CHAPTER 2

ROUTING PROTOCOLS

2.1 INTRODUCTION

Routing is an act of moving information across the internet from a source to destination. The problem of finding efficient routing algorithms has been a fundamental research area in the field of data network (Gallager 1997). Routing has been covered in computer science literature for more than two decades, but routing achieved commercial popularity only as late as the mid-1980’s. Routing involves two basic activities, one determining optimal routing paths, and the other transporting information packets through the internet. In optimal routing, path determination routing protocols are used with different metrics. A metric is a standard of measurement used by routing algorithms to determine optimal path to evaluate what path will be the best for a packet to travel. To aid the process of path determination, routing algorithms initialize and maintain routing tables, which contain route information. Route information varies depending on the routing algorithm used (Forouzan 2006, Lippmann 1985).

Routing algorithms fill routing tables with a variety of information regarding destination. Hop associations tell a router that a particular destination can be reached optimally by sending the packet to a particular router, the next hop on the way to the final destination. When a router receives an incoming packet, it checks the destination address and attempts to associate this address with the next hop (Huitema 1995).
Routing tables also can contain other information, such as data about the desirability of a path. Routers compare metrics to determine optimal routes, and these metrics differ depending on the design of the routing algorithm used.

Switching algorithms is relatively simple as it is the same for most routing protocols. In most cases, a host determines that it must send a packet to another host. Having acquired a router’s address by some means, the source host sends a packet addressed specifically to a router’s physical address, this time with the protocol address of the destination host. As it examines the packet’s destination protocol address, the router determines how to forward the packet and it typically drops the packet. If the router knows how to forward the packet however, it changes the destination’s physical address to that as the next hop and transcribes the packet. The next hop may be the ultimate destination host.

2.2 ROUTING PROTOCOL CRITERIA

The main ingredients of a good routing algorithm are dependent on the objective function that one is trying to optimize (Zhang 1993). In general, the routing algorithm should seek one or more of the following goals (Richard 2000, Turner 1992).

Optimality: Optimality refers to the capability of the routing algorithm to select the best route, which depends on the metrics and metric weightings used to make the calculation (Tsitsiklis 1986).

Simplicity and low over head: Routing algorithms also are designed to be as simple as possible and the routing algorithm must offer its functionality efficiently, with a minimum of software and utilization over
head. Efficiency is particularly important when the software implementations of the routing algorithm must run on a computer with limited physical resources (Hwang 1995).

*Robustness:* Routing algorithms must be robust, which means that they should perform correctly in the unusual or unforeseen circumstances, such as hardware failures, high load conditions and incorrect implementations. Because routers are located at network junction points, they can cause considerable problems when they fail. The best routing algorithms are stable under a variety of network conditions.

*Convergence:* Routing algorithms must converge rapidly. Convergence is the process of agreement, by all routers, on optional routes. When a network even causes routers to either go down or become available, routers distribute routing update messages on the permanent networks stimulating recalculating of optimal routes and eventually causing all routers to agree on these routes. Routing algorithms that converge slowly can cause routing loops.

*Flexibility:* Routing algorithms should also be flexible, which means that they should quickly and accurately adapt to a variety of network circumstances. For example, if a network segment has gone down, and many routing algorithms become aware of the problem, they should quickly select the next-best path for all routes normally. Segment routing algorithms should be programmed to adapt to changes in network bandwidth, router queue size and network delay, among other variables.
Routing algorithms use many different metrics to determine the best route. Sophisticated routing algorithms can use route selection on multiple metrics. The following metrics have been used in routing algorithms (Spragins 1991, Bertsekas 1987).

**Path length:** Path length is the most common routing metric. Some routing protocols allow network administrators to assign arbitrary costs to each network link. In this case, path length is the sum of the costs associated with each link traversed. Other routing protocols define hop count, a metric that specifies the number of passes through internet working products, and the routes that a packet must take from a source to a destination.

