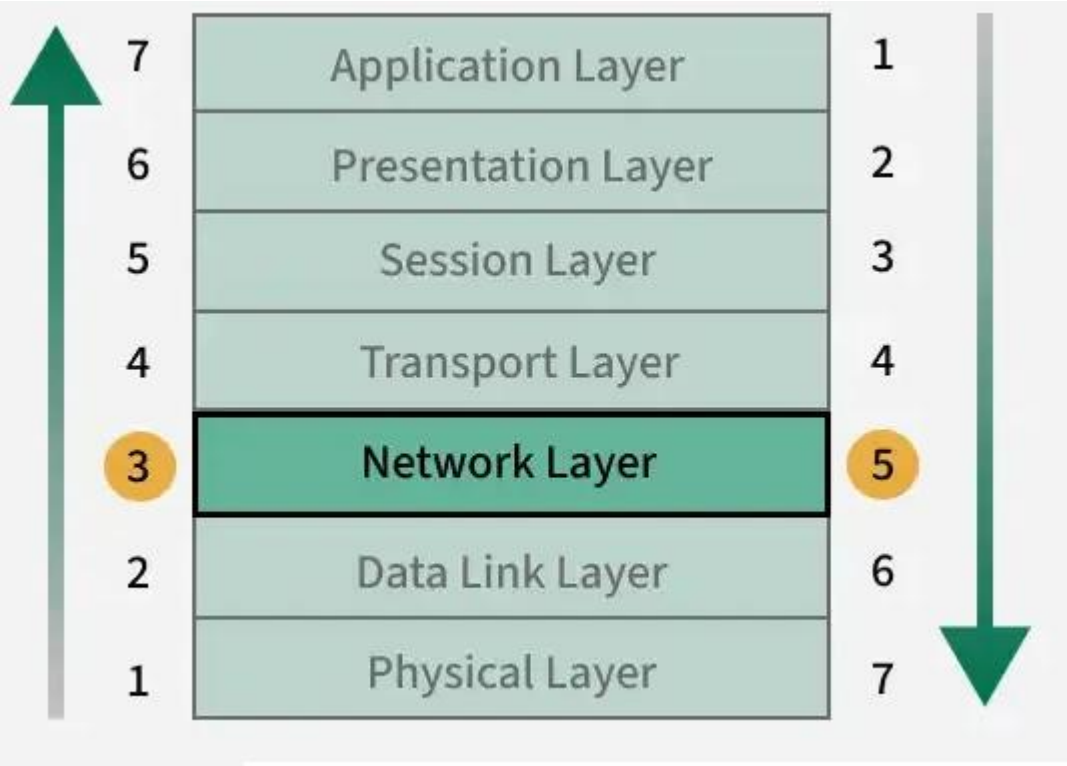
Network Layer :

Network Layer is the third layer from the bottom (Layer 3) and the fifth layer from the top in the OSI (Open Systems Interconnection) Model. It is responsible for ensuring end-to-end packet delivery across multiple interconnected networks.



Key Responsibilities of the Network Layer

Logical Addressing

Packetization

Host-to-Host Delivery

Forwarding

Routing

Fragmentation and Reassembly

Subnetting

Network Address Translation (NAT)

Logical Addressing: Assigns unique IP addresses to devices, ensuring accurate identification and communication across networks.

Packetization: Encapsulates transport layer segments into packets for efficient transmission.

Host-to-Host Delivery: Ensures reliable delivery of packets from the sender to the intended receiver across diverse networks.

Forwarding: Moves packets from the input interface of a router to the appropriate output interface based on the destination IP.

Routing: Determines the optimal path for packets to travel across multiple networks using routing algorithms and protocols.

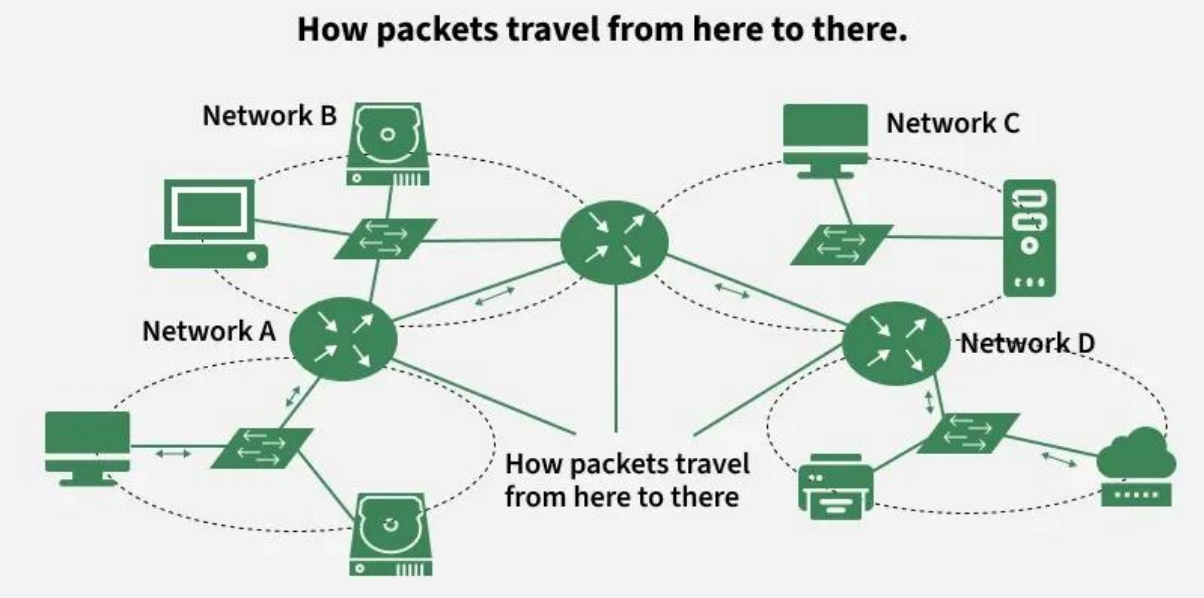
Fragmentation and Reassembly: Splits large packets into smaller fragments to match the maximum transmission unit (MTU) of a network, and reassembles them at the destination.

