# Design of Survivable Networks: A Survey

**Hervé Kerivin***
*Institute for Mathematics and its Applications, University of Minnesota, 400 Lind Hall, 207 Church Street S.E., Minneapolis, Minnesota 55455*

**A. Ridha Mahjoub**
*Laboratoire LIMOS, CNRS UMR 6158, Université Blaise Pascal, Clermont-Ferrand II, Complexe Scientifique des Cézeaux, 63177 Aubière Cedex, France*

**For the past few decades, combinatorial optimization techniques have been shown to be powerful tools for formulating and solving optimization problems arising from practical situations. In particular, many network design problems have been formulated as combinatorial optimization problems. With the advances of optical technologies and the explosive growth of the Internet, telecommunication networks have seen an important evolution and therefore designing survivable networks has become a major objective for telecommunication operators. Over the past years, much research has been carried out to devise efficient methods for survivable network models, and particularly cutting plane based algorithms. In this paper, we attempt to survey some of these models and the optimization methods used for solving them. © 2005 Wiley Periodicals, Inc. NETWORKS, Vol. 46(1), 1–21 2005**

**Keywords:** survivable network; heuristic; polyhedral approach; hop-constraint

## 1. INTRODUCTION

The concept of robust networks is among the most frequently recurring one in the problems of designing telecommunication networks. There exist several ways to express the network robustness, yet it can be defined as the continued ability of the network to perform its function in the face of damage and outages. The network design process is extremely complicated because it manages the traffic, the performance, and the resources of the network together so one cannot consider it as a single optimization problem. Therefore, it should be broken down into several optimization problems (topology computation, traffic prediction, and modeling, dimensioning, etc.), which may have their own robustness component. In this article, we are only interested in the topology computation problem where a network is represented as a collection of nodes (switches, routers, hubs, multiplexers, satellites, base stations, etc.) and connections between them by edges (optical fibers, electrical wires, etc.), and the robustness of the network topology will come from its *reliability*. The latter depends on the equipment (i.e., link or node) reliability, but also on the manner in which nodes are connected together. Therefore, the network reliability can be characterized by many parameters such as degree of each node, average distance between every pair of nodes, connectivity, etc. In this article, we base the network reliability on the presence of alternate paths (i.e., the connectivity parameter) and then consider the *survivability* of the network, which can be described as below.

Telecommunication networks, whatever is the nature of the particular layer (e.g., SDH/SONET, ATM, WDM, IP), have to be immune to equipment failures. This concept of survivability allows networks to remain functional when links are severed or nodes fail, that is, network services can be restored in the event of catastrophic failures. Therefore, one of the main concerns when designing telecommunication networks is to compute network topologies that provide protection against network equipment failures. The topology computation problem is usually the first stage of the overall network design optimization process; the following ones involve some traffic and routing issues.

The introduction of new control plans [e.g., Generalized MultiProtocol Label Switching (GMPLS) in optical networks] has created, over the last years, a movement toward networks that should have more complex topologies than rings. This fact leads to the specification of certain survivability conditions, usually modeled in terms of

Received June 2004; accepted April 2005
*Correspondence to*: A.R. Mahjoub; e-mail: Ridha.Mahjoub@math.univ-bpclermont.fr
*Present address: Laboratoire LIMOS, CNRS UMR 6158, Université Blaise Pascal, Clermont-Ferrand II, Complexe Scientifique des Cézeaux, 63177 Aubière Cedex, France
Contract/grant sponsor: Université Libre de Bnuxelles, Belgium
DOI 10.1002/net.20072
Published online in Wiley InterScience (www.interscience.wiley.com).

node or link connectivity, which should be ensured. Thus, the topology computation problem we are interested in, called the *survivable network design problem* and denoted by *SNDP*, consists of selecting links so that the sum of their costs is minimized and some given requirements for the number of paths between every pair of nodes are satisfied. To respond to node (respectively link) failures scenarios, the paths between two nodes should fulfill the additional property that they cannot share any other node (respectively any link), implying the so-called *node-survivability conditions* (respectively *link-survivability conditions*). The survivable network design problem can then be stated in two slightly different versions according to which of the two kinds of survivability conditions should be considered.

