Integrated IS-IS

- Integrated IS-IS for both CLNP and IP protocols is described in RFC 1195 while the original IS-IS protocol was described in.
- IS-IS comes with its own terminology that is different from OSPF. For example, routers are referred to as intermediate systems; thus, the name intermediate systems-to-intermediate systems means router-to-router.
- For consistency, will use the term routers instead of intermediate systems.
- LSAs are called link state protocol data units, or LSPs, in short. A broadcast network is referred to as a pseudo node; a designated intermediate system is elected from all the ISs to represent a broadcast network.
- An address to identify an intermediate system is called a network service access point (NSAP).
- IS-IS runs directly over layer-2 protocols, unlike OSPF that runs over IP.
- Similar to OSPF, IS-IS has also been extended to provide traffic engineering capabilities.

Its Key Features:

AREAS

IS-IS provides two-level network hierarchy using areas that are similar to OSPF. The routers in the backbone area are called L2 routers; the internal routers in low-level areas are called L1 routers. A network that has any low-level (L1) areas must also have at least one L1/L2 router that sits in the L1 area but is connected to the L2 (backbone) area by a link. Note that in IS-IS, a router is entirely within an area, unlike OSPF, where a router can sit on the border between two areas; connectivity between areas is only through a link.

ADDRESSING IN IS-IS

Addressing in IS-IS is based on OSI-NSAP addressing and is compatible with USA GOSIP version 2.0 NSAP address format.

As you can see, several fields are used for setup in IP networks. The important ones are: RDI, Area, and System ID. When the last byte, N-selector, is set to zero, there is no upper-layer user, and the address is meant purely for routing; such routing-layer-only NSAP addresses are called Network Entity Titles (NET). In effect, IS-IS for IP networks uses NET addressing. It is important to note that NET is a router identifier, not an interface identifier.

PSEUDONODES AND NONPSEUDONODES

IS-IS allows handling of different network types. For example, a broadcast network is treated as a pseudonode where one of the routers serves as the pseudonode, which is labeled the designated intermediate system (DIS), with links to each attached router. For links that are not for broadcast networks but are for point-to-point networks and stub networks, a nonpseudonode is created. Essentially, a nonpseudonode is similar to a router LSA in OSPF.

SHORTEST PATH CALCULATION

Shortest path calculation is based on Dijkstra's algorithm. Once a router receives a new LSP, it waits for 5 sec before running the shortest path calculation. There is a 10 sec hold-down timer between two consecutive shortest-path calculations within the same area. However, L1/L2 routers that reside in L1 areas must run separate shortest path calculations, one for the L1 area and the other for the L2 area. Link metric in IS-IS has been originally limited to 6 bits and, thus, the value ranges from 0 to 63 and the total path cost in an IS-IS domain can have a maximum value of 1023.

IS-IS defines four categories of protocol packets, or protocol data units (PDUs):

- hello packet
- link state PDUs (LSP)
- complete sequence number PDUs (CSNP)
- partial sequence number PDUs (PSNP).

There are two types of LSPs—one for level 1 and the other for level 2. Each LSP contains three key pieces of information: LSP ID (8 bytes), the sequence number (4 bytes), and the remaining lifetime (2 bytes). In addition to this, an LSP uses a Type-Length-Value (TLV) format to include information such as a list of connected IP prefixes along with subnet masks. This makes it possible to determine destination networks to which the domain is connected so that the routing table can properly list such destinations. CSNPs are like database description packets in OSPF and are used for link state database synchronization. A router creates CSNPs with all LSPs in its local link state database. A PSNP is created when upon receiving a CSNP from a neighbor, it realizes that some parts are missing; this means that this router has received certain other LSPs that are in its location link state database but the neighbor's CSNP did not include them. Thus, the receiving router generates a PSNP to request newer copy of the missing LSPs. In essence, PSNPs are similar to the link state request packet in OSPF.