Subnetting: Divides larger networks into smaller subnetworks for efficient addressing and traffic management.

Network Address Translation (NAT): Maps private IPs to public IPs for internet communication, conserving address space and adding security.

Read more about Functions of Network Layer.

How the Network Layer Works



Each device is assigned a unique logical address (IP address).

Data from the transport layer is encapsulated into packets, with source and destination IPs attached.

Routers analyze the destination address and determine the best available path.

Packets traverse the network hop-by-hop, moving across routers until reaching the destination.

If the packet size exceeds the MTU(Maximum Transmission Unit), it is fragmented into smaller units.

At the destination, the fragments are reassembled into the original data.

If errors occur (e.g., unreachable destination), protocols like ICMP send error messages back to the source.

Protocols Operating at the Network Layer

IP (Internet Protocol – IPv4/IPv6)

ICMP (Internet Control Message Protocol)

ARP (Address Resolution Protocol)

RARP (Reverse Address Resolution Protocol)

NAT (Network Address Translation)

IPSec (Internet Protocol Security)

MPLS (Multiprotocol Label Switching)

Routing Protocols

RIP (Routing Information Protocol)

OSPF (Open Shortest Path First)

BGP (Border Gateway Protocol)

Advantages of the Network Layer

Enables end-to-end communication across multiple networks.

Supports scalability by allowing subnetting and hierarchical addressing.

Efficiently routes packets using shortest-path and dynamic routing algorithms.

Provides inter-networking by connecting heterogeneous networks.

Limitations of the Network Layer

No flow control mechanism; congestion may occur if too many datagrams are in transit.

Limited error control; mainly relies on upper layers for reliability.

Routers may drop packets under heavy load, leading to possible data loss.

Fragmentation increases processing overhead and may affect performance.

Design Issues in Network Layer

The Network layer is majorly focused on getting packets from the source to the destination, routing error handling, and congestion control. Before learning about design issues in the network layer, let's learn about its various functions.

Addressing: Maintains the address at the frame header of both source and destination and performs addressing to detect various devices in the network.

Packeting: This is performed by Internet Protocol. The network layer converts the packets from its upper layer.

Routing: It is the most important functionality. The network layer chooses the most relevant and best path for the data transmission from source to destination.

Inter-networking: It works to deliver a logical connection across multiple devices.

Network Layer Design Issues

The network layer comes with some design issues that are described as follows:

1. Store and Forward packet switching
2. Services provided to the Transport Layer
3. Implementation of Connectionless Service
4. Implementation of Connection oriented Service

1. Store and Forward packet switching

The host sends the packet to the nearest router. This packet is stored there until it has fully arrived once the link is fully processed by verifying the checksum then it is forwarded to the next router till it reaches the destination. This mechanism is called "Store and Forward packet switching."

2. Services provided to the Transport Layer

Through the network/transport layer interface, the network layer transfers its patterns services to the transport layer.:-

Offering services must not depend on router technology.

The transport layer needs to be protected from the type, number, and topology of the available router.

The network addresses for the transport layer should use uniform numbering patterns, also at LAN and WAN connections.

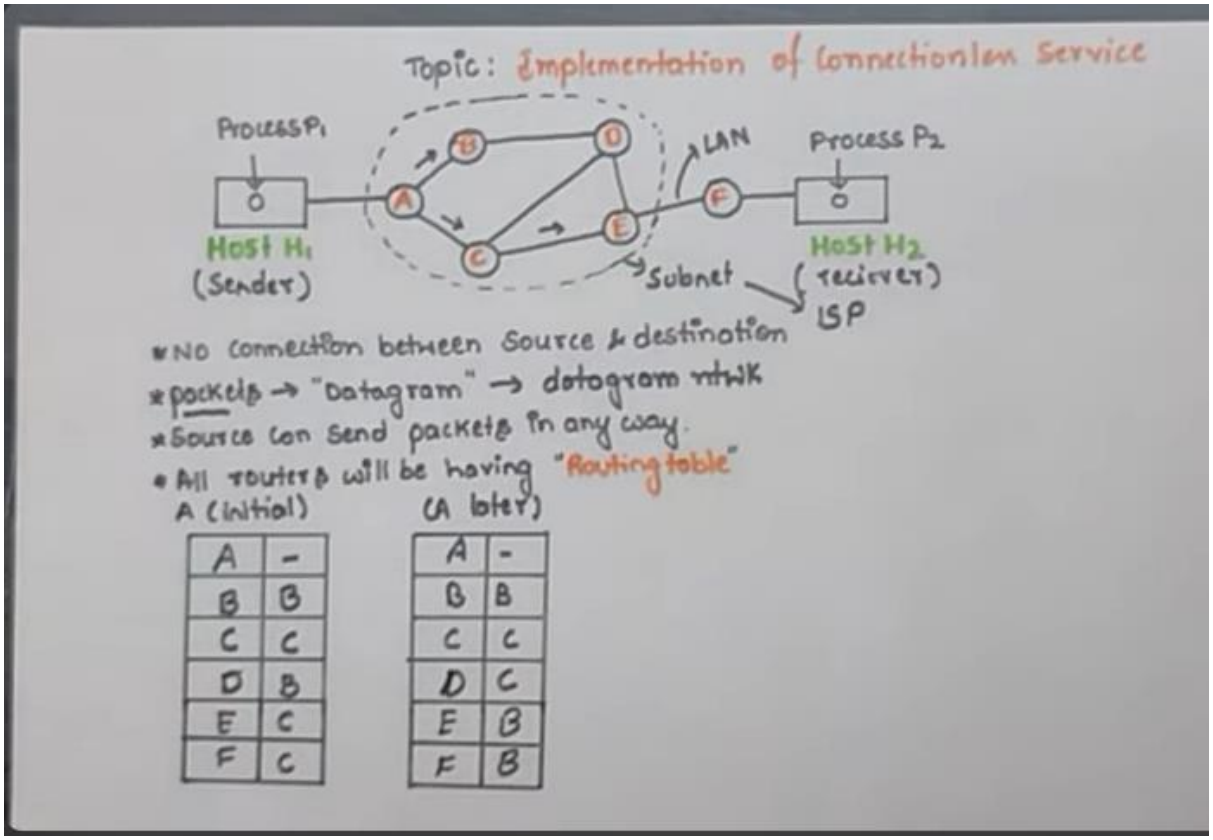
Based on the connections there are 2 types of services provided :

