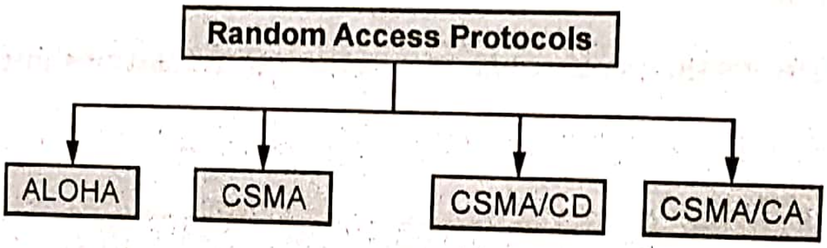
3.5 RANDOM ACCESS PROTOCOLS

We studied earlier that medium sharing techniques are broadly classified into two i.e. static medium access control and dynamic medium access control. We also found in earlier discussions that static medium access control techniques are highly inefficient for bursty traffic networks as LAN so we mostly use dynamic method. Dynamic medium access control is again categorised into two random access and control access. Here we will discuss random access protocols. They are the simplest methods used for transmission. They are known as **random access methods** because there is no scheduled time for a station to transmit, it is random among stations. They are also known as **contention methods** because there are no rules specifying which station should transmit next, all just compete with one another to access the medium.

In random access method, each station can use medium without being controlled by any other station. But if more than one station tries to send. There is an access conflict or **collision** and frames are either destroyed or modified.



The simplest protocol using this method is **ALOHA** where each individual user transmits at any arbitrary time where there is data to be sent. If more than one user attempts to transmit at the same time, the messages collide and have to retransmitted later. It is completely asynchronous as there is no coordination among users. Also high interference possibility gives maximum achievable channel utilization as only about 18 percent.

To improve efficiency, **slotted ALOHA** was developed where channel is divided into discrete uniform time slots, each equal to one frame transmission time. All users are synchronized by a central clock so that transmission can only start at the beginning of a time slot. Thus partial overlapping of colliding messages is eliminated and utilization increased to about 37 percent.

Further improvement was done by making user listen the channel to detect if it is free before transmitting a message packet. If channel is idle user transmits and if it is busy packet transmission is delayed. This scheme is called as **Carrier Sense Multiple Access (CSMA), or listen Before Talk (LBT).** It greatly reduces the chance of collision.

Although CSMA was better than ALOHA but still collisions occur, so further improvements were done by shortening the collision time duration. User can listen to the channel while sending out a message and immediately stop transmission if collision is detected. This is known as **CSMA with Collision Detection (CSMA/CD) or Listen While Talk (LWT).**

Figure 3.13 shows the throughput S versus load G for 1-Persistent and Non-Persistent CSMA for three values of a.

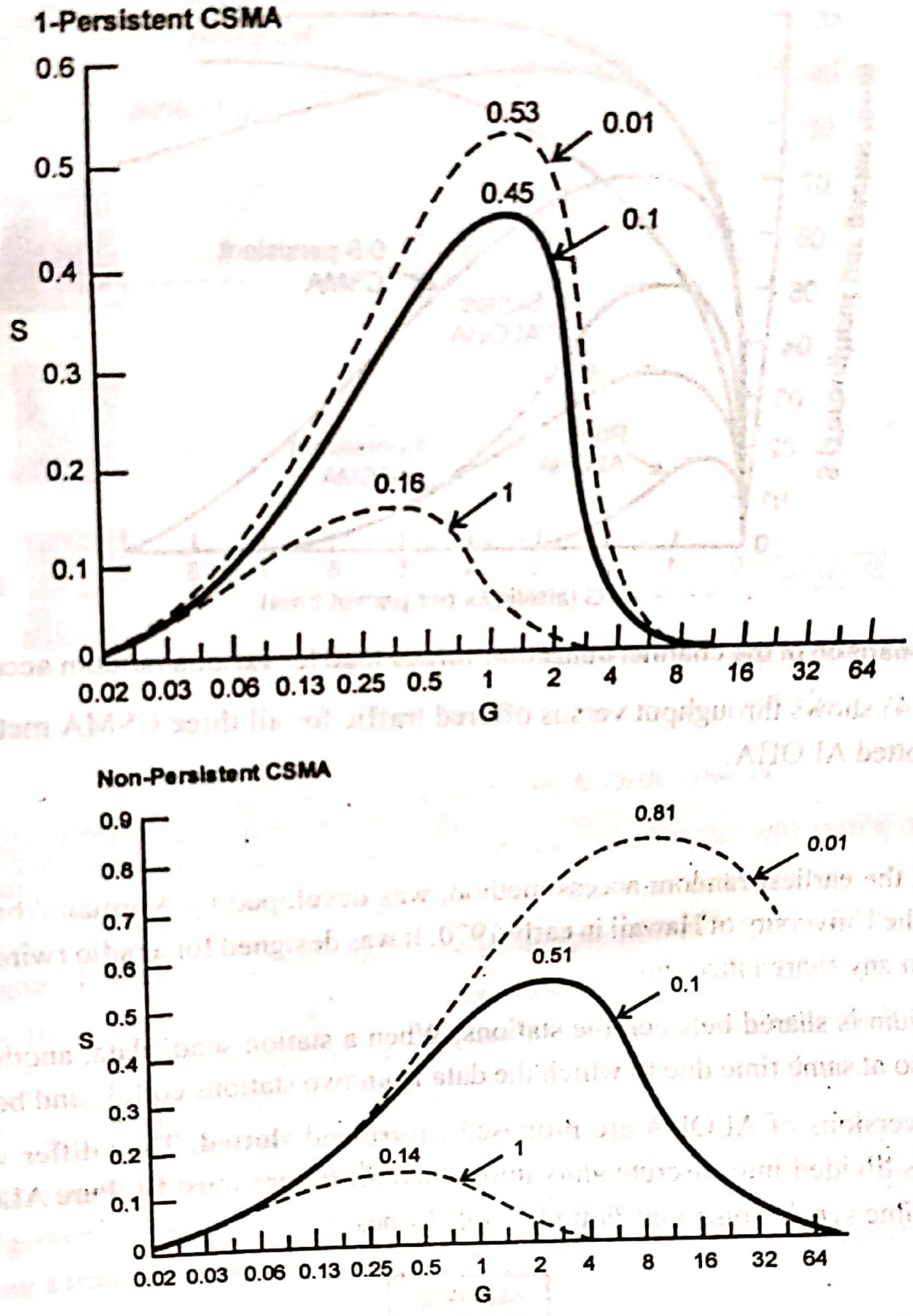


Fig. 3.13 : Throughput S versus load G for 1-Persistent and Non-Persistent CSMA. The curves are for different values of a.

The throughput of 1-Persistent CSMA drops off much more sharply with increased G. Also the normalized propagation delay $a = T_P/X$ (X is length of frame) has a significant impact on the maximum achievable throughput. Non-Persistent CSMA achieves a higher throughput than 1-persistent CSMA over a broader range of G. For small values of a, Non-persistent CSMA has a relatively high maximum achievable throughput. However as a$\rightarrow$1, both 1-Persistent and Non-Persistent CSMA have maximum achievable throughputs even lower than ALOHA schemes.

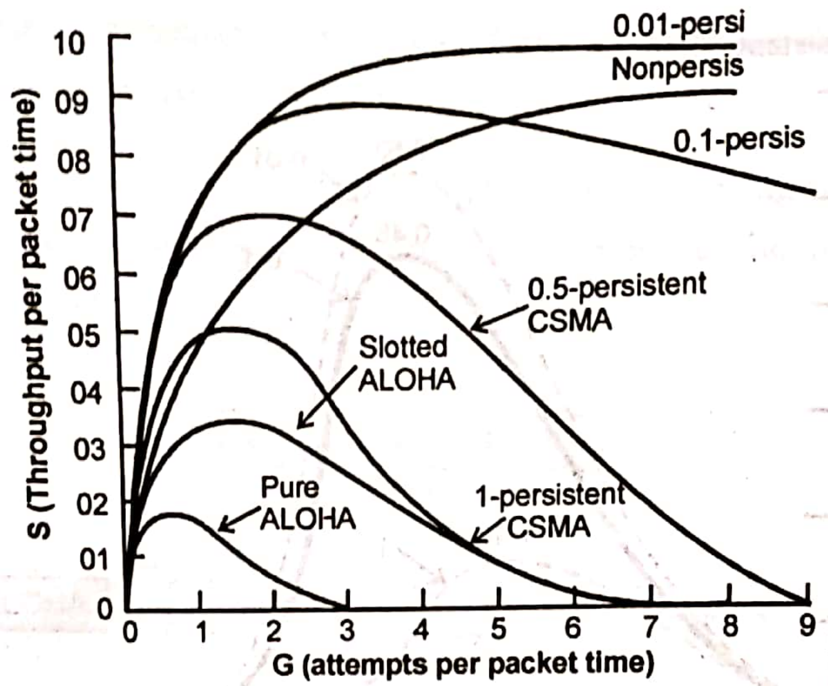


Fig. 3.14 : Comparison of the channel utilization versus load for various random access protocols

