BCS5O2 – COMPUTER NETWORKS

Faculty:

Prof. Ashok Herur

ashok. herur @eastpoint. ac. in

Module 2

Data Link Layer (DLL or L2)

Main topics in Module 2

Data link layer:

- Functions of DLL
- Framing & Synchronization
- Error detection and correction methods
- Elementary data link protocols, Sliding window protocols.

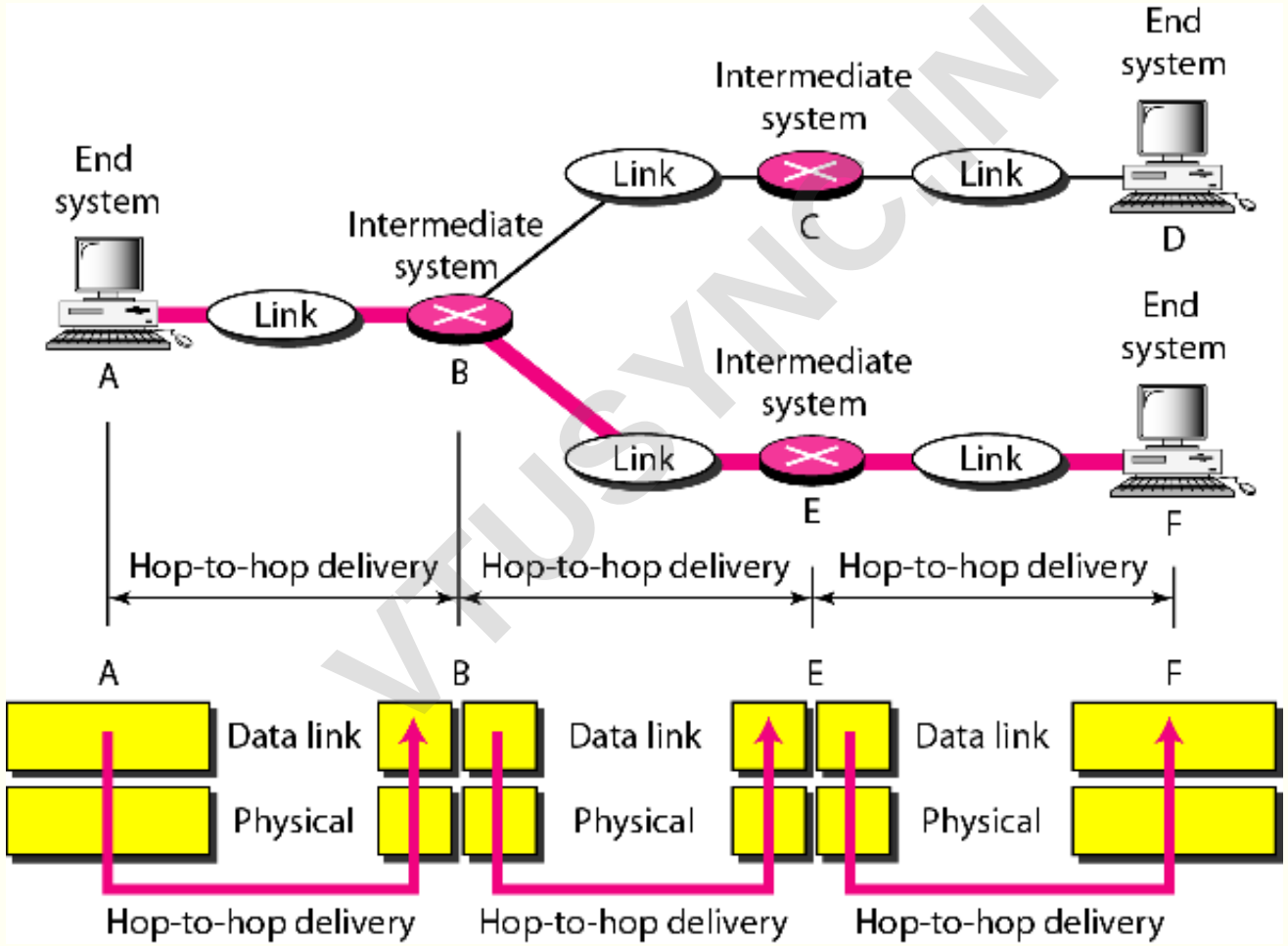
Medium access control sub-layer:

- Multiple access protocols.
- MAC in common LANs (Ethernet, Token Ring, Wireless LAN)

Functions of the Data Link Layer

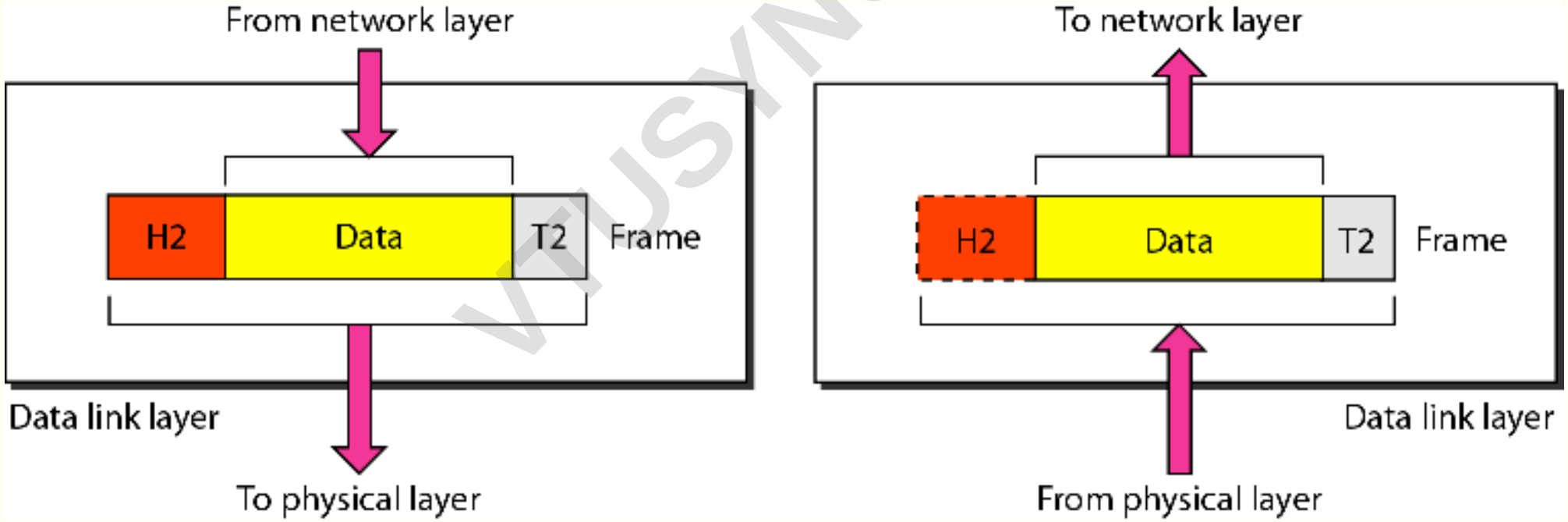
- The data link layer is responsible for moving frames from one node to the next one.
  - Framing & Synchronization
  - Error control over a link
    - Error Detection and Correction codes
  - Flow control over the links

Functions of the Data Link Layer



Framing

- It receives the “Packet” from the Network layer and adds a Header and a Trailer to form a “Frame”



Framing

- A typical frame consists of the following overheads:
- Synchronization bits (typically alternating 1's and 0's) right at the beginning;
- Special Patterns to indicate the start and end of the frame;
- Redundant bits to help the receiver to deal with transmission errors;
- Physical (MAC or Medium Access Control) addresses of the two devices across the link.

Framing - Synchronization

- The receiving device should be in perfect sync with the sending device.
  - This means it should know the start of a bit duration and the exact duration of a bit.
  - If not perfectly synchronised, they may lose track, due to a small difference (drift) in the clock rates of the transmitting and receiving nodes.
  - This will lead to bit-gain or bit-loss and even complete loss of the message.

Framing – Start and End of the frame

- Special bit patterns, called Flags, are used to indicate the beginning and the end of each frame.
- The popular HDLC (High-level Data Link Control) protocol used the same pattern of 01111110 as the Start and End flags.
- One problem here is that the Flag pattern may occur in the data since it is a bit-oriented protocol (non-text data like picture, audio, video, etc)
- This problem can be overcome by "bit-stuffing".
  - Sender scans the data and inserts an extra '0' after 5 consecutive 1's.
  - Receiver removes the 0 after 5 consecutive 1's.

Framing – Start and End of the frame

- Another protocol that is used is the Point-to-Point protocol (PPP).
- This also uses the same pattern of 01111110 as the Start and End flags.
- Unlike HDLC, this protocol supports both byte-oriented and bit-oriented messages.
- It defines the:
  - frame format of the data to be transmitted.
  - procedure of establishing link between two points and exchange of data.
  - method of encapsulation of network layer data in the frame.
  - authentication rules of the communicating devices.

Framing

- Framing also involves computing the checksum (parity) bits and putting them in the frame, to enable the receiver node to check for transmission errors.
- It also includes the Physical (MAC) addresses of the sending node and the receiving node.
  - The MAC address of a network device is a permanent address that is hardwired during its manufacture.
  - A MAC address comprises of six groups of two hexadecimal digits (48 bits), separated by hyphens or colons.
    - Example - 00:0A:89:5B:F0:11
- The IP address is used by routers to bring the packet to the right network while the MAC address is used to deliver the frame to the device within the network.

Error detection and correction codes

Error control

- When error detection codes are used, the receiver can only find out that something is wrong but will not be able to say which bit(s) are incorrect.
  - The only way out, is to ask the transmitter to retransmit the frame.
  - Example codes: Character Parity, CRC
- When error correction codes are used, the receiver can pin-point the erroneous bits and hence correct them.
  - However, the amount of redundancy (overhead) here is much more than in error detection codes.
  - Example codes: Hamming code, Convolutional code.
- Error correction codes are also called Forward Error Correction (FEC) codes.

Error Detection codes versus Correction codes?

- When the error rate is very low (optical fibre), error detection codes are better since the number of retransmissions will be very few.

- However, in real-time applications (process control, and even speech / video too), we have to use error correction codes (if at all we wish to incorporate error control).
  - Do we need error control in audio / video?