Consider an undirected graph $G = (V, E)$, where $V$ represents the node set, and $E$ the set of edges or potential links. To express the survivability conditions, we need to introduce the following graph-theoretic concepts. Given two distinct nodes $s$ and $t$ of $V$, an *st-path* is a sequence $P = (v_0, e_1, v_1, \ldots, e_k, v_k)$, where $k \geq 1$, $v_0, v_1, \ldots, v_k$ are distinct nodes, $v_0 = s$, $v_k = t$, and $e_i$ is an edge connecting $v_{i-1}$ and $v_i$ (for $i = 1, \ldots, k$). A collection $P_1, P_2, \ldots, P_l$ of $st$-paths is called *node-disjoint* (respectively *edge-disjoint*) if any node except for $s$ and $t$ (respectively, any edge) appears in at most one path. A subgraph $H$ of $G$ is called *node-survivable* (respectively, *edge-survivable*) if for any $s, t \in V$, $H$ contains at least a prespecified number of node-disjoint (respectively, edge-disjoint) $st$-paths. Suppose that each edge $e \in E$ has a certain cost $c(e) \in \mathbb{R}_+$ (e.g., the cost of digging down a cable and the price of the equipment facilitating communication), then the *node-survivable network design problem*, denoted by *NSNDP*, consists of finding a node-survivable subgraph of $G$ with minimum total cost, where the cost of a subgraph is the sum of the cost of its edges. Similarly, the *link-survivable network design problem*, denoted by *LSNDP*, consists of finding a minimum-cost edge-survivable subgraph of $G$.

Polyhedral combinatorics is a well-established approach to combinatorial optimization problems (see, i.e., Schrijver [110]), which may lead to new exact and approximate solution methods. This article provides a review and synthesis of polyhedral approaches to the two versions of the survivable network design problem. The article is divided into 10 sections. In the second section, we precisely present the model that will be studied throughout the article, and we discuss the complexity issue. In Section 3, we overview some polynomially-solvable cases as well as some of the main heuristics and approximation algorithms devised for the survivable network design problem. Section 4 is dedicated to an integer linear programming formulation for the SNDP and to a brief description of the so-called polyhedral approach. Some facet-defining inequalities are then presented in Section 5, where their separation problems are also mentioned. Section 6 concerns some linear descriptions of the polytopes associated with the problem on some special classes of graphs. The concept of critical extreme points is then discussed in Section 7, along with their algorithmic implications. Some of the theoretical results presented in the first seven sections were then used in a branch-and-cut algorithm that is presented in Section 8, together with a discussion of computational results. Finally, before giving some concluding remarks in a last section, we are interested in the survivable network design problem with length constraints in Section 9.

The rest of this introduction is devoted to additional definitions and notation that will be used in this article. The graphs we consider are finite, loopless, and connected.