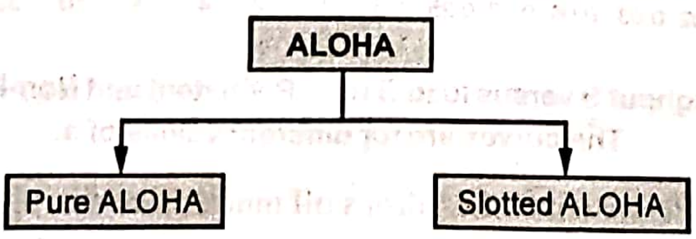
Figure (3.14) shows throughput versus offered traffic for all three CSMA methods as well as for pure and slotted ALOHA.

3.5.1 Aloha

ALOHA, the earliest random access method, was developed by Norman Abramson and his colleagues at the University of Hawaii in early 1970. It was designed for a radio (wireless) LAN, but can be used on any shared medium.

The medium is shared between the stations. When a station sends data, another station may attempt to do so at same time due to which the data from two stations collide and become garbled.

The two versions of ALOHA are proposed : **pure** and **slotted**. They differ with respect to whether time is divided into discrete slots into which all frames must fit. Pure ALOHA does not require global time synchronisation; slotted ALOHA does.



(A) PURE ALOHA :

The original ALOHA protocol is called **pure ALOHA.** This is based on the simplest idea that each station sends a frame whenever it has a frame to send. But since one channel is being shared by multiple users, there is possibility of collision between frames from different stations. Figure

(3.15) shows an example of frame collisions in pure ALOHA.

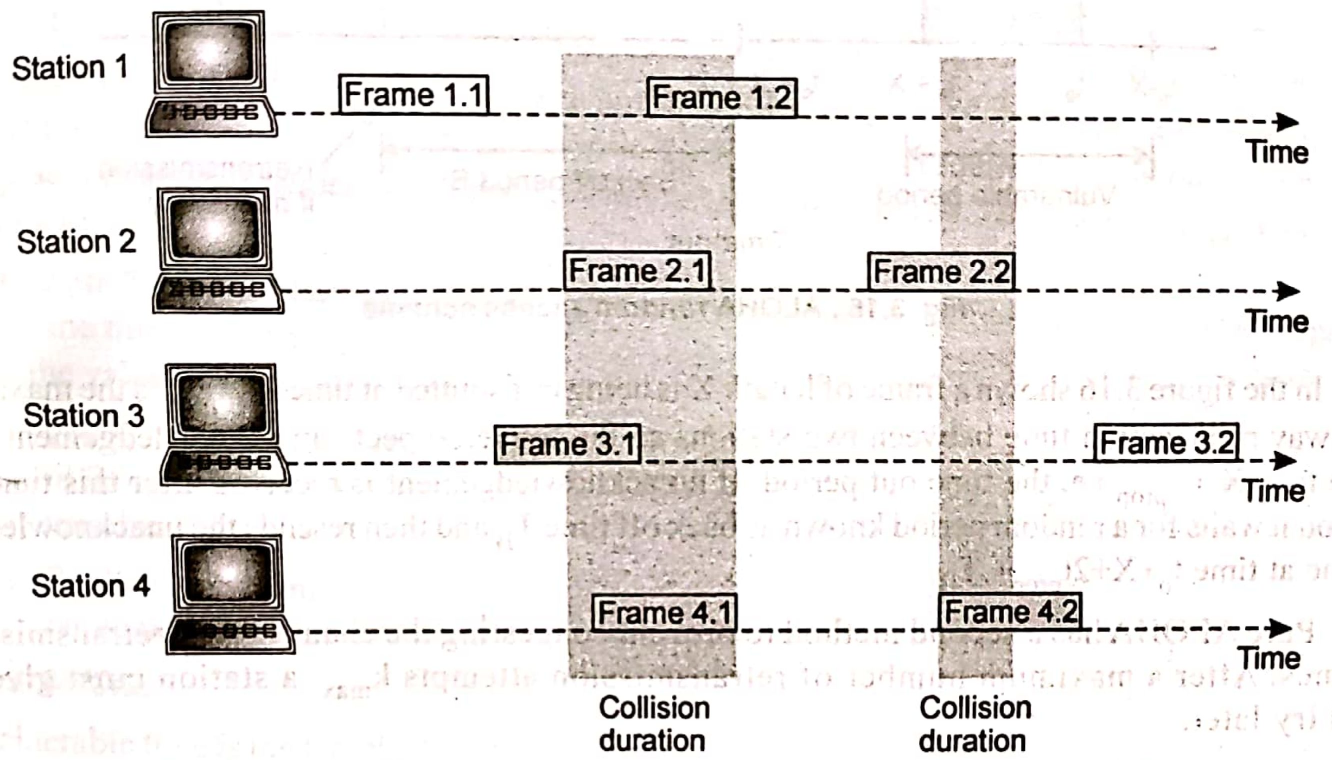


Fig. 3.15 : Frames in a pure ALOHA network

The figure 3.15 shows that there are four stations that compete with one another for access to the shared channel. Each station sends two frames so there are total of eight frames on the shared medium. Some of these frames collide because multiple frames are in contention for the shared channel. The figure 3.15 shows that only two frames survive : frame 1.1 from station 1 & frame 3.2 from station 3. Even if one bit of a frame coexists on the channel with one bit from another frame, there is a collision and both will be destroyed.

The frames which are destroyed are then resend. The pure ALOHA protocol depends on acknowledgements from the receiver. If acknowledgement is not received after a time out period after sending a frame, the station assumes that the frame (or acknowledgement) has been destroyed and it then resends the frame.

Since the collision involves two or more station, then if all these stations try to resend their frames after time out, the frames will collide again.

So in pure ALOHA when the time out period passes, each station waits a random amount of time before resending its frames which helps in avoiding more collisions. This time is called as **back off time T_B.**

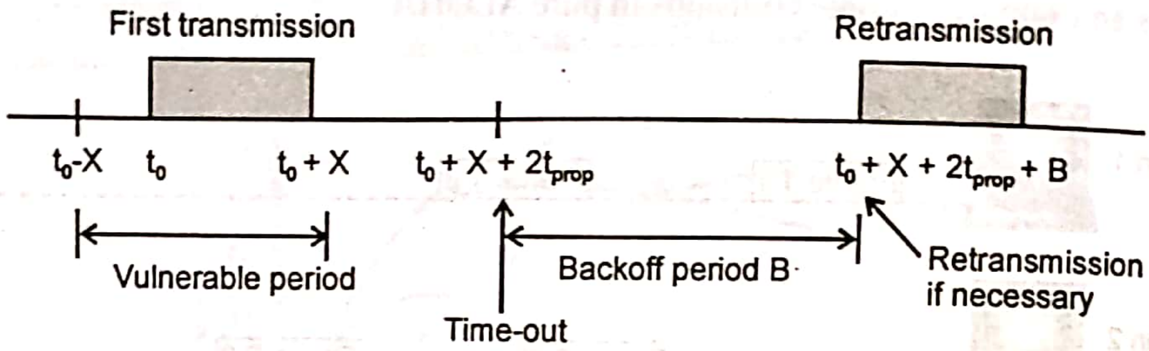


Fig. 3.16 : ALOHA random access scheme

In the figure 3.16 shown a frame of length X is being transmitted at time t_0 . t_{prop} is the maximum one way propagation time between two stations so the sender expects an acknowledgement after time to + X + t_{prop} i.e. the time out period. If no acknowledgement is received after this time out period it waits for a random period known as back off time T_B and then resends the unacknowledged frame at time t_o+X+$2t_{prop}$+ T_B.

Pure ALOHA has a second method to prevent congesting the channel with retransmission frames. After a maximum number of retransmission attempts k_{max}, a station must give up and try later.