Connectionless - The routing and insertion of packets into the subnet are done individually. No added setup is required.

Connection-Oriented - Subnet must offer reliable service and all the packets must be transmitted over a single route.

3. Implementation of Connectionless Service

Packets are termed as "datagrams" and corresponding subnets as "datagram subnets". When the message size that has to be transmitted is 4 times the size of the packet, then the network layer divides into 4 packets and transmits each packet to the router via. a few protocols. Each data packet has a destination address and is routed independently irrespective of the packets.



A	-
B	B
C	C
D	B
E	C
F	C

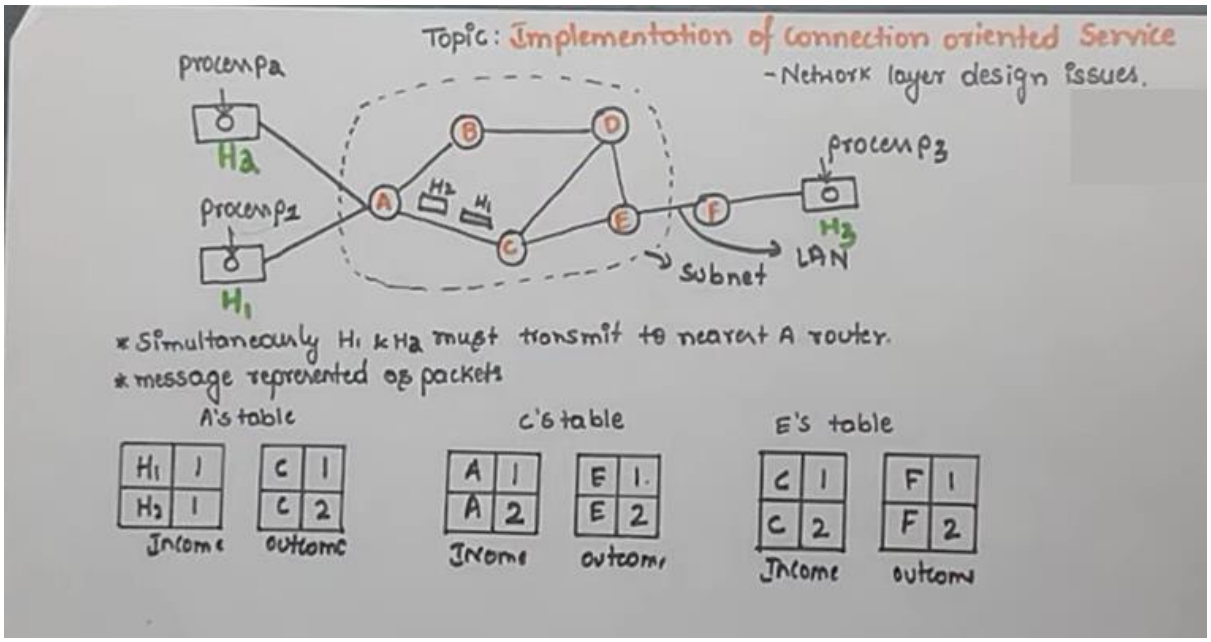
A	-
B	B
C	C
D	C
E	B
F	B

4. Implementation of Connection-Oriented service:

To use a connection-oriented service, first, we establish a connection, use it, and then release it. In connection-oriented services, the data packets are delivered to the receiver in the same order in which they have been sent by the sender. It can be done in either two ways :

Circuit Switched Connection - A dedicated physical path or a circuit is established between the communicating nodes and then the data stream is transferred.

Virtual Circuit Switched Connection - The data stream is transferred over a packet switched network, in such a way that it seems to the user that there is a dedicated path from the sender to the receiver. A virtual path is established here. While, other connections may also be using the same path.



Difference Between Virtual Circuits and Datagram Networks

Criteria	Virtual Circuit Networks	Datagram Networks
Connection Establishment	Prior to data transmission, a connection is established between sender and receiver.	No connection setup is required.
Routing	Routing decisions are made once during connection setup and remain fixed throughout the duration of the connection.	Routing decisions are made independently for each packet and can vary based on network conditions.
Flow Control	Uses explicit flow control, where the sender adjusts its rate of transmission based on feedback from the receiver.	Uses implicit flow control, where the sender assumes a certain level of available bandwidth and sends packets accordingly.
Congestion Control	Uses end-to-end congestion control, where the sender adjusts its rate of transmission based on feedback from the network.	Uses network-assisted congestion control, where routers monitor network conditions and may drop packets or send congestion signals to the sender.
Error Control	Provides reliable delivery of packets by detecting and retransmitting lost or corrupted packets.	Provides unreliable delivery of packets and does not guarantee delivery or correctness.
Overhead	Requires less overhead per packet because connection setup and state maintenance are done only once.	Requires more overhead per packet because each packet contains information about its destination address and other routing information.

