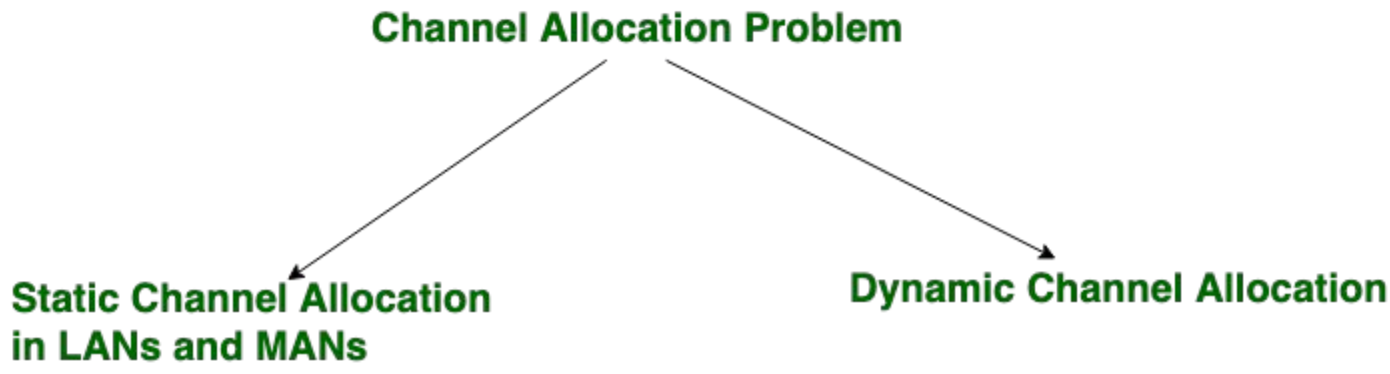
Channel Allocation Problem in Computer Network

- **Channel allocation** is a process in which a single channel is divided and allotted to multiple users in order to carry user specific tasks. There are user’s quantity may vary every time the process takes place. If there are N number of users and channel is divided into N equal-sized sub channels, Each user is assigned one portion. If the number of users are small and don’t vary at times, then Frequency Division Multiplexing can be used as it is a simple and efficient channel bandwidth allocating technique.

Channel allocation problem can be solved by two schemes: Static Channel Allocation in LANs and MANs, and Dynamic Channel Allocation.



These are explained as following below.

1. **Static Channel Allocation in LANs and MANs:** It is the classical or traditional approach of allocating a single channel among multiple competing users using Frequency Division Multiplexing (FDM). if there are N users, the frequency channel is divided into N equal sized portions (bandwidth), each user being assigned one portion. since each user has a private frequency band, there is no interference between users.

   However, it is not suitable in case of a large number of users with variable bandwidth requirements. It is not efficient to divide into fixed number of chunks.

T = 1/(U*C-L) **T(FDM)** = N*T(1/U(C/N)-L/N) Where,

T = mean time delay,

C = capacity of channel,

L = arrival rate of frames,

1/U = bits/frame,

N = number of sub channels,

T(FDM) = Frequency Division Multiplexing Time

2. Dynamic Channel Allocation:

In dynamic channel allocation scheme, frequency bands are not permanently assigned to the users. Instead channels are allotted to users dynamically as needed, from a central pool. The allocation is done considering a number of parameters so that transmission interference is minimized.

This allocation scheme optimises bandwidth usage and results is faster transmissions.

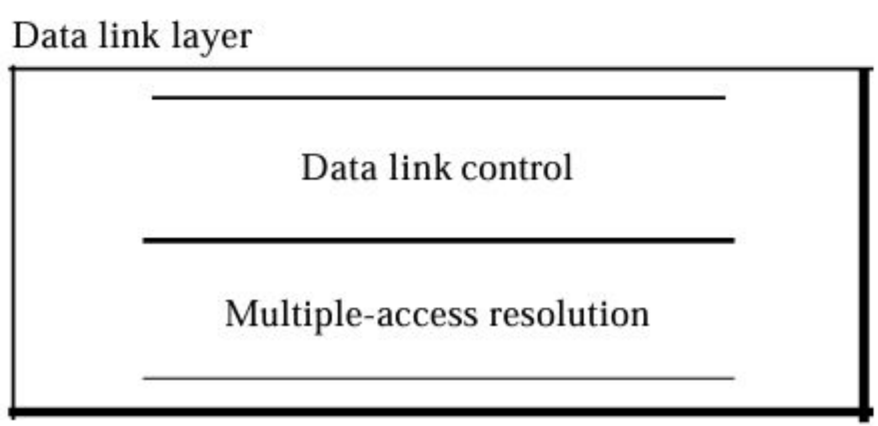
Dynamic channel allocation is further divided into:

- **Centralised Allocation**
- **Distributed Allocation**

Multiple Access

- We can consider the data link layer as two sub layers. The upper sub layer is responsible for data link control, and the lower sub layer is responsible for resolving access to the shared media. If the channel is dedicated, we do not need the lower sub layer. Figure 12.1 shows these two sub layers in the data link layer.

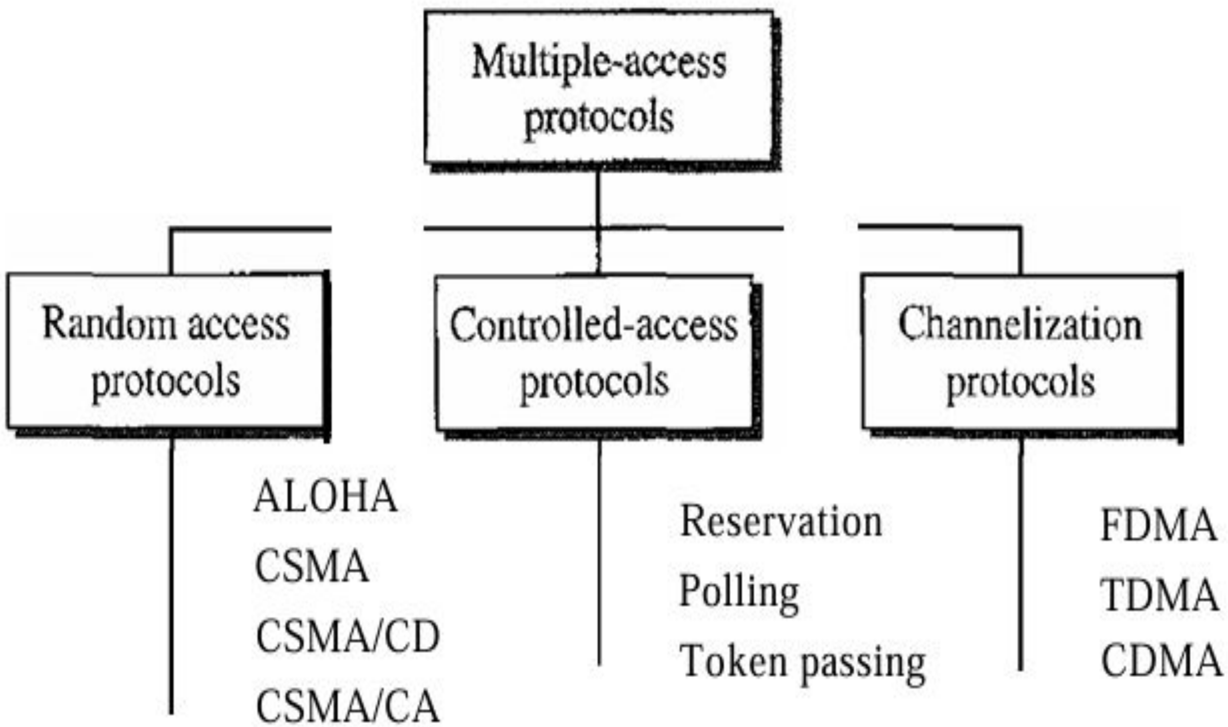
Figure 12.1 *Data link layer divided into two functionality-oriented sublayers*



- When nodes or stations are connected and use a common link, called a multipoint or broadcast link, we need a multiple-access protocol to coordinate access to the link. The problem ofcontrolling the access to the medium is similar to the rules of speaking in an assembly. The procedures guarantee that the right to speak is upheld and ensure that two people do not speak at the same time, do not interrupt each other, do not monopolize the discussion, and so on.
- The situation is similar for multipoint networks. Many formal protocols have been devised to handle access to a shared link. We categorize them into three groups. Protocols belonging to each group are shown in Figure 12.2.