We consider a graph $G = (V, E)$, and we denote by $n$ the number of nodes of $G$, that is, $n = |V|$. For $W \subseteq V$, we let $\overline{W} = V \setminus W$. Given two distinct nodes $u$ and $v$ of $V$, an edge between both is denoted by $uv$. For a nonempty node subset $W \subsetneq V$, the set of edges having exactly one endnode in $W$ is called a *cut* or a *cutset*, and is denoted by $\delta_G(W)$. Moreover, if $s \in W$ and $t \notin W$, then $\delta(W)$ is called an *st-cut*. The sets $W$ and $\overline{W}$ are called the *shores* of the cut $\delta(W)$. If $W = \{u\}$, we then write $\delta_G(u)$ for $\delta_G(\{u\})$. If $W_1$ and $W_2$ are two disjoint subsets of $V$, then $[W_1, W_2]$ denotes the set of edges having one endnode in $W_1$ and the other in $W_2$. A *partition* of $V$ is a collection of disjoint subsets of $V$ with union $V$. The elements of the partition are called its *classes*. Given a partition $\{V_1, \ldots, V_p\}$ of the node set $V$, we denote by $\delta_G(V_1, \ldots, V_p)$ the set of edges with endnodes in two different classes. For all our notation, we do not use the subscript $G$ whenever the graph $G$ can be deduced from the context. For $F \subseteq E$, we denote by $V(F)$, the set of nodes that are spanned by the edges in $F$. For $W \subseteq V$, we denote by $E(W)$, the set of edges with both endnodes in $W$, and by $G(W) = (W, E(W))$, the subgraph induced by $W$. Given $e = uv \in E$, *contracting* $e$ means deleting $e$, identifying $u$ and $v$, deleting the resulting loops and keeping the new parallel edges. If $F \subseteq E$, then $G/F$ denotes the graph obtained from $G$ by contracting $F$, that is, by contracting all edges in $F$. If $Z \subseteq V$, $G \backslash Z$ is then the graph obtained from $G$ by deleting $Z$ and the edges incident to $Z$. If $Z = \{u\}$, we then write $G \setminus u$ for $G \setminus \{u\}$. The *dimension* of a polyhedron $P$, denoted by $\dim(P)$, is the maximum number of affinely independent points in $P$ minus 1. Let $a \in \mathbb{R}^n$ be a row vector. An inequality $ax \geq \alpha$ is said to be valid for $P$ if $P \subseteq \{x \mid ax \geq \alpha\}$; the set $F = \{x \in P \mid ax = \alpha\}$ is called the *face* defined by $ax \geq \alpha$. If $\dim(F_a) = \dim(P) - 1$, and $F_a \neq \emptyset$, then $F_a$ is called a *facet*, and $ax \geq \alpha$ is called a *facet-defining inequality*. Given a polytope $P \subseteq \mathbb{R}^n$, the *dominant* of $P$ is the polyhedron given by $P + \mathbb{R}^n_+$.

## 2. A MODEL FOR THE SURVIVABLE NETWORK DESIGN PROBLEM

The survivable network design problem has received considerable attention in the past, and two models precisely specifying the survivability conditions have been mainly considered. The first one, originally formulated by Steiglitz et al. [111] and later called the *generalized Steiner problem* by Winter [122], is as follows.

Given an undirected graph $G = (V, E)$ and a cost vector $c \in \mathbb{R}^E_+$ on the edges, the node-survivability (respectively, link-survivability) conditions are specified by a symmetric integer $n \times n$ matrix $R = [r_{st}]$, where the entry $r_{st}$ prescribes the number of node-disjoint (respectively, edge-disjoint) $st$-paths needed for $s, t \in V$. This model has been extensively investigated (see, i.e., [26, 28, 55, 94]), and is a special case of a more general model introduced by Grötschel and Monma [75], where, for any $s, t \in V$, survivability is also measured by a minimum number $r_{st}$ of disjoint $st$-paths remaining after the deletion of any node subset having a given cardinality $k_{st}$. This model is a kind of general framework for the SNDP, and it requires the knowledge of considerable amount of data to specify particular connectivity requirements for every pair of nodes. Yet, some data may not be available in real-world applications, and therefore, a slightly more restricted and realistic model was introduced by Grötschel et al. [76–78] (see also Stoer [112]). This second model is based on the specification of node types to model the survivability conditions as described below, and captures the important aspects of practical problems.

A generic telecommunication network consists of access networks that connect the terminals (e.g., user nodes) to concentrators (e.g., switches, multiplexers) and a backbone network that interconnects these concentrators or connects them to a central unit. The access and backbone networks can be fully or partially connected according to the level of survivability that is required, and their topologies may differ. The backbone network can also be partitioned into smaller subsets of nodes, which in turn, can be partitioned into even smaller subsets of nodes, implying eventually different levels of survivability requirements. The partitioning of the network creates a hierarchical structure and makes some nodes more important than some other ones, because of their specific functions. The lowest level of the hierarchy corresponds to the terminals that only require being connected to the network, whereas for a node of the backbone network, the higher the level of the hierarchy to which it belongs, the "higher" the degree of survivability it requires. This hierarchical structure of telecommunication networks leads us to grade the nodes in the order of their relative importance as described below.