Criteria	Virtual Circuit Networks	Datagram Networks
Example Protocol	ATM, Frame Relay	IP (Internet Protocol)

Shortest Path Algorithm in Computer Network

What is Shortest Path Routing?

It refers to the algorithms that help to find the shortest path between a sender and receiver for routing the data packets through the network in terms of shortest distance, minimum cost, and minimum time.

Common Shortest Path Algorithms

Dijkstra's Algorithm

Bellman Ford's Algorithm

Floyd Warshall's Algorithm

Dijkstra's Algorithm

The Dijkstra's Algorithm is a greedy algorithm that is used to find the minimum distance between a node and all other nodes in a given graph. Here we can consider node as a router and graph as a network. It uses weight of edge .ie, distance between the nodes to find a minimum distance route.

Algorithm:

1: Mark the source node current distance as 0 and all others as infinity.

2: Set the node with the smallest current distance among the non-visited nodes as the current node.

3: For each neighbor, N, of the current node:

Calculate the potential new distance by adding the current distance of the current node with the weight of the edge connecting the current node to N.

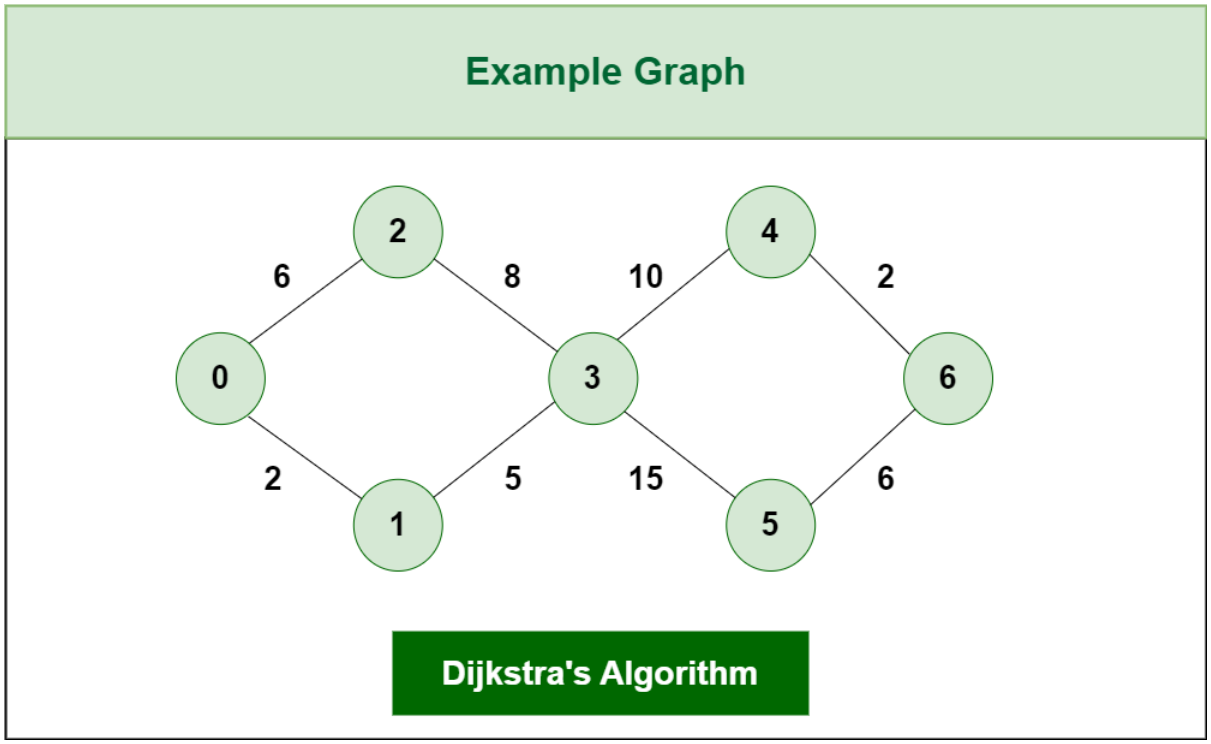
If the potential new distance is smaller than the current distance of node N, update N's current distance with the new distance.

4: Make the current node as visited node.

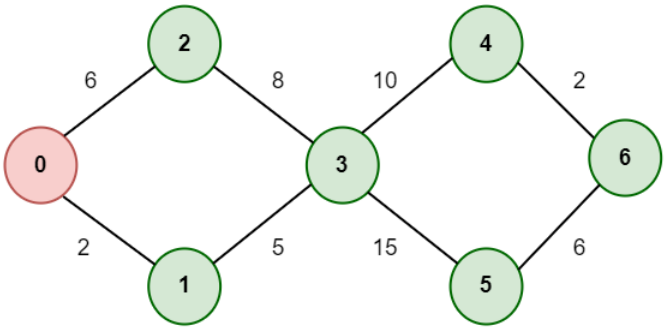
5: If we find any unvisited node, go to step 2 to find the next node which has the smallest current distance and continue this process.