We categorize them into three groups. Protocols belonging to each group are shown in Figure 12.2

Figure 12.2 *Taxonomy ofmultiple-access protocols discussed in this chapter*



12.1 RANDOMACCESS

- In random access or contention methods, no station is superior to another station and none is assigned the control over another. No station permits, or does not permit, another station to send. At each instance, a station that has data to send uses a procedure defined by the protocol to make a decision on whether or not to send. This decision depends on the state of the medium (idle or busy). In other words, each station can transmit when it desires on the condition that it follows the predefined procedure, including the testing ofthe state ofthe medium.

Two features give this method its name.

- First, there is no scheduled time for a station to transmit. Transmission is random among the stations. That is why these methods are called random access.
- Second, no rules specify which station should send next. Stations compete with one another to access the medium. That is why these methods are also called contention methods.

Random Access

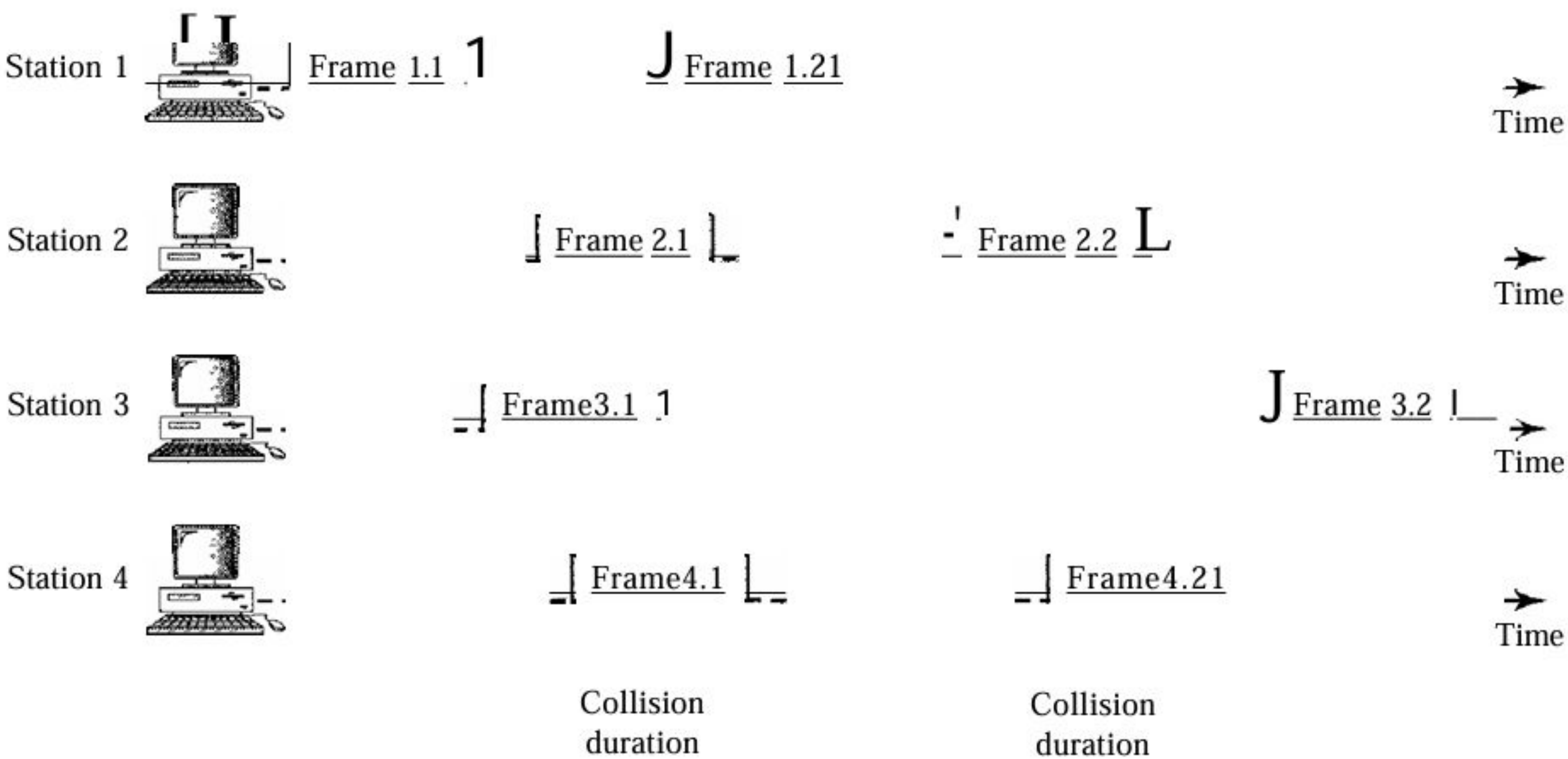
- In a random access method, each station has the right to the medium without being controlled by any other station. However, if more than one station tries to send, there is an access conflict-collision-and the frames will be either destroyed or modified. To avoid access conflict or to resolve it when it happens, each station follows a procedure that answers the following questions:
  - When can the station access the medium?
  - What can the station do if the medium is busy?
  - How can the station determine the success or failure of the transmission?
  - What can the station do if there is an access conflict?

- The random access methods we study in this chapter have evolved from a very interesting protocol known as ALOHA, which used a very simple procedure called multiple access (MA). The method was improved with the addition of a procedure that forces the station to sense the medium before transmitting. This was called carrier sense multiple access. This method later evolved into two parallel methods: carrier sense multiple access with collision detection (CSMAlCD) and carrier sense multiple access with collision avoidance (CSMA/CA). CSMA/CD tells the station what to do when a collision is detected. CSMA/CA tries to avoid the collision.

ALOHA

- ALOHA, the earliest random access method, was developed at the University of Hawaii in early 1970. It was designed for a radio (wireless) LAN, but it can be used on any shared medium.
- It is obvious that there are potential collisions in this arrangement. The medium is shared between the stations. When a station sends data, another station may attempt to do so at the same time. The data from the two stations collide and become garbled.
- PureALOHA The original ALOHA protocol is called pure ALOHA. This is a simple, but elegant protocol. The idea is that each station sends a frame whenever it has a frame to send. However, since there is only one channel to share, there is the possibility of collision between frames from different stations. Figure 12.3 shows an example of frame collisions in pure ALOHA.

