3 unit of time. • This case leads to A successfully retransmitting its data after the 2^{nd} collision.

Example

From here,

- Probability of station A to successfully retransmit its data after the 2nd collision = 5 / 8
- Probability of station D to successfully retransmit its data after the 2nd collision = 1 / 8
- Probability of occurrence of collision again after the 2nd collision = 2 / 8 = 1 / 4

In the similar manner, the procedure continues.

Back Off Algorithm

With each successive collision-

- Back off time increases exponentially.
- Collision probability decreases exponentially.

Back Off Algorithm is also known **as Binary Exponential Back Off Algorithm** because-

- It works for only two stations.
- The back off time increases exponentially.
- Collision probability decreases exponentially.

Back Off Algorithm

- One disadvantage of Back Off Algorithm is that it shows capture effect.
- It means if a particular station wins the collision one time, then its probability of winning the successive collisions increases exponentially.