CHAPTER I

INTRODUCTION TO VARIOUS NETWORKS (OR) TOPOLOGIES
1.1. Introduction to Networks:

Nowadays usage of computers in the field of Information Technology (I.T.) is growing fast and their utility has increased enormously in almost all fields. In earlier days computers were very big in size and used to occupy large space as they were built with vacuum tubes and hence used to be installed in large rooms with good ventilation. Because of their size, cost and limitations in applications, their usage was restricted to very limited fields like Scientific Research and Development, Military etc. In the early 1960s, in the place of vacuum tubes, Integrated Circuits (ICs - Chips) and later Microprocessors were introduced which are very small in size and versatile in function at the same time generate less heat. This new technology has changed the look, shape and size of the Computers. The computers became small in sizes which were easy to carry. There was a two pronged advancement in Computer Technology - as the size of the computers was becoming smaller and smaller, their applications have grown wider and wider. Along with its reduction in size the functional capabilities have grown dramatically from 'an automatic calculating device' to a device that is able to 'store, process and retrieve large volumes of data'. Later technological developments have made it possible to transmit the stored/processed data to any part of the globe [75].

Since the 70s, the Computers have started effecting/influencing our day-to-day lives in many ways. Computers have become an important part of our lives with rapid developments taking place in its Technology. To-day, Computers are playing significant role in many important fields like Education, Research and Development, Space Technology, Medical Instrumentation, Weather Forecasting, Telecommunications and even Entertainment etc., besides the routine
jobs like computations, data storage, processing and retrieval. Thus, the number of ways the computers are affecting human lives has become countless.

With the reduction of size to a desktop (PC) level system, proliferation of computers has taken place in leaps and bounds, during the last three decades. Naturally, a method to interconnect the computers located at different places has become a necessity, for exchanging data/information between those computer systems. A new Technology to interconnect computer systems located at different places has emerged which is called the ‘Computer Networking Technology’. This Technology has also grown from simple interconnection between a few computers in single location through copper wires, to an elaborate network encompassing computers located at far and wide locations, even separated by continents/oceans, through opticfibre, wireless microwaves or satellite technology.

In a sense, the communication systems developed due to interconnection of computers through Wide Area Networks (WANs) has shrunk the world to such an extant hitherto never imagined, thus enabling one to exchange data/information/communications from one corner of the globe to another, at the flick of a button almost instantly, as the data flows at the speed of the light.

With the above facts in mind, at this juncture we are naturally interested in certain enquiries like,

i) How Fast One Can Transmit The Data ?

ii) Which Is the Media through which one can transmit more rapidly and More Accurately?
Such type of questions can be answered effectively through studies in depth about networks, calculation of their performance measures and critical comparisons among these performance measures with respect to different networks, which is set as the aim of the Research for this thesis.

Basically, a Network represents interlink between several systems. In particular, if these systems are autonomous computers such networking is known as 'Computer Networks'. Two computers are said to be inter-connected if they are able to exchange information and the connection media between the computers may be a copper wire, fiber optics or micro-waves. The Speed and volume of data transmission depends upon the type of the connection. One can also exchange the data from satellite to satellite through micro-waves. Such type of networks play key role in present day Information Technology.

Efficiency of any Network depends on the Network design and transmission technology [75]. Based on the transmission methodology, the Networks can be of two types namely,

i). Broadcast Networks

ii) Point–to–Point Networks.

i) Broadcast Networks

Broadcast Networks have a single communication channel that is shared by all the systems on the networks, through these networks messages usually called as packets are transmitted from this communication channel to all other systems. The receiving client system checks for the address field and if the packet is intended for itself, that client system processes the packet otherwise it just ignores. This mode of operation is called Broadcasting. Some Broadcasting systems also support the
transmission to a subset of systems. Such transformation of information is known as **Multicasting** [75].

ii) **Point-to-Point Networks**

In **Point-to-Point Networks**, many connections between individual pairs of systems and transmission of information is done from **Source to Destination systems**. In this type of Network a *packet* many have to visit one or more intermediate systems. Thus there exist multiple routes with different lengths and hence there is a necessity for **routing algorithms** in these types of networks. Usually broadcasting networks cover geographically smaller area than point-to-point networks, which cover larger geographical area.

Based on ............... the Networks are classified as of two types namely:

i. **Logical Networks**

ii. **Multihop Networks (Point-to-Point Logical Networks)**

These networks are briefly explained as follows.

i) **Logical Networks**:

A logical network is a virtual network with nodes connected by logical links on the optical networks. Logical networks are required because of the connectivity bottlenecks in all optical networks and the limit on the distance of optical path. The features of logical networks include the end systems communicate via., virtual paths, and alternative logical paths between the end systems, and the **topology of the logical networks independent to the physical topology**. The design issues of logical networks include the **logical layer design** (routing and capacity-assignment), the **physical layer design** (embedding the logical topology to the physical topology).

There are two approaches for the embedding, they are, realizing a logical topology on
a given physical topology or choosing a physical topology for a given logical topology.

The logical topology can be viewed as an interface for matching a set of routing requests to the physical topology.

ii) Multihop Networks (Point-to-Point Logical Networks)

There are three layers in multihop networks, namely routing requests (change dynamically), logical topologies (static), and Physical topologies (fixed). The logical topology is expressed by a directed graph (digraph) with nodes for logical switch nodes and arcs for logical links. Regular Graphs are often used for the logical topology. The advantages of regular logical topologies for multihop networks include simple routing / congestion control, small network diameter and average distances, symmetric property, identical structures for network node, and simple network implementation.

The performance issues on the logical topology include the connectivity, throughput, logical path length, and so on.

The present thesis concentrates on Point-to-Point Networks.