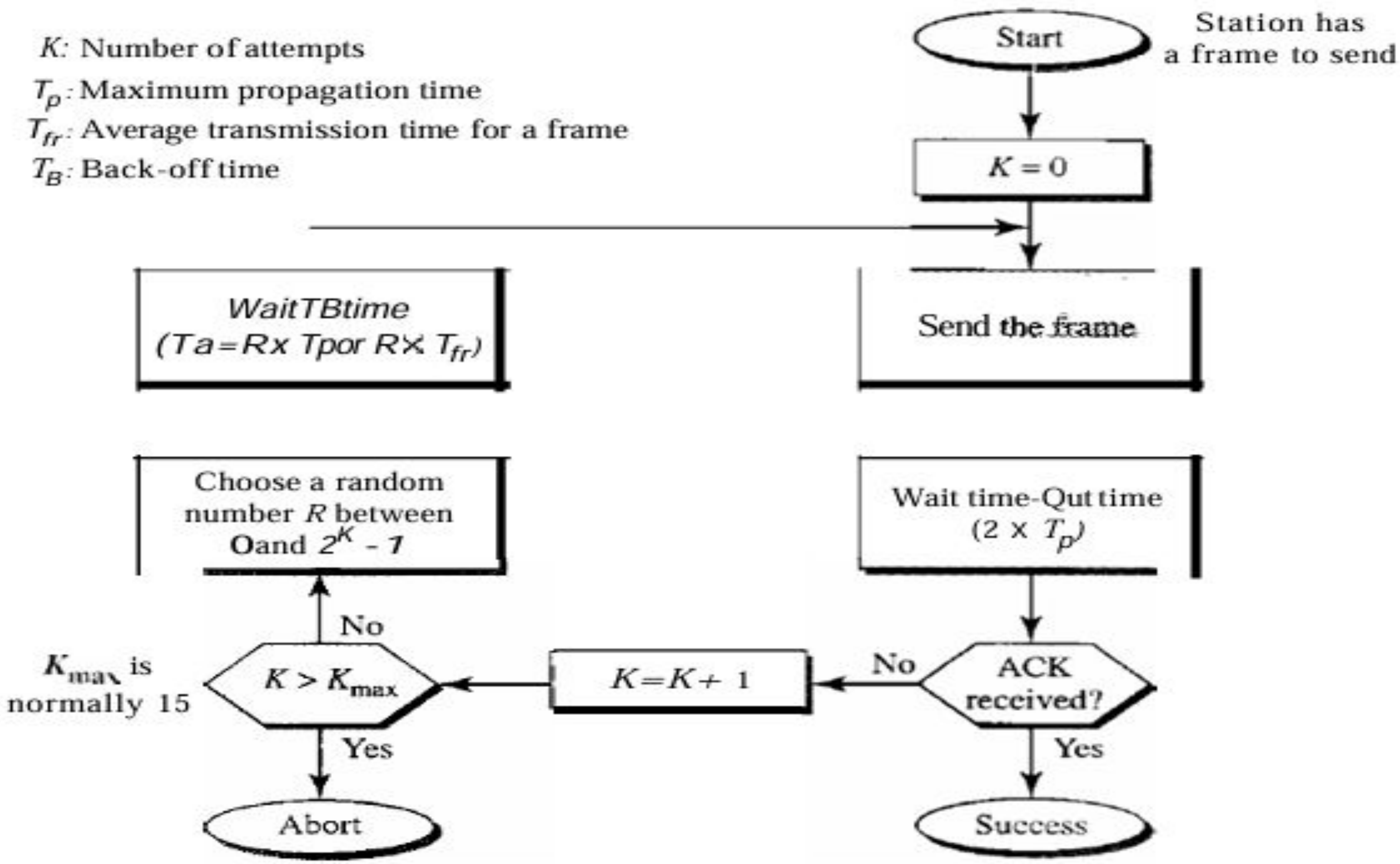
Figure 12.3 *Frames in a pure ALOHA network*



There are four stations (unrealistic assumption) that contend with one another for access to the shared channel. The figure shows that each station sends two frames; there are a total of eight frames on the shared medium. Some of these frames collide because multiple frames are in contention for the shared channel. Figure 12.3 shows that only

Pure ALOHA has a second method to prevent congesting the channel with retransmitted frames. After a maximum number of retransmission attempts 'Kmax ' a station must give up and try later. Figure 12.4 shows the procedure for pure ALOHA based on the above strategy.

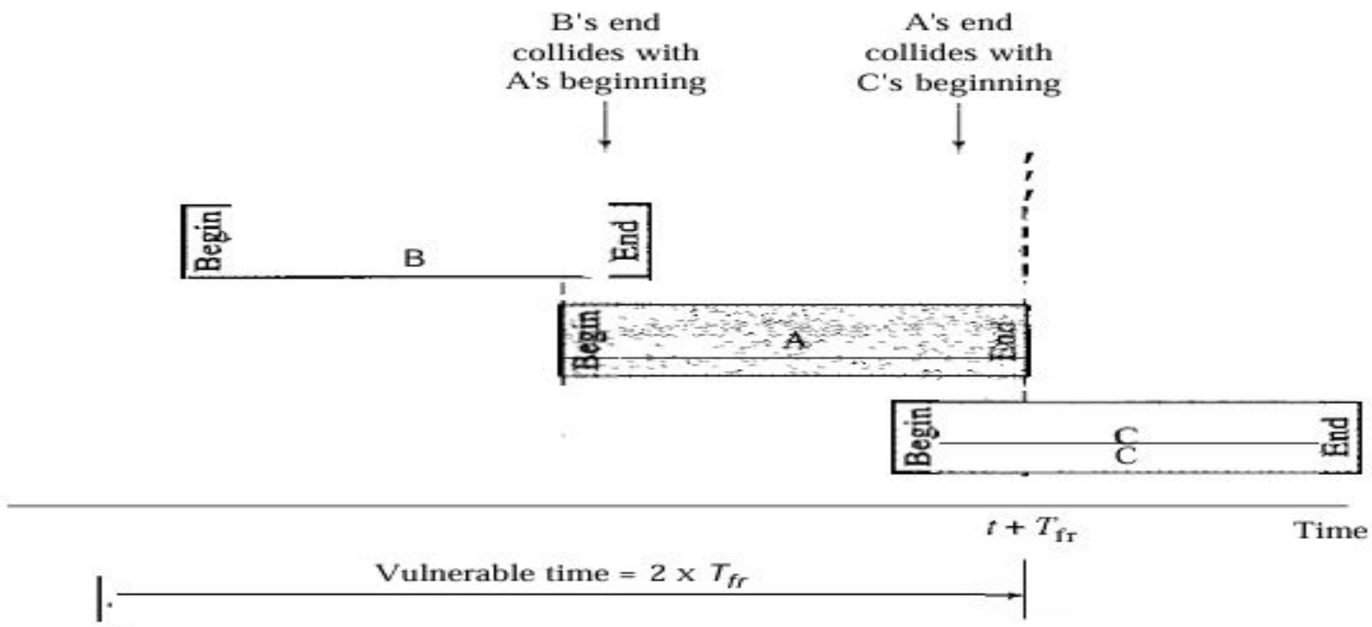
Figure 12.4 *Procedure for pure ALOHA protocol*



ALOHA: Vulnerable time

Vulnerable time Let us find the length of time, the vulnerable time, in which there is a possibility of collision. We assume that the stations send fixed-length frames with each frame taking T_{fr} S to send. Figure 12.5 shows the vulnerable time for station A.

