QoS routing as a tool of MPLS Traffic Engineering

Karol Kowalik and Martin Collier

Research Institute for Networks and Communications Engineering (RINCE)
Dublin City University,
Dublin 9, Ireland,
e-mail: {kowalikk, collierm}@eeng.dcu.ie

Abstract

Multimedia traffic and real-time e-commerce applications can experience quality degradation in traditional networks such as the Internet. These problems can be overcome in networks which feature dynamically set up paths with bandwidth and delay guarantees. Multi Protocol Label Switching (MPLS) shows promise as a networking protocol which can provide such capabilities. However, existing routing protocols need to be enhanced or replaced by QoS-aware algorithms if this potential is to be realised. This paper presents a survey of emerging QoS routing methods for unicast flows suitable for usage within MPLS networks. We discuss basic algorithms such as Widest Shortest Path (WSP) and more complicated ones such as the Minimum Interference Routing (MIRA) or Profile-Based Routing. Then we propose a new QoS routing algorithm for Diff-Serv flows in MPLS networks. The method has two phases: off-line routing - based on daily statistics, which achieves efficient global network utilisation; on-line routing - based on off-line routing and actual network state, which results in fast routing. The on-line algorithm can use information obtained from extended versions of link state routing protocols such as OSPF or IS-IS and the off-line version will use traffic descriptors obtained from service level agreements (SLAs). Our algorithm uses a fall-through mechanism whereby it attempts to establish a path with progressively lower QoS guarantees before abandoning a setup attempt. This method allows the network to transparently transport multimedia traffic whilst maintaining network efficiency.

1 Introduction

It is hard to retrofit the Internet with QoS capabilities. In the current Internet, data packets belonging to a flow may follow different paths to the destination. This leads to situation where throughput is not guaranteed and delay is not bounded. Simple best-effort service is not suitable for multimedia or real-time e-commerce applications. To provide QoS guarantees new service models [1, 2, 3] and mechanisms [4] need to be implemented.

The motivation of our work is to quickly set-up bandwidth-guaranteed paths in ISP (Internet Service Providers) networks. This requires Traffic Engineering. Traffic Engineering can be defined as the process of arranging traffic flows. To make the traffic engineering process automatic we can use QoS routing or more generally constraint-based routing (see Figure 1).

The basic function of QoS routing is to select a path that is likely to be able to meet the required QoS requirements. Constraint-based routing [5] extends QoS routing by considering other constraints of the network such as policy.

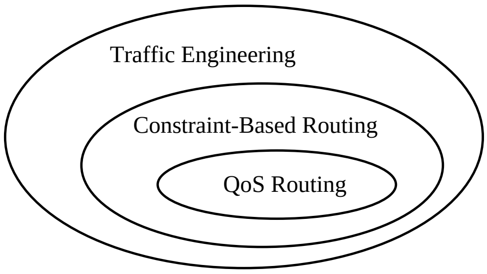

Fig. 1: Traffic Engineering Hierarchy

The emerging Multi-Protocol Label Switching (MPLS) [6] supports Traffic Engineering with mechanisms to establish explicit switched paths. Explicit routing allows the traffic flows to receive the requested QoS level, what results in better use of the network infrastructure.

In this paper we consider mechanisms of setting up Label Switched Paths (LSPs) with bandwidth guarantees. Following Kodialam and Lakshman [7] we assume that other constrains such as delay, delay jitter can be converted into effective bandwidth requirements for the LSP.

Then we present a new QoS routing algorithm for Diff-Serv flows in MPLS networks. We assume that incoming traffic has a quasi-static [7] nature (i.e. the ingress-egress pairs are known). This information is used in the off-line phase of our algorithm based on daily statistics (we here assume that the traffic profile will be similar to the profile from the previous day). The on-line phase uses predetermined paths from off-line phase as well as the link residual capacities which could be provided by extended versions of link state routing protocols such as OSPF [8] or IS-IS [9].

2 Related Work

A variety of QoS routing techniques have been presented recently [10, 7, 13]. Those algorithms are designed to use resources efficiently while providing the desired QoS level. The simplest techniques attempt to limit resource consumption while balancing the network load. The easiest way to limit resource consumption is to choose the shortest path (the path with the least number of links) and the network load can be balanced by selecting the path with higher residual bandwidth. It is straightforward to see that these two criteria conflict. The most commonly cited examples of QoS routing [10] algorithms are:

Widest-shortest path [11] - selects the shortest feasible path but if there are several such paths, the one with the largest reservable bandwidth is chosen.

Shortest-widest path [12] - selects the path with largest reservable bandwidth but if there are several such paths, the one with the minimum hop count is chosen.

Shortest-distance path [10] - selects the path with the shortest *distance*. The *distance* is defined as the sum of the inverse bandwidths of all links along the path.