Example:

Consider the graph G:



STEP 1 Start from Node 0 and mark Node 0 as Visited and check for adjacent nodes

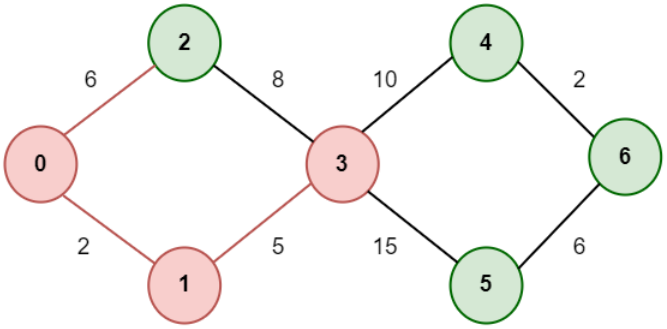


Unvisited Nodes
{0,1,2,3,4,5,6}

Distance:
0: 0 ✅
1: ∞
2: ∞
3: ∞
4: ∞
5: ∞
6: ∞

Dijkstra's Algorithm

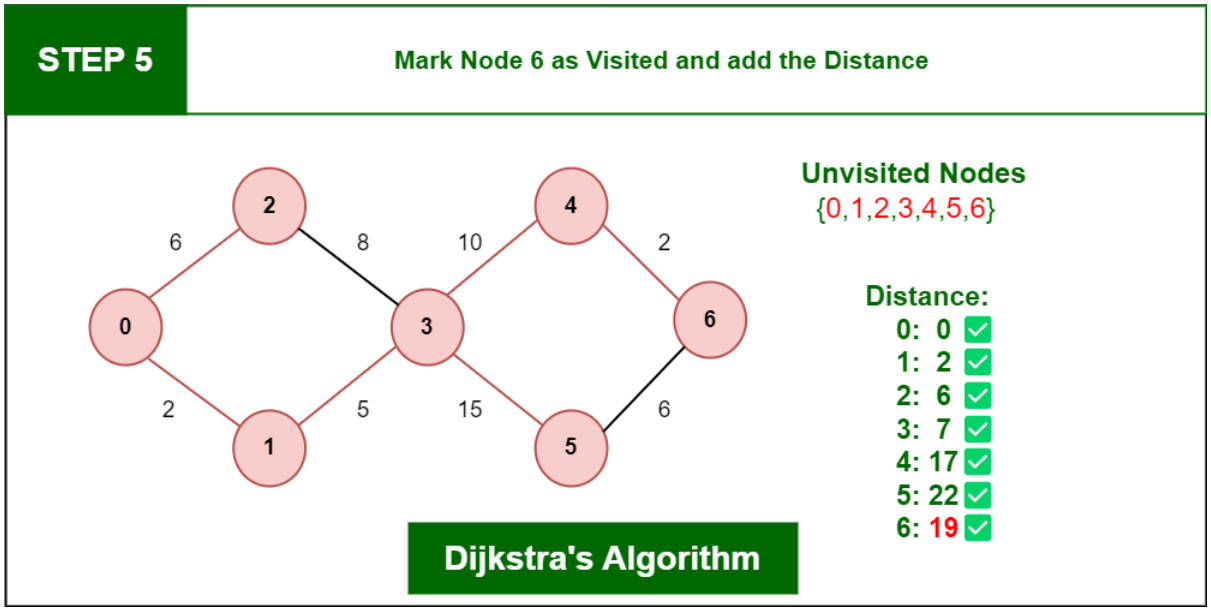
STEP 3 Mark Node 3 as Visited after considering the Optimal path and add the Distance



Unvisited Nodes
{0,1,2,3,4,5,6}

Distance:
0: 0 ✅
1: 2 ✅
2: 6 ✅
3: 7 ✅
4: ∞
5: ∞
6: ∞

Dijkstra's Algorithm



Congestion Control in Computer Networks

congestion occurs when too much data is sent through the network at the same time, causing delays, packet loss, or even network collapse.

Congestion control is the set of techniques and mechanisms used to prevent, detect, and manage congestion. Its main goal is to maintain smooth data flow, fair bandwidth allocation, and efficient utilization of network resources, ensuring a stable and reliable network performance.

Effects of Congestion Control

Improved Network Stability: Prevents overload and keeps the network running smoothly.

Reduced Latency and Packet Loss: Ensures data is delivered faster with fewer retransmissions.

Enhanced Throughput: Maximizes the volume of data successfully transferred in a given time.

Fairness in Resource Allocation: Distributes bandwidth evenly so no user monopolizes resources.

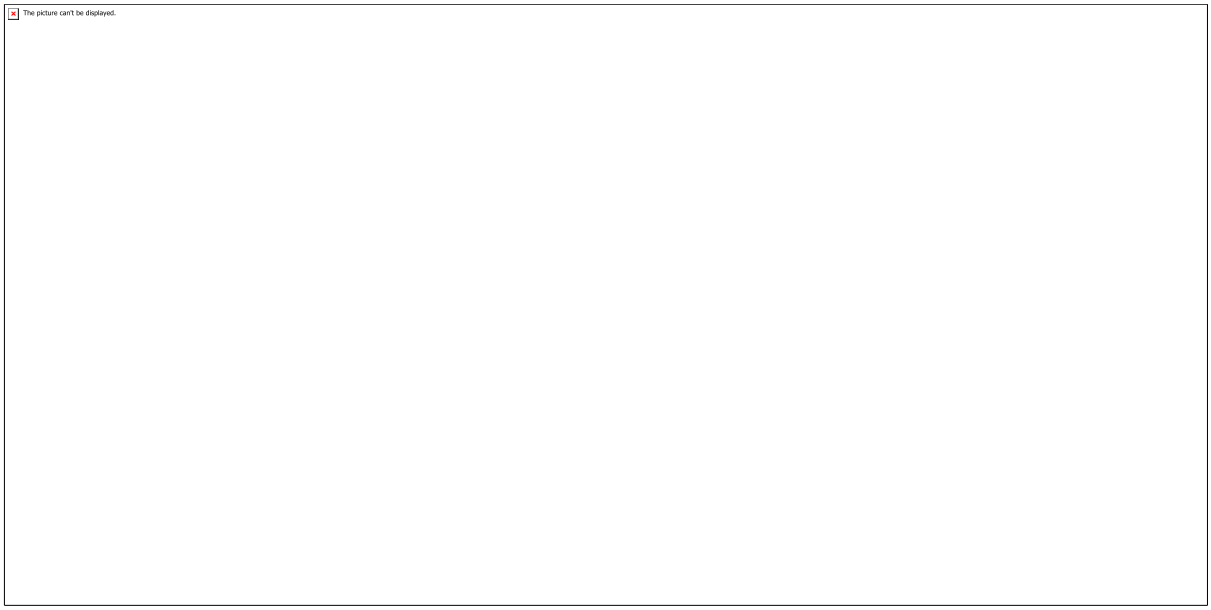
Better User Experience: Provides faster access to websites, applications, and services.

