International Journal of Electronics and
Communication Engineering (IJECE)
ISSN(P): 2278-9901; ISSN(E): 2278-991X
Vol. 3, Issue 3, May 2014, 87-94

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

MPLS-TRAFFIC ENGINEERING LOAD BALANCE ALGORITHM USING UNCOMMON PATH

RAHUL P. SAKHARKAR & PANKAJ P. TASGAONKAR

Department of Electronics and Tele-Communication Engineering, College of Engineering, Pune, Maharashtra, India

ABSTRACT

This paper presents a new path computing algorithm called Uncommon Path Algorithm (UPA) to implement Load Balance in Multiple Protocol Label Switching – Traffic Engineering (MPLS-TE). This algorithm finds an alternate path, called Uncommon Path (UP), for a selected flow, having no common path to other flow paths. Then we use explicit routing technology of MPLS to route selected flow via Uncommon Path. We used Network Simulator-2 (NS-2) for simulation. The simulation results show that UPA effectively balances traffic load between different links and improves network performance giving lower packet drop rate, lower end to end packet delay, lower variation in packet arrival and higher throughput.

KEYWORDS: Multiple Protocol Label Switching (MPLS), Traffic Engineering (TE), Uncommon Path Algorithm (UPA), Uncommon Path (UP)

INTRODUCTION

Use of internet is increasing at a tremendous rate and hence there is an explosive growth in internet traffic in recent years. The existing network infrastructure and capacity is hardly able to satisfy all the needs and it leads to increase in the frequency of network congestion. Congestion occurs when network resources are inefficient or inadequate to accommodate data traffic and when traffic streams are incorrectly routed via available resources, causing some part of network underutilized and some over utilized. Though there are many factors affecting congestion, the primary cause of network congestion is the unbalanced distribution of network traffic. Therefore how to balance the network traffic and improve the internet quality of service is becoming the most crucial issue [1].

TRAFFIC ENGINEERING

To optimize the utilization of resources in a network, an important network optimization technique - Traffic Engineering (TE) - came into existence [2]. TE aims to optimize resource utilization and network performance by controlling how traffic flows through desired network. Load balancing is an important aspect of TE. In load balancing traffic streams from congested network area are routed via alternate paths available in network, if such paths exists and thus traffic in a particular area is relocated to another area reducing congestion.

Traditional IP routing algorithms like Open Shortest Path First (OSPF) and Intermediate System-Intermediate System (ISIS) compute the shortest way to the destination only based on the destination address and don't take capacity constraints and traffic characteristics into account when routing decisions are made. So some segments of a network can be very congested while other segments along alternative routes are under-utilized. TE came into existence to address this problem [2].

MPLS TRAFFIC ENGINEERING

In practice, there are many strategies that can be used for TE like IP-over-ATM [3], constraint based routing [4] and others. But Multiple Protocol Label Switching – traffic engineering (MPLS-TE) overcomes the limitations of these approaches by providing strong technical support. Hence MPLS is best suited for TE. Explicit routing technology of MPLS is extensively used in TE [5].

Load balancing is an important aspect of MPLS-TE. In load balancing, traffic flows are mapped on multiple paths to reduce load on a particular path. These paths need not be the shorted paths. There are many load balancing algorithms proposed so for to improve performance of MPLS-TE [6]. We propose Uncommon Path Algorithm (UPA) for load balance in MPLS-TE and compare it's performance with Shortest Path Algorithm (SPF).

UNCOMMON PATH ALGORITHM

Network traffic is unpredictable and hence adaptive adjustment capability in a network is very essential. UPA adjusts network traffic according to incoming requests for data flow.

Concept of UPA

The concept of UPA is as follows: Suppose there are many flows through a Label Switched Path (LSP). Select any two flows from congested LSP. Find shortest paths for these two flows. Then find common path between these two shortest paths. This common path can include a single link or a number of consecutive links. If there is a single node common between two paths, it is not considered as a common path. If there is common path available, try to find a new path which will not include this common path. This path is called as Uncommon Path (UP). If UP exists, switch flow on UP using explicit routing technology of MPLS; otherwise route by using SPF algorithm. If there is no common path between two flows, there is no need to apply any algorithm. Check if LSP is still congested. If yes, repeat above procedure, otherwise stop. Figure 1 shows flow chart of UPA.