**Reliability:** The reliability in the context of routing algorithms, refers to the dependability usually described in terms as the bit-error rate of each network link. Some network links might go down more often than others. After a network fails, certain network links might be repaired more easily or more quickly than other links. Any reliability factor can be taken into account in the assignment of the reliability ratings, which are arbitrary numeric values usually assigned to network links by network administrators.

**Delay:** Routing delay refers to the length of time required to move a packet from source to destination through the inter network. Delay depends on many factors, including the bandwidth of intermediate network links, the port queues at each router along the way, network congestion on all intermediate network links and the physical distance to be traveled.

**Band width:** Band width refers to the available traffic capacity of a link. This is a rating of the maximum attainable throughput on a link. Routes through links with greater bandwidth do not necessarily provide better routes than routes through slower links.
Load: The load refers to the degree to which a network resource, such as a router, is busy. Load can be calculated in a variety of ways, including CPU utilization and packets processed per second. Monitoring these parameters on a continual basis can be resource intensive itself (Chang 1986).

Communication cost: Communication cost is another important metric, especially because some companies may not care about performance as much as they care about operating expenditures. Although line delay may be longer, they will send packets over their own lines rather than through the public lines that cost money for usage. Time routing is accomplished by means of routing protocols that establish routing table in every router in the network (Vakil 1987). The routing table contains at least two columns: firstly, the address of a destination node and secondly, the address of the network element, that is, the next hop in the optimal path to the destination node. Figure 2.1 shows how the packets are routed from node A. Table 2.1 shows the next hop for each destination node.

Figure 2.1 Routing the packet in a network with 9 nodes
Table 2.1 Routing packet at node A

<table>
<thead>
<tr>
<th>Destination</th>
<th>Next hop</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>-</td>
</tr>
<tr>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>D</td>
<td>D</td>
</tr>
<tr>
<td>E</td>
<td>D</td>
</tr>
<tr>
<td>F</td>
<td>C</td>
</tr>
<tr>
<td>G</td>
<td>C</td>
</tr>
<tr>
<td>H</td>
<td>D</td>
</tr>
<tr>
<td>I</td>
<td>B</td>
</tr>
</tbody>
</table>

The node ‘A’ specifies path (A, D, H) in the packet to the destination node ‘H’. The node ‘D’ may decide that the better path to H is directly to H or through node D.

2.4 ROUTING ALGORITHM CLASSIFICATIONS

One can classify routing algorithms in several ways based on their responsiveness. Link-state algorithm and distance vector algorithm are the current routing protocol in the internet. Link-state algorithms are also known as the shortest path algorithm. Each router, however, sends only the portion of the routing table that describes the state of its own links. In link-state algorithms, each router builds a picture of the entire network in its routing table. Distance vector algorithm also known as Bellman-Ford algorithms call for each router to send all or some portion of its routing table, but only to its neighbors. In essence, link-state algorithms send small updates everywhere,
while distance vector algorithms send larger updates only to neighboring routers. Distance vector algorithms know only about their neighbors. Because they converge more quickly, link-state algorithms are somewhat less prone to routing loops than distance vector algorithms (William 2002). On the other hand, link-state algorithms require more CPU power and memory than distance vector algorithms. Link-state algorithms, therefore, can be more expensive to implement and support. Link-state protocols are generally more scalable than distance vector protocols.

The routing algorithm is that part of the network layer software responsible for deciding which output line an incoming packet should be transmitted on. Routing algorithms can be grasped into two major classes, such as non adaptive and adaptive routing algorithms. Non adaptive algorithms do not base their routing decisions on measurements or estimates of the current traffic and topology called static routing. Adaptive algorithms, in contrast, change their routing decisions to reflect changes in the topology, and usually the traffic as well, called dynamic routing algorithms (Jean 2000)(Kelly 1991).

