Efficient Algorithms for Computing Disjoint QoS Paths

CCIT Report #434

June 2003

Ariel Orda* and Alexander Sprintson†[1]

*Department of Electrical Engineering, Technion – Israel Institute of Technology, Haifa, Israel 32000
Email: ariel@ee.technion.ac.il

†Parallel and Distributed Computing Group, California Institute of Technology, Pasadena, California, USA 91125
Email: spalex@caltech.edu

Abstract

Networks are expected to meet a growing volume of requirements imposed by new applications such as multimedia streaming and video conferencing. Two essential requirements are support of Quality of Service (QoS) and resilience to failures.

In order to satisfy these requirements, a common approach is to use two disjoint paths between the source and the destination nodes, the first serving as a *primary* path and the second as a *restoration* path. Such approach, referred to as *path restoration*, has several advantages, the major one being the ability to switch promptly from one path to another in the event of a failure. A major issue in this context is how to identify two paths that satisfy the QoS constraints imposed by network applications. Since network resources, e.g., bandwidth, are allocated along both primary and restoration paths, we need to consider also the overall network performance. Accordingly, in this paper we study the fundamental problem of finding two disjoint paths that satisfy the QoS constraints at minimum cost. We present approximation algorithms with provable performance guarantees for this fundamental network problem.

Index Terms: Routing, Restoration, Disjoint Paths, Quality of Service.

I. Introduction

Networks are expected to meet *Quality of Service* (QoS) requirements imposed by new applications, such as multimedia streaming and video conferencing. This is facilitated by current efforts to provide resource reservations and explicit path routing, e.g., *MultiProtocol Label Switching* (MPLS). On the other hand, physical network infrastructures may be prone to failures. Therefore, a major challenge in this context is to develop adequate network mechanisms for establishing connections that satisfy QoS requirements and are also resilient to failures. It has been recognized that, for many practical settings, the speed and capacity of links do not allow to provision restoration paths *after* the failure. Thus, the restoration paths must be provisioned in advance, i.e., *before* a failure occurs.

This goal can be achieved by provisioning two disjoint QoS paths between the source and destination nodes. This approach is widely used because of its ability to switch promptly from one path to another in the event of a failure. The disjoint path strategy has many additional advantages. First, it allows to use various protection schemes, such as 1+1 protection or 1:1 protection [1]. With 1+1 protection, traffic is simultaneously transmitted on both paths, which allows instantaneous recovery from link failures. Alternatively, with 1:1 protection, traffic is transmitted along a *primary* path, and, upon a failure of one of its links, the traffic is switched to a *restoration* path. Second, the disjoint path strategy requires minimal network support, because failure detection and restoration can be implemented at the application level of the source. Finally, the disjoint path strategy provides a greater flexibility to application designers, as they can choose a protection scheme (e.g., 1+1 or 1:1) that is most adequate for each particular application.

To facilitate seamless recovery to a restoration path in the event of a failure, it is necessary to reserve network resources (e.g., bandwidth) on both the primary and restoration paths. Such resources should be consumed in a networkwide efficient manner. A common way for modelling the impact of such resource consumption on each link is by associating "costs" with the links. Accordingly, a major problem is to find two disjoint paths between source and destination nodes that satisfy end-to-end QoS constraints at minimum cost. This problem is the subject of this study.

QoS constraints occur naturally in a number of practical settings, involving bandwidth and delay-sensitive applications, such as voice over IP, audio and video conferencing, multimedia streaming, etc. QoS constraints can be divided into *bottleneck* constraints, such as bandwidth, and *additive* constraints, such as delay or jitter. Bottleneck QoS constraints can be efficiently handled by pruning links that do not satisfy them. The problem is then effectively reduced to finding two disjoint paths of minimum cost; this problem was extensively investigated in the literature [2]. Accordingly, in this study we focus on additive QoS constraints, which are more difficult to handle.

QoS routing has been the subject of several recent studies and proposals (see [3], [4] for comprehensive surveys). However, the problem of finding two disjoint QoS paths got little attention. Similarly, path restoration and routing over alternate paths has also attracted a large body of research (see, e.g., [5], [6]). Most of the proposed solutions, however, considered only bottleneck

[1] Part of this work was done while A. Sprintson was with the Department of Electrical Enginnering, Technion.