Explanation with Example

Consider IP network topology as shown in Figure 2, in which all IP nodes are having circular shape, all links are having 2Mbps bandwidth and 10ms delay. Suppose there are two requests coming – First from IP(1) to IP(3) and second from IP(2) to IP(4) each with bandwidth of 1.5 Mbps.

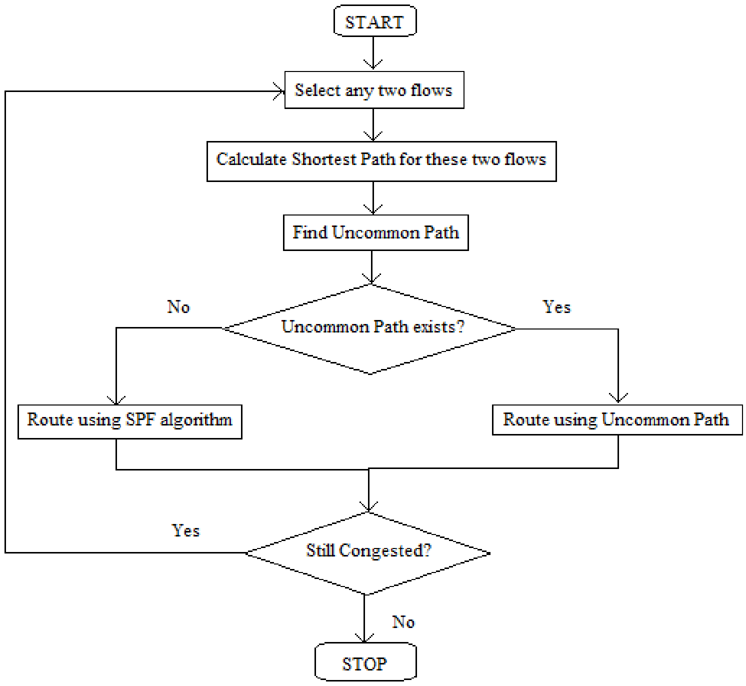

Figure 1: Flow Chart of UPA

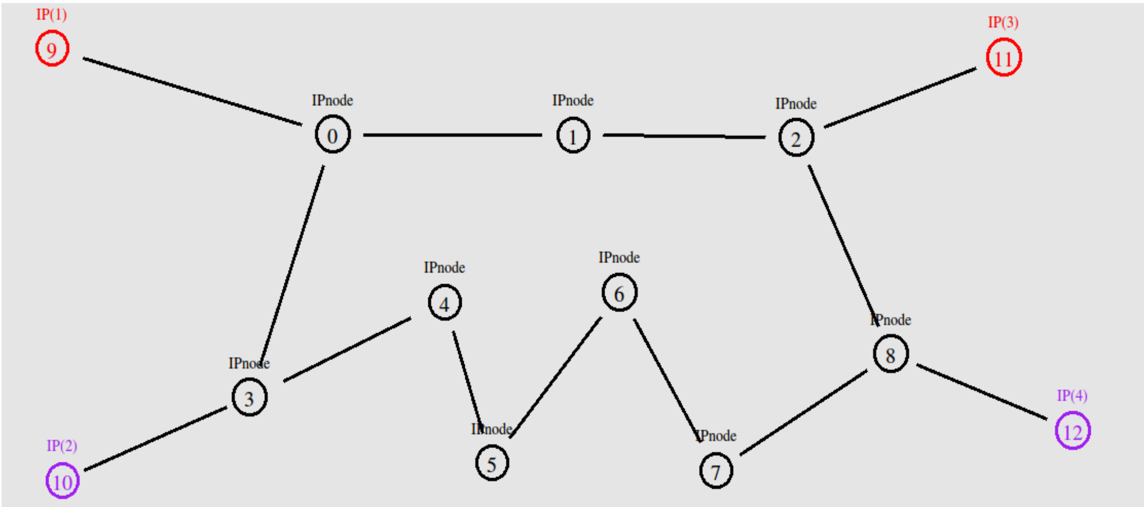

Figure 2: IP Network Topology

When IP routing is used, flow 1 – corresponding to request 1 – will follow route 9-0-1-2-11 and flow 2 – corresponding to request 2 – will follow route 10-3-0-1-2-8-12, according to SPF routing algorithm. This will lead to congestion on common path 0-1-2, because combined bandwidth requirement of two flows is 3Mbps and links 0-1 and 1-2 are having 2Mbps bandwidth each. Therefore when buffer at node 0 is filled completely, packets will be dropped from tail at node 0 as shown in Figure 4.