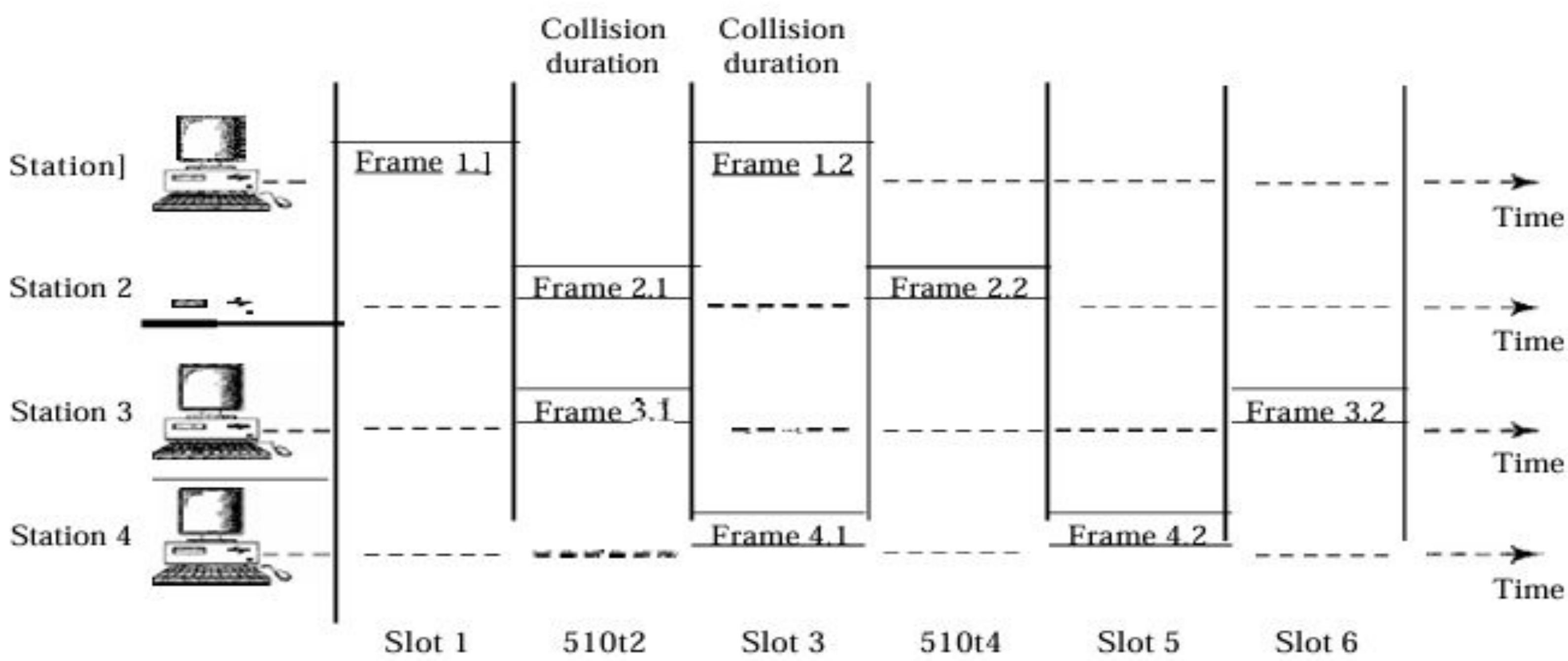
Figure 12.5 *Vulnerable time for pure ALOHA protocol*



Slotted ALOHA

- Pure ALOHA has a vulnerable time of 2 x T_{fr} . This is so because there is no rule that defines when the station can send. A station may send soon after another station has started or soon before another station has finished. Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- In slotted ALOHA we divide the time into slots of T_{fr} s and force the station to send only at the beginning of the time slot. Figure 12.6 shows an example of frame collisions in slotted ALOHA.

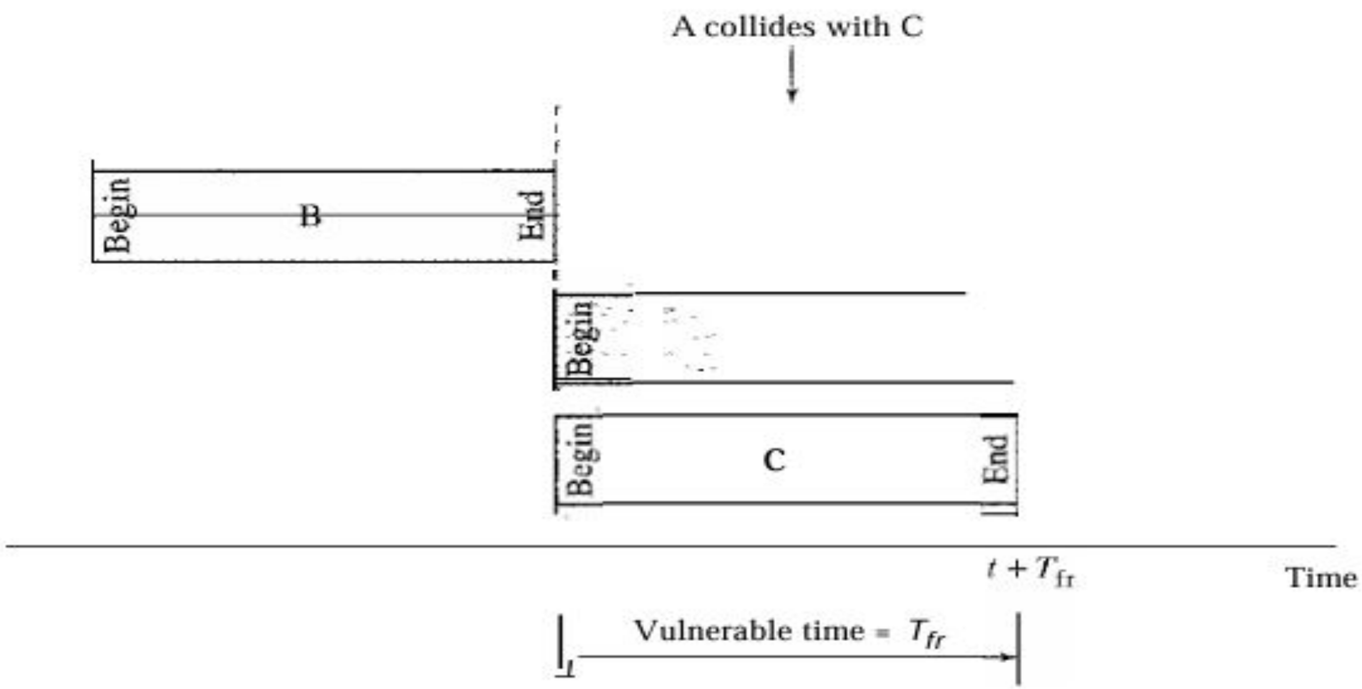
Figure 12.6 *Frames in a slotted ALOHA network*



- Because a station is allowed to send only at the beginning of the synchronized time slot, if a station misses this moment, it must wait until the beginning of the next time slot. This means that the station which started at the beginning of this slot has already finished sending its frame. Of course, there is still 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 T_{fr}. Figure 12.7 shows the situation. Figure 12.7 shows that the vulnerable time for slotted ALOHA is one-half that of pure ALOHA.

  Slotted ALOHA vulnerable time $= T_{fr}$

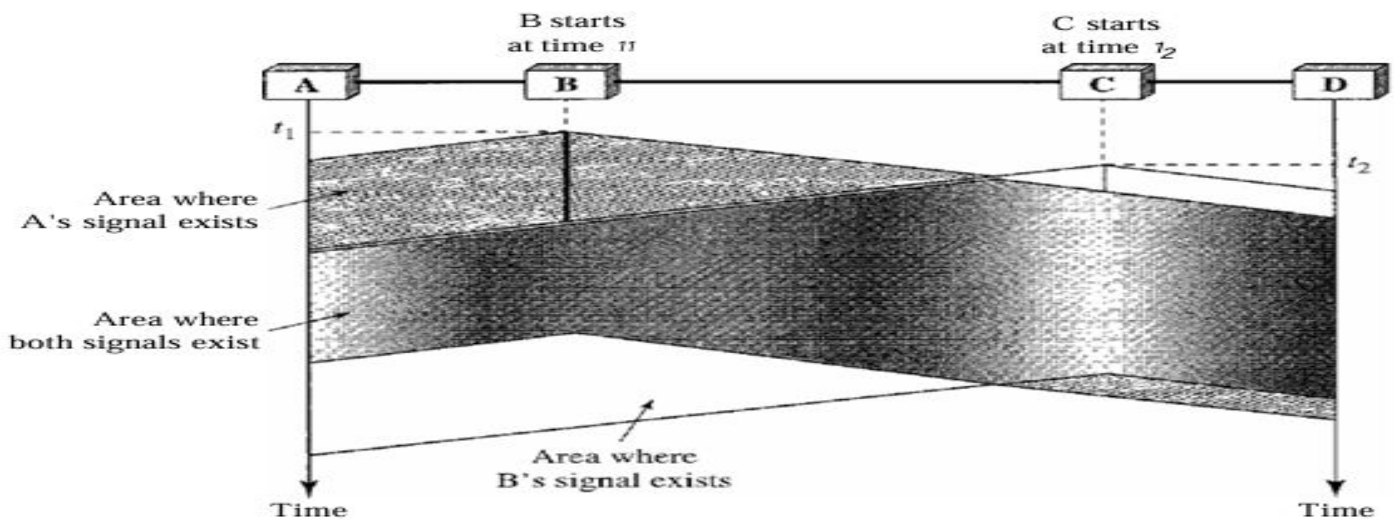
Figure 12.7 *Vulnerable time for slotted ALOHA protocol*



Carrier Sense Multiple Access (CSMA)

- To minimize the chance of collision and, therefore, increase the performance, the CSMA method was developed. The chance of collision can be reduced if a station senses the medium before trying to use it. Carrier sense multiple access (CSMA) requires that each station first listen to the medium (or check the state of the medium) before sending. In other words, CSMA is based on the principle "sense before transmit "or” listen before talk." CSMA can reduce the possibility of collision, but it cannot eliminate it. The reason for this is shown in Figure 12.8, a space and time model of a CSMA network. Stations are connected to a shared channel (usually a dedicated medium).