Prevention of Congestion Collapse: Avoids severe breakdowns where the network becomes nearly unusable.

Congestion Control Algorithms

1. Leaky Bucket Algorithm

The leaky bucket algorithm controls the flow of packets by sending them out at a constant, fixed rate, regardless of incoming burst traffic. Packets are placed into a bucket (queue), and if the bucket overflows, excess packets are discarded. This ensures smooth traffic but wastes bandwidth during idle times.



Steps:

Packets arrive -> placed into the bucket.

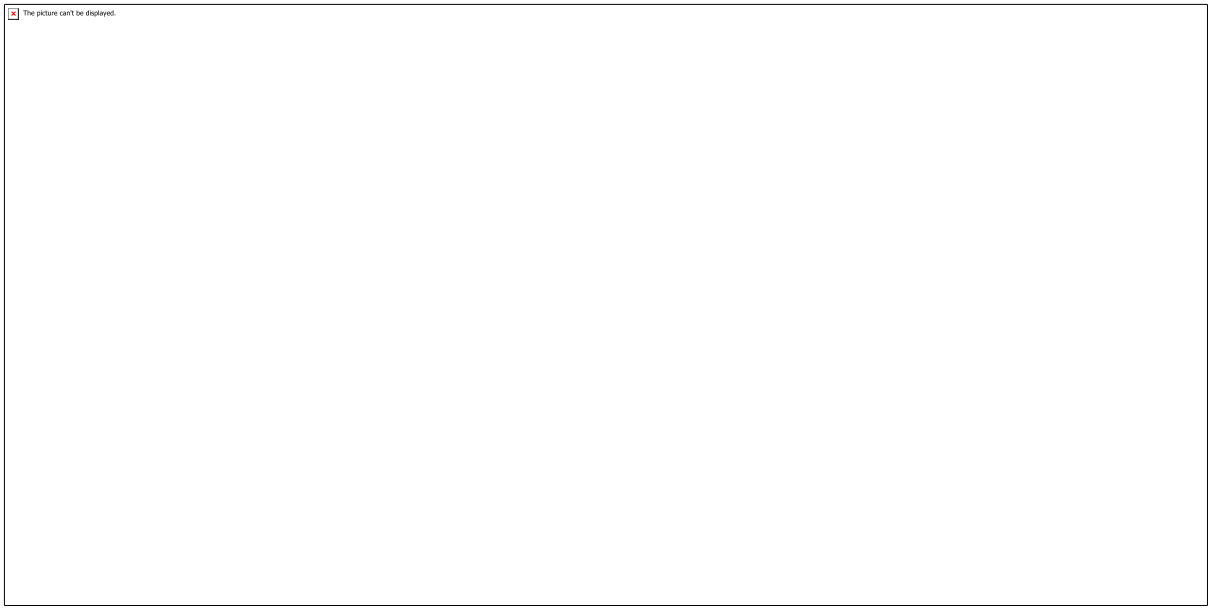
Bucket leaks (transmits) at a constant rate.

Bursty traffic is smoothed into uniform traffic.

Limitation: Too rigid, wastes available bandwidth if traffic is bursty.

2. Token Bucket Algorithm

The token bucket algorithm allows more flexibility by generating tokens at a fixed rate. A packet can only be transmitted if a token is available, and each packet consumes one token. If enough tokens accumulate, burst traffic can be sent quickly, making it better for handling variable traffic patterns without unnecessary drops.



Steps:

Tokens are added to the bucket at regular intervals.

Each token permits sending one packet.

If tokens exist -> packets can be transmitted immediately.

If no tokens -> packets must wait.

Advantage: Handles bursty traffic efficiently without unnecessary data loss.

Leaky Bucket vs Token Bucket

Parameter	Leaky Bucket	Token Bucket
Output	Fixed, constant rate	Variable, allows bursts
Flexibility	Rigid	Flexible
Packet Loss	Possible during bursts	Avoids loss if tokens available

Advantages of Congestion Control

Ensures stable and reliable operation of networks.

Reduces delays and retransmissions.

Minimizes data loss.

Optimizes resource utilization.

Scales well with growing networks.

Adapts to changing traffic conditions.

Disadvantages of Congestion Control

Adds complexity to network design.

May introduce overhead in processing.

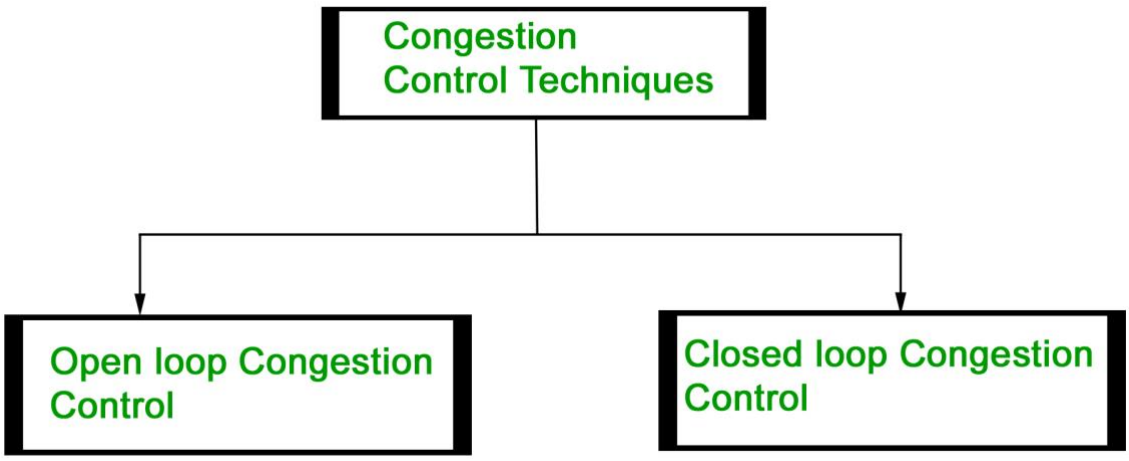
Can be sensitive to network conditions.