Now consider MPLS network topology, as shown in Figure 3, in which backbone network is MPLS network to which IP nodes are connected. Here MPLS nodes are shown by square shape to distinguish them from IP nodes. Each MPLS node is a Label Switched Router (LSR). Other considerations are same as IP network.

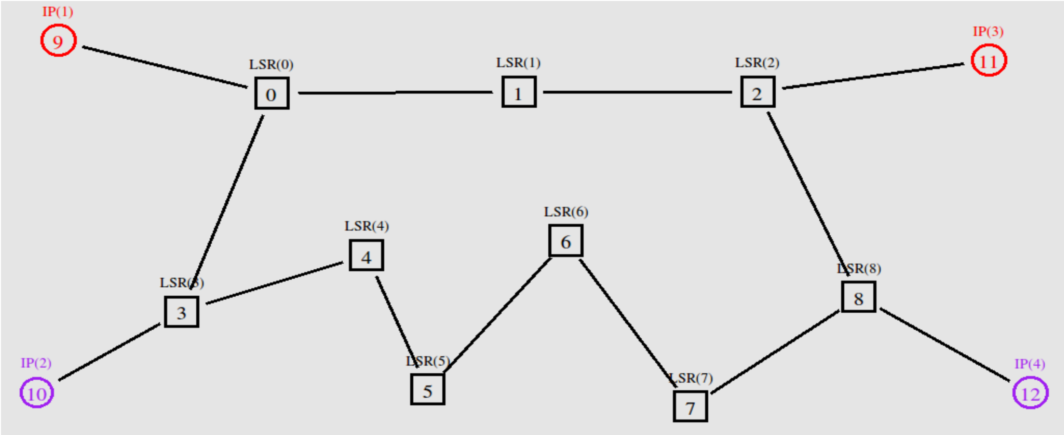

Figure 3: MPLS Network Topology

We now use UPA. When we apply UPA, the algorithm will first calculate shortest path for each request. UPA is applied only in MPLS domain. Shortest path for request 1 is 9-0-1-2-11 and for request 2 is 10-3-0-1-2-8-12. Here common path is 0-1-2. Also shortest path for request 1 is shorter than shortest path for request 2 if we consider node as a distance metric. Here it is better to calculate UP for request 2 because request 1 path is shortest. Hence UPA will find UP for request 2, which is 3-4-5-6-7-8. It is clear that though this path is somewhat longer than shortest path for request 2, it is avoiding congestion, as shown in Figure 5. Then using explicit routing technology of MPLS, flow 2 is routed through path 10-3-4-5-6-7-8-12. Thus congestion is avoided and network performance is improved.

SIMULATION RESULTS

For simulation an open source network simulator version 2 (NS-2) is used. Network topologies for simulation are shown in Figure 2 and Figure 3. A trace file is generated for each simulation and graphs are drawn using trace files. To show effectiveness of UPA over SPF, we have compared characteristics of flow 1 using SPF with that of characteristics of flow 1 using UPA. The effectiveness can also be checked by drawing graphs for flow 2 but results for flow 1 are sufficient to prove feasibility and effectiveness of UPA. Similar conclusions can be drawn using results for flow 2.

Figure 4 shows packet dropping when SPF routing is used. At node 0, combined bandwidth of incoming flows is 3 Mbps (1.5 Mbps for flow 1 + 1.5 Mbps for flow 2). But bandwidth of link 0-1-2 is 2 Mbps only. Hence when buffer at node 0 is full, packets are dropped.