1.2 Introduction to Routing Algorithms

The basic function of a network is routing packets from the Source system to the Destination system. The Source system is the system at which the packet is generated and Destination System is the one, where the packet has to be reach. There may be intermediary systems through which the packet is deflected; finally the packet is to be absorbed at the Destination system only. Thus it is necessary for the system to identify whether to deflect the packet or to absorb it. If the packet is meant
for a particular Destination system, it has to be absorbed at that system it self. Otherwise, it has to be deflected to the next intermediary system [75].

Routing Algorithms:
Routing algorithms can be classified in to two major classes namely,

a) Adaptive Routing Algorithms
b) Non-adaptive Routing Algorithms.

In Adaptive Routing Algorithms, Routing decisions are changed, to reflect the corresponding changes in the topology and also the traffic. Where as in Nonadaptive Routing Algorithms, routing decisions does not depend on the topology or the traffic. That is in Nonadaptive Algorithms, Routing is static where as in Adaptive Routing Algorithms it is Stochastic or Dynamic.

In this thesis we focus on the Dynamic or Stochastic (Adaptive) Routing Algorithms only. The Routing Algorithm is that part of the network Software, which is responsible for deciding on the route/output line of an incoming packet, has to be transmitted in order to reach its destination system. Naturally, there exist multiple ways/paths in between the Source System and the Destination System. Hence, there is a necessity to identify Shortest Path Routing for quick and efficient way of transmitting the data/packet from source to destination. Such algorithms are known as ‘Shortest Path Routing Algorithms’, which play a vital rôle in identifying the shortest paths in different networks. The present thesis makes an effort in identifying the shortest paths with respect to various networks like:

I) Shuffle Net
II) De Bruijn Graphs
III) Kautz Graphs
IV) Modified De Bruijn Graphs
Detailed discussions on the above mentioned networks are given in the forthcoming chapters.

The Concept of Shortest Path is identified by using any one of the following parameters namely:-

i. **Number of Hops** that a *packet* can take from a Source System to Destination System, denoted by \(H_i\).

ii. **Geographic Distance in kms** between different systems from Source System to Destination System, denoted by \(G_i\).

iii. **Transmission Delay** in different routes from Source System to Destination System, denoted by \(D_i\).

iv. **Transmission Time** taken in hrs to transmit from Source System to Destination System\(T_i\).

Among the above parameters, here, in this thesis we consider the shortest path in terms of number of hops/transmission time i.e., number of intermediary systems required from source to destination systems/Shorter transmission. Thus the shortest path is defined as fastest path rather than the path with other parameters. In other words, we are basically interested to calculate total transmission time required to transmit a *packet* from Source system to Destination system having smaller number of hops \(H_i\)/Shorter transmission time but not the distance of that path what we have chosen. For instance, some paths may be shorter in distance but because of heavy traffic and frequent traffic jams that occur in that path result in increasing total transmission time required to travel from ‘A’ to ‘B’. Thus one may choose a possible route from ‘A’ to ‘B’ which may be longer in distance but has less traffic and hence result in shorter travel time. Even though the second route is longer in distance, we
consider it as shortest distance based on the travel time required. Thus shortest path is that route which has lesser number of hops / smallest transmission time [67].

Thus the following parameters like Shortest path, Transmission time, Bandwidth, Average traffic, Average number of hops, Maximum edge load on the system, communication cost, mean queue length, measure the delay and network utilization. These are some of the parameters on which we are interested, with respect to the different routing schemes.

Based on the number of hops and transmission time, we choose to calculate the shortest path algorithms which are to be carefully written to measure the shortest path between the systems in a given graph.

It is important to note that the shortest path between 'A' to 'B', need not be shortest from 'B' to 'A' in directed graphs, whereas it will be the same in undirected graphs. In this thesis we concentrate on directed graphs only and hence, the shortest path (in time dimension) between 'A' to 'B' need not be equal to 'B' to 'A'. Hence, we are talking with respect to time rather than geographical distance between different systems. For example, day time traffic is different from night time traffic and hence, travel time may change from 'A' to 'B' and 'B' to 'A' depending upon the time of travel, we make the journey. Same way transmission time also changes from 'A' to 'B' and 'B' to 'A' depending upon the number of packets generated and transmitted.

Usually while writing algorithms one has to take into account the topology, load, traffic from 'A' to 'B', the distance and so on.
Thus routing schemes are of different types [67] namely: -

i) Flooding

ii) Flow–Based Routing

iii) Distance–Vector Routing

iv) Link–state Routing

v) Hierarchical Routing

vi) Routing for mobile hosts

vii) Broadcast Routing

viii) Multicast Routing.

These routing schemes are discussed in detail in the section [1.4] of this chapter below.

In this thesis, we have considered the following performance measures [75] for different networks which are explained in the following sections.

1.3. Performance Measures of Networks:

Performance issues are very important and play a vital role in Networks when hundreds and thousands of systems are connected together. Complex interactions, with unforeseen consequences are very common in practice. These complexities lead to poor performance of the Network and no one will be able to know the specific cause for the poor performance "thus understanding Network Performance is an Art rather than a Science". Thus there exists very little theory about network performance which will be helpful to be used in practice. The best we can do is to give some thumb rules gained from Hard Experience and Practice. The performance issue is not only the issue related to the transport layer but related to
other layers like **Physical layer, Data link layer, Network layer, Presentation layer** and **Application layer**. A detailed discussion about these layers is done in [1.4]

Measuring Network Performance is a complicated [76] issue and hence in this thesis we have identified some parameters closely related to the Performance of any network namely:

i) Total Number of Nodes (N)

ii) Average Hop lengths (H)

iii) Average Edge Loading (L)

iv) Maximum Edge Loading (L_{\text{MAX}})

v) Average Network utilization (U_{\text{AVG}}) and

vi) Throughput (\lambda)

### 1.3.1. Notations, Definitions of various Parameters of Networks:

Let ‘N’ represent Total number of independent systems / nodes / machines / computers connected in a Network. This ‘N’ varies from Network to Network depending upon two factors namely the **Degree of the Network** denoted by ‘\Delta’ and **diameter of the network** denoted by ‘d’. ‘\Delta’ and ‘d’ are called the **parameters of the graph**. Total Number of Nodes ‘N’ depends on the values of ‘\Delta’ and ‘d’ in different networks.