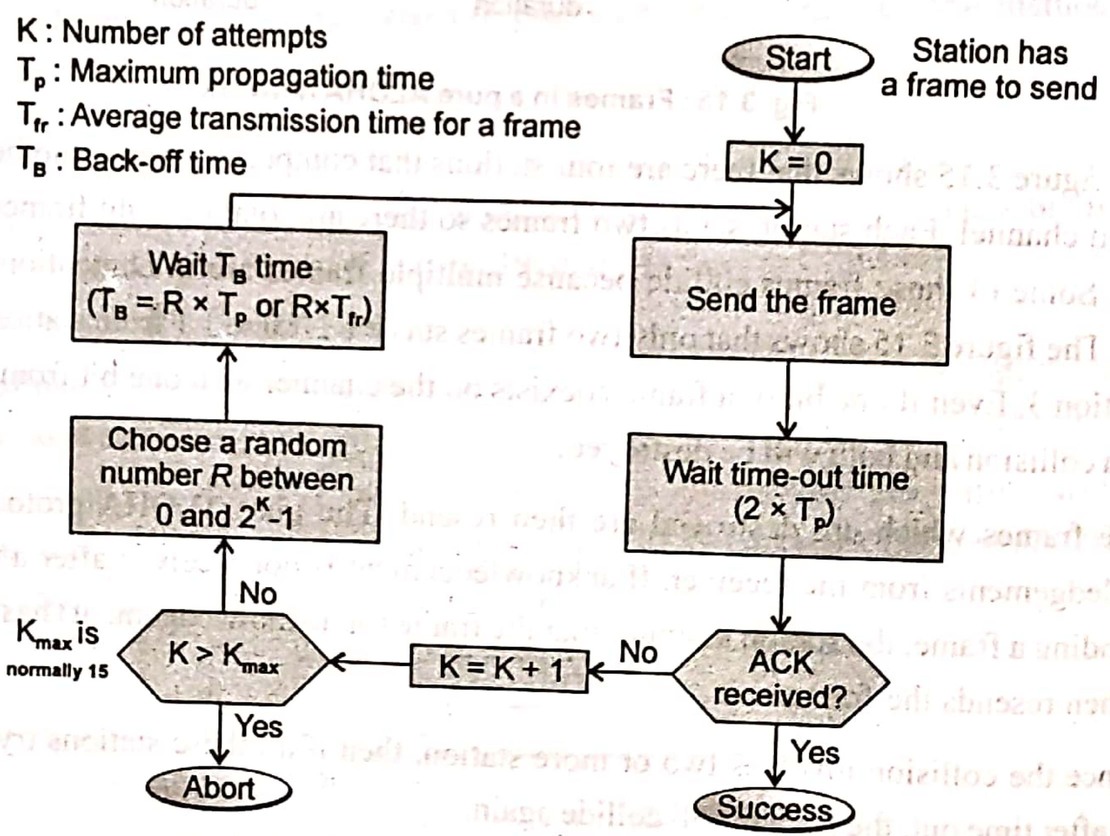


Fig. 3.17 : Procedure for pure ALOHA protocol

The figure (3.17) shows the procedure for pure ALOHA based on this second strategy. The time out period is equal to the maximum possible round trip propagation delay which is twice the amount of time required to send a frame between two stations (2×T_P). The back off time T_B is a random value that depends on K (the number of attempted unsuccessful transmissions). Formula for T_B depends on implementation, where one commonly used is the **binary exponential back off.**

In this method for each retransmission, a multiplier in range 0 to $2^k - 1$ is randomly chosen and multiplied by T_P (maximum propagation time) or T_{fr} (average time required to send out a frame) to find T_B. So the range of random numbers increases after each collision. The value of k_{mcx} is usually chosen as 15.

Example 3.1 : The stations on a wireless ALOHA network are a maximum of 600 km apart. If we assume that signals propagate at 3×10^8 m/s, we find $T_P = (600\times10^5)/3\times10^8 = 2$ ms. Now we can find the value of T_B for different values of K.

(a) For K = 1, the range is (0, 1). The station needs to generate a random number with a value of 0 or 1. So $T_B = R\times T_P = (0\times2)$ 0ms or (1×2) 2ms, based on the outcome of the random variable.

(b) For K = 2, the range is (0, 1, 2, 3). So T_B can be 0, 2, 4 or 6ms based on the outcome of the random variable.

Vulnerable Time :

Vulnerable time is the length of 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 (3.18) shows the vulnerable time for station A.

Station A sends a frame at time *t*. Station B has already sent a frame between *t*-T_{fr} and *t* which leads to a collision between the frames from station A & station B. The end of B's frame collides with the beginning of A's frame. Also if another station C sends a frame between *t* and $t + T_{fr}$ there is a collision between frames from station A & C. The beginning of C's frame collides with the end of A's frame. Thus we found that vulnerable time during which a collision may occur in pure ALOHA is two times the frame transmission time.

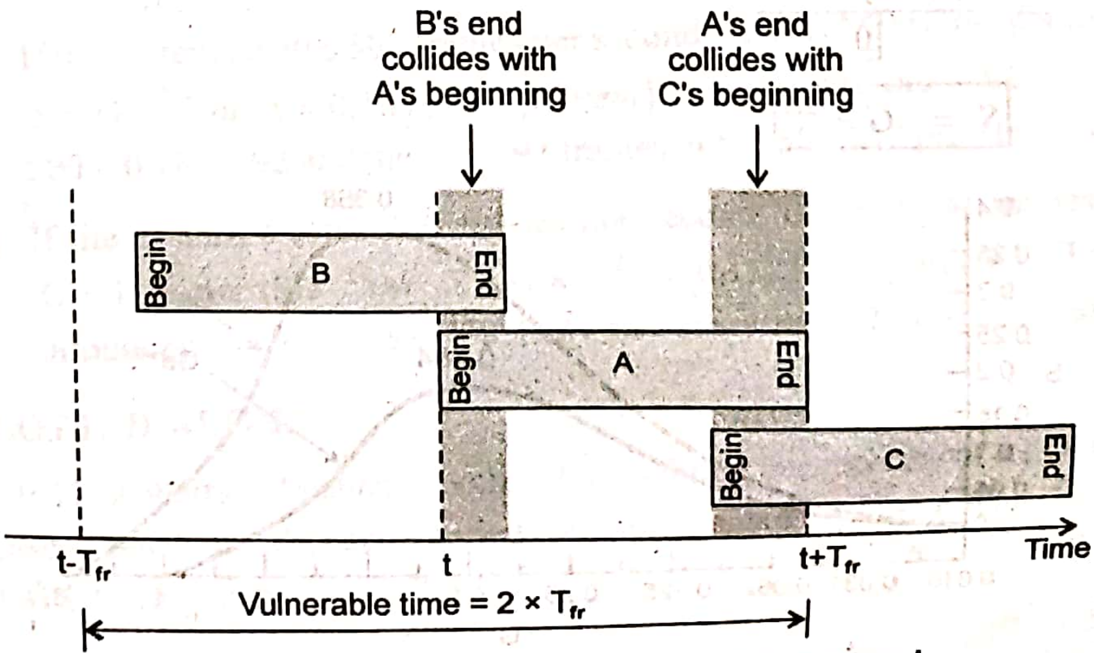


Fig. 3.18 : Vulnerable time for pure ALOHA protocol

Pure ALOHA vulnerable time = $2 \times T_{fr}$.

Example 3.2 : A pure ALOHA network transmits 200 bit frames on a shared channel of 200 Kbps. What is the requirement to make this frame collision-free ?

Solution : Average frame transmission time T_{fr} is 200 bits/200 kbps **or 1ms**. The vulnerable time is 2×1ms = 2ms. This means no station should send later than 1ms before this station starts transmission and no station should start sending during the one 1ms period that this station is sending.