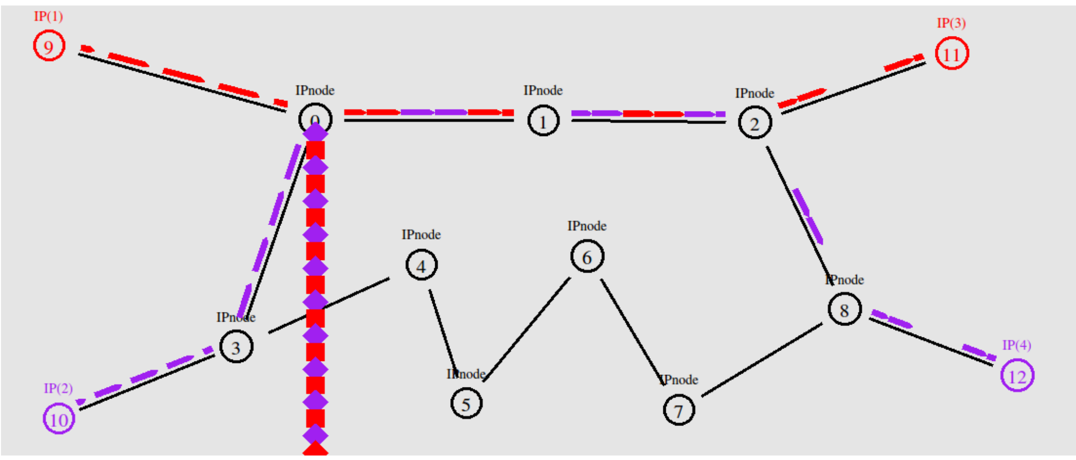

Figure 4: Packet Dropping in SPF Routing

Figure 5 shows that when UPA is used, congestion is avoided and there is no packet drop. Figure 6 shows graph for packet drop. From graph also, it is clear that no packet is dropped ever since we apply UPA.