Figure 12.8 *Space/time model of the collision in CSMA*



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

- The CSMA method does not specify the procedure following a collision. Carrier sense multiple access with collision detection (CSMA/CD) augments the algorithm to handle the collision.
- In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again. To better understand CSMA/CD, let us look at the first bits transmitted by the two stations involved in the collision. Although each station continues to send bits in the frame until it detects the collision, we show what happens as the first bits collide. In Figure 12.12, stations A and C are involved in the collision

Figure 12.12 *Collision ofthe first bit in CSMAICD*

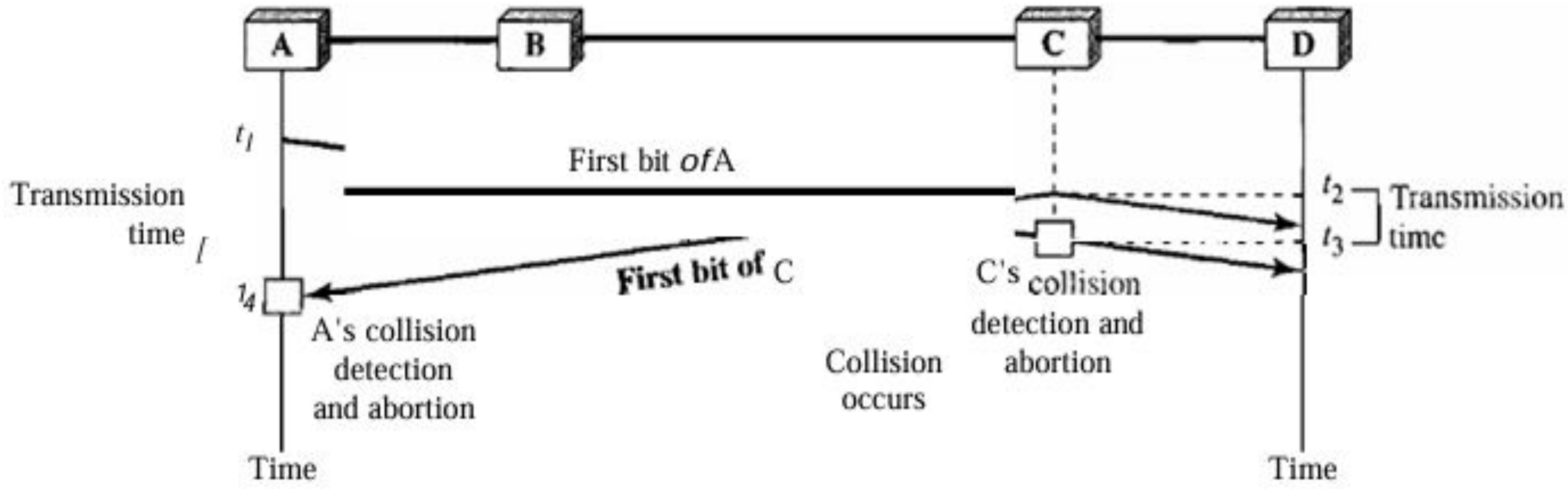


Figure 12.13 *Collision and abortion in CSMAICD*

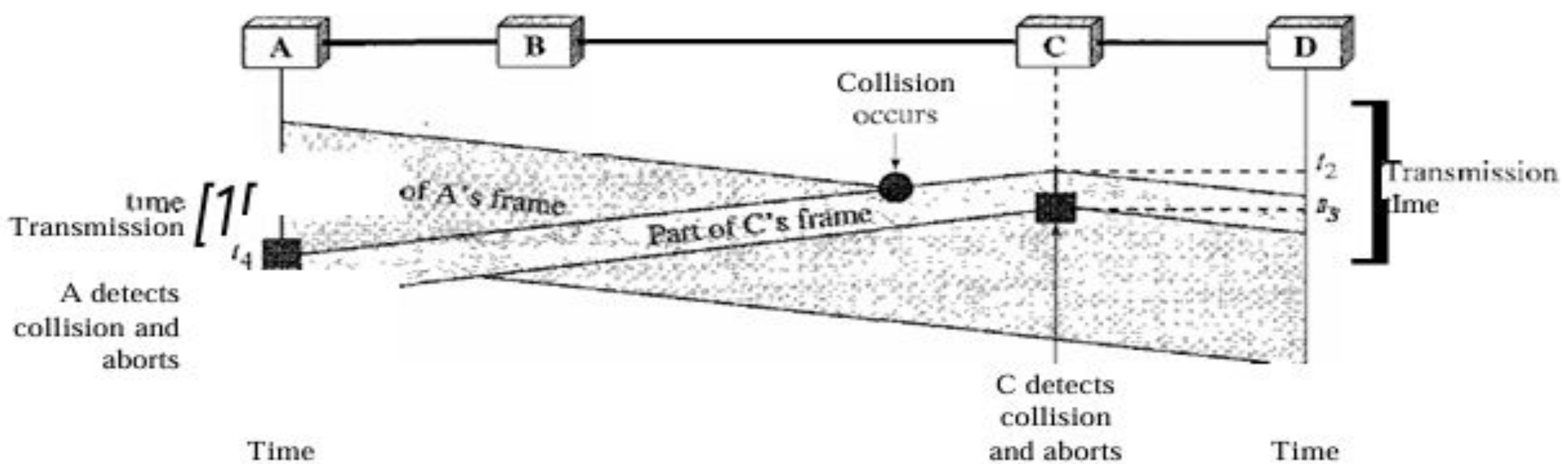
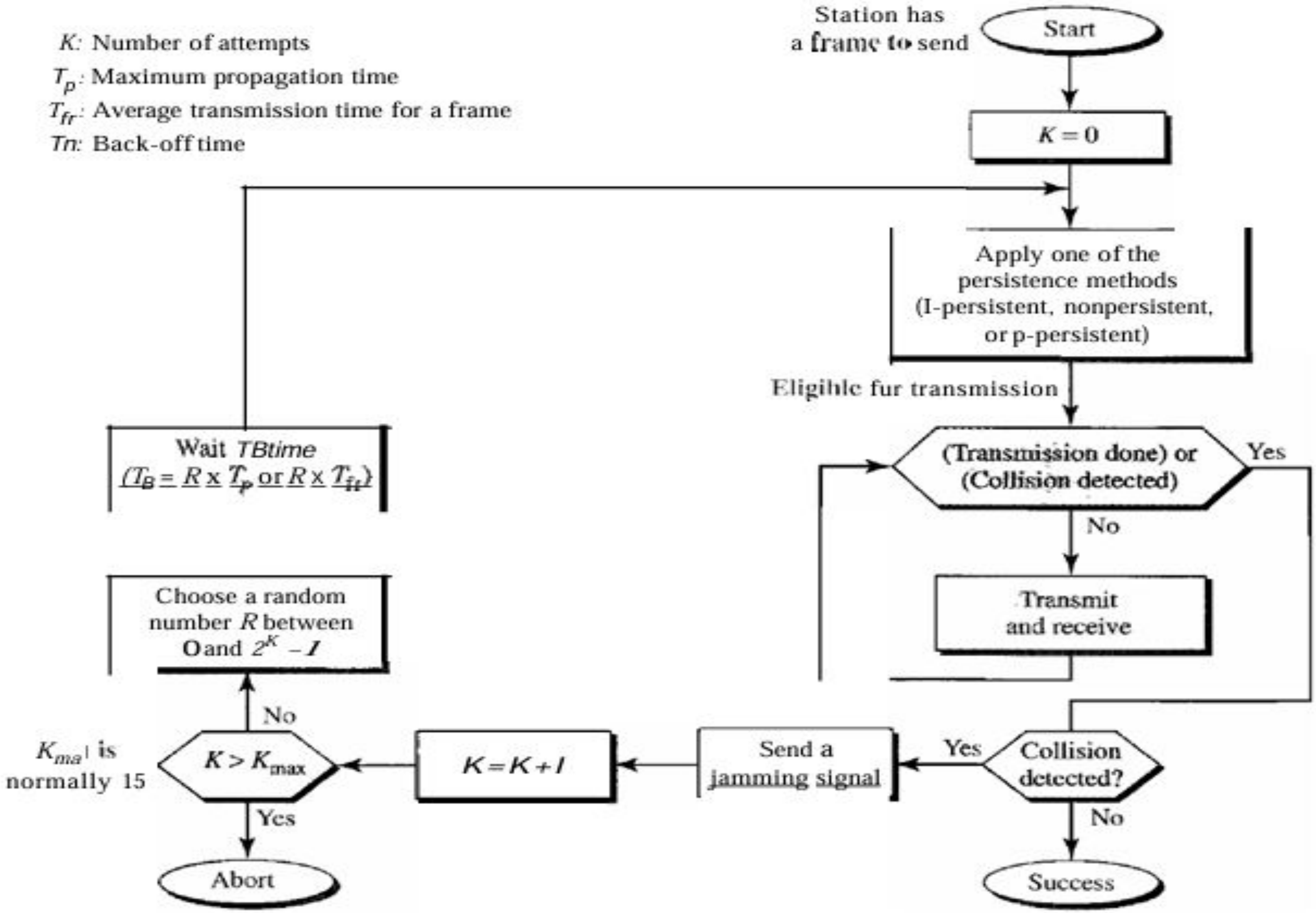