- Transmission errors can be isolated (once in a while) or can occur in bursts; The latter type are almost impossible to correct (though they can be detected).

Character Parity coding

- An Even- or an Odd-parity scheme can be chosen.
- The number of 1's after adding the encoding (Parity) bit should be Even/Odd.
- The code of 'A' : 0100 0001 will be
  - 0100 0001 0, in case of Even-parity scheme
  - 0100 0001 1, in case of Odd-parity scheme
- Receiver checks for the required parity.
- Overhead is 12.5% (one extra bit for 8 message bits).
- Used for asynchronous transmission of characters (typically at a low bit-rate)

Cyclic Redundancy Check (CRC) code

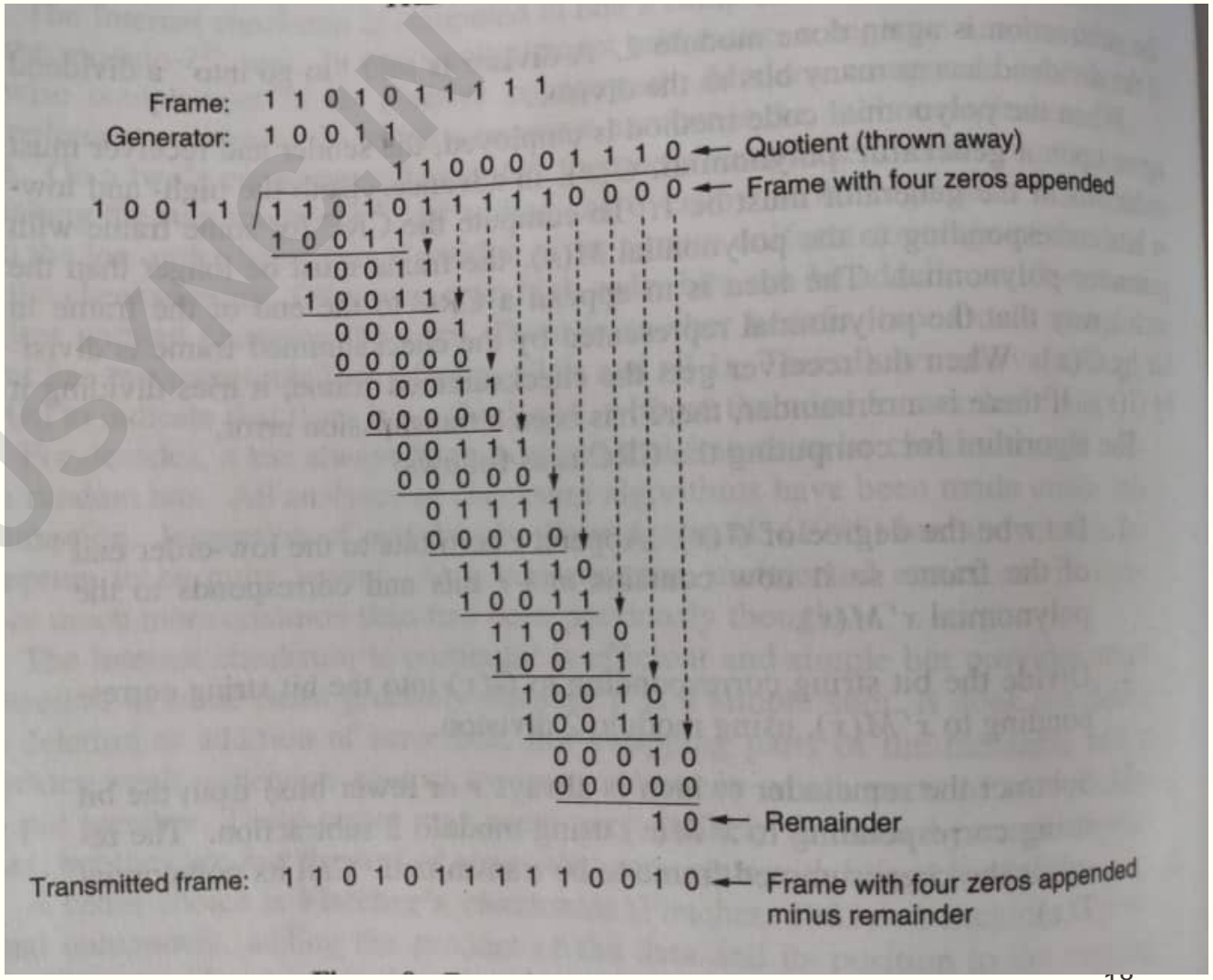
- These are powerful error detection codes, used for synchronous transmission of large frames at high bit-rates.
  - At high bit rates, the same impulsive noise will affect many bits and hence the length of a burst error will be more.
- The Checksum is computed from the message using a Generator Polynomial, and is then appended to the message.
- The receiver does the same computation and verifies the Checksum.
- For example, CRC-32 will generate 32 checksum bits, and can detect the error even when 32 bits of a frame are corrupted.

CRC coding method – At the transmitter

- The coefficients of the Generator Polynomial (GP) are used to form a GP string.
- Let the degree of the GP be "N"
- Then, N zeroes are appended to the message.
- The appended message is then divided by the GP string, using Modulo-2 arithmetic.
  - No concept of Borrow during subtraction. Just XOR operation is done.
- The N-bit remainder is the Checksum. This is then appended to the original message and is then transmitted.
- The receiver does the same computation and verifies the Checksum.

CRC code – Example 1

- Compute the Checksum for the message, 1101011111, using a Generator Polynomial, $x^4 + x + 1$



CRC coding method – At the receiver

- The receiver divides the received bit sting by the same GP string used by the transmitter.
- If the remainder is Zero, it means that there are no errors. Else, it means that the frame has some errors.
- Note that this method can detect the presence of up to N errors.

CRC code – Example 2

- Example:

1. Compute the Checksum for the message, 10010, using a Generator Polynomial, $x^3 + x + 1$
2. Show how the receiver verifies the Checksum when there is no error.
3. Show how the receiver detects the presence of a 2-bit error.

Hamming code

- Hamming code is an error correction code, that can detect and correct single-bit errors.
- Hamming code uses a block parity mechanism; The data is divided into blocks, and parity is added to the block.
- The number of parity bits added to Hamming code is given by the formula $2^p \geq d + p + 1$, where p is the number of parity bits and d is the number of data bits.
- For example, if you wanted to transmit 8 data bits, the formula would be $2^4 \geq 8 + 4 + 1$. So 4 parity bits are required, for a total of 12 bits.
  - This code is called Hamming (12, 8)

Hamming code

- These parity bits occupy the 2^N bit positions, and are calculated by using standard equations. The other bit places are filled by the bits representing the character.
- Its transmission efficiency (code rate = k/n) goes up as the block size increases.
  - In Hamming(12, 8), the code rate (data bits / total) is only 0.67, while Hamming(255, 247) is 0.969.
  - However, as the block size increases, the chance of multiple errors in the block also increases, and hence it will not be effective.
- It has a larger overhead compared to Character parity method.
  - It needs 4 parity bits for each character (50% overhead) while Character parity method required just one parity bit (12.5% overhead).

Example: Hamming (11, 7)

- The 4 equations for the computation of parity bits are:
  - $P_1 + D_3 + D_5 + D_7 + D_9 + D_{11} = 0$ (Even parity)
  - $P_2 + D_3 + D_6 + D_7 + D_{10} + D_{11} = 0$
  - $P_4 + D_5 + D_6 + D_7 = 0$
  - $P_8 + D_9 + D_{10} + D_{11} = 0$
- The receiver computes the value of the 4 expressions (LHS only)
  - If all of them give a value of '0', it implies that there are no errors.
  - If not, take the value of the 4 expressions and write them down in the reverse order, and convert it to its decimal equivalent (let's say, it's E)
  - The erroneous bit is the one in the E^{th} bit position.

Example: Hamming (11, 7)

1. Compute the Hamming parity bits for the message, 1001101.
2. Show how the receiver verifies that there is no error.
3. Show how the receiver detects and corrects the error in the 10^{th} bit position (of the received bit stream).

Solution:

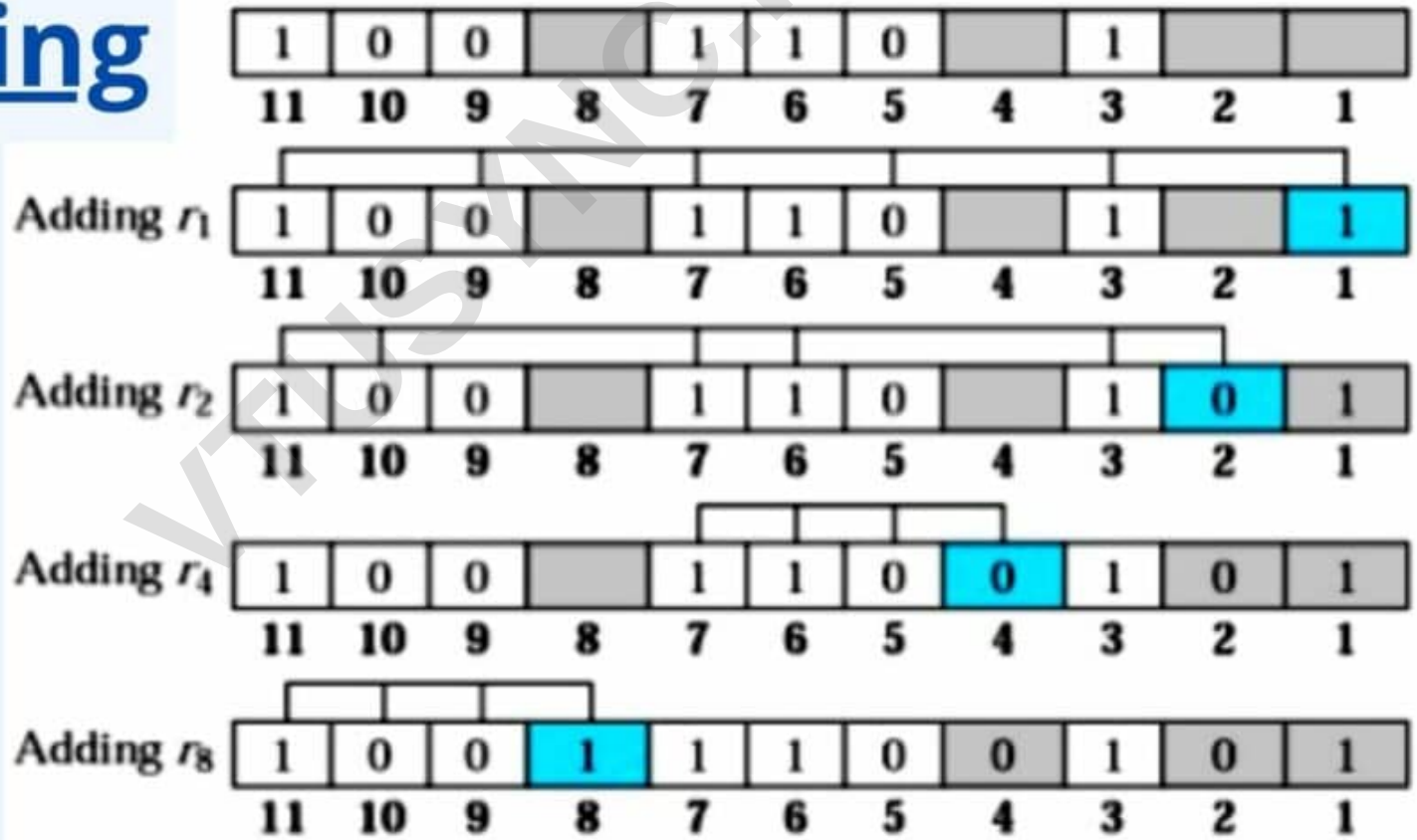
- Since the message is 7 bits long, we need 4 parity bits, as per the formula: $2^p \geq d + p + 1$.
- These parity bits will occupy the 1^{st}, 2^{nd}, 4^{th} and 8^{th} bit positions.