Similarities and Differences Between IS-IS and OSPF

SIMILARITIES

- Both protocols provide network hierarchy through two-level areas.
- Both protocols use Hello packets to initially form adjacencies and then continue to maintain them.
- Both protocols have the ability to do address summarization between areas.
- Both protocols maintain a link state database, and shortest path computation performed using Dijkstra's algorithm.
- Both protocols have the provision to elect a designated router for representing a broadcast network.

DIFFERENCES

- With OSPF, an area border router can sit on the boundary between the backbone area and a low-level area with some interfaces in the area while other interfaces are in the other area. In IS-IS, routers are entirely within one or the other area—the area borders are on links, not on routers.
- While OSPF packets are encapsulated in IP datagrams, IS-IS packets are directly encapsulated in link layer frames.
- The OSPF dimension-less link metric value is in the range 1 to 65,535, while IS-IS allows the metric value to be in the range 0 to 63 (narrow metric), which has been extended to the range 0 to 16,777,215 (wide metric).
- IS-IS being run directly over layer 2 is relatively safer than OSPF from spoofs or attacks.
- IS-IS keepalives can be used for MTU detection since they are MTU-sized TLVs that are explicitly checksummed and need to be verified as such.
- IS-IS allows overload declaration through an overload bit by a router to other routers. This is used, for example, by other routers to not consider an overloaded router in path computation.

IP Traffic Engineering: Traffic, Stochasticity, Delay, and Utilization

IP Network Traffic

An IP network provides many services such as web and email; there are also interactive services such as telnet, ssh for terminal services. In current IP networks, the predominant traffic is due to applications that

use TCP for transport layer; it has been reported that on a backbone link approximately 90% of traffic is TCP based [350]. A message content created by applications is broken into smaller TCP pieces, called TCP segments, by including TCP header information, which are then transmitted over the IP network after including IP header information; the data entity at the IP level is IP datagrams, while packet is also a commonly used term. Thus, traffic in an IP network is IP datagrams generated by various applications, without wondering which among the applications it is for.

Traffic and Performance Measures

- Interestingly, there is an analogy between road transportation networks and IP networks. In road transportation networks, delay depends on the volume of traffic as well as the number of street lanes (and speed limit) imposed by the system.
- Similarly, delay in an IP network can depend on the amount of traffic as well as the capacity of the system; thus, the following functional relation can be generally written:
  Delay = F(Traffic volume data rate, Capacity).
- To be specific, the above relation is true only in a single-link system. When we consider a network where routing is also a factor, then a more general functional relation is as follows:
  Delay = F(Traffic volume data rate, Capacity, Routing).

Average Delay in a Single-Link System

First, we assume that packet arrival to a network link follows a Poisson process with the average arrival rate as λ packets per sec. The average service rate of packets by the link is assumed to be μ packets per sec. We consider here the case in which the average arrival rate is lower than the average service rate, i.e., $\lambda < \mu$ otherwise, we would have an overflow situation. If we assume that the service time is exponentially distributed (see Appendix B.10), in addition to packet arrival being Poissonian, then the average delay is: $\tau = 1/ \mu - \lambda$.

Now consider that the average packet size is κ Megabits, Then, there is a simple relation between the link speed c (in Mbps), the average packet size κ, and the packet service rate μ, which can be written as: $c = \kappa\mu$

$h = \kappa\lambda$

if we multiply the numerator and the denominator by κ, we can then transform eq. as follows:

$\tau = \kappa / \{\kappa(\mu - \lambda)\} = \kappa / \{c - h\}$

the link utilization parameter, $\rho = h / c = \kappa\lambda / \kappa\mu = \lambda / \mu$

In regard to traffic engineering, there are two important points to note :

- First, there is a direct relation between delay and utilization; because of this, requiring a network to maintain a certain delay can be recast as requiring the utilization to be kept below an appropriate level.
- Second, since there is no simple formula to consider delay for self-similar traffic, being conservative on the requirement on utilization can often be sufficient for the purpose of traffic engineering.

Applications' View

Since applications are the ones that generate IP traffic, it is helpful to understand the requirements in regard to applications. Since most commonly used applications such as web, email are TCPbased, from an application point of view, not only should the delay perception be minimized, but the throughput of data rate transfer is also an important consideration; this is necessary since TCP uses an adaptive sliding window mechanism to regulate how much data to be pumped based on perception of congestion.

