Journal of Discrete Algorithms 30 (2015) 29–44

Contents lists available at ScienceDirect

Journal of Discrete Algorithms

www.elsevier.com/locate/jda

Parameterized complexity of spare capacity allocation and the multicost Steiner subgraph problem

Tibor Jordán [a], Ildikó Schlotter [b,*]

[a] Department of Operations Research and the MTA-ELTE Egerváry Research Group on Combinatorial Optimization, Eötvös Loránd University, 1117 Budapest, Pázmány Péter sétány 1/C, Hungary
[b] Department of Computer Science and Information Theory, Budapest University of Technology and Economics, 1117 Budapest, Magyar Tudósok körútja 2, Hungary

ARTICLE INFO

Article history:
Received 26 October 2012
Received in revised form 3 November 2014
Accepted 23 November 2014
Available online 28 November 2014

Keywords:
Network design
Steiner forest
Parameterized complexity
Tree-width

ABSTRACT

We study the computational complexity of the Spare Capacity Allocation problem arising in optical networks that use a shared mesh restoration scheme. In this problem we are given a network with edge capacities and point-to-point demands, and the goal is to allocate two edge-disjoint paths for each demand (a working path and a so-called restoration path, which is activated only if the working path fails) so that the capacity constraints are satisfied and the total cost of the used and reserved bandwidth is minimized. We focus on the setting where we deal with a group of demands together, and select their restoration paths simultaneously in order to minimize the total cost. We investigate how the computational complexity of this problem is affected by certain parameters, such as the number of restoration paths to be selected, or the treewidth of the network graph. To analyze the complexity of the problem, we introduce a generalization of the Steiner Forest problem that we call Multicost Steiner Subgraph. We study its parameterized complexity, and identify computationally easy and hard cases by providing hardness proofs as well as efficient (fixed-parameter tractable) algorithms.

1. Introduction

In this paper, we give efficient combinatorial algorithms as well as hardness results for optimization problems arising in restoration planning strategies of optical networks. An important aspect of Generalized Multi-Protocol Label Switching (GMPLS) networks, which has been extensively studied in the last decade [1,2], is fast restoration of service after a network failure. We focus on restoration path selection in the design of a shared mesh restoration scheme, which is a key component of such strategies, since it determines the spare bandwith needed and hence also contributes to the required network resources and its total cost.

A restorable connection (Label-Switched Path, or LSP) in a GMPLS network supporting shared mesh restoration has a working path as well as a protection path. During normal network operation, the connection is established along the working path, with resources reserved along the protection path, which is activated when some link on the working path fails. A subset of links in the network that share the risk of failure at the same time are said to belong to a Shared Risk Link Group (SRLG): a failure of an SRLG means the failure of all links in the group. SRLGs can be used to model several types of failures, including single-link or single-node failures. For a connection to be restorable, the working path and the protection

* Corresponding author.
E-mail addresses: jordan@cs.elte.hu (T. Jordán), ildi@cs.bme.hu (I. Schlotter).

http://dx.doi.org/10.1016/j.jda.2014.11.005
1570-8667/

path have to be SRLG-disjoint, i.e., no SRLG can contain links of both the working path and the protection path of the connection.

To minimize the total bandwidth needed on the links of the network, shared restoration schemes allocate the bandwidth necessary for protection paths in a shared manner: a certain amount of bandwidth ensures protection for several demands at the same time. However, the bandwidth reserved along the protection paths must be sufficient to recover all affected restorable connections in the event of any single SRLG-failure. Hence, to realize shared restoration, bandwidth is reserved along the protection paths in such a way that two protection paths can be assigned the same bandwidth on a link only if the corresponding working paths are SRLG-disjoint, that is, they are not expected to fail simultaneously.

Most path selection algorithms first select the working path as the shortest path between the endpoints of the demand, with respect to appropriately defined edge-costs, and then select the protection path, trying to maximize bandwidth sharing and hence minimize the additional bandwidth needed. Several protection path selection algorithms have been developed for the situation when one protection path needs to be determined for a single additional demand [23,25]. These solutions provide different performance guarantees—some of them may overestimate the bandwidth that needs to be reserved on some links.

The algorithm most relevant to our approach is the Full Information Restoration (FIR) algorithm of Li, Wang, Kalmanek, and Doverspike [23]. Their algorithm is able to find an optimal solution for the single demand situation, where all working paths have been fixed. It can also be used to improve an existing solution (i.e. a complete list of path pairs for all demands) in a local search type algorithm, which replaces protection paths by better ones, one by one, whenever possible.

Our goal is to analyze the more general scenario, when we need to select protection paths for k new demands simultaneously, given that all the working paths as well as the protection paths of the existing demands are fixed. This approach has the following advantages:

- First, this simultaneous allocation problem can be thought of as a local search task: given a complete realization of the network (that is, a working and a protection path for each demand), is it possible to change the protection paths for a *subset of the demands* in a way that the total cost decreases? As modifying the working paths is usually infeasible, re-allocating some of the protection paths is probably the most natural approach in this setting. By repeating this procedure and re-allocating the protection paths for groups of demands iteratively, we can expect a significant decrease in the total cost of the network.
  Solving a hard optimization problem step-by-step through a sequence of such local improvements is the central idea of local search, a heuristic that is extremely useful in many real-world routing problems. In particular, it has been successfully applied in different capacity allocation problems [12,29]. To reduce costs using this method in our model as well, as a subtask we have to solve the above problem repeatedly.
- Second, this simultaneous allocation problem can also be considered as the core task of a spare capacity allocation procedure in networks where demands appear in an on-line fashion and, after fixing the new working paths, we may deal with the protection paths in groups of k without violating time constraints. Allocating spare capacity for the protection paths in larger groups may lead to solutions which are better than what we can achieve by doing it one by one.
- Third, this problem also arises in the case when some SRLG fails. In such a situation, the demands whose working paths failed activate their protection paths. Thus, these paths become unprotected, and we have to find new protection paths for them. Furthermore, the failure might effect some protection paths directly as well, leading again to simultaneous re-allocation.

We shall explore the complexity status of several versions of this simultaneous allocation problem from the *fixed-parameter tractability* point of view, focusing on the cases where the number k of new demands and/or the *treewidth* of the graph is considered to be constant. We provide hardness results wherever the problem remains intractable even if some parameter is fixed, and develop efficient algorithms in the remaining cases. For example, we give a linear-time algorithm in the case when k and the treewidth are both small.

To analyze the simultaneous allocation problem, we also introduce the Multicost Steiner Subgraph problem. This problem is an extension of the well-known Steiner Forest problem, and may be of independent interest. Its input is an undirected graph with a set of terminal pairs, and different edge costs defined for each terminal pair. The task is to connect each terminal pair by a path, minimizing the total cost under the following assumption: if an edge e is used by several paths connecting different terminal pairs, each having a different cost on the edge e, then the cost of e is defined as the maximum among these values. We show how this problem is related to the aforementioned local search variant of the Spare Capacity Allocation problem we investigate. We examine its computational complexity and give positive as well as negative results for it.

The organization of the paper is the following. Section 2 describes the notation and provides the necessary definitions. Section 3 deals with the simultaneous allocation problem and its connection to the Multicost Steiner Subgraph problem. Sections 4.1 and 4.2 contain our contribution regarding Multicost Steiner Subgraph; in Section 4.1 we present two FPT-algorithms, while Section 4.2 discusses some hardness results. We finish with some concluding remarks and some ideas for future research in Section 5.