Example: Hamming (11, 7)

Data: 1001101 **Code:** 10011100101

Hamming Code

Example:



Example: Hamming (11, 7)

- The transmitted code is: 10011100101
- The received code (10^{th} bit in error): 11011100101

  - $P_1 + D_3 + D_5 + D_7 + D_9 + D_{11} = 0$
  - $P_2 + D_3 + D_6 + D_7 + D_{10} + D_{11} = 1$
  - $P_4 + D_5 + D_6 + D_7 = 0$
  - $P_8 + D_9 + D_{10} + D_{11} = 1$

- So, the number $1010_2 = 10$ (decimal) indicates that bit number 10 is incorrect. Hence, it is flipped from the received value of '1' to '0'.

Convolutional code

- Convolutional code is an error correction code, that can detect and correct multiple bit errors in a synchronous transmission frame.
- The computed parity bits depend not only on the current block but on the past few blocks as well.
- It is quite complex and requires a lot of computation.
- It is also called as Viterbi coding.

Sliding window protocols

ARQ protocols

- When an error occurs, there is a retransmission request when error detection codes are used.
- Common ARQ (Automatic Retransmission reQuest) protocols:
  - Stop-and-wait protocol
  - Go-back-N protocol
  - Selective Repeat protocol
- The last two protocols are classified as “Sliding window protocols”

Stop-and-wait protocol

- Transmitter transmits a frame and waits for the acknowledgement (ACK), which is sent by Receiver only if it receives the frame without any errors.
  - It does not transmit the next frame till it receives the ACK for the previous one.
- If ACK is not received within a specified time, the Transmitter times-out and retransmits the frame.
  - Results in a duplicate frame if ACK frame was lost.
  - Need sequence numbers (0,1) to deal with this situation.
- Main drawback is the huge waiting time, and consequently the very low rate of transmission (low utilization of bandwidth)

Stop-and-wait protocol

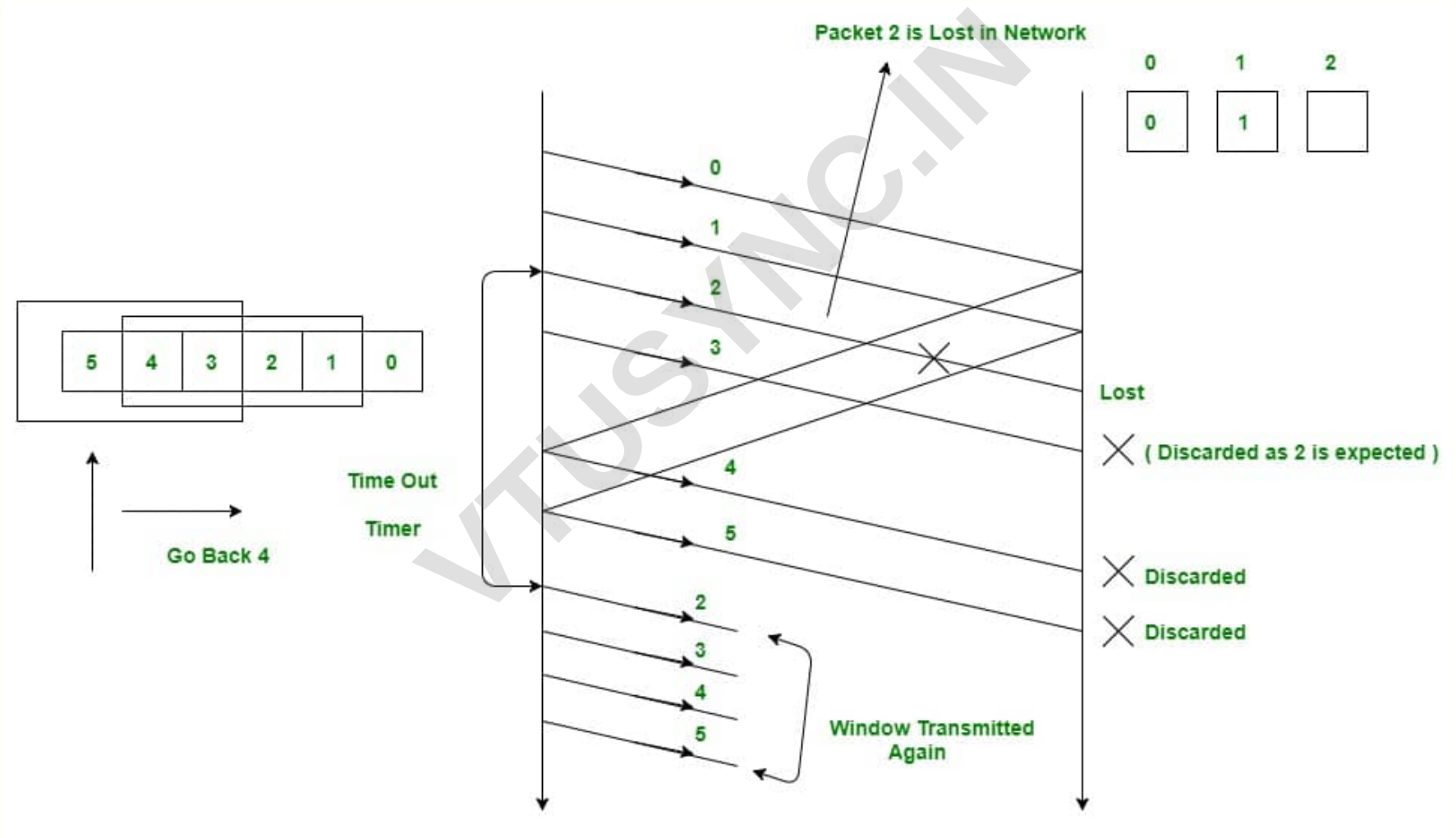
Sliding window protocols

- In these protocols, the sender is allowed to send up to N frames without waiting for the acknowledgement of the earlier frames, where 'N' is the window size.
- The sender maintains a set of sequence (frame) numbers corresponding to frames sent but as yet not acknowledged. These frames are said to fall within the 'sending window'
- Similarly, the receiver maintains a 'receiving window' corresponding to frames that it is permitted to accept.
- The two windows need not be of the same size.

Go-back-N protocol

- Sender's window size = N, Receiver's window size = 1
- The sender can send N frames without waiting for the acknowledgement, but the receiver will only receive the one it is expecting to receive next.
  - Subsequent frames, received after a faulty one, will be discarded till the retransmitted frame is received.
- Whenever a frame is faulty, it has to be therefore retransmitted along with all the subsequent frames that were sent earlier.
- No need to buffer frames at the receiver.

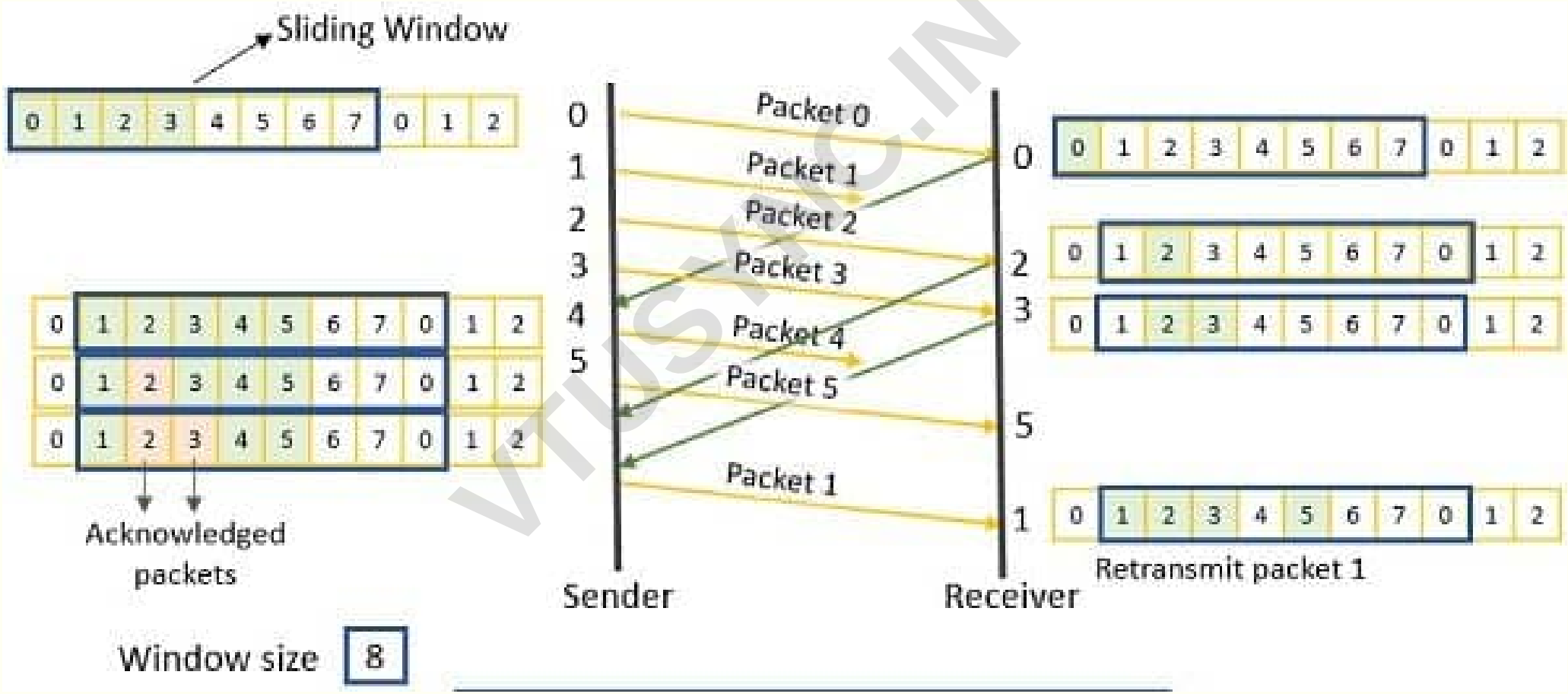
Go-back-N protocol (N_t = 4)