TCP Throughput and Possible Bottlenecks

It has been noted that TCP throughput depends primarily on three factors: the maximum segment size (S), the round-trip time (RTT), and the average packet loss probability (q).

TCP throughput = $1.22\ S / \{RT\ T \times \sqrt{q}\}$

An important question is: from the traffic engineering perspective, where and how does an IP network fit in the three factors and the relation shown in above formula. So following are the factors:

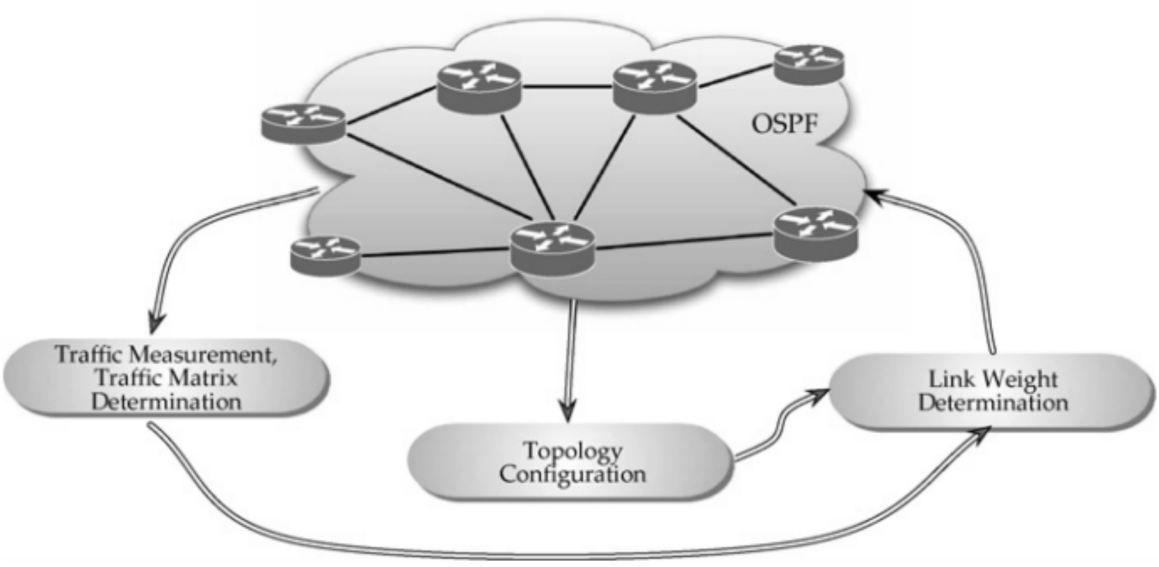
- First, we see that the segment size should be as large as possible. However, note that the maximum segment size is not entirely within the control of the network since it is negotiated by the end hosts; at the same time, this tells us that the network link should be set for the maximum transmission unit possible so that the network link itself does not become the bottleneck in reducing the TCP throughput of end applications
- The second factor that affects TCP throughput is the round-trip time. Round-trip time should be minimized, which means that one-way delay must be minimized. While many factors, including processing at the end hosts can impact delay, from the point of view of the network, it is important that the delay on a network link be minimized.
- The third factor is the average packet loss probability. The average packet loss can depend on many points along a TCP connection; the end hosts may drop a packet, the edge network may drop a packet, there may be bit error rate, and so on.

Bandwidth-Delay Product

The term bandwidth-delay product means exactly what it says—that is, to take the product of the bandwidth and the delay. In case of a network link, the bandwidth then refers to the link speed and the delay refers to what the network would like to account for. For example, if the link speed is given in Mbps and the round-trip time delay in seconds, then the product will result in a quantity in Megabits. What does this quantity signify? This is none other than the amount of data the network link needs to handle in-flights, often referred to as the window size. To put it formally, if c is the data rate of a link ("bandwidth") and RT T is the round-trip time delay, then the bandwidth-delay product defines the window W given by $W = c \times RTT$

Traffic Engineering: An Architectural Framework

Since a network consists of a number of routers, it is important to estimate source-destination traffic volume rather than on a link basis to obtain a traffic matrix that can be used for traffic engineering. First and foremost, traffic engineering occurs outside the actual network. This can be illustrated through an architectural framework of the traffic engineering system as shown in figure:



- From the actual network, traffic measurements are collected to estimate the traffic matrix; furthermore, topology and configuration are also obtained from the network.