Consider an undirected graph $G = (V, E)$, where each edge $e \in E$ has a cost $c(e) \in \mathbb{R}_+$. To each node $u \in V$ is associated a nonnegative integer $r(u)$, called its *connectivity type*, which represents the importance of communication from and to that node. The integer vector $r = (r_u,\ u \in V)$ will then be called the *connectivity type vector*. The node-survivability (respectively, link-survivability) conditions are then stated as the requirement of the existence of at least

$$r(s,t) = \min\{r(s), r(t)\} \tag{1}$$

node-disjoint (respectively, edge-disjoint) paths in the subgraph of $G$ for any pair of nodes $s, t \in V$. We remark that modeling the survivability conditions using node types is a particular case of the generalized Steiner problem; it corresponds to the case where $r_{st} = \min\{r(s), r(t)\}$ for any $s, t \in V$. Yet, as mentioned above, such modeling is particularly suitable for telecommunication networks. Moreover, this connectivity type-based model can be used to model numerous other applications having hierarchical structures, such as distribution networks (involving, i.e., major central facilities, minor regional depots), wireless networks (involving, for instance, base transceiver stations, base station controllers, transcoder/rate adapter units, mobile switching centers), etc.

Let us denote by $r_{\max}$ the maximum connectivity type, that is,

$$r_{\max} = \max\{r(u) \mid u \in V\}.$$

If $r_{\max} \leq 2$, we then deal with the *low-survivability case* which was shown to be cost effective and provides an adequate level of survivability for telecommunication networks [103]. In fact, as failures are not very common in practice, telecommunication network designers consider protection strategies that will withstand single network equipment failures. This fact implies the classification of the nodes into three kinds: specific nodes, which must be protected from single equipment failures [i.e., nodes $u \in V$ with $r(u) = 2$]; ordinary nodes, which simply have to be connected to the network [i.e., nodes $u \in V$ with $r(u) = 1$]; and optional nodes, which may be considered in the network, depending only on some design considerations [i.e., nodes $u \in V$ with $r(u) = 0$]. However, this traditional protection for telecommunication networks tends to be now outdated (it was essentially dedicated to circuit-switched telephone networks), and the new generation networks (e.g., packet based data networks carrying voice, video, or data traffic) require more complicated and adaptive protection strategies to face the competitive environment. This practical motivation, combined with the interesting theoretical framework of the model, leads us to consider also the *high-survivability case*, that is, where $r_{\max} \geq 3$.

Expressing the survivability requirements using the connectivity types allows us to model a wide variety of well-known combinatorial optimization problems that have been intensely studied for several decades. For instance, if the connectivity type vector $r$ is uniform, say $r(u) = k$ for all $u \in V$ where $k$ is a positive integer, then the NSNDP (respectively, LSNDP) is nothing but the *k-node connected network problem* denoted by *kNCNP* (respectively, *k-edge connected network problem* denoted by *kECNP*). Furthermore, the 2-node connected network problem includes the *traveling salesman problem* (i.e., find a simple circuit, also called tour, passing through all the nodes, for which the cost is minimized) as a special case. In fact, if a large constant is added to the cost of each edge, any optimal solution of the 2NCNP will have a minimum number of edges, and such a solution will be an optimal traveling salesman tour. Another famous version of the SNDP is where the entries of the connectivity type vector $r$ take their values in {0,1}. This is the *Steiner tree problem*, which consists of finding a minimum tree of $G$, spanning the so-called terminal nodes [i.e., the nodes $u \in V$ with $r(u) = 1$]. [The nodes $u \in V$ with $r(u) = 0$ are called *Steiner nodes*.] We notice that this equivalence between the Steiner tree problem and the survivable network design problem with $r \in \{0, 1\}^V$ holds only if the cost of each edge is positive.