Selective Repeat Request protocol

- Sender's window and the Receiver's window are of the same size = N
- The sender can send N frames without waiting for the acknowledgement, while the receiver can receive up to N frames.
- Whenever a frame is faulty, only the faulty frame has to be retransmitted.
- The receiver buffers the correct frames, while waiting for the retransmission of the faulty frame.
- It then puts the frames in the correct sequence before passing them on to the Network layer.

Selective Repeat Request protocol (N = 8)



Medium Access Control (MAC) protocols

Channel allocation problem

- The Channel allocation problem is about how to allocate a single broadcast channel among many competing users.
- The network channel may be a single cable or optical fibre connecting multiple nodes, or a portion of the wireless spectrum.
- Channel allocation algorithms allocate the wired channels and bandwidths to the users, who may be base stations, access points or terminal equipment.
- The channel allocation can be Static or Dynamic.

Static Channel allocation

- In static channel allocation scheme, a fixed portion of the bandwidth is allotted to each user.
- For N competing users, the bandwidth is divided into N channels, and each portion is assigned to one user.
- In this allocation scheme, there is no interference between the users since each user is assigned a fixed channel.
- However, it is not suitable in case of a large number of users (varying with time) or when the traffic is bursty.
  - It results in an inefficient use of the available bandwidth.

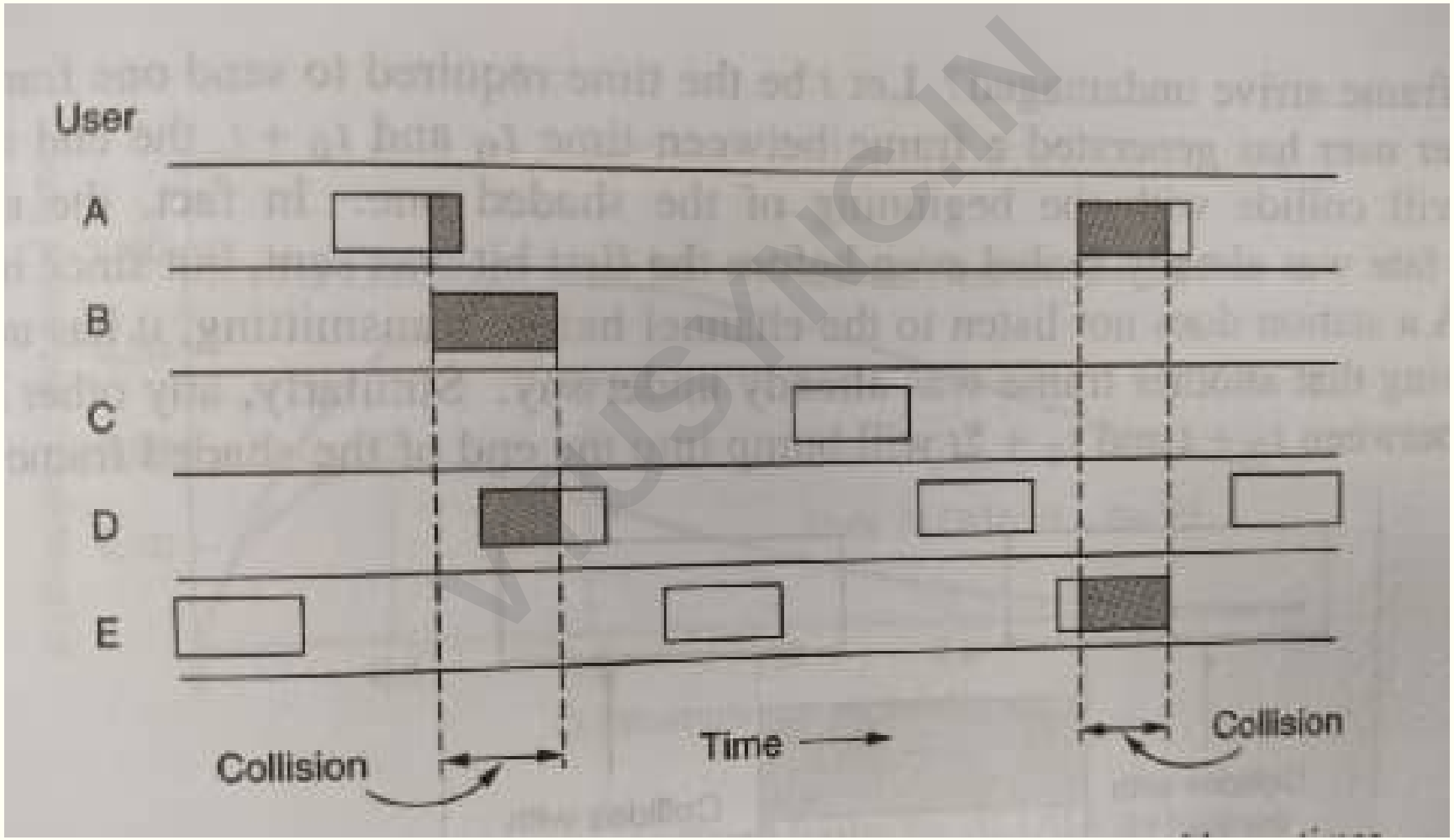
Dynamic Channel allocation

- In dynamic channel allocation scheme, users get to use the channels as and when they are needed.
- The allocation protocols consider a number of parameters to minimize the transmission interference from other users.
- This allocation scheme optimises bandwidth usage and results in faster transmissions.
- Dynamic channel allocation is further divided into centralised (cell-phone networks) and distributed allocation (LAN).

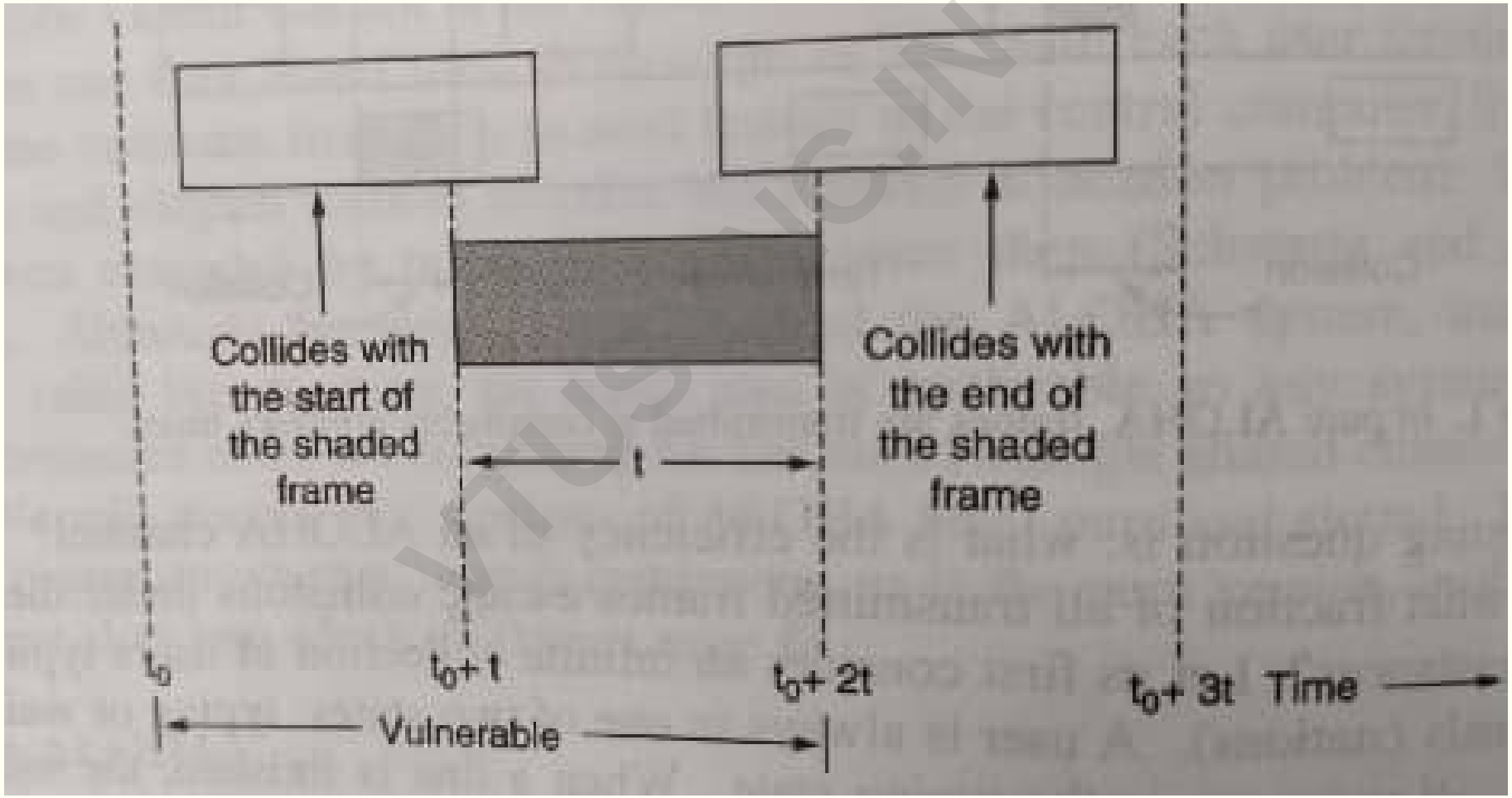
MAC protocols – Pure ALOHA

- The ALOHA protocol was developed at the University of Hawaii in the early 1970s for packet radio networks.
- However, it can be used in any situation where multiple devices share a common communication channel.
- This protocol allows devices to transmit data at any time, without a set schedule. There is no coordination between devices.
- When multiple devices attempt to transmit data at around the same time, it can result in a collision. A collision is assumed when no ACK is received.
- In this case, each device will simply wait a random amount of time before attempting to transmit again.