#### 1.3.1.1 Degree of the Network (‘\Delta’)

Is the number of edges with the vertex as an end-point.

#### 1.3.1.2 Diameter of the Network (‘d’)

Largest distance/Maximum number of hops between any source and destination nodes.
1.3.1.3 **Average number of hops of a network** is defined as the distance measured in number of hops between any source and Destination nodes. It is calculated as the average of all Source and Destination pairs of Nodes and is denoted by (H).

1.3.1.4 **Edge** :- The path connecting two nodes is known as an Edge / Link.

1.3.1.5 **Edge Loading** :- The loading on edge is defined as the number of Source Destination pairs that use that Edge 'i' for communication. 'i\textsuperscript{th}' edge loading is denoted by (L\textsubscript{i}).

1.3.1.6 **Maximum Edge loading**: The maximum value among all L\textsubscript{i}'s called maximum edge load denoted by (L\textsubscript{MAX}).

Where, \( L_{\text{MAX}} = \text{Max} \{ L_1, L_2, \ldots, L_m \} \), where 'm' is the Number of edges in the network.

1.3.1.7 **Average Edge Loading** is denoted by (\( \overline{L} \)) which is average of edge loadings of (1.3.1.6) all edges M.

1.3.1.8 Using M/M/1 queuing Model [76][44] the **Average queue delay** experienced by a packet along a link is defined as throughput denoted by '\( \lambda \)'.

Where, \( \lambda = 1/L_{\text{MAX}} \)

1.3.1.9 In a Queuing system **Traffic intensity utilization factor** is denoted by 'p'.

On similar lines the Network utilization here is denoted by \( U_{\text{AVG}} \). Where \( U_{\text{AVG}} \) is defined as \( U_{\text{AVG}} = \frac{\overline{L}}{L_{\text{MAX}}} \)

1.4. **Literature Review** :-

In literature review we focused detailed discussion of all layers of the Standard Open Systems Interconnection (OSI) Reference Model and focused on different routing schemes and also the developments of Network.
1.4.1 Open Systems Interconnection (OSI) Reference Model:

The OSI models are based on a proposal developed by the International Organization for Standardization (ISO) as a first step towards international standardization of the protocols used in the various layers (Day and Zimmermann, 1983). The Model is popularly known as OSI (Open Systems Interconnection) Reference Model because it deals with connecting open systems – that is, systems that are open for communication with other systems. It is usually referred as OSI model for short.

The ISO/OSI Reference Model separates computer-to-computer communication into seven protocol layers or levels. Each layer builds and relies upon the standards contained in the levels below it [76]. The lowest of the seven layers (Physical Layer) deals solely with hardware links, while the highest layer (Application Layer) deals with software interactions at the application program level. Functions of Different layers are explained below.

The Seven Layers of ISO/OSI Reference Model are:

7. Application Layer
6. Presentation Layer
5. Session Layer
4. Transport Layer
3. Network Layer
2. Data Link Layer
1. Physical Layer
Various guiding principles that were applied to arrive at the above seven layers are as follows:

a) A layer should be created where a different level of abstraction is needed.

b) Each layer should perform a well defined function.

c) The function of each layer should be chosen by keeping an eye on internationally standardized protocols.

d) The layer boundaries should be chosen to minimize the information flow across the interfaces.

e) The number of layers should be large enough so that distinct functions need not be thrown together in the same layer out of necessity, and small enough that the architecture does not become unwieldy [76].
1.4.1.1 Physical Layer:

The Physical layer is concerned with transmitting raw bits over communication channel. The design issues have to do with making sure that when one side sends a ‘1’ bit, it is received by the other side as ‘1’ bit, but not as a ‘0’ bit. Typical Questions here are how many volts should be used to represent a ‘1’ and how many for ‘0’, how many microseconds a bit lasts, whether transmission may proceed simultaneously in both directions, how the initial connection is established and how it is torn down when both sides are finished, and how many pins the network connector has and what each pin is used for. The design issues here, largely deal with mechanical, electrical, and procedural interfaces, and the Physical transmission medium, which lies below the physical layer.

1.4.1.2 Data Link Layer:

The main task of the data link layer is to take a raw transmission facility and transform it into a line that appears free of undetected transmission errors to the network layer. It accomplishes this task by having the sender break up the input data into data frames (typically a few hundred or a few thousand bytes), transmit those frames sequentially and process the acknowledgement frames sent back by the receiver. Since the physical layer merely accepts and transmits a stream of bits without any regard to meaning or structure, it is up to the data link layer to create and recognize frame boundaries.

Another issue that arises in the data link layer (and most of the higher layers as well) is how to keep a fast transmitter from drowning a slow receiver in data. Some traffic regulation mechanism must be employed to let the transmitter know how much
buffer space the receiver has at that moment. Frequently, this flow regulation and the error handling are integrated.

If the line can be used to transmit data in both directions, this introduces a new complication that the data link layer software must deal with. The problem is that the acknowledgement frames for 'A' to 'B' traffic complete for the use of the line with data frames for the 'B' to 'A' traffic.

1.4.1.3 Network Layer:

The Network Layer is one level above the data-link layer and ensures that information arrives at its intended destination correctly. The network layer is concerned with controlling the operation of the subnet. A key design issue is determining how packets are routed from source to destination. Routes can be based on static tables that are 'wired into' the network and rarely change. They can also be determined at the start of each conversation, for example a terminal session. Finally, routes can be highly dynamic, being determined a new one for each packet, to reflect the current network load.