Throughput : To find the probability that there is no collision with the reference frame we follow this approach. Let S be the arrival rate of new frames to the system in units of frames/X seconds. We assume that all frames eventually make it through the system, so S also represents the throughput of the system. But the actual arrival rate to the system consists of new arrivals and retransmissions. Let G be the total arrival rate in units of frames/X seconds. G is also called the total load. Again an assumption is made that the back off algorithm spreads the retransmissions so that frame transmissions, new and repeated are equally likely to occur at any instant in time. Thus the number of frames transmitted in a time interval has a Poisson distribution with average number of arrivals of 2G arrivals/2X seconds,

i.e. $$P(k \text{ transmission in } 2\,X \text{ seconds}) = \frac{(2G)^k\, e^{-2G}}{\underline{|k}} \quad k = 0,1,2$$

The throughput S is equal to the total arrival rate G times the probability of a successful transmission, i.e.

$$S = GP(\text{no collision})$$

$$= GP(0 \text{ transmissions in } 2X \text{ seconds})$$

$$= \frac{G(2G)^0\, e^{-2G}}{\underline{|0}}$$

$$\boxed{S = G\,e^{-2G}}$$

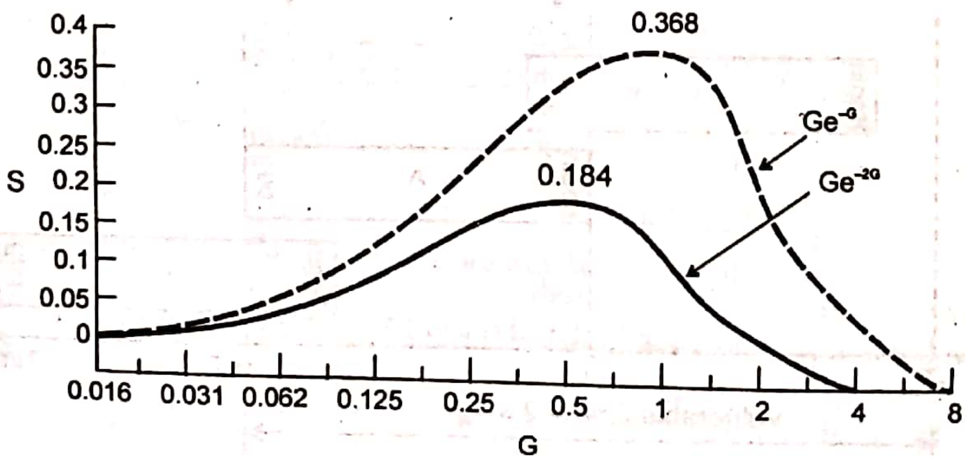


Fig. 3.19 : Throughput S versus load G

Figure (3.19) shows a graph of S versus G. Starting with small G, S increases and reaches a peak value at G = 0.5 of S = 1/2e and then declines back towards 0. For a given values of S, say S =0.05, there are two associated values of G. Thus it explains that system has two modes : one associating a small value of G with S i.e. $S \approx G$ which means low contention when all frames arriving are successfully transmitted and another associating a large value of G with S i.e. G >>S when many stations are backlogged. Values of S beyond 1/2e are not attainable so pure ALOHA can achieve **maximum throughput of 1/2e or 18.4 percent.** This maximum value occurs at $G = \frac{1}{2}$ which corresponds to a total arrival rate of exactly one frame per vulnerable period, as two or more arrivals in a valnerable period result in a collision. Thus if a station generates only one frame in its vulnerable time (and no other station generates a frame during this time), the frame will reach its destination successfully.

Example 3.3 : A pure ALOHA network transmits 200 bit frames on a shared channel of 200 kbps. What is the throughput if the system (all stations together) produces

(a) 1000 frames per second.

(b) 500 frames per second.

(c) 250 frames per second.

Solution : The frame transmission time is $200\ bit/200\ kbps$ or 1ms.

(a) If the system creates 1000 frames per second, it gives G = 1 frame/1ms ($\because$ G is average number of frames generated by system during one frame transmission time (i.e. 1ms)). So $S = G \times e^{-2G}$ or $S = 0.135\,(13.5\ percent)$. This means that throughput is $1000 \times \frac{13.5}{100} = 135$ frames. Only 135 frames out of 1000 will probably survive.

(b) If the system creates 500 frames per second, this is (1/2) frame per millisecond i.e. G= ½. $S = Ge^{-2G}$ or $S = 0.184\,(18.4\ percent)$. This means that the throughput is $500 \times 0.184 = 92$ and that only 92 frames out of 500 will probably survive.

(c) If the system creates 250 frames per second, this is (1/4) frames per millisecond i.e. $G = 1/4$. In this case $S = G \times e^{-2G} = 0.152\,(15.2\ percent)$. This means that the throughput is $250 \times 0.152 = 38$. Only 38 frames out of 250 will probably survive.

(B) SLOTTED ALOHA

In 1972, Roberts published a method for doubling the capacity of ALOHA system by dividing time into discrete intervals, each interval corresponding to one frame. The assumptions made for the slotted ALOHA protocol were following.

- Time is divided into slots of size of one frame.
- Nodes start to transmit frames only at the beginning 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.

Pure ALOHA has a vulnerable time of $2 \times T_{fr}$ because any station may send soon after another station has started or soon before another station has finished.

But slotted ALOHA improves the efficiency since time is divided into slots of T_{fr} seconds & stations can transmit only at the beginning of the time slot. Figure (3.20) shows an example of frame collisions in slotted ALOHA.

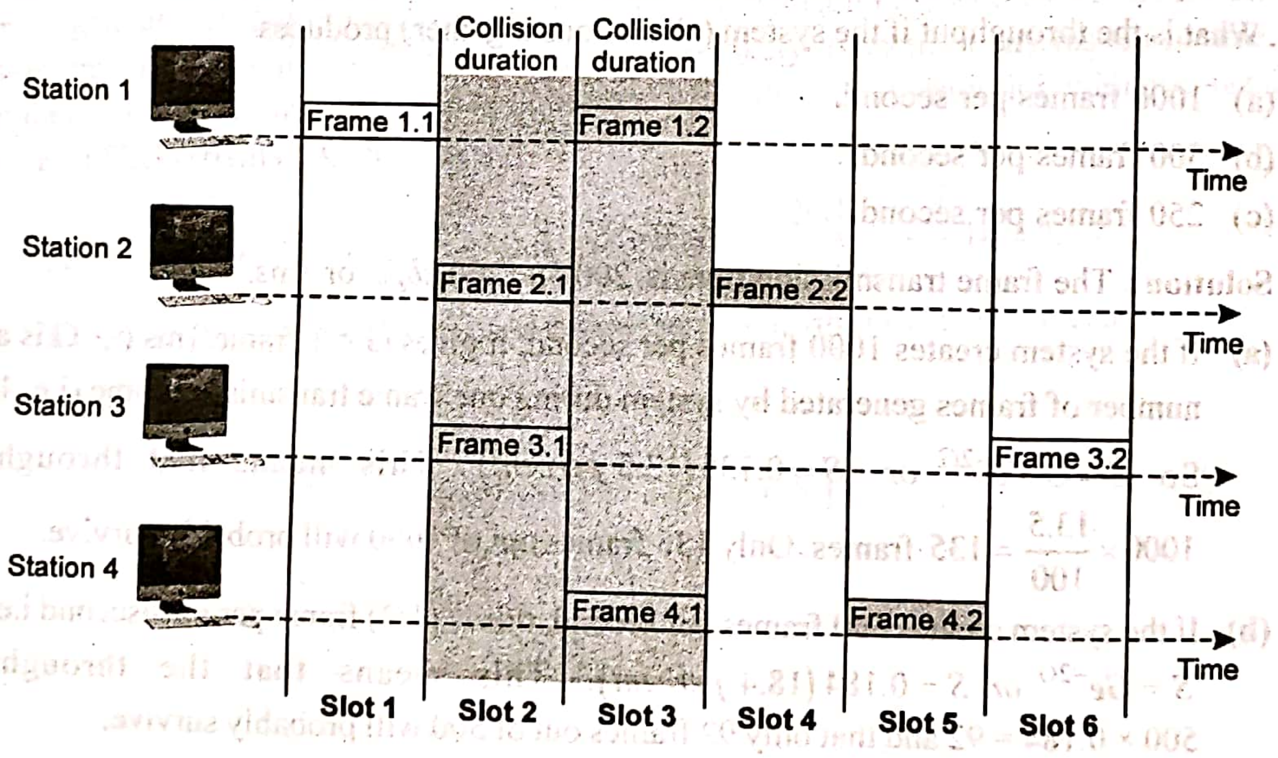


Fig. 3.20 : Frames in a slotted ALOHA network

So since a station must wait to transmit until the beginning of next time slot, another station which started earlier has already finished transmitting, thus avoiding collision. But there is still the probability of collision if two stations try to send at the beginning of same time slot. However the vulnerable time is now reduced to one-half i.e. equal to T_{fr}.