However these algorithms take into account only the actual available bandwidth and don't care if accepting one LSP can result in blocking many future flows. This situation is well illustrated in Figure 2 (presented first in [13]). It is easy to see that setting up an LSP between source *S0* and destination *D0* can block LSPs between *S0-D0*, and others. Such a problem

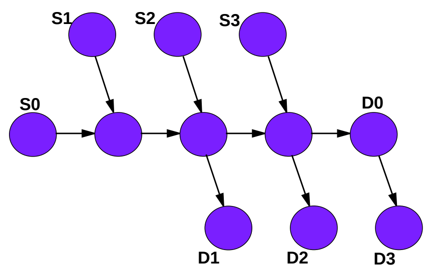

Fig. 2: The *parking lot* topology

can be avoided if we consider also the location of ingress-egress nodes. Kodialam and Laksman [7] first realised that such a problem exists and they proposed MIRA (Minimum Interference Routing Algorithm) as a solution. The main idea of MIRA is to choose a path that minimize interfere with future LSP requests. However the method used to compute *interference* between flows is quite complex and could be difficult to implement in real time.

A simpler approach called *Profile Based Routing* was described in [13]. It uses a preprocessing phase to assign every link in the network with some amount of bandwidth for every traffic class. Then in the on-line phase it uses the pre-allocated bandwidth to check if the request can be accepted as a part of that traffic class. The main limitation of *Profile Based Routing* algorithm is that it reject flows if they cannot fit into the pre-allocated amount of bandwidth. An algorithm without that limitation will now be described.

3 Algorithm

We assume that our algorithm operates in an MPLS network supporting Differentiated Services. Each traffic flow belongs to one of *k* Diff-Serv classes with priorities ranging from *0* (highest) to *k-1* (lowest). Traffic flows will reserve bandwidth on a basis other than the mean bandwidth required. For example, the bandwidth may be requested on the basis of peak bandwidth or equivalent bandwidth. In such circumstances, the link loading will be less than 100% when the aggregate reserved bandwidth matches the link capacity. We can avoid this by allowing links to be "oversubscribed". This means that the total assigned bandwidth of flows across the link may exceed the link's physical bandwidth. This results in efficient link utilisation at the expense of QoS. By allowing different levels of oversubscription among the Diff-Serv classes, we can ensure that the high priority classes receive adequate bandwidth, whilst overall link utilisation remains high. A possible oversubscription schedule is shown in Table 1. A flow which cannot be routed in the requested class without exceeding the allowed level of oversubscription may be accepted in a class of lower priority.

Algorithm has two phases:

1. **off-line routing**- based on daily statistics;
2. **on-line routing** - based on off-line routing results and actual network state.

Off-line routing is a preprocessing phase for the on-line routing. Network is represented by

class:	oversubscription:
0	50%
1	60%
2	75%
3	90%
4	110%
5	130%
6	160%
7	200%

Tab. 1: Link oversubscription

a weighted graph model *G=(V,E)* (where *V* is the set of nodes *n* and *E* is the set of links *e*). In that graph every link *e* has a cost equal to *1*, and each LSP has a weight $w_j = k - j$, where *j* is the Diff-Serv class priority (we could assign to $w_j = j$ but then class *k* would have the highest priority). The main goal of the off-line routing is to maximise the sum:

$$\sum_e \sum_{j=0}^{k-1} w_j \sum_i b_{i,j}(e)$$

where:

$b_{i,j}(e)$	is the amount of bandwidth routed thought link *e* within LSP *i* which belongs to class *j*;
and w_j	is the weight of traffic class *j*.

The LSPs accepted in the off-line phase are then used in on-line routing as the pre-established paths. The set of values $\{b_{i,j}(e)\}$ is used in the on-line routing phase as the graph weights.

On-line routing computes routes for real time requests. For each traffic class it uses a different graph $G_J = (V,E)$ (where J corresponds to the traffic class) to represent the network. When an ingress router receives a request first it checks whether the new LSP can be accommodated by one of the pre-established paths calculated off-line. If not it uses the graph $G_C = (V,E)$ (where C is the requested traffic class) with modified links costs. Every link with residual bandwidth smaller than that requested is assigned a weight of ∞. For every other link *e* the cost function is given by:

$$cost(e) = \sum_{j=0}^{J-1} w_j \sum_i b_{i,j}(e).$$

We search for the shortest path on graph G_C using Dijkstra's algorithm. The decision whether to assign the LSP to this path is made using the algorithm of Figure 3. This algorithm uses a parameter γ to inhibit the assignment of LSPs to pre-established paths.

The algorithm described here has the following advantages:

- it doesn't necessarily reject flows if they don't fit into pre-allocated paths;
- by modifying the parameter γ we can change the level of protection given to pre-established paths, allowing us to achieve better response to network dynamics;
- the on-line phase is simple, with a complexity similar to that of the *widest-shortest path* algorithm.