If too many packets are present in the subnet at the same time, they will get in each other's way, forming bottlenecks. The control of such congestion also belongs to the network layer.

Since the operators of the subnet may well expect remuneration for their efforts, there is often some accounting function built into the network layer. At the very least, the software must count how many packets or characters or bits are sent by each customer, to produce billing information. When a packet crosses a national border, with different rates on each side, the accounting can become complicated.
When a packet has to travel from one network to another many problems can arise to get to its destination. Though the addressing is used but, the second network may have a different system of addressing from the first one. The second one may not accept the packet at all because it is too large. The protocols may differ, and so on. It is up to the network layer to overcome all these problems to allow heterogeneous networks to be interconnected.

In broadcast networks, the routing problem is simple, so the network layers are often thin or even nonexistent in them.

1.4.1.4 Transport Layer:

The Transport Layer is one level above the network layer and is responsible for both quality of service and accurate delivery of data through error detection and correction. The basic function of the transport layer is to accept data from the session layer, split it up into smaller units if needed pass these to the network layer, and ensure that all the pieces arrive correctly at the other end. Further, all this must be done efficiently, and in a way that insulates the upper layers from the inevitable changes in the hardware technology.

If the transport connection requires a high throughput, however, the transport layer might create multiple network connections, dividing the data among the network connections to improve throughput. On the other hand, if creating or maintaining a network connection is expensive, the transport layer might multiplex several transport connections into the same network connection to reduce the cost. In all the cases, the transport layer is required to make the multiplexing transparent to the session layer.
The transport layer is a true end-to-end layer, from source to destination. In other words, a program on the source machine carries on a conversation with a similar program on the destination machine, using the message headers and control messages. In the lower layers, the protocols are between each machine and its immediate neighbours, and not by the ultimate source and destination machines, which may be separated by many routers.

The transport layer must take care of establishing and deleting connections across the network. This requires some kind of naming mechanism, so that a process on one machine has a way of describing, with whom it wishes to converse. There must also be a mechanism to regulate the flow of information, so that a fast host cannot overrun a slow one. Such a mechanism is called flow control and plays a key role in the transport layer as well as in all in other layers. Flow control between hosts is distinct, from flow control between routers, although we will later see that similar principles apply to both.

1.4.1.5 Session Layer:

The fifth of seven layers in ISO/OSI Reference Model, the session layer allows users on different machines to establish sessions between them. A session allows ordinary data transport, so does the transport layer, but it also provides enhanced services useful in some applications. A session might be used to allow a user to log into a remote time sharing system or to transfer a file between two machines.

One of the services of the session layer is to manage dialogue control. Sessions can allow traffic to go in both directions at the same time, or in only one
direction at a time. If traffic can only go one way at a time, analogous to a single-track railroad, the session layer can help keep track of whose turn it is.

Another session service is Synchronization. Consider the problems that might occur when trying to do a 2-hours file transfer between two machines with a 1-hour mean time between crashes. After each transfer was aborted, the whole transfer would have to start-over again and would probably fail again the next time as well. To eliminate this problem, the session layer provides a way to insert check-points into the data stream, so that after a crash, only the data transferred after the last check-point have to be repeated.

1.4.1.6 Presentation Layer:

This presentation layer deals with text formatting and display, code conversion for printing etc. It performs certain functions that are requested quite often to warrant finding a general solution for them, rather than letting each user to solve the problems themselves. In particular, unlike all the lower layers, which are just interested in moving bits reliably from here to there, the presentation layer is concerned with the syntax and semantics of the information transmitted.

A typical example of a presentation service is encoding data into a standard agreed upon format. Most user programs do not exchange random binary bit strings. They exchange things such as people's name, dates, amounts of money, and invoices. These items are represented as character strings, integers, floating-point numbers, and data structures composed of several simpler items. The present layer manages these abstract data structures and converts from the representation used inside the computer to the network standard representation and back.
1.4.1.7 Application Layer:

The highest layer in this OSI Reference Model performs useful tasks like file transfer and remote access to a computer. The application layer contains a variety of protocols that are commonly needed. For example, there are hundreds of incompatible terminal types in the world. Consider the plight of a full screen editor that is supposed to work over a network with many different terminal types, each with different screen layouts, escape sequences for inserting and deleting text, moving cursor, etc.

Transferring a file between two different systems requires handling these and other incompatibilities [76]. This work, too, belongs to the application layer, as does electronic mail, remote job entry, directory lookup, and various other general purpose and special-purpose facilities.

1.4.2 Routing Schemes:

Further, it is already mentioned that routing is the basic function related to the network layer and various routing techniques exists based on different schemes as discussed in (1.2). Now we proceed to discuss in depth some details about various routing schemes as follows.

1.4.2.1 Flooding

Flooding is the most extreme form of isolated routing. Every incoming packet is sent out on every outgoing line except the one it arrives on. This method generates a large number of duplicate packets, an infinite number if damping is not used. One way of damping is to have a hop counter embedded in the header of each packet, which is decremented at each hop with the packet being discarded when the counter reaches zero. Ideally, the hop counter should be initialized to the length of the path.
from source to destination. If the sender is unaware of the length then the counter is initialized to the diameter of the network.

Another method for damping is through the use of sequence numbers. The source node attaches a sequence number with every packet it receives. Each node then needs a list per source node telling which sequence numbers originating at that source have been already received. For the sake of brevity, each list is augmented by a counter, ‘k’ (‘k’ is the number of packets a node received) indicating that all sequence numbers through ‘k’ have been seen. When a packet comes in, it is easy to check if the packet is a duplicate. If it is found to be one, then that packet is discarded.

Flooding is not practical in most applications. It is used in certain critical applications, like military, where its tremendous robustness is desirable.

A variation of flooding that finds more use is selective flooding, which is similar to flooding that transmits incoming packets to specific output lines.

