Computers & Operations Research 64 (2015) 159–167

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

On the hop-constrained survivable network design problem with reliable edges

Quentin Botton [a], Bernard Fortz [b,*], Luis Gouveia [c]

[a] Louvain School of Management, Université catholique de Louvain, Louvain-la-Neuve, Belgium
[b] Départment d' Informatique, Faculté des Sciences, Université Libre de Bruxelles, Bruxelles, Belgium
[c] DEIO, CIO, Faculdade de Ciências da Universidade de Lisboa, Lisboa, Portugal

ARTICLE INFO

Available online 9 June 2015

Keywords:
Network design
Survivability
Hop constraints
Benders decomposition
Branch-and-cut algorithms

ABSTRACT

In this paper, we study the hop-constrained survivable network design problem with reliable edges. Given a graph with non-negative edge costs and node pairs Q, the hop-constrained survivable network design problem consists of constructing a minimum cost set of edges so that the induced subgraph contains at least K edge-disjoint paths containing at most L edges between each pair in Q. In addition, we consider here a subset of reliable edges that are not subject to failure. We study two variants: a static problem where the reliability of edges is given, and an upgrading problem where edges can be upgraded to the reliable status at a given cost. We adapt for the two variants an extended formulation proposed in Botton, Fortz, Gouveia, Poss (2011) [1] for the case without reliable edges. As before, we use Benders decomposition to accelerate the solving process. Our computational results indicate that these two variants appear to be more difficult to solve than the original problem (without reliable edges). We conclude with an economical analysis which evaluates the incentive of using reliable edges in the network.

1. Introduction

Given an undirected graph $G=(V,E)$ with a nonnegative cost c_{ij} associated with each edge $ij \in E$, and a set Q of node pairs, we want to find a set of edges of minimum cost such that the induced subgraph contains at least K edge-disjoint L-paths between each pair in Q. A path in G is a L-path if it contains no more than L edges. Given two distinct nodes $o, d \in V$, an od-path is a sequence of node-edges $P=(v_0, e_0, v_1, \ldots, e_{l-1}, v_l)$, where $l \geq 1$, $v_0, v_1, \ldots, v_l$ are distinct nodes, $v_0 = o$, $v_l = d$, and $e_i = v_i v_{i+1}$ is an edge connecting v_i and v_{i+1} (for $i=0,\ldots,l-1$). A collection $P_1, P_2, \ldots, P_k$ of od-L-paths is called edge-disjoint if any edge ij appears in at most one path. The problem considered is NP-hard [10] and was first studied by Huygens et al. [9] who only consider $L \leq 4$ and $K=2$. Recently, an extended formulation combined with a Benders decomposition approach to efficiently handle the large number of variables and constraints of the formulation has been proposed and tested in Botton et al. [1]. They consider instances with $1 \leq K \leq 3$ and $3 \leq L \leq 5$.

The survey by Kerivin and Mahjoub [11] describes several variants of this problem as well as techniques for solving them. There are several reasons for requiring edge-disjoint paths and including a limit on the number of edges for each path of each commodity. Briefly, the requirement for using disjoint paths guarantees the existence of a working path for each commodity after $K-1$ edge failures and the hop constraints impose a certain level of quality of service, since in most of the routing technologies, delay is caused at the nodes, and thus, it is usual to measure the delay in a path in terms of its number of intermediate nodes, or equivalently, its number of edges (or hops).

In this paper, we study two different versions of the problem with different degrees of reliability associated with the edges of the underlying graph. In the first variant we assume that the edges in a subset $E_R \subset E$ are more reliable but more costly. Essentially, these edges are not prone to failure and the edge-disjoint property must be guaranteed only for the remaining edges. Thus, for a given commodity, all the K paths can simultaneously use each reliable edge.