Figure 12.14 *Flow diagramfor the CSMAICD*

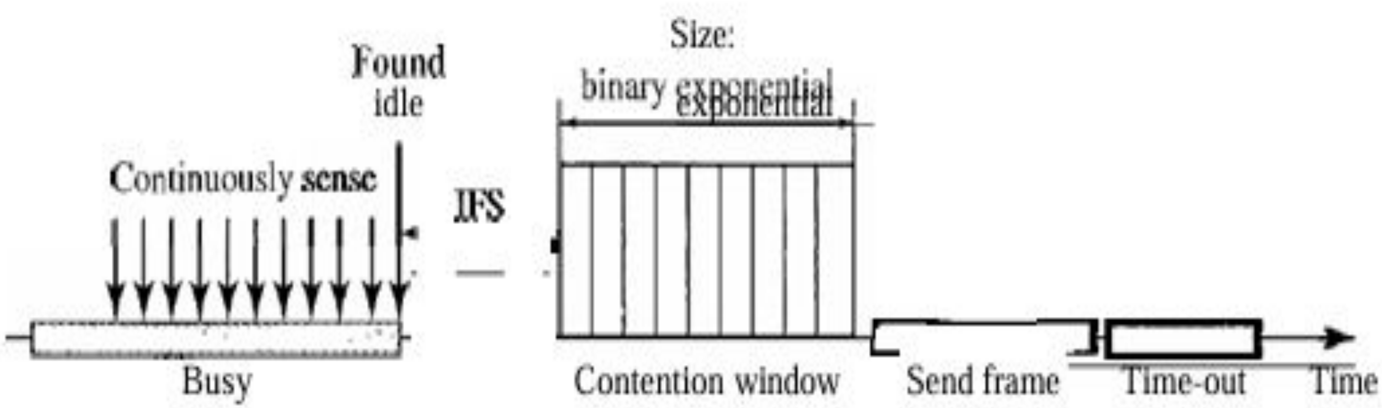


Carrier Sense MultipleAccess with Collision Avoidance (CSMA/CA)

- The basic idea behind CSMA/CD is that a station needs to be able to receive while transmitting to detect a collision. When there is no collision, the station receives one signal: its own signal. When there is a collision, the station receives two signals: its own signal and the signal transmitted by a second station. To distinguish between these two cases, the received signals in these two cases must be significantly different.
- In other words, the signal from the second station needs to add a significant amount of energy to the one created by the first station. In a wired network, the received signal has almost the same energy as the sent signal because either the length of the cable is short or there are repeaters that amplify the energy between the sender and the receiver. This means that in a collision, the detected energy almost doubles.
- However, in a wireless network, much of the sent energy is lost in transmission. The received signal has very little energy. Therefore, a collision may add only 5 to 10 percent additional energy. This is not useful for effective collision detection.

- We need to avoid collisions on wireless networks because they cannot be detected. Carrier sense multiple access with collision avoidance (CSMAICA) was invented for this network. Collisions are avoided through the use of CSMAICA's three strategies: the inter frame space, the contention window, and acknowledgments, as shown in Figure 12.16.

Figure 12.16 *Timing in CSMAICA*



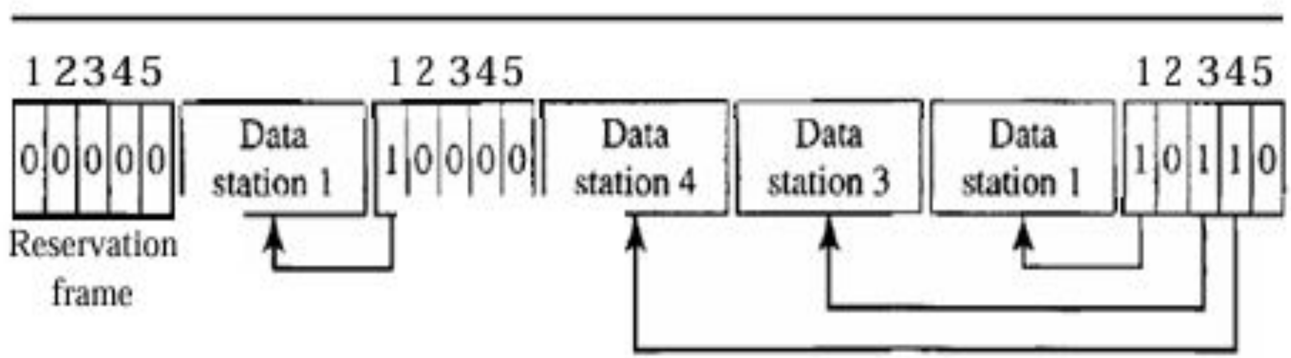
- Interframe Space (IFS):
  - First, collisions are avoided by deferring transmission even if the channel is found idle. When an idle channel is found, the station does not send immediately. It waits for a period of time called the interframe space or IFS. Even though the channel may appear idle when it is sensed, a distant station may have already started transmitting. The distant station's signal has not yet reached this station. The IFS time allows the front of the transmitted signal by the distant station to reach this station. If after the IFS time the channel is still idle, the station can send, but it still needs to wait a time equal to the contention time (described next). The IFS variable can also be used to prioritize stations or frame types.
- Contention Window
  - The contention window is an amount of time divided into slots. A station that is ready to send chooses a random number of slots as its wait time. The number of slots in the window changes according to the binary exponential back-off strategy. This means that it is set to one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time.
- Acknowledgment
  - With all these precautions, there still may be a collision resulting in destroyed data. In addition, the data may be corrupted during the transmission. The positive acknowledgment and the time-out timer can help guarantee that the receiver has received the frame.