Another robust non-adaptive procedure is the random walk [64]. The node just picks up a link at random and forwards the packet on it. The node can also be made to direct the packet towards the destination if some knowledge of the topology is available. If the subnet is richly interconnected, this algorithm makes excellent use of the alternative routes.

1.4.2.2 Flow Based Routing

In some networks, the mean data flow between each pair of nodes is relatively stable and predictable. In such networks it is possible to analyze the network flows mathematically to optimize the routing as discussed by Bertsekas and Gallanger (1992) [9]. The principal idea behind the analysis being that for a given link, if the
capacity and average flow are known, it is possible to compute the mean packet delay on that link using queuing theory. Using the mean delay on all the links, a flow-weighted average can be directly calculated to get the mean packet delay for the whole network. The problem then reduces to finding the routing algorithms that produces the minimum average delay for the network.

In order to utilize the technique, certain parameters must be known in advance. This includes the network topology, the traffic matrix and the line capacity matrix. For the given values of these parameters the algorithm that minimizes the average delay is deemed to be the efficient algorithm.

1.4.2.3 Distance Vector Routing:

Modern computer networks generally use dynamic routing algorithms rather than the static ones described above. Two dynamic algorithms in particular, distance vector routing and link state routing, are the most popular. In this section we will look at the former algorithm.

Distance vector routing algorithms operate by having each router maintain a table (i.e. a vector) giving the best known distance to each destination and which line to use to get there. These tables are updated by exchanging information with the neighbors. The distance vector routing algorithm is sometimes called by other names, including the distributed Bellman-Ford routing algorithm and the Ford-Fulkerson algorithm, after the researchers who developed it. It was the original ARPANET routing algorithm and was also used in the Internet.

In distance vector routing, each router maintains a routing table indexed by, and containing one entry for, each router in the subnet. This entry contains two parts:
the preferred outgoing line to use for that destination, and an estimate of the time or
distance to that destination. The metric used might be number of hops, time delay in
milliseconds, total number of packets queued along the path, or something similar.

The router is assumed to know the distance to each of its neighbors. If the
metric is hops, the distance is just one hop. If the metric is queue length, the router
simply examines each queue. If the metric is delay, the router can measure path is of
length ‘N’ hops, within ‘N’ exchanges everyone will know about newly revived lines
and routers.

1.4.2.4 Link State Routing :

Distance vector routing was used in the ARPANET until 1979, when it was
replaced by link state routing. Two primary problems caused its demise. First, since
the delay metric was queue length, it did not take line bandwidth into account when
choosing routes. Initially, all the lines were 56 kbps, so line bandwidth was not an
issue, but after some lines had been upgraded to 230 kbps and others to 1.544 Mbps,
not taking bandwidth into account was a major problem. Of course, it would have
been possible to change the delay metric to factor in line bandwidth, but a second
problem also existed, namely, the algorithm often took too long to converge, even
with tricks like split horizon. For these reasons, it was replaced by an entirely new
algorithm now called link state routing. Variants of link state routing are now
widely used.

The idea behind link state routing is simple and can be stated as five parts.
Each router must

a) Discover its neighbors and learn their network addresses.

b) Measure the delay or cost to each of its neighbors.
c) Construct a packet telling all it has just learned.

d) Send this packet to all other routers.

e) Compute the shortest path to every other router.

In effect, the complete topology and all delays are experimentally measured and distributed to every router. The Dijkstra’s algorithm can be used to find the shortest path to every other router.

1.4.2.5 Hierarchical Routing:

In large networks, it is not economical to maintain routing table entries corresponding to every node in the network. In such networks hierarchical routing [38] is used. The network is divided into regions made of a cluster of nodes, each node knowing all the details about how to route packets to the destinations within its own region, but ignorant about the internal structure of other regions. This method of routing is also followed in interconnected networks.

For such huge networks, a multi-level hierarchy is formed by grouping regions into clusters, clusters into zones, zones into groups and so on.

1.4.2.6 Routing for Mobile Hosts:

Millions of people have portable computers nowadays, and they generally want to read their email and access their normal file systems wherever in the world they may be. These mobile hosts introduce a new complication: to route a packet to a mobile host, the network first has to find it. The subject of incorporating mobile hosts into a network is very young, but in this section we will sketch some of the issues and give possible solutions.
1.4.2.7 Broadcast Routing:

Some applications may require that every node should send message to every other node, as in distributed data bases. The simultaneous transmission of a packet to all destinations is called broadcasting [16].

1.4.2.8 Multicast Routing:

For some applications, widely-separated processes work together in groups, for example, a group of processes implementing a distributed database system. It is necessary frequently, for one process to send a message to all the other members of the group. If the group is small, it can just send each other member a point-to-point message. If the group is large, this strategy is expensive. Sometimes broadcasting can be used, but using broadcasting to inform 1000 machines on a million-node-network is inefficient because most receivers are not interested in the message or worse yet, they are definitely interested but or not supposed to see it. This situation needs a way to send messages to well-defined groups that are numerically large in size but, relatively small compared to the network as a whole.

Sending a message to such a group is called multicasting, and its routing algorithm is called multicast routing.

To do multicasting, group management is required. Some method is needed to create and destroy groups, and for the processes to join and leave groups. How these tasks are accomplished is not of concern to the routing algorithm. What is of concern is that, when a process joins a group, it informs its host of this fact. It is important that routers know which of their hosts belong to which groups. Either host must inform their routers about changes in group membership, or routers must query their hosts periodically. Either way, routers learn about which of their hosts are in
which groups. Routers tell their neighbors, so the information propagates through the subnet.

1.4.2.9 Self Routing:

In this class, we put routing techniques that either avoid routing tables completely or use static, possibly incomplete, routing tables which seldom or never updated during the normal operation of the network. With self routing, a switch accepting an incoming packet is able to determine its path locally without consulting the network’s database in its centralized or distributed form. The price paid for simplicity of the routing algorithm is its sub-optimal character. The gain is in the low cost of routing, the simplicity of the switch, and the absence of administrative traffic in the network [12].