May struggle with resource prioritization in critical scenarios.

Effectiveness depends on modern infrastructure.

Approaches to congestion control

Congestion occurs when the demand for network resources exceeds the available capacity, leading to delays and packet loss. Congestion control refers to the techniques used to control or prevent this issue, which can be broadly classified into two categories:



Open Loop Congestion Control

Open loop congestion control policies are applied to prevent congestion before it happens. The congestion control is handled either by the source or the destination.

1. Retransmission Policy: When a packet is lost or corrupted, the sender may retransmit it. However, excessive retransmissions can worsen congestion. To avoid this, retransmission timers must be carefully designed to balance efficiency and congestion prevention

2. Window Policy: The type of sliding window used by the sender impacts congestion. In Go-Back-N ARQ, multiple packets are retransmitted even if only one is lost, which increases load on the network. To minimize congestion, the Selective Repeat ARQ policy is preferred since it retransmits only the missing packets.

3. Discarding Policy: Routers may apply discarding policies to manage congestion. For example, less sensitive or corrupted packets can be selectively discarded without affecting the overall quality of service. In multimedia applications like audio streaming, dropping some packets may still preserve acceptable quality while reducing congestion.

4. Acknowledgment Policy: Acknowledgments themselves contribute to network load. To reduce this overhead, receivers can use strategies such as:

Sending one acknowledgment for multiple packets (ACK for N packets).

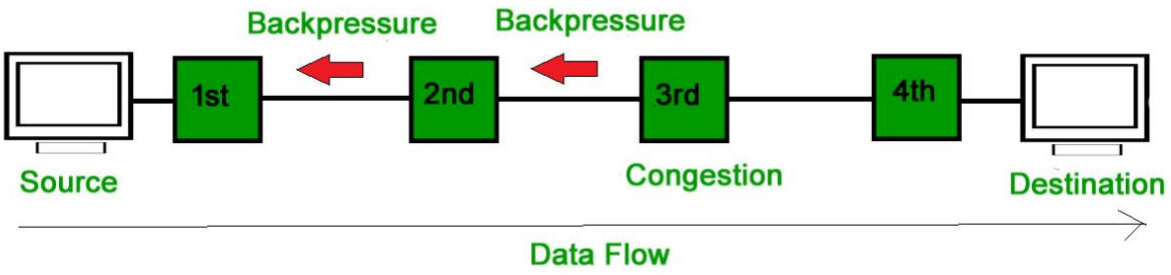
Delaying acknowledgments until a packet needs to be sent or a timer expires.

5. Admission Policy: Admission control prevents congestion by allowing new connections only if resources are available. Before establishing a virtual connection, switches verify whether sufficient bandwidth and buffer space exist. If congestion is likely, the request is denied to protect existing flows.

Closed Loop Congestion Control

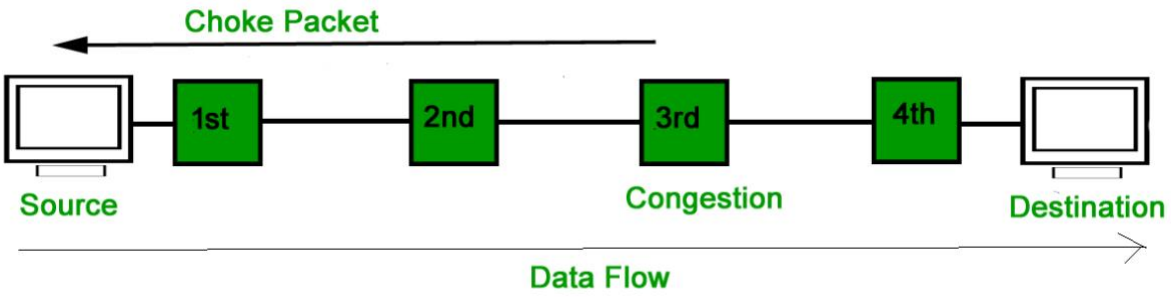
Closed loop congestion control techniques are used to treat or alleviate congestion after it happens. Several techniques are used by different protocols; some of them are:

1. Backpressure : In this technique, a congested node temporarily stops accepting packets from its upstream neighbor. This may propagate backward through the network, slowing traffic up to the source. Backpressure works only in virtual circuit networks, where nodes know their upstream neighbors.



In above diagram the 3rd node is congested and stops receiving packets as a result 2nd node may be get congested due to slowing down of the output data flow. Similarly 1st node may get congested and inform the source to slow down.

2. Choke Packet Technique: In this method, a congested router sends a special choke packet directly to the source, instructing it to reduce traffic. Routers monitor resource usage, and when utilization exceeds a threshold, the choke packet is triggered. Only the source is notified, while intermediate nodes remain unaware of congestion.



3. Implicit Signaling: Here, no direct message is exchanged. The source infers congestion when, for example, acknowledgments are delayed or missing for a long time.

4. Explicit Signaling: A congested node directly informs the source or destination by embedding a congestion signal inside data packets, unlike choke packets which use separate control packets.

Forward Signaling: Congestion information is sent to the destination, which then applies preventive measures.