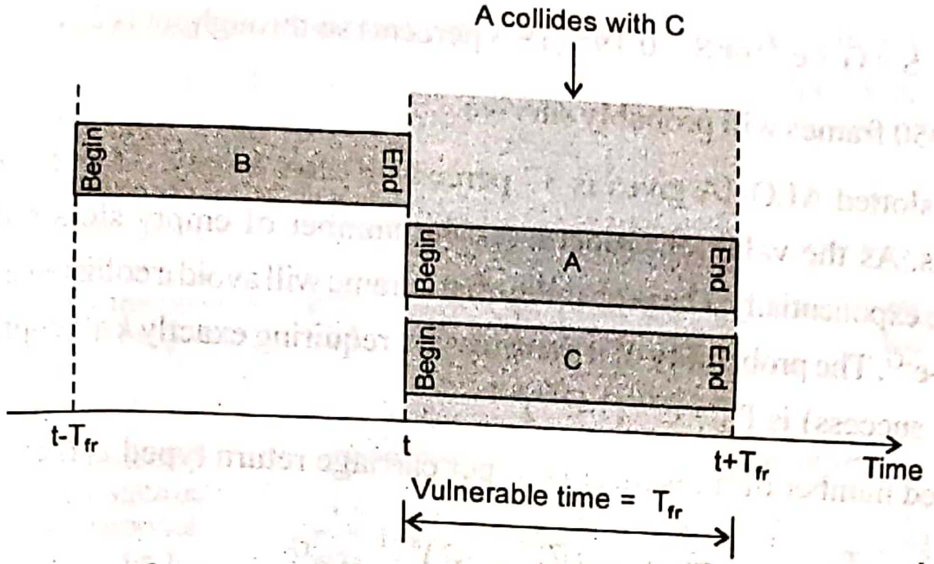


Fig. 3.21 : Vulnerable time for slotted ALOHA protocol

This figure (3.21) shows that the vulnerable time for slotted ALOHA is one-half that of pure ALOHA.

Throughput : Since vulnerable period is now halved, the probability of no other traffic during the same slot is e^{-G} which gives the average number of successful transmissions for slotted ALOHA as, $S = G \times e^{-G}$. The maximum throughput S_{max} is 0.368 when G = 1. It means that if a frame is generated during one frame transmission time then 36.8 percent of these frames reach their destination successfully, because the vulnerable time is equal to the frame transmission time.

Therefore, if a station generates only one frame in this vulnerable time and no other station generates a frame during this time, the frame will reach its destination successfully.

Example 3.4 : A slotted ALOHA network transmits 200 bit frames using a shared channel with a 200 kbps bandwidth, find the throughput if the system (all stations together) produces.

(a) 1000 frames per second.

(b) 500 frames per second.

(c) 250 frames per second

Solution : The frame transmission time is 200 bits/200 kbps or 1ms.

(a) In this case G is 1 so $S = G \times e^{-G}$ or S=0.368 (36.8 percent) i.e. throughput is $1000 \times 0.38 = 368$ frames. Only 368 out of 1000 frames will probably survive.

(b) G is $\frac{1}{2}$ so $S = G \times e^{-G}$ or S = 0.303 (30.3 percent). Thus throughput is $500 \times 0.303 = 151$. Only 151 out of 500 frames will probably survive.

(c) G is $\frac{1}{4}$. $S = G \times e^{-G}$ or $S = 0.195$ (19.5 percent) so throughput is $250 \times 0.195 = 49$. Only 49 out of 250 frames will probably survive.

So the best slotted ALOHA gives is 37 percent of slots empty, 37 percent successes and 26 percent collisions. As the value of G increases the number of empty slots reduces,& number of collisions increase exponentially. The probability that a frame will avoid a collision is e^{-G} & so probability of a collision is $1-e^{-G}$. The probability of a transmission requiring exactly k attempts (i.e., k-1 collisions followed by one success) is $P_k = e^{-G}(1-e^{-G})^{k-1}$

The expected number of transmission, E per carriage return typed is then

$$E = \sum_{k=1}^{\infty} kP_k = \sum_{k=1}^{\infty} k\, e^{-G}\left(1-e^{-G}\right)^{k-1} = e^{G}$$

Thus E depends exponentially on G which means that small increase in channel load (G) can drastically reduce its performance.

3.5.2 Carrier Sense Multiple Access (CSMA)

In both slotted and pure ALOHA, a node's decision to transmit is made independently of the activity of the other nodes attached to the broadcast channel. A node neither pays attention to whether another node is transmitting when it begins to transmit, nor stops transmitting if another node begins to interfere with its transmission. So it results in a highly collided network. To avoid this congestion some rules are introduced as :

Listen before speaking :

- This is called **carrier sensing** in networking where a node listens to the channel before transmitting. If a frame from another node is currently being transmitted into the channel, it waits a random amount of time and then again senses the channel. If channel is sensed to be idle, the node begins transmission, otherwise it waits another random amount of time and repeats this process. This rule is used in **carrier sense multiple access (CSMA)**.
- **If someone else begins talking at the same time, stop talking** : This is called as **collision detection** in networking where a transmitting node listens to the channel while it is transmitting. If it detects another node is transmitting an interfering frame, it stops transmitting and uses some protocol to determine when it should next attempt to transmit. This rule is used in CSMA with collision detection (CSMA/CD).

Here we will discuss **carrier sensing multiple access** technique. The wastage of transmission bandwidth because of frame collisions can be reduced by sensing the medium for presence of a carrier signal from other stations.

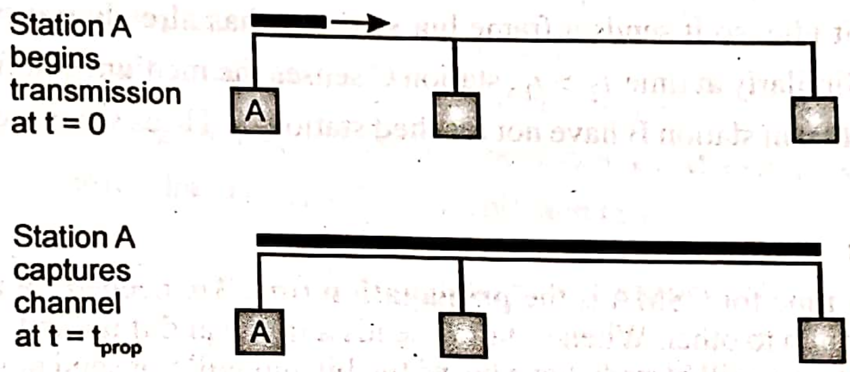


Fig. 3.22 : CSMA random access scheme

In the figure (3.22) shown above, at time t = 0, station A begins transmission at one extreme end of a broadcast medium. As the signal propagates through the medium, stations become aware of the transmission from station A. At the time $t = t_{prop}$, the transmission from station A reaches the other end of the medium. By this time all stations are aware of the transmission from station A thus it captures the channel because no other station will transmit thereafter.

CSMA can reduce the possibility of collision, but it cannot eliminate it. The reason is shown in figure (3.23) a space and time model of a CSMA network.

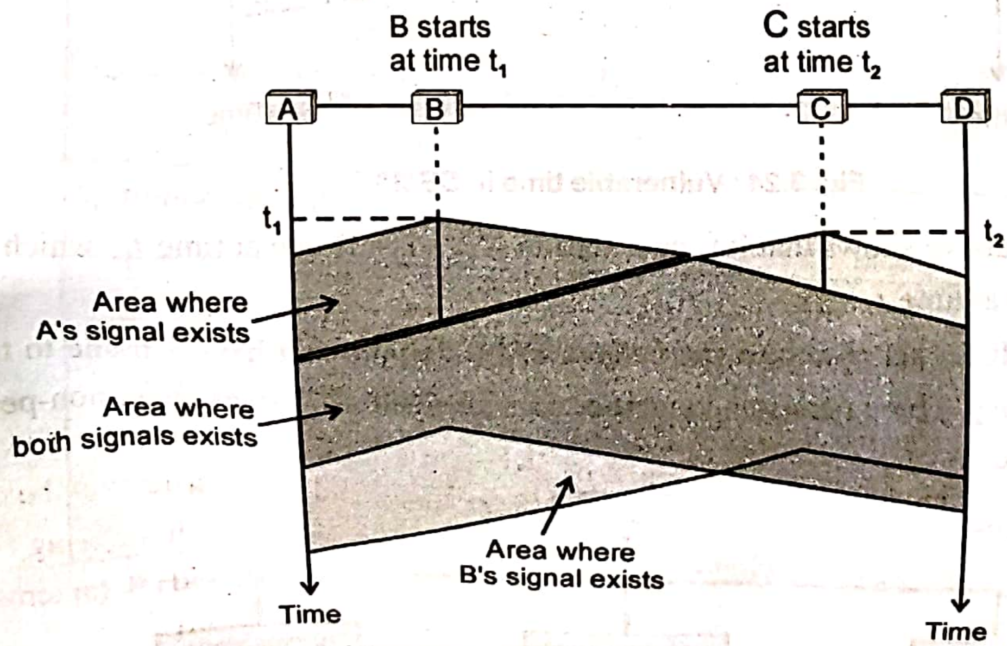


Fig. 3.23 : Space/time model of the collision in CSMA

Stations are connected to a shared medium. The possibility of collision still exists because of propagation delay; when a station sends a frame, it still takes time for the first bit to reach every station and for every station to sense it. As the station may sense the medium and find it idle. Only because the first bit sent by another station has not yet been received. At time t_1 station B senses the medium and finds it idle, so it sends a frame but station A has already transmitted which not yet reached station B. Similarly at time $t_2 > t_1$, station C senses the medium and finds it idle because at this time the first bit from station B have not reached station C. Thses signals collide & thus frames are destroyed.