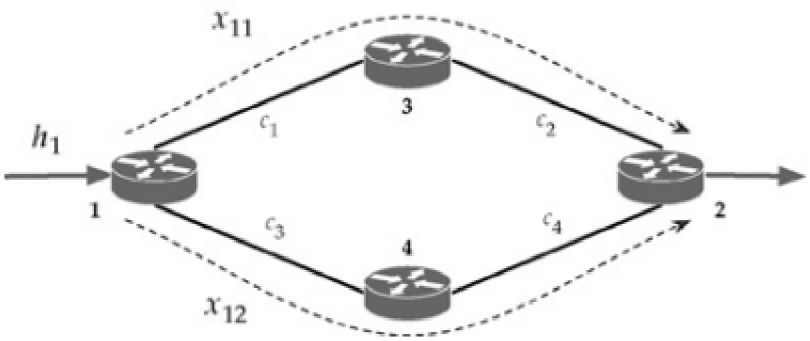
- Based on topology and configuration, along with the traffic matrix, a link weight determination process determines link weights keeping in mind that OSPF/IS-IS uses shortest path routing.
- The computed link weight for each link is then injected into the network; that is, each router receives metrics for its outgoing links through this external process.
- Once a router receives these link metrics, it then disseminates through flooding of link-state advertisements (LSAs) to other routers through the normal OSPF/IS-IS flooding process.
- This would mean that if no new link weights are obtained from the traffic engineering system when the age field of an LSA expires, the router will generate a new LSA by continuing to use the link metric value it received last from the the traffic engineering system.

Traffic Engineering: A Four-Node Illustration

We will first discuss the traffic engineering problem in a network by considering a four-node network. Assume that in this four-node network, there is traffic volume for only a single demand pair; this is then a single-commodity problem.

Network Flow Optimization

We will consider traffic volume to exist for the demand pair 1:2; this pair will be identified as demand identifier 1. We will denote the path from node 1 to 2 via 3 as path number 1, and the path from 1 to 2 via node 4 as path number 2, and denote the flow variables as x11 and x12, respectively
(see the Figure : "A four-node network example" shown below):



Thus, to carry the traffic volume h1 for demand identifier 1, i.e., from node 1 to node 2, the following must be satisfied:

x11 + x12 = h1

Certainly, we require that flow on each path is non-negative, i.e., x11 ≥ 0, x12 ≥ 0. Let the link be identified as 1 for 1-3, 2 for 3-2, 3 for 1-4, and 4 for 4-2. Then, we can list the flows to satisfy the capacity constraints as follows:

x11 ≤ c1, x11 ≤ c2, x12 ≤ c3, x12 ≤ c4

Note that we can combine constraints x11 ≤ c1, and x11 ≤ c2 to a single constraint by considering whichever capacity is more stringent, i.e., as x11 ≤ min{c1, c2}; this is similar for the other two constraints. However, we will list them all since this is the general representation, unless we consider specific values of capacity. Suppose our goal is to minimize maximum link utilization. Then, we can write the optimization problem as:

$$\begin{aligned} &\textit{minimize}_{\{x\}} && F = \max\left\{\frac{x_{11}}{c_1}, \frac{x_{11}}{c_2}, \frac{x_{12}}{c_3}, \frac{x_{12}}{c_4}, \right\} \\ &\textit{subject to} && x_{11} + x_{12} = h_1 \\ & && x_{11} \le c_1, \quad x_{11} \le c_2, \quad x_{12} \le c_3, \quad x_{12} \le c_4 \\ & && x_{11} \ge 0, \quad x_{12} \ge 0. \end{aligned}$$