12.2 CONTROLLED ACCESS

- In controlled access, the stations consult one another to find which station has the right to send. A station cannot send unless it has been authorized by other stations. We discuss three popular controlled-access methods.
- Reservation:
  - In the reservation method, a station needs to make a reservation before sending data. Time is divided into intervals. In each interval, a reservation frame precedes the data frames sent in that interval.
  - If there are N stations in the system, there are exactly N reservation minislots in the reservation frame. Each minislot belongs to a station. When a station needs to send a data frame, it makes a reservation in its own minislot. The stations that have made reservations can send their data frames after the reservation frame.

- Figure 12.18 shows a situation with five stations and a five-minislot reservation frame. In the first interval, only stations 1, 3, and 4 have made reservations. In the second interval, only station 1 has made a reservation.

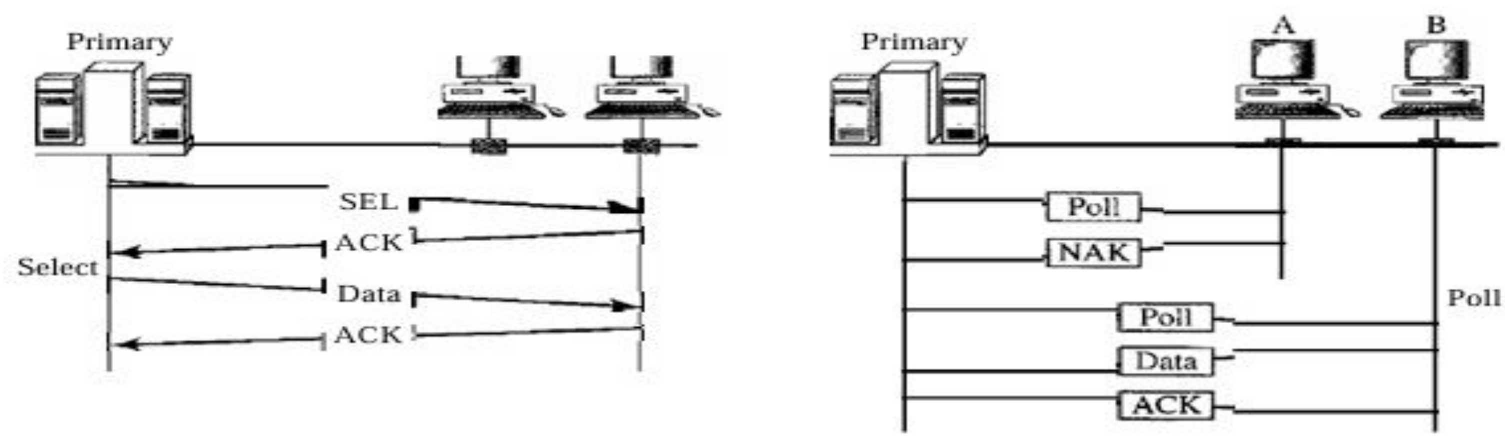
Figure 12.18 *Reservation access method*



Polling

- Polling works with topologies in which one device is designated as a primary station and the other devices are secondary stations. All data exchanges must be made through the primary device even when the ultimate destination is a secondary device. The primary device controls the link; the secondary devices follow its instructions. It is up to the primary device to determine which device is allowed to use the channel at a given time. The primary device, therefore, is always the initiator of a session (see Figure 12.19).

Figure 12.19 *Select and poll functions in polling access method*



If the primary wants to receive data, it asks the secondaries if they have anything to send; this is called poll function. If the primary wants to send data, it tells the secondary to get ready to receive; this is called select function.

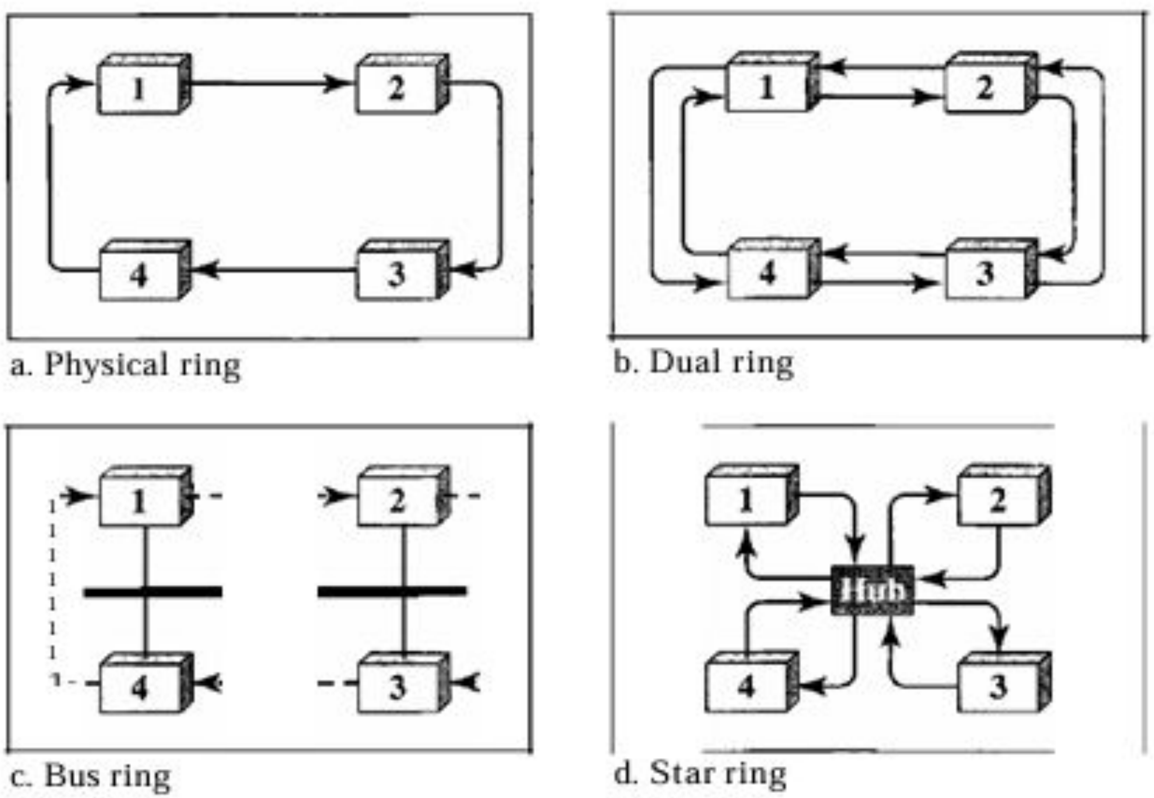
- Select
  - The select function is used whenever the primary device has something to send. Remember that the primary controls the link. If the primary is neither sending nor receiving data, it knows the link is available.
  - If it has something to send, the primary device sends it. What it does not know, however, is whether the target device is prepared to receive. So the primary must alert the secondary to the upcoming transmission and wait for an acknowledgment of the secondary's ready status. Before sending data, the primary creates and transmits a select (SEL) frame, one field of which includes the address of the intended secondary.
- Poll
  - The poll function is used by the primary device to solicit transmissions from the secondary devices. When the primary is ready to receive data, it must ask (poll) each device in turn if it has anything to send. When the first secondary is approached, it responds either with a NAK frame if it has nothing to send or with data (in the form of a data frame) if it does. If the response is negative (a NAK frame), then the primary polls the next secondary in the same manner until it finds one with data to send. When the response is positive (a data frame), the primary reads the frame and returns an acknowledgment (ACK frame), verifying its receipt.