Vulnerable Time :

The vulnerable time for CSMA is the **propagation time T_P**, needed for a signal to propagate from one end of medium to other. When a station sends a frame and if its first bit reaches the end of the medium, every station will already have heard the bit and will not send any frame.

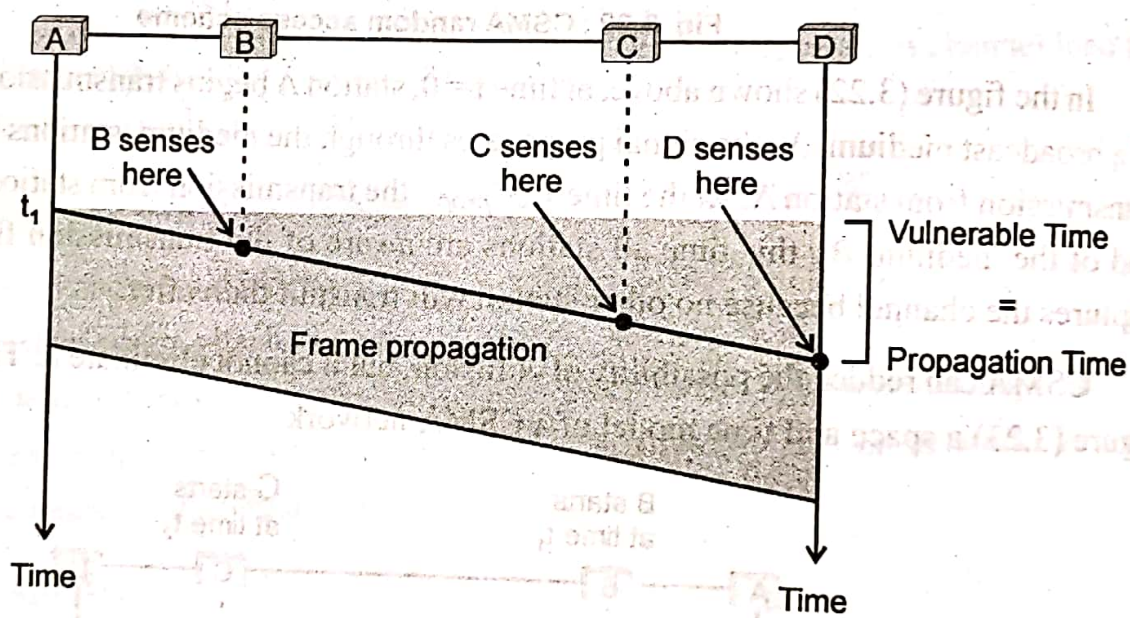
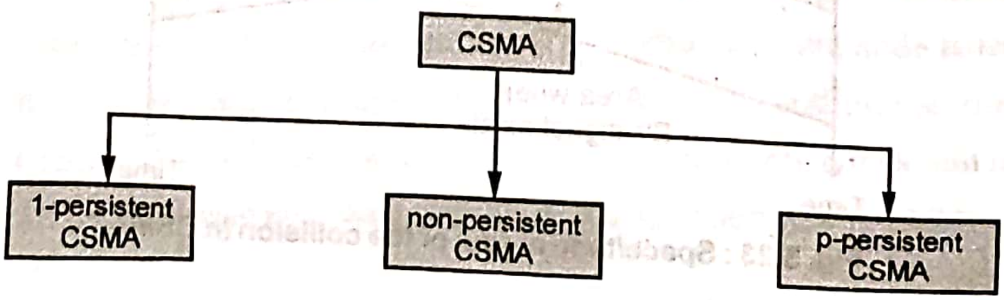


Fig. 3.24 : Vulnerable time in CSMA

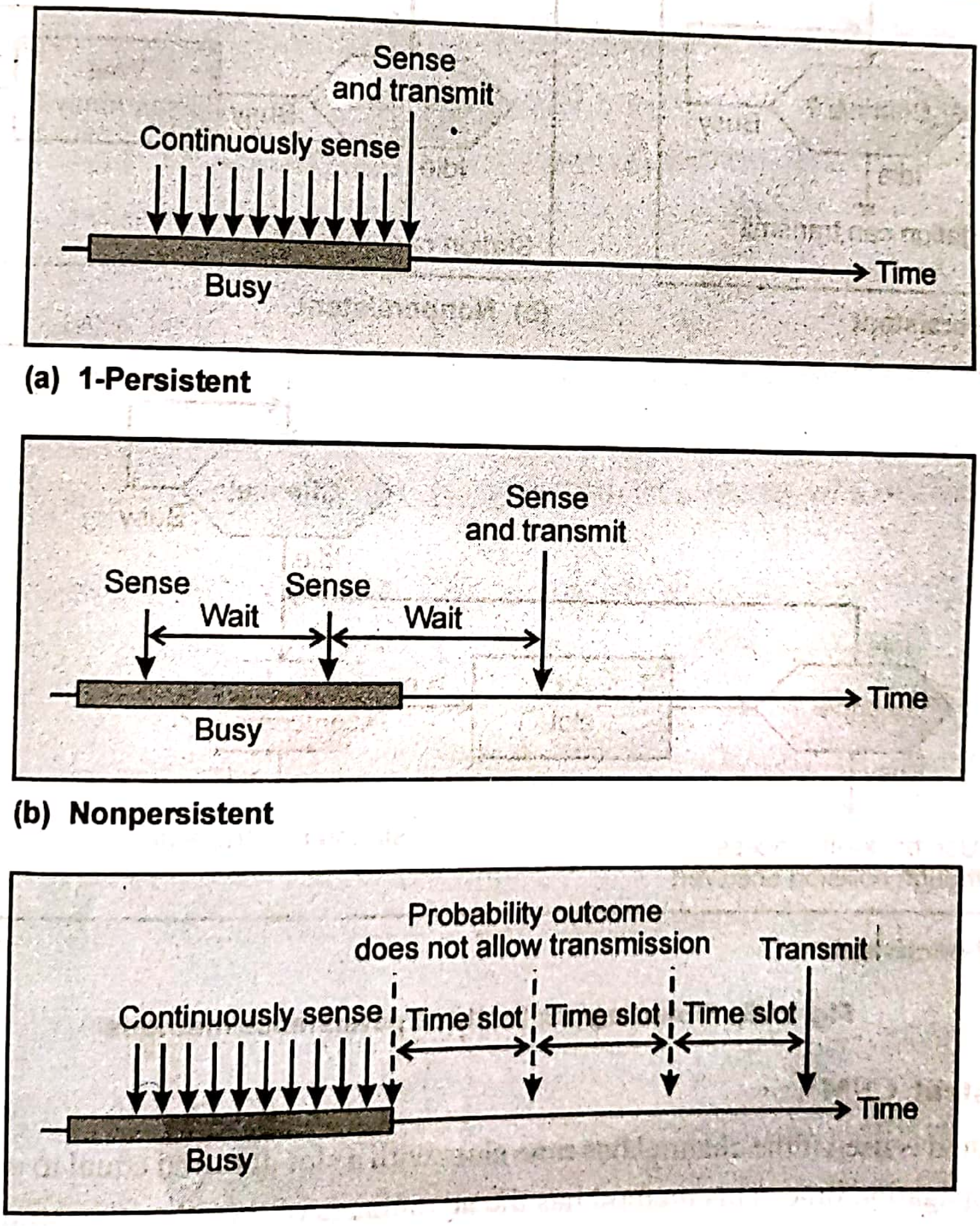
The figure (3.24) above shows that leftmost station A sends a frame at time t_1, which reaches the rightmost station D at time $t_1 + T_P$.