This variant arises in the context of multi-layered telecommunication networks, for example, a SONET/SDH layer over an optical network layer. A link of the SONET/SDH layer can be considered as a demand that should be routed through a path in the optical layer. If this path is protected against failures, then the link corresponding to this path in the SONET/SDH layer can be viewed as "reliable." Otherwise the link can fail and hence requires some protection against failures at the SONET/SDH layer (see [13]). Zotkiewicz et al. [14] present a polynomial time algorithm to solve the particular case of this variant where $K=2$.

* Corresponding author.
E-mail addresses: quentin.botton@uclouvain.be (Q. Botton), bernard.fortz@ulb.ac.be (B. Fortz), legouveia@fc.ul.pt (L. Gouveia).

http://dx.doi.org/10.1016/j.cor.2015.05.009
0305-0548/

In the second variant, we give the network provider the option of choosing whether an edge should be reliable or not. That is, the subset E_R corresponds to edges ij that could be upgraded to more reliable edges at a higher cost.

These two variants are obviously NP-hard since they contain the original problem without reliable edges as particular case.

The paper is organized as follows: In Section 2, we first review the model proposed in [1] for the original case where all edges are assumed to be unreliable and then we show how to adapt this original model to accommodate the new specifications. In Section 3 we show the Benders reformulation of both problems and in Section 4 a branch-and-cut algorithm is described. Computational results are reported in Section 5, and we provide concluding remarks in Section 6.

2. Problem definition and formulations

In this section, we start by describing the model for the problem without reliable edges proposed in [1] and then, we show how to modify the model for the two variants previously described.

2.1. Original problem (no reliable edges)

The main idea of the formulation is to model the subproblem associated with each commodity with a directed graph composed of $L+1$ layers as illustrated in Fig. 1.

From the original undirected graph $G=(V,E)$, we create a directed layered graph $G^q=(V^q,A^q)$ for each commodity q, where $V^q=V_1^q \cup \ldots \cup V_{L+1}^q$ with $V_1^q=\{o(q)\}$, $V_{L+1}^q=\{d(q)\}$ and $V_l^q=V\backslash\{o(q)\}$, $l=2,\ldots,L$. Let v_l^q be the copy of $v \in V$ in the l-th layer of graph G^q. Then, the arc sets are defined by $A^q=\{(i_l^q,j_{l+1}^q) \mid ij \in E, i_l^q \in V_l^q, j_{l+1}^q \in V_{l+1}^q, l \in \{1,\ldots,L\}\} \cup \{d(q)^l, d(q)^{l+1}, l \in \{2,\ldots,L\}\}$, see Fig. 1. In the sequel, an (undirected) edge in E with endpoints i and j is denoted by ij while a (directed) arc between $i_l^q \in V_l^q$ and $j_{l+1}^q \in V_{l+1}^q$ is denoted by (i,j,l) (the commodity q is omitted in the notation as it is often clear from the context).

Note that each path between $o(q)$ and $d(q)$ in the layered graph G^q is composed of exactly L arcs (hops), which correspond to a maximum of L edges (hops) in the original graph (the horizontal-loop arcs at the bottom of Fig. 1 allow paths with less than L arcs). This transformation was first proposed by Gouveia [7] and is motivated by the fact that in the layered graph any path is feasible with respect to the hop-constraints. A set of K-paths satisfying the hop limit can be obtained by sending K units of flow in the layered graph, leading to the following extended formulation:

$$\text{(Hop)} \quad \min \sum_{ij \in E} c_{ij} z_{ij} \text{ s.t.}$$

$$\sum_{j:(j,i,l-1) \in A^q} x_{ji}^{l-1,q} - \sum_{j:(i,j,l) \in A^q} x_{ij}^{l,q} = \begin{cases} -K & \text{if } (i=o(q)) \text{ and } (l=1) \\ K & \text{if } (i=d(q)) \text{ and } (l=L+1) \\ 0 & \text{else,} \end{cases}$$

$$i \in V^q, l \in \{2,\ldots,L+1\}, \quad q \in Q, \tag{1}$$

$$\sum_{l \in \{1,\ldots,L\}} \left(x_{ij}^{l,q}+x_{ji}^{l,q}\right) \le z_{ij}, \quad ij \in E, \quad q \in Q, \tag{2}$$

$$z_{ij} \in \{0,1\}, \quad ij \in E, \tag{3}$$

$$x_{ij}^{l,q} \ge 0 \text{ and integer}, \quad (i,j,l) \in A^q, \quad q \in Q. \tag{4}$$