1.4.2.10 Optimal Routing:

This class of algorithms are based on the optimality principle, which states that if a node ‘A’ lies on the optimal path from node ‘X’ to node ‘Y’, then the optimal path from node ‘A’ to node ‘Y’ falls along the same route. The direct consequence of the optimality principle is that the set of optimal routes from all sources to a given destination form a tree, rooted at the destination. Such a tree is called the ‘Sink tree’.

If traffic from node ‘X’ passes through node ‘Y’ as it flows along the sink tree to the destination, ‘X’ is said to be upstream from ‘X’, and ‘Y’ is said to be downstream from ‘X’ when a link goes down, blocking the path to a certain destination, a node cannot simply divert its traffic to another node that is upstream from it with respect to destination. The algorithm fails if none of the nodes upstream
from the failed link can make contact with any node on another branch of the 'sink tree'. This situation arises when the subnet has been broken into two separate components.

1.4.3 ARPANET and INTERNET

A major development in the network in the early 1950s was the use of communication links to connect central computers to remote terminals. The Large Wide Area Network (WAN) created by the U.S. Department of Defense, ARPANET (Advanced Research Projects Agency Network) and TYMNET (Timeshare Incorporated Network) introduced in 1970s were the first large-scale general-purpose computer networks connecting geographically distributed computer systems.

ARPANET was the Network from which the INTERNET has evolved. Eventually the Internet became the largest network of computer systems large and small, distributed over the entire globe trespassing the physical boundaries of the countries, continents and oceans. Internet has grown in leaps and bounds from its birth in 1969 to an enormously large Network of networks with practically an agglomeration of millions of computers, hosts and servers from all over the world, communicating with each other and sharing the data/information. It is difficult even to make a guess, how many millions of computers are connected through the Internet at any given moment. The proliferation of computers as well as the growth application software over the last 50 years has added to the stupendous growth of the Internet.

The Success of the Internet has lead to the creation of even more versatile INTERNET2 in 1996, by a Consortium of Universities in US working in tandem
with the Industry and the Government there. The Internet2 is not yet been open to the
general public for use.

1.4.4 Some Salient Features of the Network Technology:

Since 1970, there has been an explosive growth in the number of wide area
networks (WANs) and local area networks (LANs). There are two approaches,
namely circuit switching and store-and-forwarded switching that can be used within a
subnet to transmit the traffic for the various sessions. Circuit switching is rarely used
for data networks because of inefficient use of the links. Typical data sessions tend to
have short bursts of high activity followed by lengthy inactive periods. Besides
circuit switching wastes the allocated rate during these inactive periods. The store
and forward approach in its subnet design, each session is initiated without
necessarily making any reserved allocation of transmission rate for the session.
Hence, this type of switching has the advantage over circuit that each communication
link is fully utilized whenever it has any traffic to send.

Most of the algorithms for computing the shortest path between two nodes of a
graph are due to Dijkstra [19] in the 1959. The routing algorithm for ARPANET
datagram routing, was first implemented in 1969 and has played an important role in
the development of routing techniques. This was a distributed, adaptive
algorithm. Further research has identified some fundamental flaws that were
corrected in 1979. In the second version of the ARPANET algorithm presented in
1980 [54] shown the shortest-path-first, in which the length of each link is calculated
by keeping track of the delay of each packet in crossing the link.
The TYMNET routing algorithm (virtual circuit routing), implemented originally in 1971, is based on the shortest path method which is adaptive but following centralized routing technique.

In some networks, the storage space at each node is limited where it requires modifying the routing algorithm so as to minimize buffer overflow resulting in loss of packets. This method suggested by Baran in 1964 [8] is popularly known as hot potato routing. In recent research, this is being known as deflection routing in which some packets may travel on long routes to their destination.

The primary objective in designing a routing algorithm is to generate a decision-making procedure that will perform the routing function at a lower cost. The various types of network routing, function at a lower cost. The various types of network routing algorithms are classified as non-adaptive or adaptive. The adaptive algorithms are further classified into centralized, isolate and distributed algorithms presented by A.S.Tanenbaum [75] in 1981. The one factor which most distinguishes a packet-switched network from the more conventional circuit-switch network is the fact that information to be transmitted is broken into small pieces or packets, which are then transmitted at high speed from node to node until the destination is researched. An excellent discussion of earlier work in the area of routing in packet-switched works was given by Fultz [24] in the year 1972.

In mid 1971, one of the largest commercial data communication network known as TYMNET [79] was introduced, which uses a centralized approach to routing. The most widely publicized research data communication is the ARPANET that uses a completely distributed approach. In TYMNET having a relatively invariant routing decisions algorithm based on an estimation of global performance, and, the scheme was implemented centrally. The ARPANET strategy is relatively
more adaptive; is based on local information added to global information and is a distributed technique.

The design of distributed adaptive routing algorithm for packet switching computer networks gives raise to many problems.

The research activities on the original routing algorithm for ARPANET are given, in the year 1977, by J.M McQuillan [53] and, in the year 1980, a new ARPANET [54] routing algorithm is implemented which is an improvement over the old procedure. This new algorithm operates on more realistic estimates of network auditorium, reacts faster to important network changes, and does not suffer from long-term loops or oscillations. In this new procedure, each node in the network maintains a database, describing complete network topology and delays on all lines. Each node, periodically measures the delay along its outgoing links and forwards the information to all other nodes. All nodes update their databases and continue to route traffic in a consistent and efficient manner. In the same year (1980), Eric C. Rosen [63] had suggested the good routing update protocol. The ISO /OSI Reference Model described in 1983 by Day and Zimmerson [17] deals with connecting open systems, that is, systems that are open for communication with other systems. Layers of the OSI reference model offer two different types of services to the layers above them namely, connection-oriented and connectionless described by Chapin in the year 1983 [13] and Bucciarelli and Caneschi [11], in the year 1984.