Token Passing

- In the token-passing method, the stations in a network are organized in a logical ring. In other words, for each station, there is a predecessor and a successor. The predecessor is the station which is logically before the station in the ring; the successor is the station which is after the station in the ring. The current station is the one that is accessing the channel now. The right to this access has been passed from the predecessor to the cur rent station. The right will be passed to the successor when the current station has no more data to send.
- Token management is needed for this access method. Stations must be limited in the time they can have possession of the token. The token must be monitored to ensure it has not been lost or destroyed. For example, if a station that is holding the token fails, the token will disappear from the network. Another function of token management is to assign priorities to the stations and to the types of data being transmitted. And finally, token management is needed to make low-priority stations release the token to high priority stations.

- Logical Ring
  - In a token-passing network, stations do not have to be physically connected in a ring; the ring can be a logical one. Figure 12.20 show four different physical topologies that can create a logical ring.

Figure 12.20 *Logical ring and physical topology in token-passing access method*



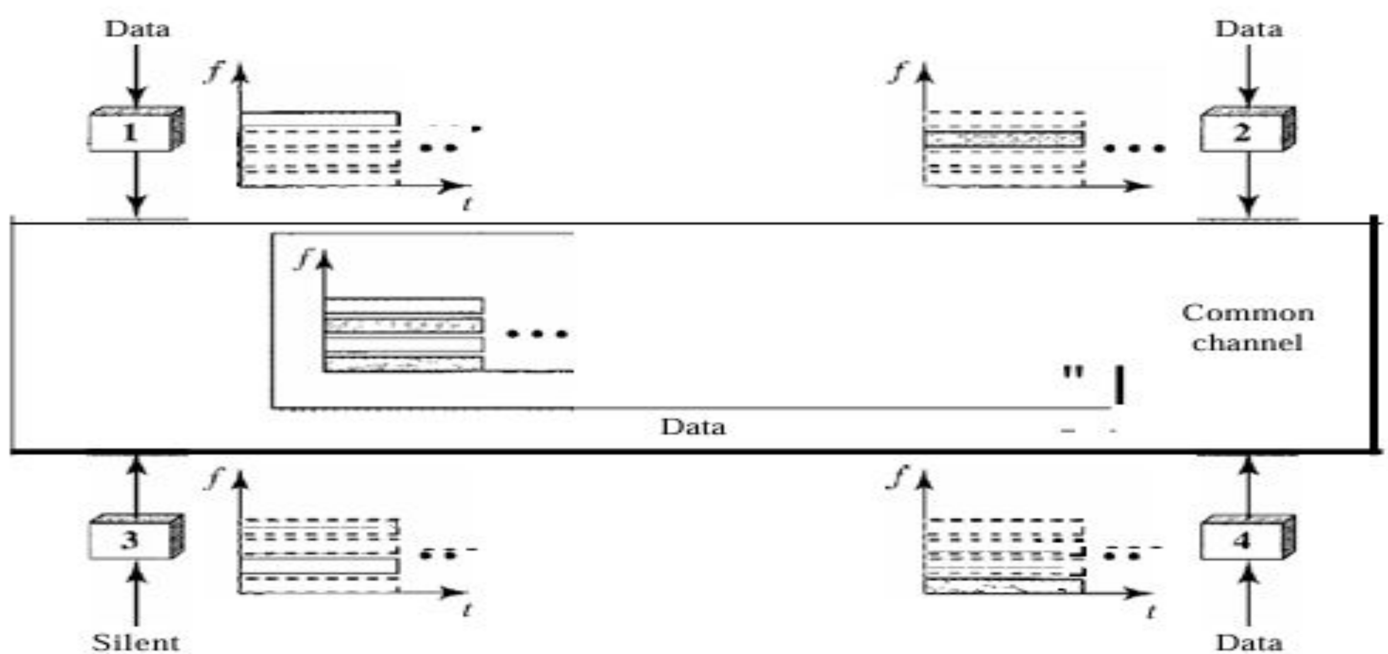
12.3 CHANNELIZATION

- Channelization is a multiple-access method in which the available bandwidth ofa link is shared in time, frequency, or through code, between different stations. In this section, we discuss three channelization protocols:
  - FDMA,
  - TDMA, and
  - CDMA.

Frequency-Division MultipleAccess (FDMA)

- In frequency-division multiple access (FDMA), the available bandwidth is divided into frequency bands. Each station is allocated a band to send its data. In other words, each band is reserved for a specific station, and it belongs to the station all the time. Each station also uses a band pass filter to confine the transmitter frequencies. To pre vent station interferences, the allocated bands are separated from one another by small guard bands. Figure 12.21 shows the idea of FDMA.

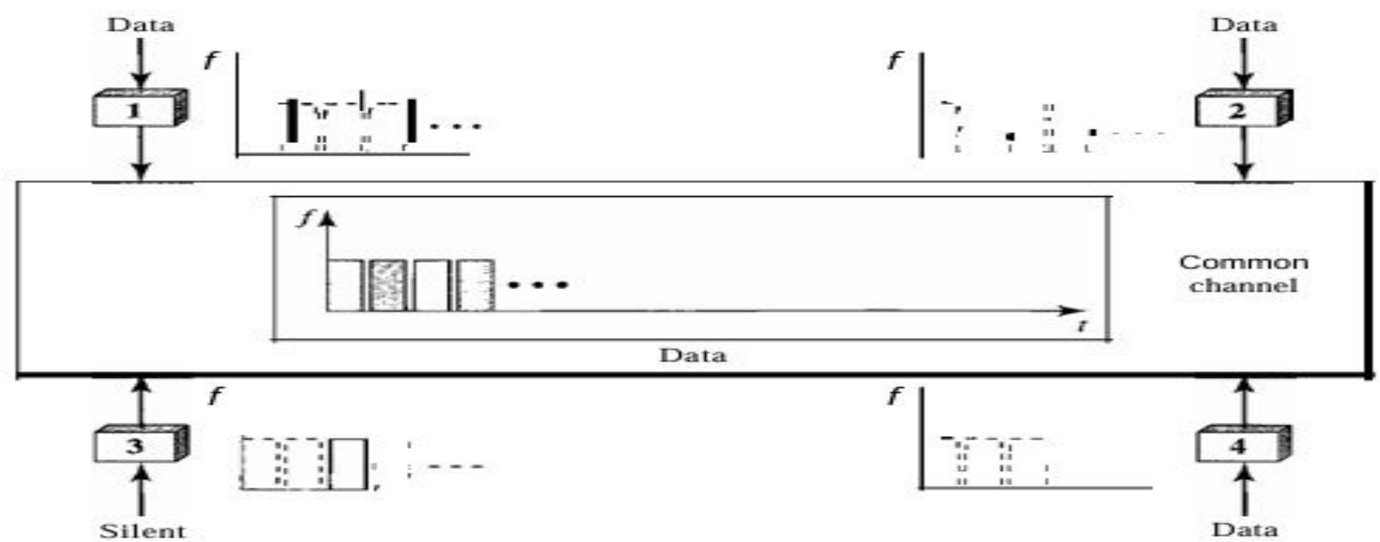
Figure 12.21 *Frequency-division multiple access (FDMA)*



Time-Division Multiple Access (TDMA)

- In time-division multiple access (TDMA), the stations share the bandwidth of the channel in time. Each station is allocated a time slot during which it can send data. Each station transmits its data in is assigned time slot. Figure 12.22 shows the idea behind TDMA.
- The main problem with TDMA lies in achieving synchronization between the different stations. Each station needs to know the beginning of its slot and the location of its slot. This may be difficult because of propagation delays introduced in the system if the stations are spread over a large area. To compensate for the delays, we can insert guard times. Synchronization is normally accomplished by having some synchronization bits (normally refened to as preamble bits) at the beginning of each slot.

Figure 12.22 *Time-division multiple access (TDMA)*



Code-Division Multiple Access (CDMA)

- Code-division multiple access (CDMA) was conceived several decades ago. Recent advances in electronic technology have finally made its implementation possible. CDMA differs from FDMA because only one channel occupies the entire bandwidth of the link. It differs from TDMA because all stations can send data simultaneously; there is no timesharing.