Because the survivable network design problem contains, as special cases, known NP-hard problems such as the traveling salesman problem and the Steiner tree problem, [58] it is clearly NP-hard in general. Moreover, the traveling salesman problem is known to be NP-hard in the strong sense, which means that it cannot be solved by a pseudopolynomial time algorithm unless $P = NP$. Therefore, we have:

**Theorem 1 [58].** *The survivable network design problem is strongly NP-hard.*

In this article, we only consider the second model to express the connectivity requirements, that is, the connectivity type-based model. Nevertheless, we may also deal with the first model in the next section, which reviews some of the most noteworthy research on the survivable network design problem, with a main focus on the polynomially-solvable cases as well as some devised heuristics and approximation algorithms.

## 3. SPECIAL CASES, HEURISTICS, AND APPROXIMATION ALGORITHMS

In addition to the problems mentioned above, some other special cases of the survivable network design problem have received considerable attention over the last century. Hence, there exists a plentiful literature on the different problems related to the SNDP, and the aim of this section is to present a brief, but as complete as possible, overview of it.

### *3.1. Polynomially Solvable Cases*

Despite the NP-hardness of the survivable network design problem, it happens that this problem may be solved in polynomial time, depending on some special connectivity type vectors, some special edge cost functions and/or some special classes of graphs.

The survivable network design problem was shown to be polynomially solvable if $r(u) = 1$ for all $u \in V$. In fact, this version of the SNDP is nothing but the minimum spanning tree problem which is a well-solved combinatorial optimization problem [3]. If we now have some nodes having their connectivity type equal to 0, we then deal with the Steiner tree problem, which is a well-known NP-hard problem. However, Lawler [92] gave two algorithms for solving the Steiner tree problem, which are either polynomial in the number of terminal nodes and exponential in the number of Steiner nodes, or vice versa. Therefore, if $r \in \{0, 1\}^V$ and either the number of nodes of connectivity type 0 or the number of nodes of connectivity type 1 is restricted, the SNDP can be solved in polynomial time.

Another famous version of the survivable network design problem is where all the nodes have connectivity types equal to 0 except for exactly two nodes, $u_1$ and $u_2$, which have $r(u_1) = r(u_2) = 1$. This is actually the shortest path problem between $u_1$ and $u_2$, for which there exist different polynomial-time algorithms provided there are no negative-cost cycles. Moreover, if the shortest path must satisfy a hop-constraint, that is, it has no more than $L$ links where $L$ is a positive integer, a dynamic programming approach then permits one to solve the problem [3]. The hop-constraint is meaningful for telecommunication networks because of some routing considerations that might be taken into account (see Section 9). If the connectivity types associated with $u_1$ and $u_2$ are now equal to $k$, where $k$ is a fixed positive integer, one deals with the $k$-shortest path problem, which can also be solved in polynomial time [114, 115].

Some other polynomially solvable cases of the survivable network design problem arise from special edge cost functions. If the edge costs are uniform, that is, they are restricted to be equal to 1, the problem then consists of finding a node-survivable or edge-survivable subgraph having a minimum number of edges. For this kind of edge cost function, the SNDP with node-survivability conditions has not been solved yet. On the other hand, the SNDP with link-survivability conditions and uniform edge costs was shown to be polynomially solvable provided $r(u) \geq 1$ for all $u \in V$, and the use of parallel edges is allowed [112]. The algorithm for solving this problem is similar to the one given by Chou and Frank [26] for the generalized Steiner problem with link-survivability conditions, uniform edge cost function, and the possibility of using parallel edges in the solution. Chou and Frank [27] considered the same version of the generalized Steiner tree problem when no parallel edges but additional nodes are allowed, and they gave a polynomial-time algorithm to solve it.