MAC protocols – Pure ALOHA



MAC protocols – Pure ALOHA



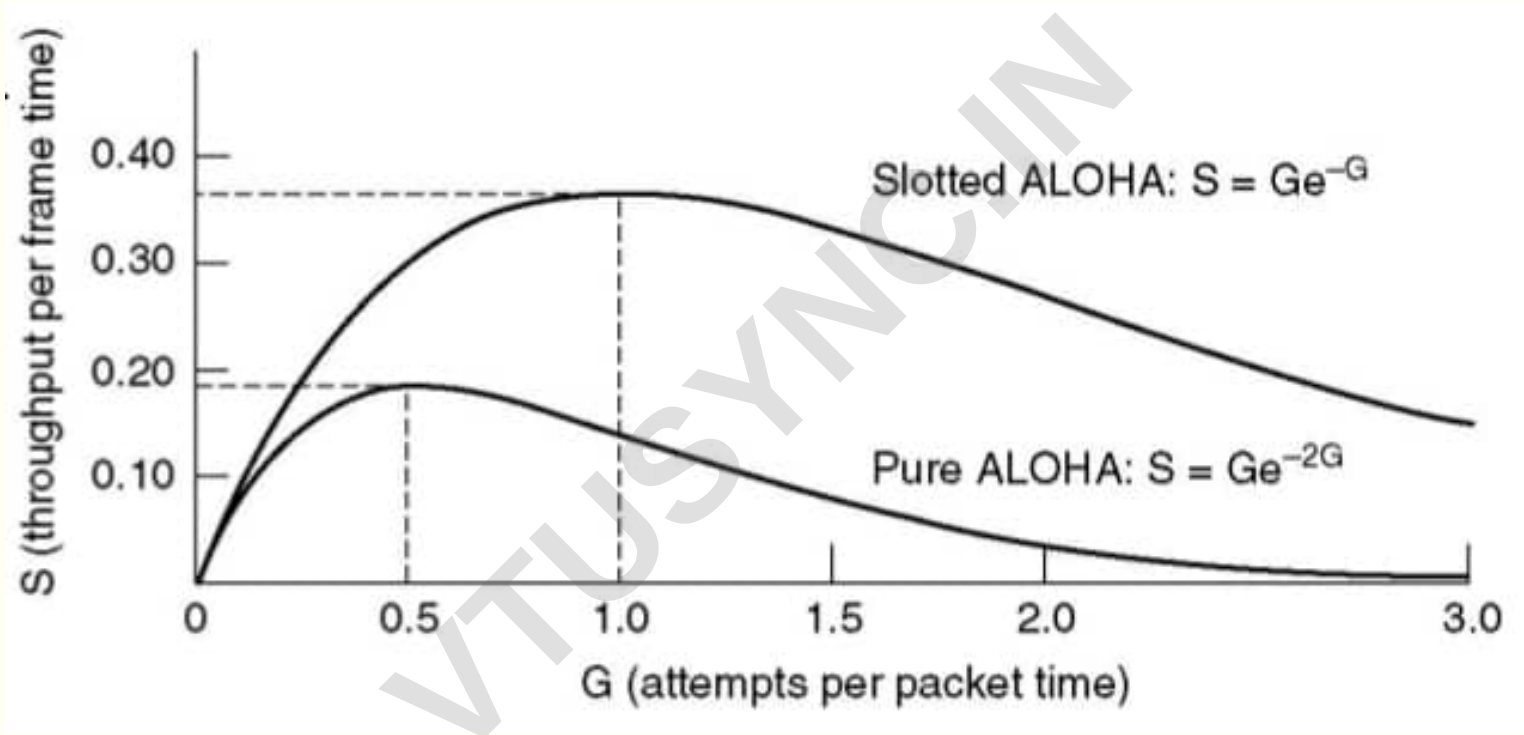
MAC protocols – Slotted ALOHA

- Here, the channel time is divided into time slots, and stations are only authorized to transmit at the beginning of a time-slot.
  - The duration of a time slot is exactly equal to the packet transmission time.

- As a result, wasted time due to collisions can be reduced to one packet time or the susceptible period can be half of that in Pure ALOHA.

- If the frame is received successfully, the receiver sends an acknowledgment.
  - If the acknowledgment is not received within a time-out period, the sender assumes that the frame was not received and retransmits the frame in the next time slot.

Throughput of Pure ALOHA and Slotted ALOHA



G = the offered load (or the number of packets being transmitted in one packet time i.e. in one time-slot). In other words, it is a measure of the number of nodes attempting to transmit in the time of a given time slot.

Carrier Sense Multiple Access (CSMA) protocols

- The maximum throughput in Slotted ALOHA is just 37%.
  - It's not surprising since the stations are transmitting at will, without knowing what the other stations are doing.
- In LANs, it is often possible for stations to detect what other stations are doing, and thus adapt their behaviour accordingly.
- Protocols in which stations "listen" for a "carrier" (transmission signal) and act accordingly are called Carrier Sense protocols.
  - In short, they don't transmit when they know that others are transmitting, thereby reducing the collisions, and improving the throughput.

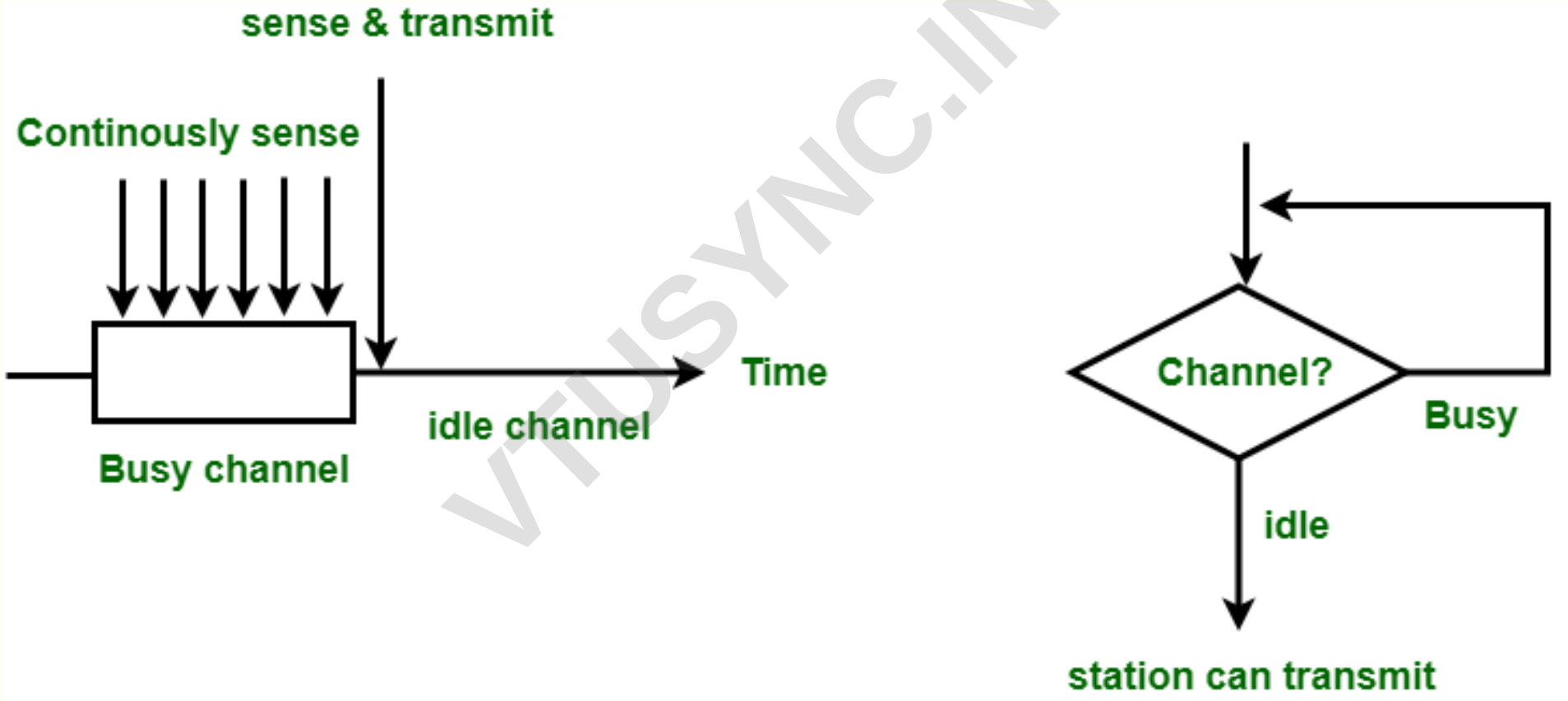
Carrier Sense Multiple Access (CSMA) protocols

- Common CSMA protocols:
  - 1-persistent protocol.
  - Non-persistent protocol.
  - p-persistent protocol.
  - CSMA / CD (Collision Detection) protocol.
- Collision-free protocols:
  - Bit-map protocol.
  - Token passing protocol.