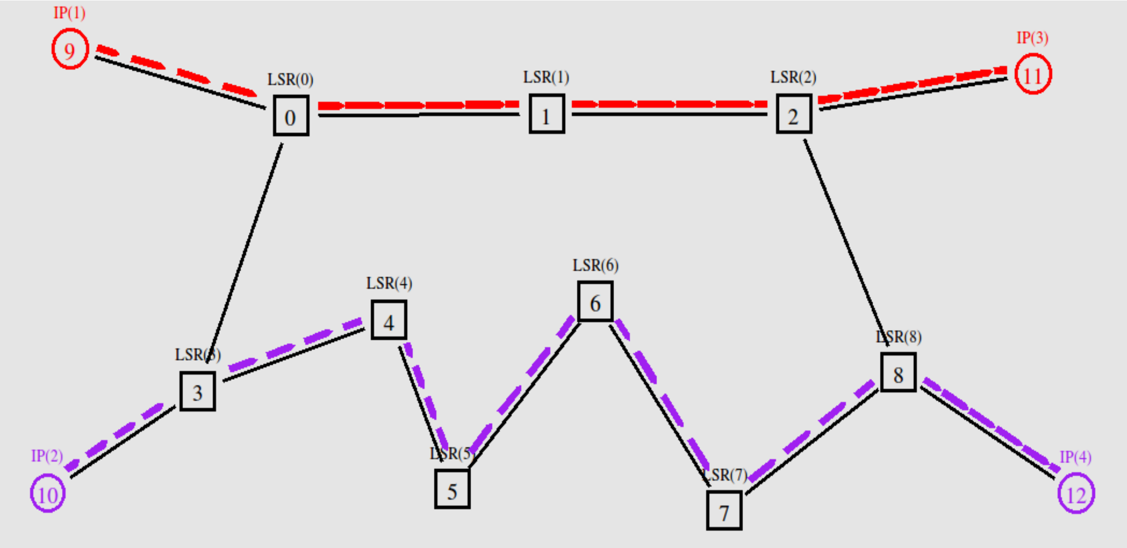

Figure 5: No Packet Dropping after Application of UPA

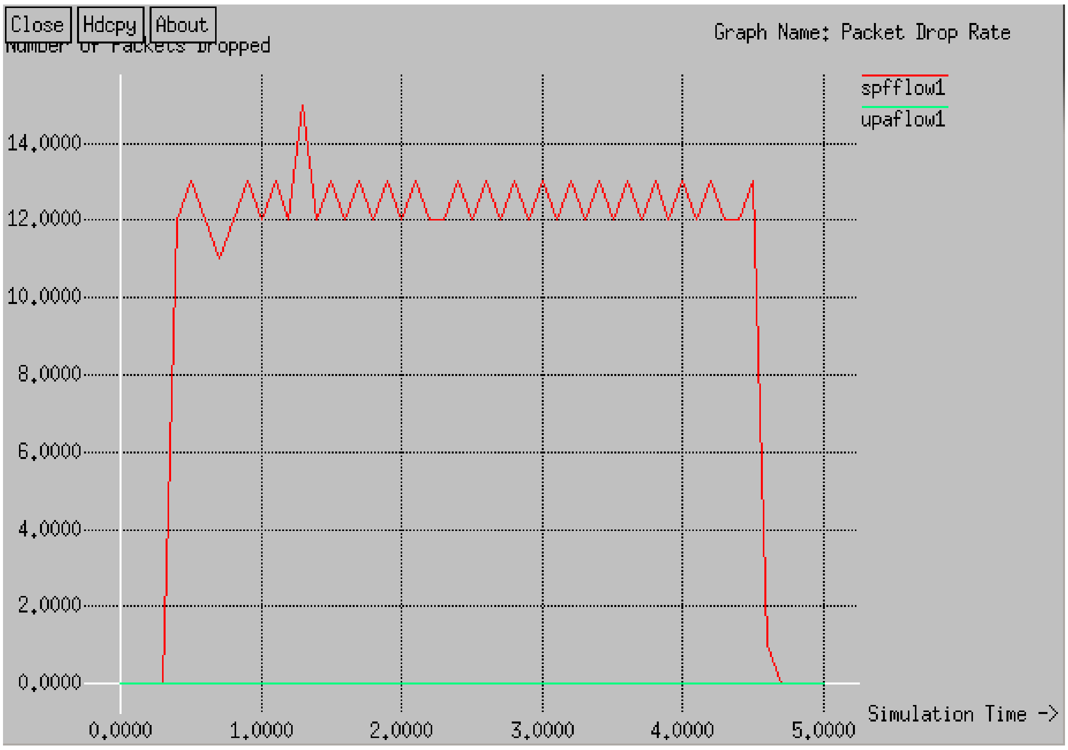

Figure 6: Graph for Packet Drop Rate

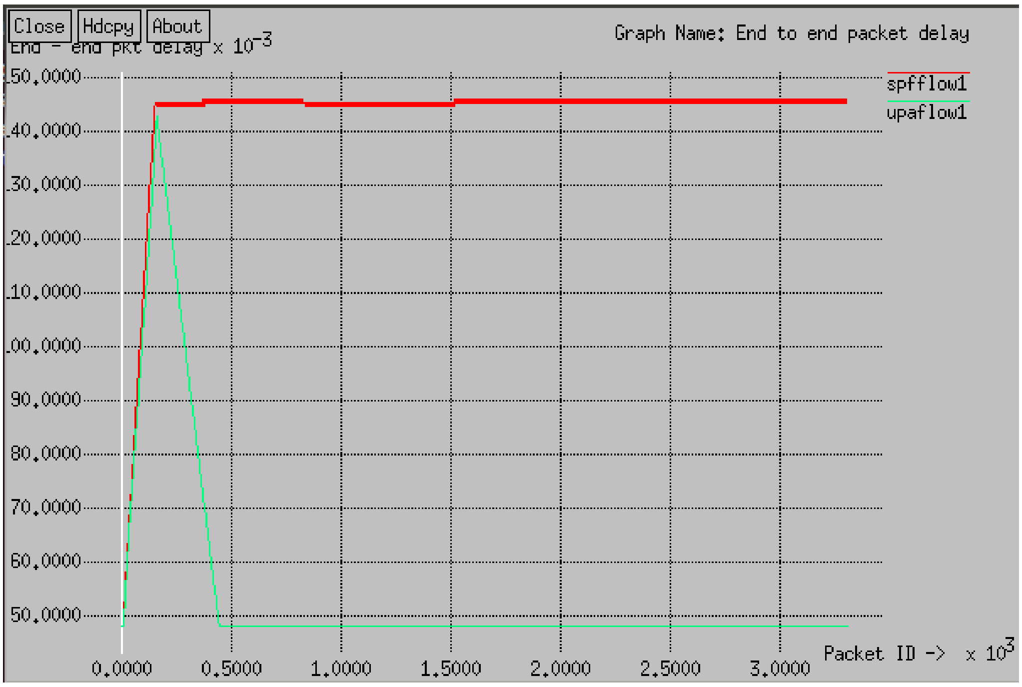

Figure 7: Graph for End to End Packet Delay

In Figure 7 graph of end to end packet delay for flow 1 is plotted. It is seen from graph that end to end packet delay increases first and when we apply UPA, it stabilizes at around 48 ms. Also as end to end packet delay is constant, variation in packet arrival time is zero which signifies a steady flow. It is also clear that as end to end packet delay is reduced and because of no congestion time required for data transmission is reduced.