2.4.1 Static Routing Algorithm

The static routing paths are pre-computed based on the network topology, link capacities, and other information. The computation is performed offline by a dedicated node. When the computation is completed, the paths are loaded to the routing table and remain fixed for a relatively long period of time. In the static routing, the network topology determines the initial paths. The pre calculated paths are then loaded to the routing table and are fixed for a longer period.
2.4.2 Shortest Path Routing

Shortest path routing is widely used in many forms because it is simple and easy to understand (Keshav 2000, Mehmet 1993). The idea is to build a graph of the subnet, with each node of the graph representing a router and each area of the graph representing a communication line called a link. To choose a route between a given pair of routers, the algorithm just finds the shortest path between them on the graph. The best route from any node to any other node is the shortest path in terms of the number of hops. The routing becomes unstable under heavy loads or traffic when the link cost metric used in the routing algorithm is related to the delays or congestion experienced over the links (Korillis 1995). This type of routing algorithm is used in the source routing. The shortest path routing is shown in Figure 2.2. The source node is A and the destination node is F. The packet travels minimum number of hops to the destination node as A-D-F.

![Diagram](figure-2.2.png)

Figure 2.2 Shortest path routing from A to F

2.4.3 Flooding

Another static algorithm is flooding, in which every incoming packet is sent out on every outgoing line except the one it arrived on.
Flooding obviously generates vast numbers of duplicate packets, in fact, an infinite number, unless some measures are taken to damp the process. One such measure is to have a hop counter contained in the header of each packet, which is decremented at each hop, with the packet being discarded when the counter reaches zero.

### 2.4.4 Flow-Based Routing

Flow-based routing algorithms use both topology and load for routing. To use this technique, certain information must be known in advance. First, the subnet topology must be known. Second, the traffic matrix, the line capacity matrix and routing algorithm must be known. If for example, there is always a huge amount of traffic from A to B in Figure 2.3, then it may be better to route traffic from A to C via AGEFC, even though this path is much longer than ABC (Keshav 1991).

![Figure 2.3 Flow-based routing](image)

### 2.4.5 Dynamic Routing

The dynamic routing algorithm changes routing decision, if there is change in topology, or traffic of routers continuously checks the network status by communicating with neighbors. Thus, a change in network topology
is eventually propagated to all the routers. Based on this information, each router computes the suitable path to the destination. The task of making routing decision is therefore the responsibility of the individual nodes in the network.

The advantages of dynamic routing over static routing are stability and adaptability. A dynamic network grows more quickly, and is larger. It is able to adopt changes in the network topology brought about by this growth or by the failure of one or more network components.

2.4.6 Distance Vector Routing

Distance vector routing algorithm is one of the most popular dynamic routing algorithms. Distance vector routing algorithms operate by having each router maintain a table 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. This algorithm is sometimes called by other names, including the distributed Bellman-Ford routing algorithm and Ford-Fulkerson algorithm. It was the original ARPANET routing algorithm and was also used in the internet.

2.4.7 Link State Routing

Distance vector routing was used in the ARPANET until 1979, when it was replaced by link-state routing (Tenenbaum 2000). Two primary problems caused its demise. First, since the delay metric was queue length, it did not take line band width into account when choosing routes. Initially, all the lines were 56 kbps, so line band width was not an issue, but after some lines had been upgraded to 230 kbps and others to 1.544 Mbps (William 2002, Leon 2000), 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 long to converse, link-state algorithms flood routing information to all nodes in the internet work. Each router, however, sends only the portion of the routing table that describes the state of links. In link-state algorithms, each router builds a picture of the entire network in its routing tables. On the other hand, link-state algorithms require more CPU power and memory than distance vector algorithms.

2.4.8 Single versus Multiple Path Routing

In single path routing, a router maintains only one path to each destination. In multiple paths routing, a router maintains a primary path to a destination, along with alternative paths. If the primary path fails for some reason, routers may send the packets on the alternative path. The single path routing is used on the internet, because the alternative paths requires more routing table space. The telephone networks usually use multiple paths routing, because this reduces the call blocking probability, which is very important for customer satisfaction (Tenenbaum 2000).