The survivable network design problem with edge costs in $\{0, 1\}$ is known as the augmentation problem, which can be stated as follows. It consists of augmenting a graph by a minimum number of edges in $V \times V$ (possibly using parallel edges) such that the survivability requirements are met. This version of the SNDP is equivalent to augmenting the graph induced by the edges having costs equal to 0 using eventually the edges having costs equal to 1. Frank [55] solved this problem for the link-survivability conditions (more precisely for the generalized Steiner problem), and he generalized many results related to the uniform connectivity type vector case. For the node-survivability conditions, some polynomial-time algorithms were given to augment a graph to a 2-node connected one [48, 80], and in those algorithms, no parallel edges are allowed. Moreover, when some nodes are subject to node-survivability conditions and the other ones are only subject to link-survivability conditions, Hsu and Kao [80] devised a polynomial-time algorithm to solve the augmentation problem if $r(u) = 2$ for all $u \in V$.

The survivable network design problem may also be solved in polynomial time for special connectivity type vectors and underlying graphs belonging to certain classes. Some of the results related to those polynomially solvable cases will be discussed in Section 6, and therefore we content ourselves with summarizing the most important ones below. Thus, let us first define the three classes of graphs we are going to consider hereafter and in Section 6, and in which the survivable network design problem has been extensively investigated. A *homeomorph* of $K_4$ (i.e., the complete graph on four nodes) is a graph obtained from $K_4$ when its edges are subdivided into paths by inserting new nodes of degree 2. A graph is called *series-parallel* if it contains no homeomorph of $K_4$ as a subgraph. A graph is called *outerplanar* if it can be drawn in the plane as one cycle with noncrossing chords. We note that outerplanar graphs are also series-parallel. A graph is said to be a *Halin graph* if it consists of a cycle and a tree without nodes of degree 2 whose leaves are precisely the nodes of the cycle.

If $r \in \{0,1\}^V$, the survivable network design problem, which is related to the Steiner tree problem in that case, can then be solved in polynomial time on series-parallel graphs as shown by Takamizawa et al. [116]. For the three classes of graphs mentioned above, Winter [119–121] gave polynomial-time algorithms to solve the SNDP with $r \in \{0,2\}^V$ and either the node-survivability or the link-survivability conditions. On Halin graphs, Winter [122] also gave polynomial-time algorithms to solve the survivable network design problem for both survivability conditions and $r \in \{0,3\}^V$. If the graph $G$ does not have $W_4$ (the wheel on five nodes) as a minor or if $G$ is a Halin graph, Coullard et al. [32] devised a linear time algorithm for the SNDP with the node-survivability conditions and $r \in \{0,2\}^V$. (A graph $H$ is a *minor* of a graph $G$ if $H$ arises from $G$ by a series of deletions and contractions of edges and deletions of nodes.) Kerivin and Mahjoub [86] showed that the link-survivable network design problem with $r(u)$ even for all $u \in V$ can be solved in polynomial time on series-parallel graphs. The $k$-edge connected network problem was shown to be polynomially solvable by Didi Biha and Mahjoub [46] on series-parallel graphs, where $k$ is a positive integer. Recently, Didi Biha et al. [45] gave a polynomial-time algorithm, based on the ellipsoid method, for solving the survivable network design problem where $r \in \{1,2\}^V$ and the underlying graph belongs to a subclass of series-parallel graphs that strictly contains all the outerplanar ones.

### 3.2. Heuristics and Approximation Algorithms

As the survivable network design problem is NP-hard, a considerable amount of research has been conducted into the design of heuristics and approximation algorithms. (Recall that a *$\rho$-approximation algorithm* is an algorithm that always delivers a solution of cost at most $\rho$ times the optimum.) The rest of this section is devoted to reviewing some among the most important results for these two non-exact approaches.

In the design of efficient heuristics, the knowledge of structural properties of the solution is often very useful because of the possible improvements in the problem formulations. If the edge cost function satisfies the triangle inequalities (i.e., $c(e_1) \leq c(e_2) + c(e_3)$ for every triplet of edges $(e_1, e_2, e_3)$ defining a triangle), Frederickson and Jájá [56] showed that a 2-edge connected graph can be transformed into a 2-node connected one without any increase of the cost. This result leads to an equivalence between the node-survivability conditions and the link-survivability ones if the connectivity types are uniform and equal to 2. For the same kind of edge cost function, Monma et al. [102] obtained a structural description of the optimal solutions of the 2-node connected network problem, and this work was later generalized to the $k$-node connected network problem by Bienstock et al. [17]. In the latter, it is also proven that optimal $k$-node and $k$-edge connected subgraphs may have different costs.