1-persistent protocol

- When the station is ready to send the frames, it will sense the channel.
  - If the channel found to be busy, the station will wait for it to be idle.
  - If the channel is found to be idle, the station transmits the frame immediately.
- It is called “persistent” because it continuously tracks a busy channel till it becomes idle.
- It is called “1-persistent” because after finding a channel to be idle, it surely transmits it’s frame immediately (with a probability of 1, i.e. 100%)

1-persistent protocol



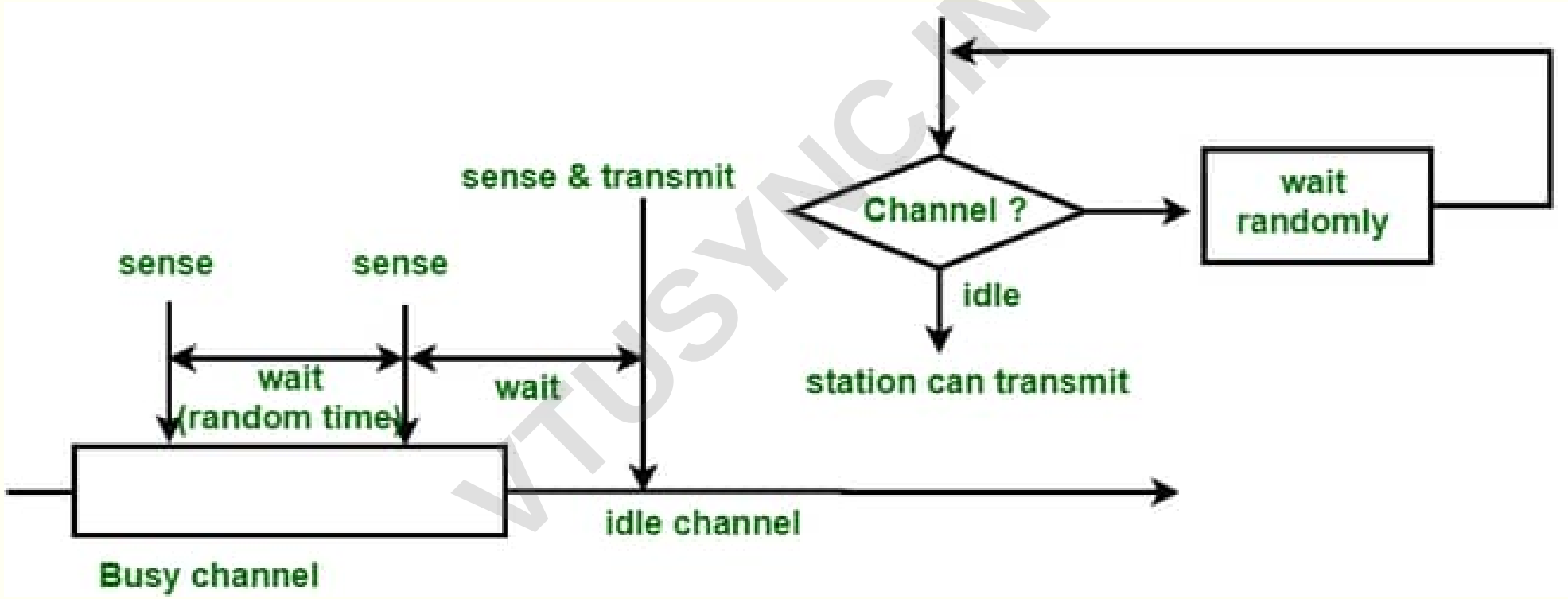
1-persistent protocol

- The problem with this method is that there is a high chance of collisions:
  - When two or more stations find the channel in an idle state and they transmit their frames at the same time.
  - When the frames of a station are being transferred, then all the other stations will sense that and won't transmit the data; But the moment they sense that the transmission has ended, they will all send their frames at the same time.
- Another drawback is that when a collision occurs, the concerned stations have to wait for a random time for the channel to be idle, and start all over again.

Non-persistent protocol

- As in 1-persistent CSMA, the station that has frames to send will sense for the channel.
  - If the channel is found to be idle, the station transmits the frame immediately.
  - However, if the channel found to be busy, the station will wait for a random time before sensing it again.
- In this method, the station does not continuously sense for the channel to get idle.
- This reduces the chances of collision, but also reduces the efficiency of the network when the load is light.

Non-persistent protocol



p-persistent protocol

- p-persistent CSMA is used when a channel has time-slots and the duration of a time-slot is equal to or greater than the maximum propagation delay time for that channel.
- When the station is ready to send the frames, it will sense the channel.
  - If the channel found to be busy, the station will wait for the next time-slot.
  - But if the channel is found to be idle, the station transmits the frame immediately with a probability p.
  - The station then waits for the beginning of the next time-slot with a probability q = (1-p).
  - If the next time-slot is also found idle, the station transmits or waits again with the probabilities p and q.
  - This process repeats until either the frame gets transmitted or another station starts transmitting.

Throughput of all CSMA protocols

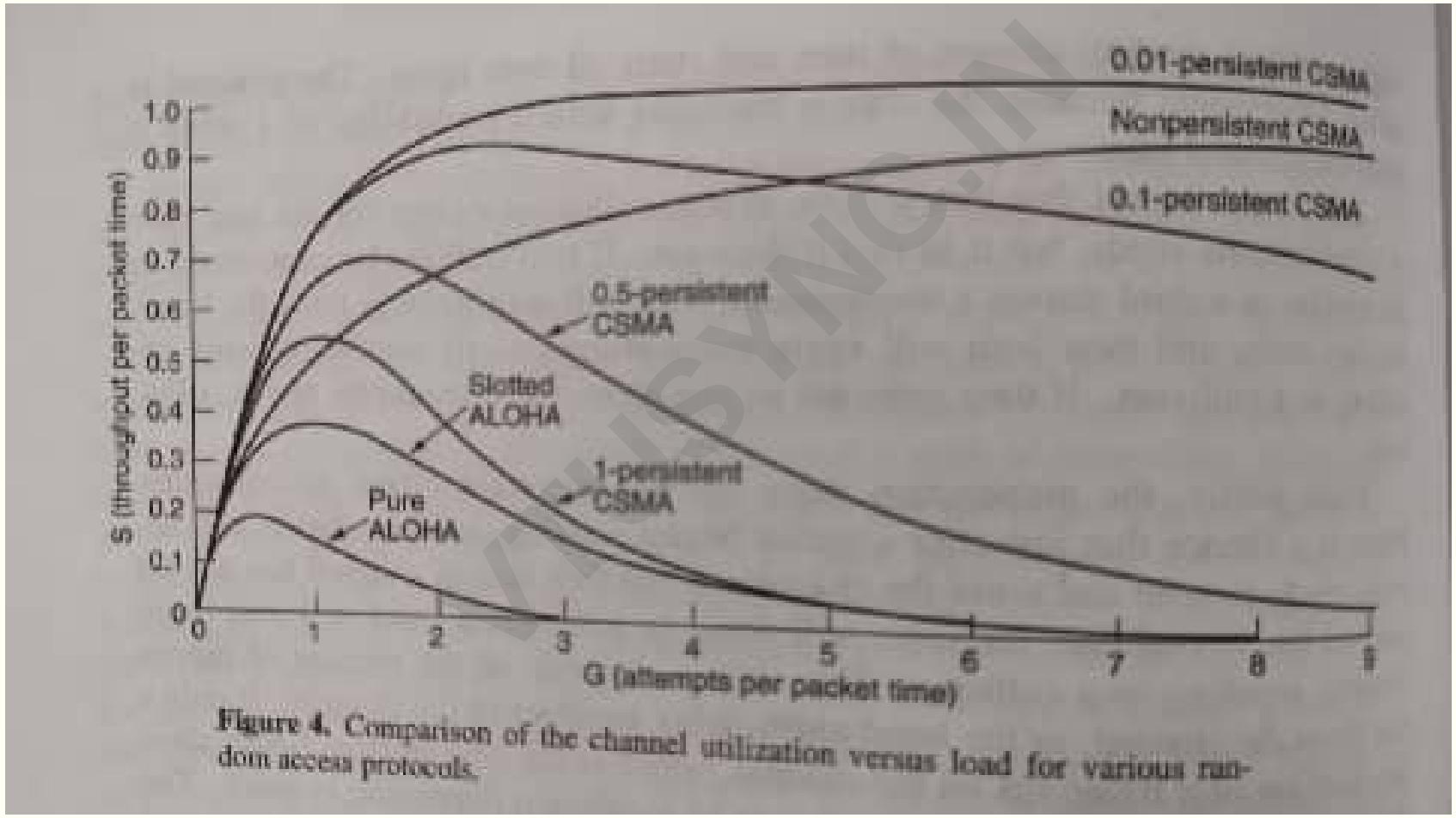


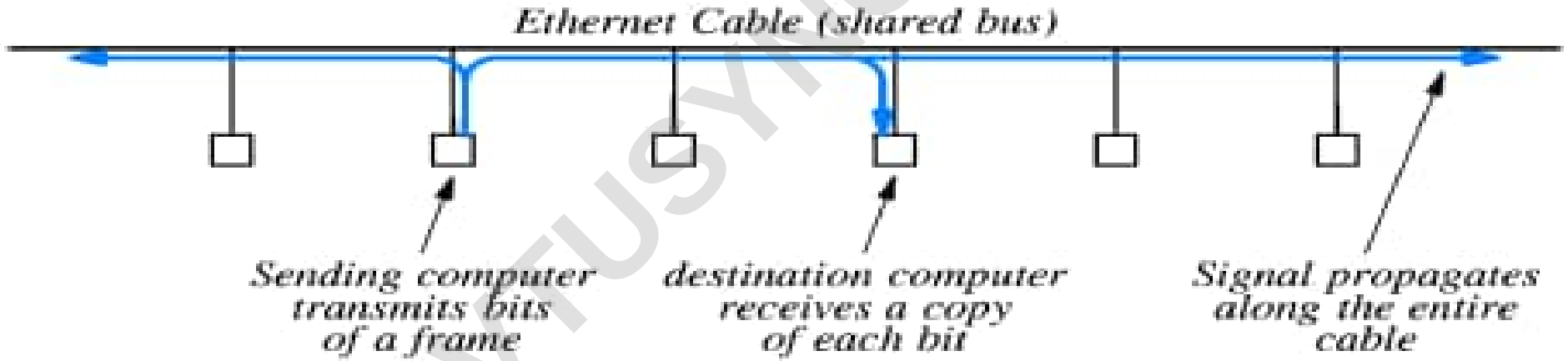
Figure 4. Comparison of the channel utilization versus load for various random access protocols.