Recent developments in optical technology have made it possible to transmit data by pulses of light. Multi-user local communication systems, based on light wave technology have become a topic of keen interest. For, long-distance point-to-point communications, light wave has emerged as the technology of choice as a result of the very high bit rate-distance product which could be achieved, basing on
the discussions by S.E. Miller and A.G. Chynoweth [56] in the year 1979; by T. Li [49] in the year 1983 and by P.S. Henry [32] in the year 1985. When applied to multi-user systems, light wave technology holds forth the potential of enormous bandwidth for each end-user.

With the emergence of applications such as speech, high resolution graphics, facsimile, video and so on, on local area networks, requiring high volume of traffic and real time delivery of data, there was a need for high bandwidth transmission medium. **Ring topologies** were implemented on fiber-optic medium to suit only point-to-point communications between neighboring stations, as presented by W. Stallings in the year 1984 [73]. Later, in the same year, Fine, M and Tobagi, F. A introduced high-performance bus-oriented local area networks through which it became feasible due to the emergence of a new class of **Demand Assignment Multiple Access (DAMA)** schemas [22], that can efficiently utilize high bandwidths. Distributed routing algorithms, based on the regular topology, were described in the years 1985 and in 1987 by N.F. Maxemchuk [50], [51]. Although the lightwave transmission medium possesses a bandwidth which might be measured in the tens of Terahertz range, a single user port on the network is limited by the rate at which light may be electro-optically modulated or demodulated as observed by the experiments conducted by B.L. Kasper et. al. [41] in 1984. Followed by this, in the year 1987, A.S. Acompore [1] proposed new approaches to achieve lightwave network concurrency, entitled “A multichannel multihop lightwave network”. With this approach, it was possible, to provide an aggregate network capacity of tens or hundreds of Gigabits/sec range via a collection of interfaces, each of which operates at only one Gigabit/sec. It is also stated that physically the multihop approach was appropriate for bus, star and tree topologies. For all the topologies, active electronic
repeater circuitry is decentralized, appearing in close proximity to the end-user locations. In an all-optical network, concurrency may be provided according to either wavelength or frequency (Wavelength Multiple Division Access - WMDA), time slots (Time Division Multiple Access – TDMA), or wave shape (Code Division Multiple Access - CDMA) which have been investigated by Brazio, J.M., and Tobagi, F.A. [10] in the year 1984. By virtue of its low-loss, low-dispersion properties, photonic transport technology has enabled reliable digital communications at very high speeds over long-unrepeated dispenses. Further, investigations in the year 1992, on fiber optic networks was driven by emerging applications. A review of the characteristics of lightwave technology that facilitates the design of WDM (Wavelength Division Multiplexing) networks and construction of WDM local networks, based on the single-hop and multi-hop approaches, as given by Biswanath Mukherjee in the year 1992 [58].

Generally, there are two classes of architectures that can be contructed for WDM local and metropolitan area networks, namely, single-hop and multi-hop. Multi-hop structures can be either irregular or regular. Irregular structures generally address the optimality criterion directly, but the routing complexity can be large, since they lack any structural connectivity-pattern as reported in literature [7], [6] and [45]. Regular structures, because of their structured node-connectivity pattern, have simplified routing schemes, however, their regularity also constrains the set of solutions in addressing the optimality problem, and the number of nodes in a complete regular structures from a discrete set of integers rather than an arbitrary integer as reported by B. Mukherjee in the year 1992 [59].

Regular structures which have received a significant amount of attention in the literature are the perfect shuffle, called Shuffle net by different authors in [1], [2]
and [33]; the De Bruijn Graph by R. Ramaswami and K. N. Sivarajan [43] and [44] in
the year 1991 and 1994; the Manhattan Street Network (MSN) by N. F. Maxemchuk
year 1992; the Hypercube by P. W. Dowd [20] in the year 1991 and later on by B. Li
S. Banerjee et. al., [5] and by T. D. Todd, et. al., [78].

WDM network operate on a broadcast and select principle. They are
classified into single-hop (also called Direct WDM) and multi-hop networks. From
the research of M. S. Goodman et. al., [26] in the year 1990 and B. Mukherjee [58],[59]
it was found that there were technological difficulties in implementing WDM in a
single-hop approach, such as requirement of rapid tenability and pre-transmission
coordination. The multi-hop approach was first proposed by Acampora [1] and then
by B. Mukherjee [59]. In the multi-hop WDM, the data may visit intermediate nodes
before reaching the destination; each node is equipped with a small fixed number, (for
example ‘2’ or ‘3’) of optical transmitting wavelengths. In order to manage the multi­
hop WDM network, it is strongly required to develop routing methods that would
minimize the number of hops, that minimizes average packet delay.

In order to enhance the performance of these architectures, it was desirable to
design a routing scheme, simple and fast, without sacrificing much in terms of
throughput and delay performance, known as deflection routing which is similar
to ‘hot-potato routing’ presented in the year 1964 by P. Baran [8] and shortest queue
plus bias proposed by G. L. Fultz and L. Kleinrock [24].

The method, deflection routing is used in high-speed multihop networks due
to its simplicity of implementation. Its basic principle is to eliminate the need for
storage at intermediate node in high-speed network, since this requires high-speed
memory, which is either not available or expensive [4]. According to the deflection routing method, the packets are forwarded through the network in fixed-size slots. The slots arriving at any node are delayed in that they arrive at the switching point simultaneously. If there is a contention for the same outgoing channel, a contention resolution is applied to resolve the conflict. In the year 1991, Robertazzi Thomas and Lazer Aural A, have presented deflection strategies based on Hop-length counter, a deflection counter or distance information [77], [14]. B. Hajek proposed Bounds for evacuation time in the application of deflection routing [31].

