Project Title	Menehune-Traffic Engineering (Menehune-TE)
Registration Number	
Engineering College	
Project Members	
Project Guide	
Academic Year	

Synopsis

Menehune-TE

Idea of project:

Our project **Menehune-TE** aims at giving **Traffic Engineering** extension to the native OSPF (Open Shortest Path First) routing protocol.

Traffic engineering can be achieved by having **additional link state metrics** to describe current status of links. These additional parameters will build traffic-engineering database (TED), which will be used to develop input to constraint based routing algorithm (CSPF). CSPF is enhancement to Dijkshtra's shortest path algorithm used in original OSPF, in which **optimal path** is calculated instead of shortest one.

The main issue under consideration is link state metrics. These values describe current status of network links. These includes –

- Available bandwidth
- Link propagation delay
- Delay jitter
- Network resource usage
- Loss probability

Thus, link state database with additional parameters is updated periodically through synchronized link state advertisements between routing elements. This database is maintained efficiently on persistent storage. Enhanced routing algorithm is developed which works on this traffic engineering constraints.

The project has been implemented on the **Linux 2.4** Operating System.

Working:

The main target of most IP routing protocols is to achieve connectivity between network routers and eliminate loops. One of the protocols in wide use in the Internet today - Open Shortest Path First (OSPF), can be upgraded to include more sophisticated dynamic routing and TE features through reasonable changes and additions. User can select any number of parameters for routing dynamically. There are two main issues handled –

1. **Traffic Engineering Database (TED)**

   Native OSPF uses very few parameters for describing the current link state. We are gathering additional traffic-engineering attributes through **link state advertisement** (LSA). This is a periodic activity **with reduced triggered time** than native OSPF that in turn improves reliability of the system. This activity is also synchronized

with all routers. These parameters are stored in persistent storage efficiently. Also database overflow is handled.

2. **Routing according to TED**

Now, according to this database, link state information for routing is retrieved through query interface. To calculate optimal path using constraint-based policy, additional parameters are passed to Dijkstra's Shortest Path algorithm. Original Dijkstra's algorithm used in native OSPF is slightly modified to fit to traffic engineering requirements. Now, routing tables are modified accordingly.

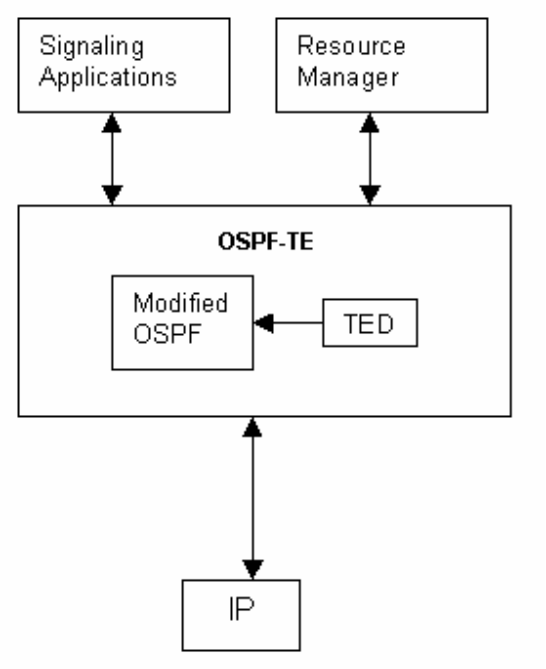

Figure 1

The problem:

Native OSPF protocol derives shortest path instead of optimal one, which leads to traffic imbalance in the network. This results in packet loss and unacceptable delays. This is often seen in many big networks like ISPs, private intranets where original routing algorithm is installed. These algorithms have little or no support for traffic engineering.

According to our knowledge, some theoretical foundation and simulations have been developed. But our efforts are towards modification of native OSPF. To do this we have setup following topology.

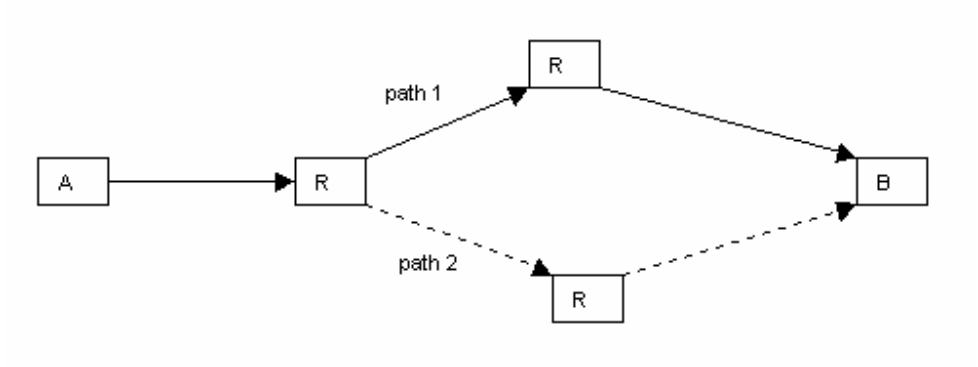

Figure 2

Above diagram (figure 2) shows the topology where multiple paths from source to destination are possible. Native OSPF will select path 1 from A to B, which is shortest. Now further traffic will converge through same path irrespective of current link state, which may change over the time. This leads to unacceptable delays or packet loss even in the presence of feasible paths over less utilized links.

Current link state can be completely described using additional parameters. If these parameters indicate that path 2 is feasible then this path will be chosen for further traffic instead of path 1. This can be achieved by using our modified routing algorithm.

Commercial Applications:

Many service providers like ISPs (Internet Service Provider) have to deliver good quality to customers without substantially upgrading network infrastructure. This kind of product will help them to satisfy their customers. ISPs can optimize resource utilization by evenly distributing or balancing traffic load on various links.

As we are providing query interface to this functionality, resource manager can utilize this product for link allocation. High data transfer applications like network games can be developed in connection with this product.

Currently in the market, some companies have started to develop such products.

Social Aspects:

As native OPSF protocol is non-proprietary, we are also making our product as open source. Hence, it will give cost-effective solution.