Figure 8 shows graph for throughput comparison. Throughput is calculated at IP(3) i.e. at node 11. Before applying UPA throughput for both algorithms was same but there is considerable increase in throughput when UPA is applied. Throughput of SPF is less than that of UPA because some packets are dropped during congestion.

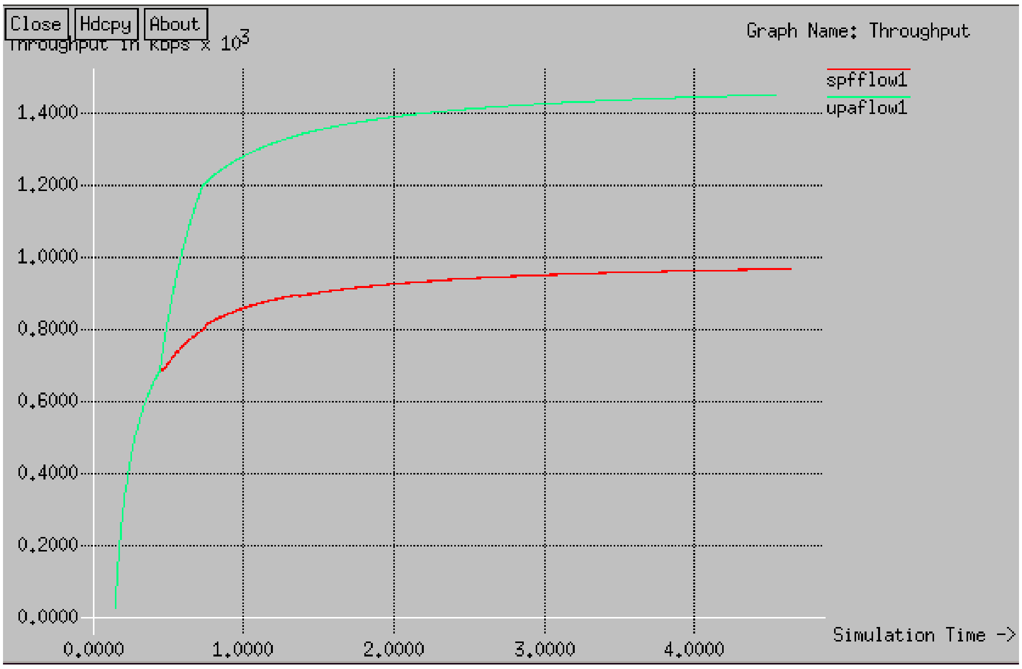

Figure 8: Graph for Throughput

Thus from simulation results and graphs, we can state that UPA considerably improves network quality of service parameters. UPA reduces end to end packet delay, packet drop rate and improves throughput. Also UPA gives steady streams than SPF. So we can conclude that UPA is effective than SPF algorithm.

CONCLUSIONS

In this paper we first discussed the need for traffic engineering and general issues of traditional routing algorithms. Then we discussed how MPLS is effectively used for TE. Then we presented concept of UPA giving explanation with example. Finally from simulation and results it is clear that UPA can effectively balance the workload between different links and improves the network performance.

REFERENCES

1. D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell and J. McManus, "Requirements for Traffic Engineering Over MPLS," RFC2702, IETF, September 1999.
2. Fenglin Li and Jianxun Chen, "MPLS Traffic Engineering Load Balance Algorithm Using Deviation Path," International Conference on Computer Science and Service System, 2012 IEEE, pp. 601-604.
3. C. Semeria, "Traffic Engineering For The New Public Network," Juniper Networks Whitepaper, September 2000.
4. B. sFortz, J. Rexford, and M. Thorup, "Traffic Engineering with Traditional IP Routing Protocols," IEEE Communication Magazine, October 2002.
5. Xipeng Xiao, Alan Hannan, Brook Bailey and Lionel Ni, "Traffic Engineering with MPLS in the Internet," IEEE Network, 2000.
6. Keping Long, Zhongshan Zhang and Shiduan Cheng, "Load balancing algorithms in MPLS traffic engineering," In 2001 IEEE Workshop on High Performance Switching and Routing, 2001, pp.175-179.