CSMA / CD protocol

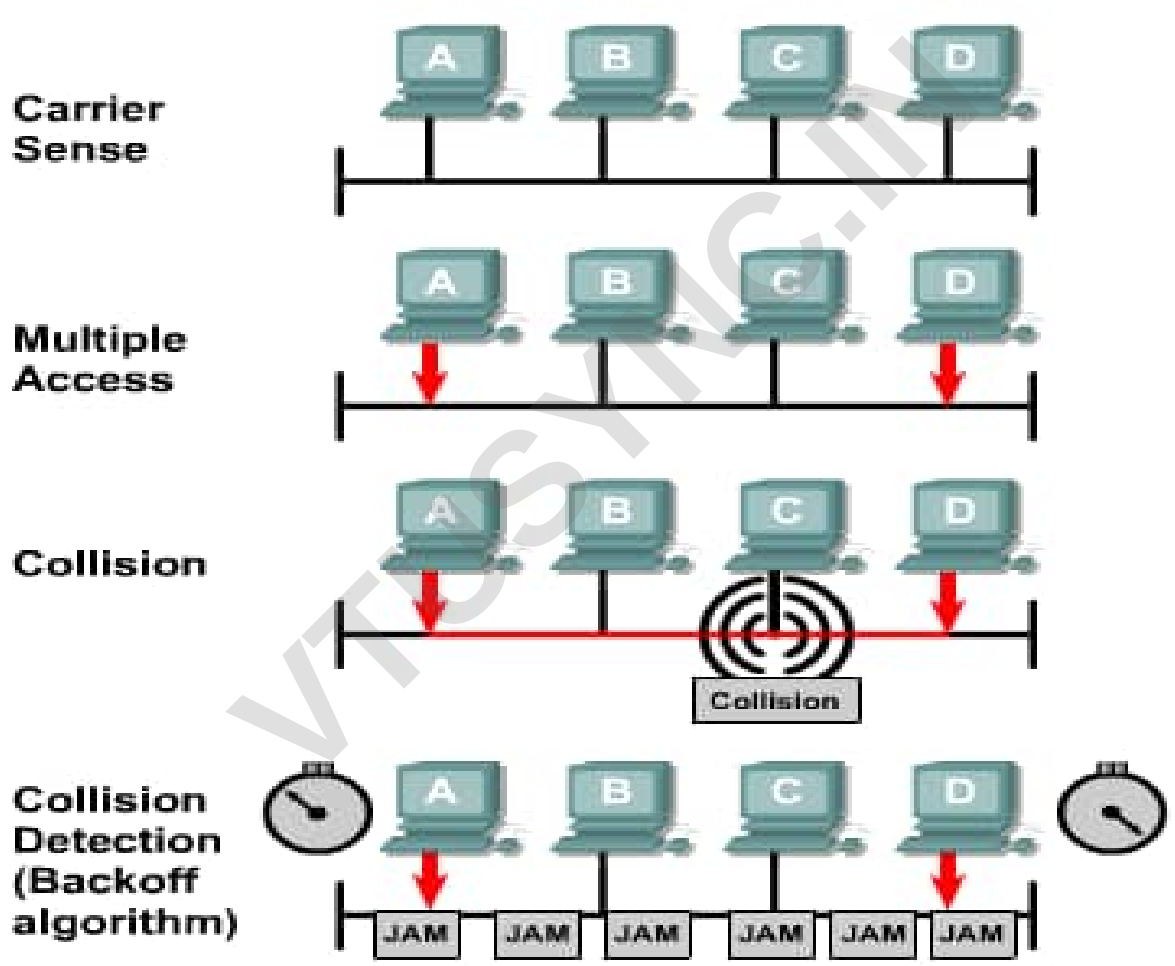
CSMA / CD

- CSMA with CD (Collision Detection) is the most popular MAC protocol used on the classic Ethernet LAN.
- Whenever a station has a frame to transmit, it will sense for a carrier (signal) on the BUS (common transmission medium).
- It will not transmit if a carrier is found.
- If two stations transmit at around the same time, it will result in a collision.
- In that case, both the stations should suspend their transmissions, and try after a random time later.

Ethernet LAN



CSMA / CD protocol



Collision detection

- The NIC of a station will compare the signal on the Bus with what it had actually transmitted.
  - If they do not match, it means that there is a collision.
- Note that once the signal representing the first bit of a frame has reached both the ends of the Bus, there will be no collisions since all the stations will be able to sense the carrier.

Performance of Ethernet LAN

- Performance can be measured by the “throughput”
- The throughput will be higher if the time lost due to collisions is lower
- Performance goes down with increasing load (many stations trying to transmit at a given time)
- Limitations:
  - There is no upper bound (limit) on the waiting time (for a transmission to be successful, since there can be repeated collissions).

Collision-free protocols

- Bit-map protocol.
- Token passing protocol.

Bit-map protocol

- Here, each contention period consists of exactly N slots. If any station has to send a frame, then it transmits a '1' bit in it's corresponding slot.
  - For example, if station 2 has a frame to send, it transmits a 1 bit in the 2^{nd} slot.
- In this way, each station has complete knowledge of which station wishes to transmit.
- There will never be any collisions because everyone agrees on who goes next.
- Protocols like this in which the desire to transmit is broadcasted, for the actual transmission, are called Reservation Protocols.

Bit-map protocol

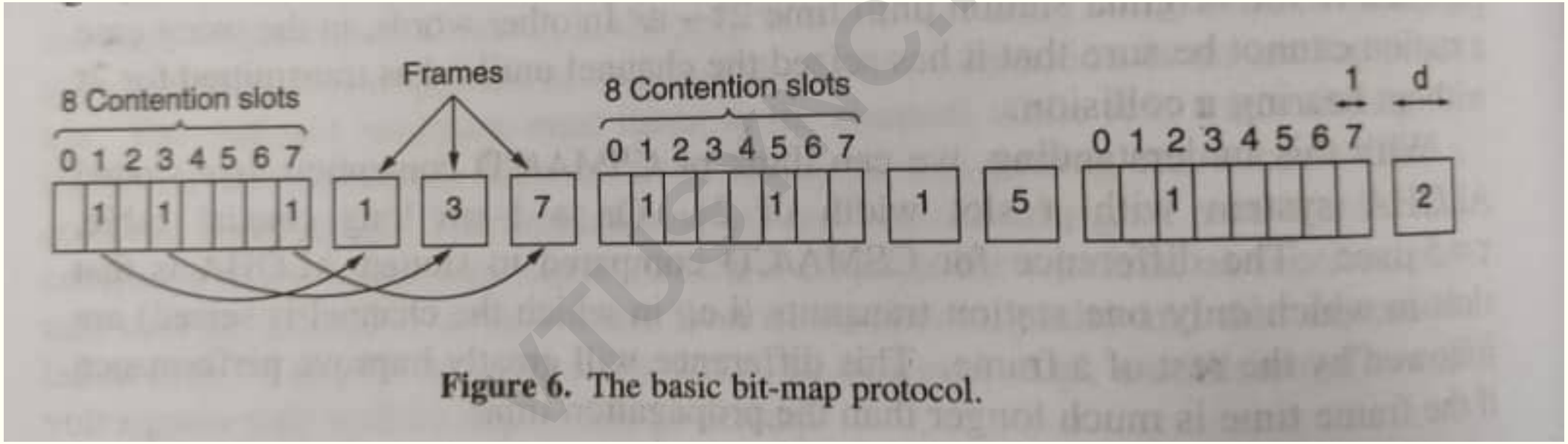


Figure 6. The basic bit-map protocol.

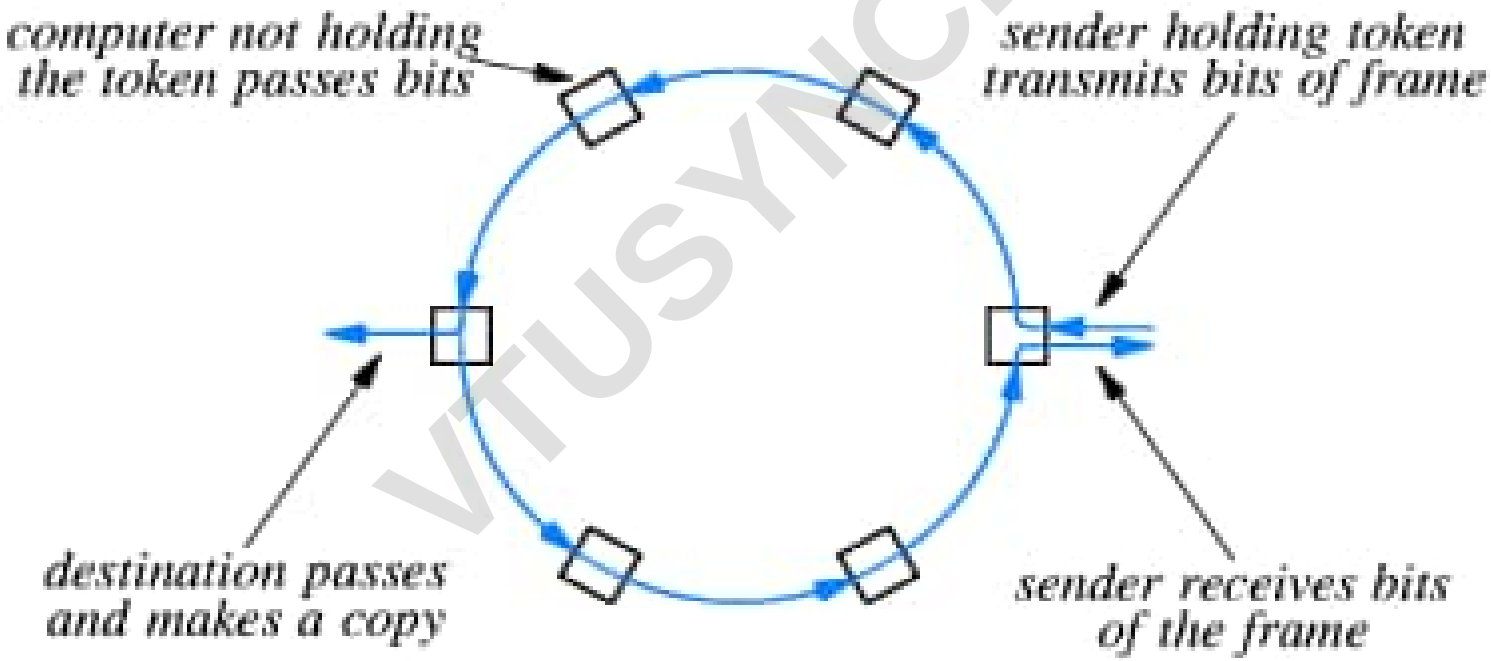
Token Ring LAN

- Token Ring protocol is used on a Token Ring LAN.
- Unlike the Ethernet, this LAN provides:
  - An upper bound on the waiting time (before a frame is successfully transmitted);
  - A priority scheme.
- Stations on a Token ring LAN are organized in a ring topology, with the data being transmitted sequentially from one station to the next.
- The message sent by any station goes round the ring and is finally drained off the ring by the transmitting station itself.