4 Conclusion and Future Work

QoS routing is an important tool for resource management in MPLS networks. Its goal is

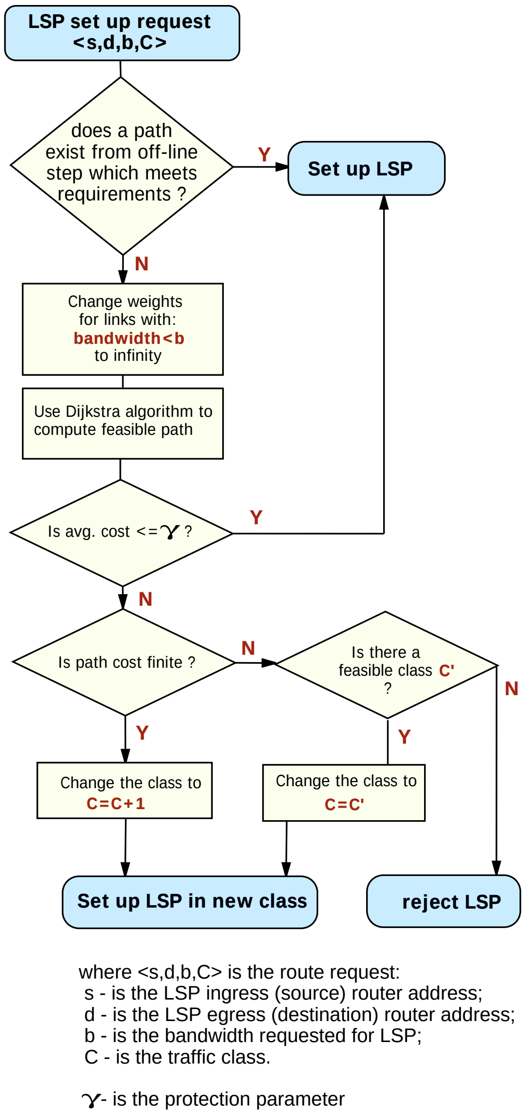

Fig. 3: Algorithm flowchart

twofold: to find a route that satisfies QoS constraints whilst efficiently using the network resources. In this paper we presented a new algorithm for routing bandwidth guaranteed Label Switched Paths (LSPs). This algorithm uses network state information as well as statistical information about route requests and statistical position of ingress-egress pairs. Using only one parameter we can control the stability of our algorithm as well as the network dynamics. The algorithm seems to be a good routing solution for networks with a quasi-static traffic profile but simulation results need to be obtained to confirm this. This will be the subject of our future work.

References

[1] B. Braden, R. Clark, and S. Shenker. Integrated Services in the Internet Architecture: An Overview. *RFC 1633*, July 1994.

[2] D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. *RFC 2475*, December 1998.

[3] K. Nichols, V. Jacobson, and L. Zhang. A Two-bit Differentiated Services Architecture for the Internet. *RFC 2638*, July 1999.

[4] B. Braden, R. Zhang, L. Berson, S. Herzog, and S. Jamin. Resource Reservation Protocol (RSVP) - Version 1 Functional Specification. *RFC 2205*, September 1997.

[5] X. Xiao and L. Ni. Internet QoS: A Big Picture. *IEEE Network Magazine*, March 1999.

[6] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. *RFC 3031*, January 2001.

[7] Murali S. Kodialam and T. V. Lakshman. Minimum Interference Routing with Applications to MPLS Traffic Engineering. In *INFOCOM (2)*, pages 884–893, 2000.

[8] Nabil Bitar, Roshan Rao, and Karthik Muthukrishnan. Traffic Engineering Extensions to OSPF. *Work in Progress, Internet Draft*, July 2001.

[9] Tony Li and Henk Smit. IS-IS extensions for Traffic Engineering. *Work in Progress, Internet Draft*, August 2001.

[10] Q. Ma and P. Steenkiste. On path selection for traffic with bandwidth guarantees. *In Proceedings of IEEE International Conference on Network Protocols*, October 1997.

[11] R. Guerin, Ariel Orda, and D. Williams. Qos routing mechanisms and ospf extensions. *In Proceedings of 2nd Global Internet Miniconference (joint with Globecom'97)*, November 1997.

[12] Z. Wang and J. Crowcroft. Quality-of-service routing for supporting multimedia applications. *IEEE JSAC*, pages 1288–1234, September 1996.

[13] Marcel Waldvogel Subhash Suri and Priyank Ramesh Warkhede. Profile-Based Routing: A New Framework for MPLS Traffic Engineering. *Washington University Computer Science Technical Report WUCS-00-21*, July 2000.