CSMA schemes differ according to the behaviour of stations that have a frame to transmit when the channel is busy. Three persistence methods are devised as : 1-persistent, non-persistent and p-persistent method.



(a) 1-Persistent CSMA

In 1-persistent CSMA stations with a frame to transmit sense the channel. If channel is busy, they sense the channel continuously waiting until it becomes idle. As soon as the channel is sensed idle, they transmit their frames. But if more than one station is waiting, a collision will occur. Also stations that have a frame arrive with T_P of the end of preceding transmission will also transmit & possibly be involved in collision. Stations that are involved in collision perform the backoff algorithm to schedule the time for resensing the channel. As in 1-persistent CSMA stations attempt to access the medium as soon as possible it has a relatively high collision rate.



(a) 1-Persistent

(b) Nonpersistent

(c) P-persistent

Fig. 3.25 : Behavior of three persistence methods

(b) Non Persistent CSMA

In this method, a station that has a frame to send senses the line. If the line is idle, it sends immediately. If the line is not idle, station immediately run the back off algorithm and wait a random amount of time before sensing the line again. It reduces the chance of collision because it is unlikely that two or more stations will wait the same amount of time and retry to send simultaneously. But this method reduces the efficiency of the network because medium remains idle where there may be stations with frames to send since they all may be waiting for a random time before sensing the channel to be idle.

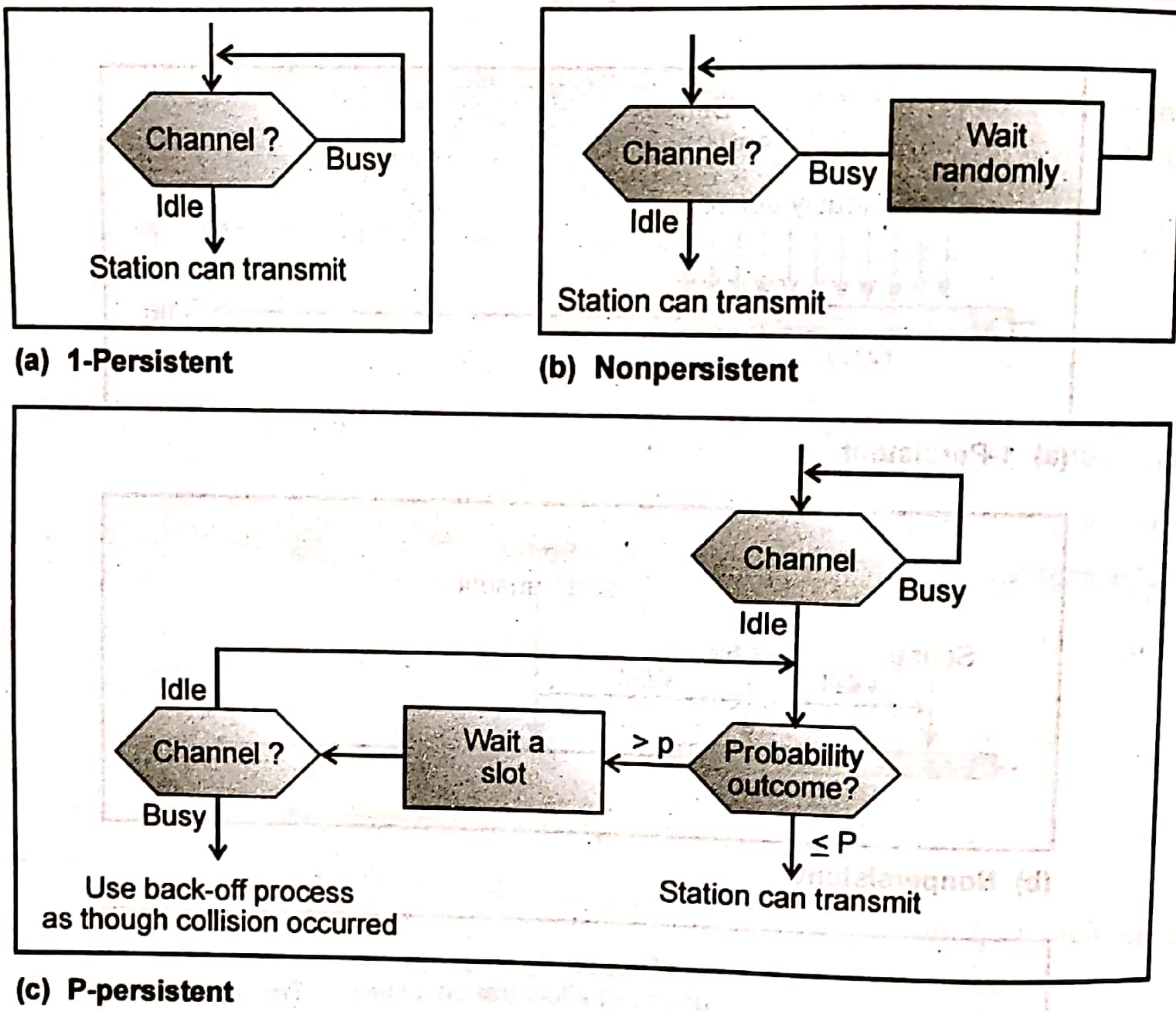


Fig. 3.26 : Flow diagram for three persistence methods

(c) P-Persistent CSMA

This method is used if the channel has time slots with a slot duration equal to or greater than the maximum propagation time. This method has the advantages of the other two strategies. It reduces the chance of collision and improves efficiency. If the channel is busy they persist with sensing until the channel becomes idle. After the station finds the line idle it follows these steps :

(i) with probability P, the station sends its frame.

(ii) with probability q = 1-P, the station waits for the beginning of next time slot and checks the line again.

- ➲ if line is idle, it goes to step (1).
- ➲ if line is busy it acts as a collision has occured and uses the back off procedure.

This behaviour spreads out the transmission attempts by the stations that have been waiting for a transmission to be completed and hence increase the possibility that a waiting station successfully captures the medium.

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

The CSMA schemes improve over the ALOHA schemes by reducing the vulnerable period from one or two frame transmission times to a single propagation delay T_P. In both ALOHA and CSMA schemes, collisions involve entire frame transmissions. If a station can determine whether a collision is taking place then the amount of wasted bandwidth can be reduced by aborting the transmission when a collision is detected. The **carrier sensing multiple access with collision detection (CSMA-CD)** scheme uses this approach as to detect the collision while transmitting by sensing the line simultaneously during transmission.

Let us take an example of transmission of first bits by the two stations involved in the collision.

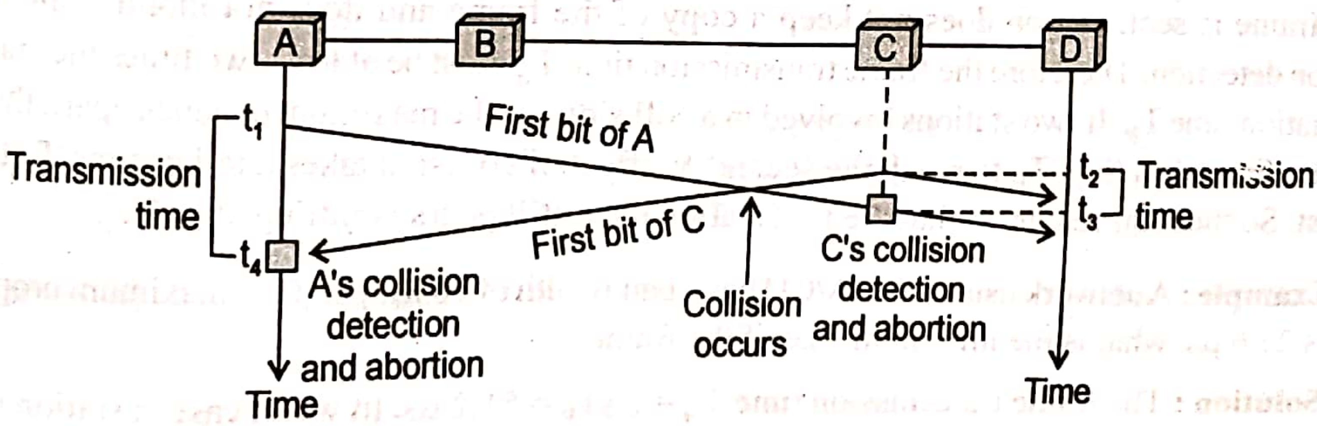


Fig. 3.27 : Collision of the first bit in CSMA/CD

At time t_1, station A has executed its persistence procedure and starts sending the bits of its frame. At time t_2, station C has not yet sensed the first bit sent by A. Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and right. The collision occurs sometimes after time t_2. Station C detects a collision at time t_3 when it receives the first bit of A's frame. Station C immediately aborts transmission. Station A detects a collision after time t_4 when it receives the first bit of C's frame ; it also immediately aborts the transmission. A transmits for duration t_4-t_1 and C for duration t_3-t_2. For the protocol to work, the length of any frame divided by the bit rate in this protocol must be more than either of these durations. At time t_4, transmission of A's frame though incomplete is aborted and at time t_3, the transmission of C's frame, though incomplete is aborted.