2.4.9 Flat versus Hierarchical Routing

In the flat routing systems, the routers are peers of all others. In the hierarchical routing system. Packets from non backbone routers travel to the backbone routers, where they are sent through the backbone until they reach the general area of the destination. At this point they travel from the last backbone router through one or more non backbone routers to the single destination.
Routing systems often designate logical groups of nodes, called domains, autonomous systems, or areas. In hierarchical systems, some routers in a domain can communicate with routers in other domains, while others can communicate only with routers within their domain. In very large networks, additional hierarchical levels may exist, with routers at the highest hierarchical level forming the routing backbone.

### 2.4.10 Host-Intelligent versus Router-Intelligent

Some routing algorithms assume that the source and node will determine the entire route. This is usually referred to as source routing. In source-routing system, routers merely act as store-and-forward devices, mindlessly sending the packet to the next stop router. Intelligent algorithms assume that hosts know nothing about routers. In these algorithms, routers determine the path through the inter network based on their own calculations. In the first system, the hosts have the routing intelligence. In the latter system, routers have the routing intelligence.

### 2.4.11 Intra domain versus Inter domain

Some routing algorithms work only within domains, others work within and between domains. The nature of these two algorithm types is different. It stands to reason, therefore, that an optimal intra domain-routing algorithm would not necessarily be an optimal inter domain-routing algorithm.
2.5 LINK STATE ROUTING ALGORITHM

Distance vector routing was used in the ARPANET, when it was replaced by link state routing. Link state routing is simple and can be stated as five tasks. Each router must satisfy the following requirements:

1. Discover its neighbors and learn their network addresses. When a router is booted, its first task is to learn who its neighbors are. It accomplishes this goal by sending a special HELLO packet on each point to point line. The router at the other end is expected to send back a reply telling who it is.

2. Measuring line cost. The link state routing algorithm requires each router to know or at least have a reasonable estimate of the delay to each of its neighbors. The most direct way to determine this delay is to send a social ECHO packet over the line that the other side is required to send back immediately. By measuring the round trip time and dividing it by two, the test can be conducted several times.

3. Building Link state packets. Once the information needed for the exchange has been collected, the next step is to build a packet containing all the data. The packet starts with the identity of the sender followed by a sequence number and age and a list of neighbors. For each neighbor, the delay to that neighbor is given.

4. Distributing the Link state packets. The fundamental idea is to use flooding to distribute the link state packets. To keep the flood in check, each packet contains a sequence number that is incremented for each packet sent. Routers keep track
of all the pairs they see. When a new link state packet comes in, it is checked against the list of packets already seen. If it is new, it is forwarded on all lines except the one it arrived on. If it is a duplicate, it is discarded. If a packet with a sequence number lower than the highest one seen so far ever arrives, it is rejected.

5. *Compute the shortest path to every other router.* Dijkstra algorithm can be run locally to construct the shortest path to all possible destinations.

The various steps involved in the algorithm are as follows:

*Step1.* Initialize the network.

*Step2.* Initialize the cost of the link.

*Step3.* Determine the path.

*Step4.* Obtain cost of the path by adding cost of the link.

*Step5.* Sort the results.

*Step6.* Print all multi path along with cost in text mode

*Step7.* Produce the best 5 optimal paths for transmission.

*Step8.* Obtain network with optimal path in Graphics mode.

### 2.6 SUMMARY

Routing algorithms, and routing protocols which should be robust, optimum, simple, with low over head, quick converge and flexibility are highlighted in this chapter. The problem of routing assignment is one of the most intensively studied areas in the field of data networks in recent years.
Different routing metrics as path length, reliability, delay bandwidth, load and communication cost are discussed. Different types of routing algorithms such as static and dynamic routing algorithms, short path routing flooding, distance vector routing, link state routing, single versus multiple path routing, flat versus hierarchical routing, host intelligent versus router intelligent routing, intra domain versus inter domain routings and link state routing algorithm are also discussed.