Backward Signaling: Congestion information is sent to the source, instructing it to slow down transmission.

Traffic-Aware Routing

Traffic-aware routing is a dynamic routing mechanism in which routers select network paths based on real-time traffic conditions rather than fixed metrics such as hop count. This technique continuously measures the load on network links, including queue length, bandwidth utilization, delay, and packet loss. Routing decisions are then adjusted to avoid congested paths and distribute traffic more evenly across the network. The goal is to maximize resource usage, minimize latency, and prevent bottlenecks.

In traffic-aware routing, routers frequently exchange traffic statistics with neighboring routers. Using this information, the routing algorithm computes the least-loaded path, even if it is longer in distance. This improves response time for applications such as video conferencing, cloud services, and online gaming. Modern networks, especially Software Defined Networks (SDN), use centralized controllers to gather traffic data and dynamically update routing tables.

The major advantage of traffic-aware routing is that it enhances overall network performance by preventing overloading of any single path. It also improves Quality of Service (QoS) by ensuring reliable packet delivery. However, frequent route changes may cause overhead due to constant updates. Despite this, traffic-aware routing remains a widely used technique in large-scale networks such as data centers, WANs, and service provider networks.

Admission Control

Admission control is a congestion-prevention technique used in networks to determine whether new traffic flows should be allowed or denied based on current network conditions. Before

accepting new connections, the system checks available bandwidth, buffer space, delay levels, and overall load. If the network is already heavily loaded, the new request is rejected to maintain performance for existing users. Admission control ensures that the network does not reach a point where congestion becomes unavoidable.

This mechanism is widely used in Quality of Service (QoS) systems such as ATM networks, MPLS networks, and multimedia communication platforms where real-time performance is critical. For example, in VoIP or video streaming services, accepting a new flow without proper resources may degrade all ongoing sessions. Admission control prevents this by guaranteeing service levels before establishing a new connection.

Admission control can be centralized or distributed. In centralized systems like SDN, a controller makes all decisions. In distributed systems, each router performs local checks. The main benefit is that it maintains predictable performance. A drawback is that some flows may be rejected even if network conditions might soon improve. Still, admission control plays a key role in ensuring stable and high-quality network service.

Traffic Throttling

Traffic throttling is a congestion-control method that intentionally limits the data transmission rate of users, applications, or services. It is used when the network experiences heavy traffic or when certain users consume excessive bandwidth. By reducing the speed of data flow, throttling helps maintain fairness and prevents the network from becoming overloaded.

Internet Service Providers (ISPs) commonly use throttling to control bandwidth usage, especially during peak hours. For example, streaming services or file downloads may be slowed down to ensure that everyone gets acceptable network performance. In enterprise networks, throttling is applied to regulate traffic generated by non-critical services, ensuring that essential applications like VoIP and business systems receive priority.

Throttling can be implemented using techniques such as rate limiting, shaping, and priority-based queuing. It helps in maintaining network stability but can cause user dissatisfaction if overused. Overall, traffic throttling is an essential tool for managing bandwidth, preventing congestion, and providing a balanced network experience for all users.

Load Shedding

Load shedding is a technique used to reduce extreme network congestion by intentionally dropping packets or refusing non-essential traffic. When a network reaches a critical level of overload, routers cannot process all incoming packets, leading to long delays and buffer overflows. Instead of allowing the entire system to collapse, load shedding selectively discards packets to keep essential services running.

Load shedding works on the principle that dropping a small amount of traffic early prevents the entire network from failing. It is particularly useful in real-time systems where old data becomes useless, such as sensor networks or streaming applications. In such cases, it is better to drop outdated packets than to waste resources processing them.

Routers may drop packets based on random early detection, priority levels, or flow-specific rules. High-priority traffic is preserved, while low-priority or bulk traffic is sacrificed. Although load shedding results in data loss, it prevents larger system failures and ensures that critical services continue functioning. It is an emergency measure used only when congestion is severe and unavoidable.

IP Protocol

The Internet Protocol (IP) is a network layer protocol responsible for addressing, routing, and delivering packets across interconnected networks. It defines the format of IP packets and provides the logical addressing mechanism required to identify devices on the internet. The most widely used versions are IPv4 and IPv6. IP operates on a connectionless and best-effort delivery model, meaning it does not guarantee delivery, order, or reliability.

IP's primary function is routing packets from the source to the destination across multiple intermediate routers. Each router examines the destination IP address and forwards the packet along the best path known at that moment. Since IP does not manage reliability, upper-layer protocols such as TCP handle retransmission and error recovery.

IP supports fragmentation when packets are too large for a particular link's MTU (Maximum Transmission Unit). It also includes essential fields such as header length, time-to-live (TTL), protocol type, checksum, and source/destination addresses. The flexibility and scalability of the IP protocol make it the foundation of modern networking, enabling communication across millions of devices worldwide.

IP Address

An IP address is a unique logical identifier assigned to each device connected to a network using the Internet Protocol. It enables devices to send and receive data packets by providing a source and destination address. IP addresses are of two main types: IPv4, which uses 32 bits and is written in dotted decimal notation (e.g., 192.168.1.10), and IPv6, which uses 128 bits written in hexadecimal format (e.g., 2001:0db8::1).

IP addresses can be public or private, static or dynamic. Public IPs are globally unique and assigned by ISPs, allowing devices to communicate over the internet. Private IPs are used within local networks (LANs), such as home or office networks, and cannot be routed on the public internet. Dynamic IPs are automatically assigned by DHCP servers, while static IPs remain constant.

IP addresses also include network and host portions, determined by the subnet mask. This allows networks to be divided into smaller subnets for efficient management and routing. IP addressing is fundamental for internet communication, network security, and resource allocation. Without IP addresses, devices would not be able to locate one another or exchange information over the network.