Seeing the collision of first bits of two frames we can show a more complete graph in figure (3.28).

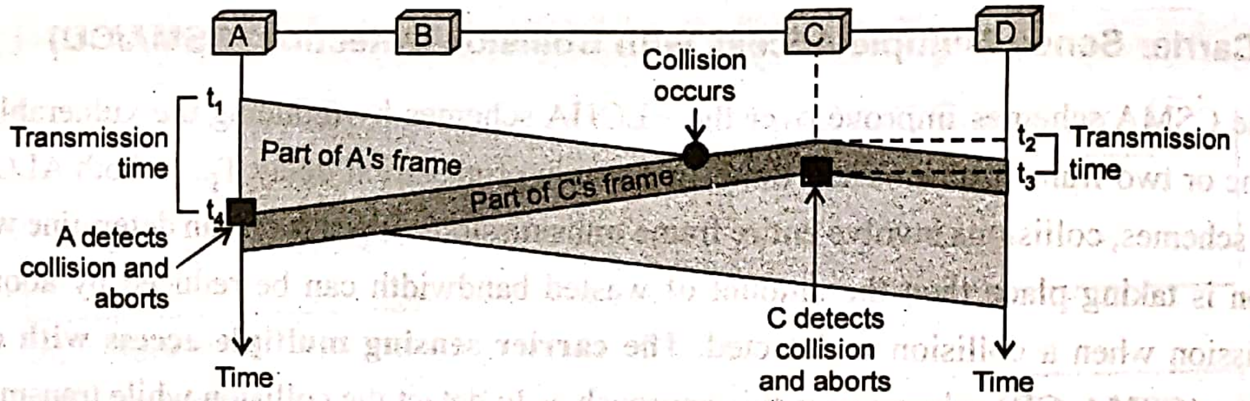


Fig. 3.28 : Collision and abortion in CSMA/CD

Minimum frame Size :

For CSMA/CD to work, a restriction on frame size is needed. Before sending the last bit of the frame, the sending station must detect a collision, if any, and abort the transmission because once the entire frame is sent, station does not keep a copy of the frame and does not monitor the line for collision detection. Therefore the frame transmission time T_{fr} must be at least two times the maximum propagation time T_P. If two stations involved in a collision are the maximum distance apart, the signal from the first takes time T_P to reach the second & effect of collision takes another time T_P to reach the first. So the requirement is that the first station must still be transmitting after $2T_P$.

Example : A network using CSMA/CD has a bandwidth of 10 Mbps. If the maximum propagation time is 25.6 μs, what is the minimum size of the frame ?

Solution : The frame transmission time $T_{fr} = 2\times T_P = 51.2$ **μs.** In worst case, a station needs to transmit for a period of 51.2 μs to detect the collision. The minimum size of the frame is 10Mbps×51.2 μs = 512 bits or 64 bytes.

Procedure : In CSMA-CD a station with a frame first senses the channel and transmits if the channel is idle. If the channel is busy, the station uses one of the possible strategies of CSMA i.e. station can persist, back off immediately or persist and attempt transmission with probability P. If a collision is detected during transmission, then a short jamming signal is transmitted to ensure that other stations know that collision has occurred before aborting the transmission and the backoff algorithm is used to schedule a future resensing time. In CSMA/CD transmission and collision detection is a continuous process unlike ALOHA in which entire frame is transmitted and then wait for an acknowledgement. In CSMA/CD line is constantly monitored to detect one of the two conditions : either transmission is finished or a collision is detected. Either event stops transmission. If collision is not detected transmission is completed i.e. entire frame is transmitted and if collision occurs a short **jamming signal** is sent which makes all stations to stop transmission immediately. This entire process is explained with this flow diagram.

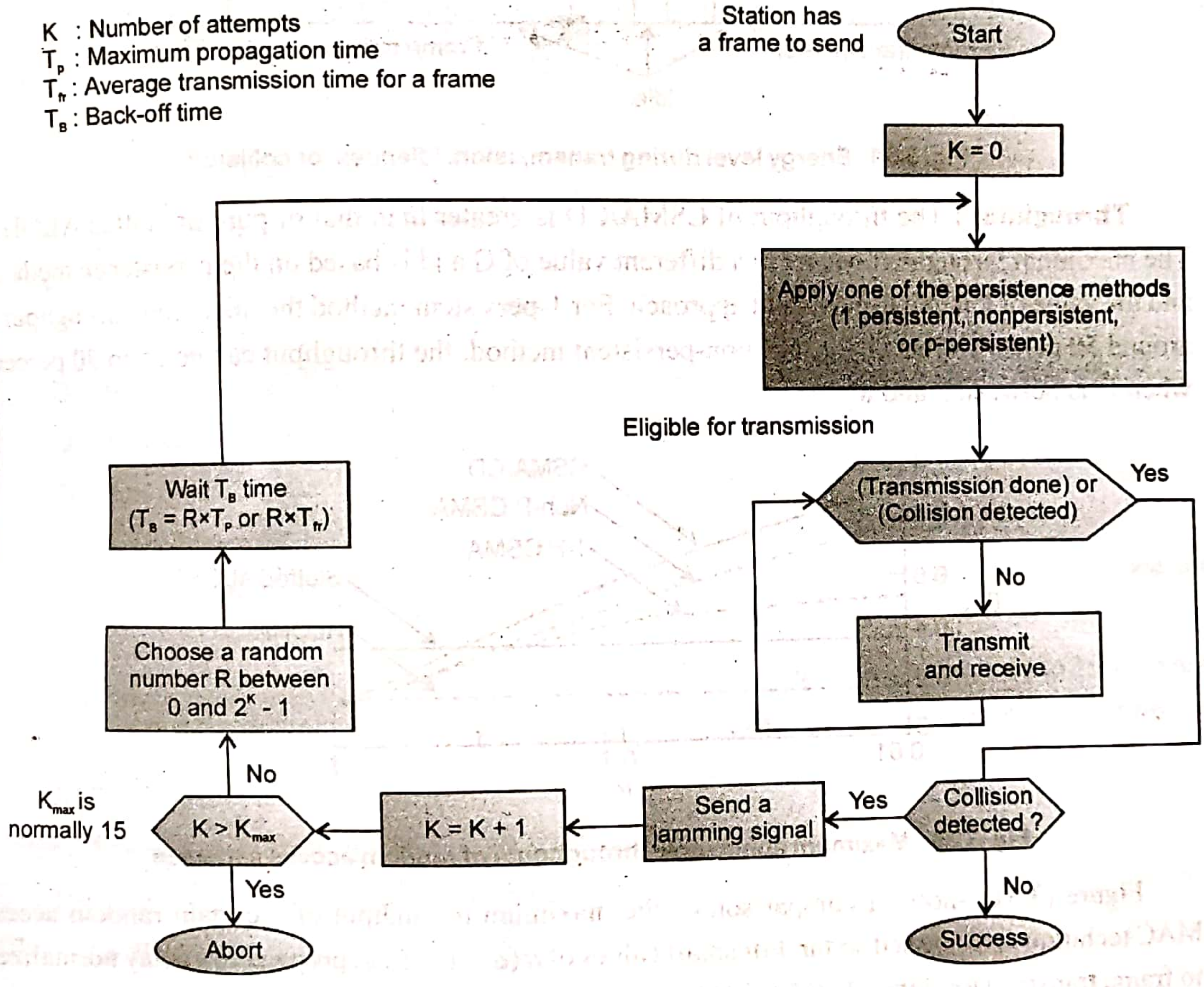


Fig. 3.29 : Flow diagram for the CSMA/CD

Energy Level :

Level of energy in a channel can have three values : zero, normal or abnormal.

- At **zero level** channel is idle.
- At the **normal level,** station has successfully captured the channel and is sending its frame.
- At **abnormal level** there is a collision & the level of energy is twice the normal level.