Token Ring LAN

- A control ‘token’ frame circulates around the ring to control access of the ring.
- A station would ‘seize’ the token, send its message frame and release the token.
- As the message frame circles around the ring, each station will observe the destination address in the frame to decide whether to make a copy.
- IEEE 802.5 defines the standard for this type of LANs

Topology and transmission



Token Ring protocol

- After the message frame circles the ring and returns to its origin, the originating station removes it from the ring and transmits a new token for another station to use.
- A station cannot hold on to a token for more than the set Token-holding time (default – 20 ms).

Performance of Token Ring LAN

- There is no loss of throughput with increasing load, due to lack of collisions.
- Even when the load is light (few stations trying to transmit), a station has to wait its turn for the token to arrive for its use.
- There is an upper bound (limit) on the waiting time.
- Priority and Reservation schemes can be implemented.

Wireless LAN

Wireless LAN

- It is a LAN where data communication occurs through space, using electromagnetic waves.
- A wireless LAN is implemented as an extension to, or as an alternative for, a wired LAN.
- Benefits:
  - Mobility of end user devices
  - Fast and flexible installation
    - No wires, no obstacles, no topology constraints
  - Scalability

WLAN architecture

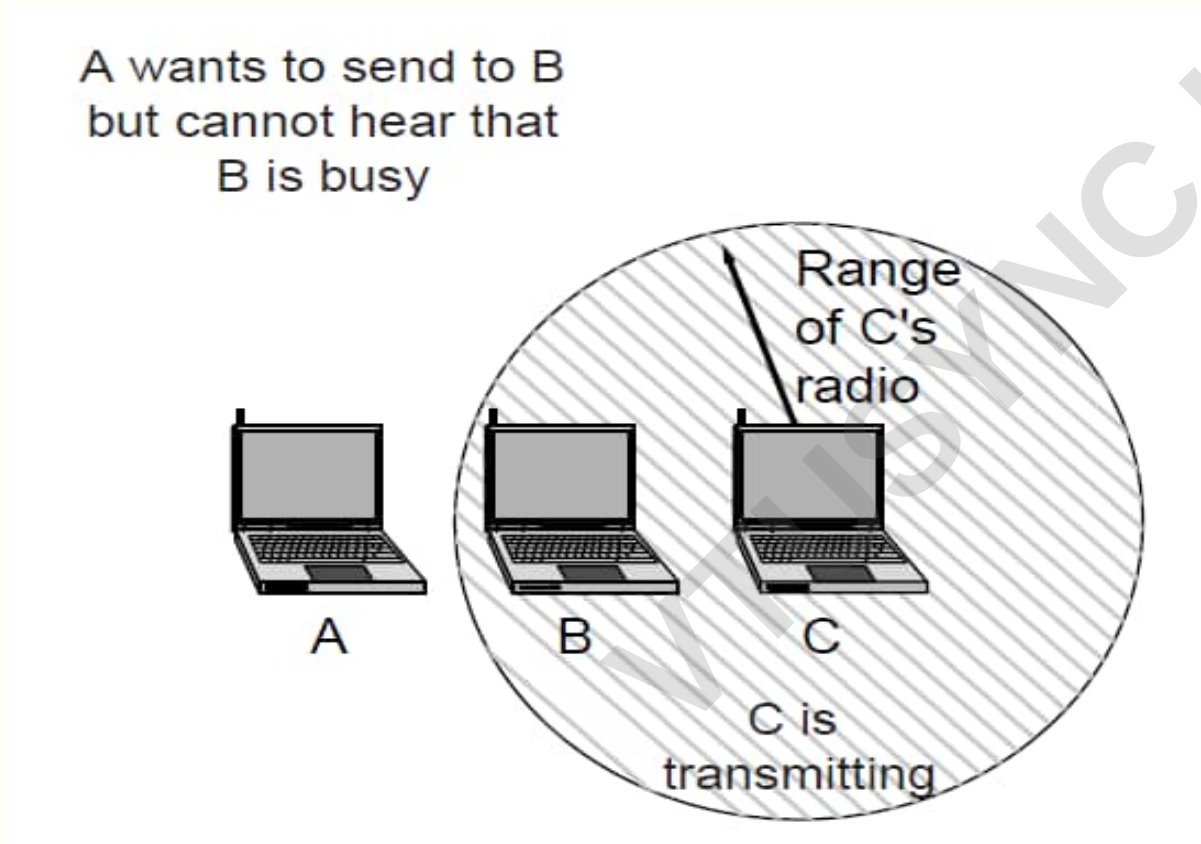
- Here clients, such as laptops and smart phones, connect to networks like the company intranet or the Internet.
- Each client is associated with an Access Point (AP) that is in turn connected to the other network.
- The client sends and receives its packets via the AP.
- The client has access to the network resources (server, shared printer, etc) as well as to other clients.

Multiple Access Points

Why not use CSMA / CD?

- The CSMA/CD protocol used on wired Ethernet LANs cannot be used in Wireless LANs, for two reasons:
- Carrier sensing is not effective because the transmission ranges of different stations is different.
  - This gives rise to problems like Hidden terminal problem.
  - In a wired Ethernet, all stations can hear each other.
- Collision detection is almost impossible because the strength of the signal received from another station can be just a millionth of the transmitted signal strength.

Hidden terminal problem



If A “senses” the channel, it will not hear C’s transmission and will falsely conclude that it can begin a transmission to B.

CSMA / Collision Avoidance

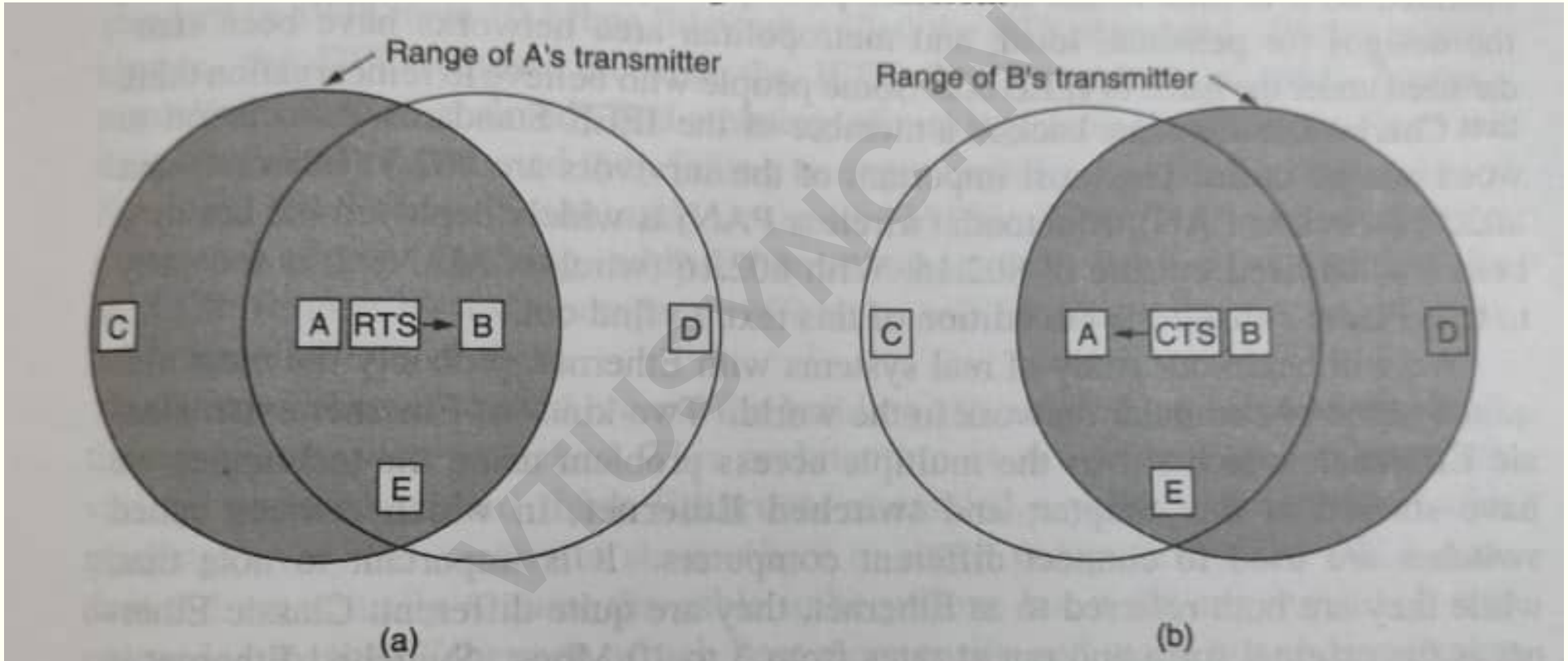


Figure 12. The MACA protocol. (a) *A* sending an RTS to *B*. (b) *B* responding with a CTS to *A*.

CSMA / Collision Avoidance

- The WLAN standard, 802.11, tries to avoid collisions by using a protocol called CSMA/Collision Avoidance (CSMA/CA)
- ‘A’ starts off by sending a short RTS (Request to Send) frame to ‘B’.
- If ‘B’ is free, it responds with a CTS (Clear To Send) frame.
- Upon receipt of CTS, ‘A starts the actual data transmission.