A number of researchers have presented the performance of deflection routing in various regular networks under the uniform traffic model, characterized through analysis by A.G.Greengerg and J.Goodman [29], [30], on Manhattan Street network in the year 1989, 1993 and by Choudhury, A.K and Li , V.O.K [15] on MSN in the year 1991.

A major factor in realizing highly reliable and efficient networks with a limited number of links is finding a graph with a minimal diameter and a maximal connectivity for a given number of nodes and a given degree as described R.S.Wilko [80] in the year 1972. The diameter of a graph is related to efficiency factors such as network delay and transmission capacity, while connectivity is directly related to the fault-tolerance capacity of networks. Several methods of constructing diagraphs with a small diameter have been proposed by N.G.De Bruijn [18] in the year 1946. M.Imase and M.Itoh, [35], [36] in the year 1981, 1983, S.M.Reddy et.al., [62] in the year 1997 and later by Matoto Imase et.al.,[37].

The De Bruijn Graph has extensively been used in various contexts, like the design of feed-back registers by A.Lempel (1970) [47] and H.Fradericksen (1982) [23], point-to-point computer networks by Pradhan, D.K(1985) [60].
The connectivity of the De Bruijn Graph has been studied by several authors H.Lee [46], M.A.Schlumberger [70], and it is known that this graph is (d-1) connected. Further research on De Bruijn Graph was presented by M.A.Sridhar [72] in the year 1988, in which the connectivity of the De Bruijn Graph is at least, (d-1). Fault-tolerant network based on the De Bruijn Graph was discussed by M.A.Sridhar and C.S.Raghavendra [71] in the year 1991.

In earlier literature [1] on multi-hop lightwave networks, have described that logical topology can be superimposed on an arbitrary physical topology, for example, consider a star topology. In the years 1991 and 1994, Kumar Sivarajan and Rajiv Ramaswami [43], [44] proposed De Bruijn Graphs as a new logical topology for multihop lightwave [28] networks. In the same year, Guoping Li et al., proposed optical networks namely Time Division Multiplexed De Bruijn network (Δ,d) [1],[33], respectively.

In the year 1995, Cesur Baransel et al., [12] surveyed networking solutions that use for high-speed packet-switched applications. They identified the specific problems resulting from high transmission rates and concluded that the solutions based on deflection routing use the most promising ones. In the same year (1995) Fabrizio et al., [21] presented an analysis of steady state behavior of regular two-connected multihop networks in uniform traffic under hot-potato and a single-buffer deflection routing technique. Most of the available logical topologies for lightwave networks in the literature are not scalable, which makes the addition of new stations to an existing network extremely difficult. On these lines, the researchers Anuradha Benkateswaran and Abhijit Sengupta [3] in the year 1996 presented a scalable topology for lightwave network called GEMNET, based mainly on Shufflenet. In the same year, Milkar, Armin R, and et al., have developed a set of heuristic decision
functions that can be used to guide messages along a near-optical path in a large network [55]. Hsu, D.Frank and Wei, David S.L. have discussed the problems of routing and storing on a De Bruijn network in the year 1997 [34]. In the year 1999, Xie, Chongjin and Le Peida [81] have proposed the Markov chain approach for the analysis of both store-and-forward and deflection routing in Manhattan Street Network and Shuffle net.

Kautz Graph architecture was proposed by Kautz for Multiprocessor interconnection in 1969[42]. Kautz graph was proposed as a multihop optical network topology by Panchapakesan, G and Senugupta [61] in the year 1995.

In the Year 1999 B. Satyanarayana Proposed a new logical topology which is a modified version of the De Bruijn Graph and being named as Modified De Bruijn Graph for multihop lightwave networks [67].
1.5. Chapter Summaries:

This thesis consists of six chapters whose broad outline is given as follows:-

Chapter 1: This is an introductory chapter where Notations, Definitions and different types of networks, routing schemes, various parameters related different networks and routing algorithm concepts are explained. A brief discussion on literature review is also given finally. The chapter is concluded with chapter summaries.

Chapter 2: In this chapter we consider a network model based on Shuffle Net. Shuffle Net Graphs are drawn for different parameters ('A' and 'd') and their properties are discussed. An algorithm is developed which is useful for calculating various parameters of the network based on Shuffle Net and obtained the results related. Conclusions are drawn based on the results obtained.

Chapter 3: In this chapter Network based on De Bruijn Graph is considered and Graphs are drawn for different parameters ('A' and 'd'). Comparisons with respect to Shuffle Net and properties are discussed. An algorithm for calculating various parameters of De Bruijn Graphs is written and calculated the required results for the network based on the De Bruijn Graphs. Conclusions are drawn based on the results obtained.

Chapter 4: In this chapter, Network based on Kautz Graphs are discussed, graphical represents of the parameters ('A' and 'd') are drawn and compared with the networks discussed in the other chapters. An algorithm is developed for calculating the various parameters of the Network based on Kautz Graphs. Calculated required results and analysed. Conclusions are drawn based on the results obtained.
**Chapter 5**: In this chapter we have proposed a network based on Modified De Bruijn Graphs. Schematic representation of graph for parameters ('A' and 'd') are drawn and compared with the Graphs of the other chapters. An algorithm is developed for calculating various parameters of the network and calculated the required results. Conclusions are drawn based on the results obtained.

**Chapter 6**: This is the last chapter and the most important chapter, where a critical comparisons between different networks introduced in this thesis in earlier chapters are made. Based on the performance measurers calculated earlier, conclusions are drawn based on the results obtained and discussions are made to select the Best /optimum network using ranking method with respect to various parameters under consideration.

The chapter is concluded with **further scope of the work**.

Finally, the thesis is concluded with an elaborate list of references and the websites we referred for this thesis.