QoS Routing for Best-effort Flows

Pawan Goyal and **Gisli Hjalmtysson**†
Ensim Corporation †AT&T Labs - Research
Mountain View, CA 94043 Florham Park, NJ 07932

In packet networks such as Internet, a simple static shortest path routing algorithm is employed. A packet is sent on the shortest path to its destination. If there are multiple shortest paths, the shortest path is chosen arbitrarily. To determine the shortest path each link is assigned a weight and the shortest path is the one with the smallest aggregate weight. The key characteristic of this routing algorithm is that the weights are assigned statically. Hence, route changes occur only when a change in the topology of the network occurs, i.e., a link is deleted or added. This topology driven static shortest path routing may suffice in networks that provide a single best effort service in which there is no guarantee about whether and when a packet will be delivered. However, it may not be adequate in networks that provide Quality of Service (QoS) guarantees to applications such as multimedia conferencing.

To observe the inadequacies of static shortest path routing, consider the network shown in Figure 1. Let the network provide bandwidth guarantees to a flow. A flow is a sequence of packets from a source to destination and for our purpose is synonymous with a connection. In the network shown in Figure 1, all flows originating from 1 and destined to 5 will be routed over path 1-2-3-4-5 (assuming each link has identical weight). If large number of flows originating from 1 and terminating at 5 request guaranteed bandwidth, then the links 2-3 and 3-4 may get saturated and flows may be denied their request. This may occur even though the alternate path 1-2-6-7-4-5 between 1 and 5 may have capacity to support the flows that have been denied their request. In order to exploit the available capacity on alternate route, the concept of QoS routing has been proposed. There are large number of QoS routing algorithms (see [1, 2, 3, 4, 5, 6] for some examples and study of such algorithms). The characteristic common to all these algorithms is that they employ dynamically obtained information about load at each of the links to determine the path over which a flow should be routed. The path of a flow may be determined by source routing in which a flow setup message carries the entire path for the flow or by hop-by-hop routing. Such QoS routing algorithms have a few limitations.

First, in order to route packets of flows with the same destination over multiple routes, they require connection oriented network layer (i.e., the ability to pin the path of a flow). Thus, it is difficult to employ them in connectionless networks such as Internet without substantial modifications to the basic architecture.

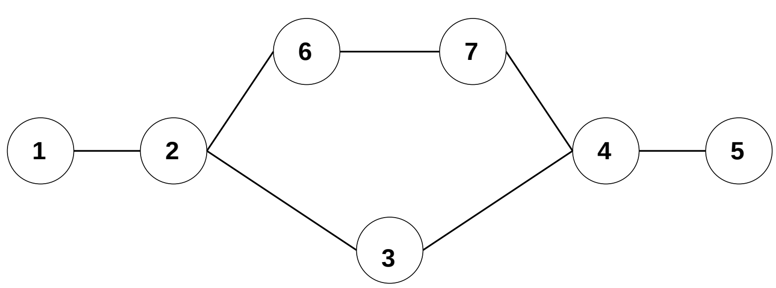

Figure 1: An example network

Second, they can introduce routing and load oscillations resulting into higher instability in the performance seen by best-effort traffic. To illustrate, when the path 1-2-3-4 becomes loaded in Figure 1, traffic will be diverted onto path 1-2-6-7-4. This will increase the load on path 1-2-6-7-4. Subsequently, when load on path 1-2-3-4 reduces, the traffic will again be diverted onto path 1-2-3-4. Thus, the best effort traffic flowing on those two paths will observe high instability in the performance which is undesirable.

To address these limitations, we develop an alternative routing technique. The technique is based on the observation that in future packet networks there will be significant amount of elastic best-effort traffic in addition to the traffic that requires QoS guarantee. Hence, it is unlikely that QoS traffic in itself will saturate the links. Therefore, if the QoS traffic is given priority in link bandwidth allocation, their requirements will be met. This technique, however, reduces the bandwidth available to the best-effort traffic. During peak loads from QoS traffic the bandwidth available for best-effort traffic may be so low that it leads to unacceptable performance for best-effort traffic. To ameliorate this problem, we reroute the best-effort traffic in case of high load from the QoS traffic. Thus, our routing algorithm is as follows:

- Use the conventional static shortest path routing for flows that require QoS guarantee.
- Use dynamically obtained load information to compute the shortest path routes for best-effort traffic. The techniques for distributing load information and using it for computing shortest paths are same as in QoS routing.

The advantages of our approach are:

- *It does not require connection oriented network layer*. Since packets of best-effort flows as well as those that require QoS are routed using the conventional shortest path datagram routing, our algorithm does not require connection oriented network layer. Thus, it can be realized in networks such as Internet.
- *It reduces route instability*. Unlike QoS routing where the path chosen was governed by the load imposed by QoS flows, in our algorithm the path chosen for best effort traffic is not governed by the best effort load. This reduces the coupling in the routing control loop. Hence, the route is expected to be more stable. This also is the key distinguishing characteristic of our algorithm from the ARPANET dynamic routing technique that made dynamic routing decisions for best effort traffic based on measured load of best effort traffic itself.
- *It can be realized at a lower cost than QoS routing*. In case of flows that require QoS guarantee, if a path that can meet the requirements can not be found, the flow is rejected which results into loss of revenue. To ameliorate this problem, dissemination of load information has to be done at a fast time scale. In contrast, in case of best-effort traffic, if the traffic is not routed over the best possible path, only performance is degraded. Thus, load information can be disseminated at a slower timescale. This will also ensure that best effort flows will be rerouted infrequently, if at all, and hence will observe very little packet reordering.

In summary, our routing algorithm is similar to conventional QoS routing algorithms; it just turns them upside down. It uses conventional routing for flows that require QoS guarantees but QoS routing techniques for best effort flows. We are currently evaluating the performance of our routing technique. Specifically, we are investigating the frequency with which the best-effort traffic has to be rerouted and the cost of our technique using traffic traces from a data and a telephone network. We will report on our results in a future publication.

References

[1] Eric Crawley, Raj Nair, Bala Rajagopalan, and Hal Sandick. A Framework for QoS-based Routing in the Internet. Internet RFC (RFC 2386), August 1998.

[2] R. Guerin, S. Kamat, A. Orda, T. Przygienda, and D. Williams. QoS Routing Mechanisms and OSPF Extensions. Internet Draft (draft-guerin-qos-routing-ospf-03.txt), work in progress, March 1998.

[3] R. Guerin, Ariel Orda, and Douglas Williams. QoS Routing Mechanisms and OSPF Extensions. In *Proceedings of IEEE GLOBECOM*, November 1997.

[4] Whay C. Lee, Michael G. Hluchyj, and Pierra A. Humblet. Routing Subject to Quality of Service Constraints in Integrated Communication Networks. *IEEE Network Magazine*, pages 46–55, July/August 1995.

[5] Qingming Ma and Peter Steenkiste. Quality-of-Service Routing for Traffic with Performance Guarantees. In *Proc. IFIP International Workshop on Quality of Service*, pages 115–126, Columbia University, New York, May 1997.

[6] Anees Shaikh, Jennifer Rexford, and Kang G. Shin. Efficient Precomputation of Quality-of-Service Routes. In *Proceedings of Workshop on Network and Operating System Support for Digital Audio and Video*, July 1998.