Each binary variable z_{ij} indicates whether edge $ij \in E$ is in the solution and each variable $x_{ij}^{l,q}$ describes the amount of flow through arc (i,j,l) for commodity q in layered graph G^q (constraints (2) together with (3) imply that $x_{ij}^{l,q} \le 1$ for $i \ne j$, and thus, these variables may be interpreted as indicating whether arc (i,j) is used in position l in one of the paths of commodity q). Constraints (1) are the flow conservation constraints at every node of the layered graph and guarantee the existence of K paths from $o(q)$ to $d(q)$. Constraints (2) guarantee edge-disjointness of the paths.

As discussed in [1], for $K=1$ and any L, for any K and $L=2,3$ and for $L=4$ and $K=2$, the integrality on x can be dropped (x will be integral as soon as z is). However, in the general case, x can be fractional even if z is integer.

2.2. The static problem

Let us first consider the version of the problem where the reliable status of an edge is given a priori. We denote by E_R the set of reliable edges and by $E_{\overline{R}}$ the set of remaining edges. The formulation (HopR1) for this variant uses the same variables and is very similar to the extended formulation (Hop) for the original problem, and thus we only specify the main differences.

To make meaningful comparisons between the formulations, we assume that the cost of a reliable edge is $p_{ij}>1$ times the cost of the unreliable version of that edge used in (Hop), i.e. a reliable edge $ij \in E_R$ has cost $p_{ij}c_{ij}$:

$$\text{(HopR1)} \quad \min \sum_{ij \in E_R} p_{ij} c_{ij} z_{ij} + \sum_{ij \in E_{\overline{R}}} c_{ij} z_{ij}$$

$$\text{s.t.} \quad (1), (3), \text{ and } (4),$$

$$\sum_{l \in \{1,\ldots,L\}} (x_{ij}^{l,q}+x_{ji}^{l,q}) \le K z_{ij}, \quad ij \in E_R, \quad q \in Q, \tag{5}$$

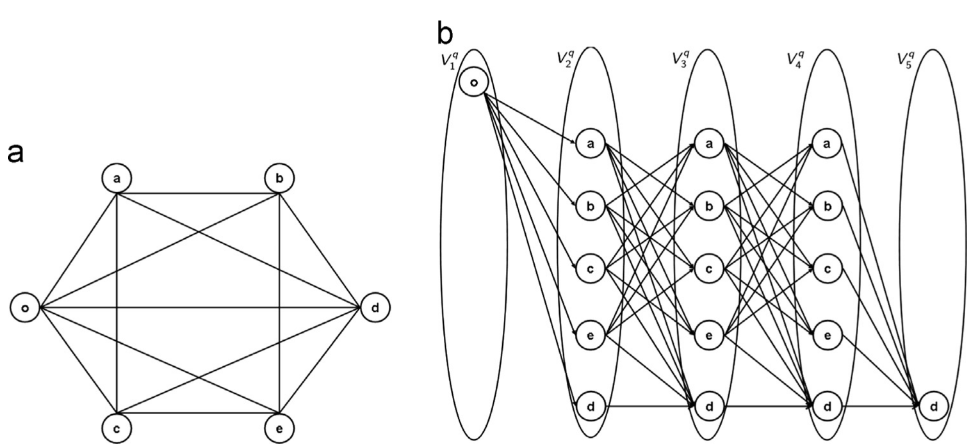

Fig. 1. A basic network (a) and its alternative layered representation (b) when $L=4$.