The first heuristics for the survivable network design problem appeared with the work of Steiglitz et al. [111]. Actually, they considered the generalized Steiner problem, and their method is based on a randomized greedy algorithm to produce an initial feasible solution that is then improved by a local-search approach. For the SNDP where $r \in \{1,2\}^V$, Monma and Shallcross [103] used a similar approach consisting of improving an initial solution generated from information given by the structural properties of the solution of the 2-node and 2-edge connected network problems. Their improvement heuristics is inspired from some local search heuristics devised from the traveling salesman problem (e.g., 2-opt, pretzel). Later, Monma-Shallcross's heuristics were modified by Ko and Monma [91] to tackle both $k$-node and $k$-edge connected network problems, and by Clarke and Anandalingam [29], who added a new heuristic to generate initial feasible solutions.

Over the last 2 decades, the question of designing approximation algorithms for the survivable network design problem has been extensively investigated. In [118], Williamson et al. (see also [57]) used a primal-dual method for approximation algorithms [63] to achieve an approximation factor of $2r_{\max}$. Later, Goemans et al. [61] refined the approach of [118], and improved their approximation factor to $2\mathcal{H}(r_{\max})$, where $\mathcal{H}(n) = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$ is the $n^{th}$ harmonic number. Jain [83] proposed a factor 2 approximation algorithm, which is based on first solving the linear relaxation of the problem and then iteratively rounding the solution. We notice that those algorithms were actually devised for a more general problem, which consists of finding a minimum cost subgraph having at least $f(S)$ edges crossing each cut $\delta(S)$, where $S \subset V$ and $f$ is a proper function. (A function $f : 2^V \to \mathbb{Z}_+$ is proper if $f(V) = 0$, $f$ is symmetric, that is, $f(S) = f(V \setminus S)$ for all $S \subseteq V$, and $f$ satisfies the maximality property, that is, $f(A \cup B) \leq \max\{f(A), f(B)\}$ for any disjoint $A, B \subseteq V$.)

For connectivity type vectors in $\{0,1,2\}^V$, Balakrishnan et al. [10] gave a $\frac{3}{2}$-approximation algorithm for the link-survivable network design problem. Ravi and Williamson [108] presented a $2\mathcal{H}(k)$-approximation algorithm for the $k$-node connected network problem, and they also gave a 3-approximation algorithm for the generalized Steiner problem where $R \in \{0,1,2\}^{V\times V}$. For the $k$-node connected network problem, a 2-approximation algorithm was devised by Khuller and Raghavachari [89], provided that the edge cost function satisfies the triangle inequalities. Using a weighted matroid intersection algorithm, Khuller and Vishkin [90] gave a 2-approximation algorithm for the $k$-edge connected network problem. There also exist approximation algorithms for the survivable network design problem where parallel edges are allowed. Goemans and Bertsimas [60] thus gave a $\min\{2\mathcal{H}(r_{\max}), 2q\}$-approximation algorithm for the SNDP with $r \in \mathbb{Z}_+$ based on a new analysis of a well-known algorithm for the Steiner tree problem, where $q$ denotes the number of distinct connectivity type values. Using a primal-dual approach, Aggarwal et al. [2] obtained a $2(\log_2 r_{\max})$-approximation algorithm for the survivable network design problem with general connectivity type vectors. Goemans and Williamson [62] used the way in which primal-dual algorithms solve combinatorial linear programs that have integer optimal solutions to devise a $2\mathcal{H}(r_{\max})$-approximation algorithm for the generalized Steiner problem where $R \in Z_{+}^{V\times V}$. Recently, Aggarwal and Garg [1] improved this result by giving a $2(\log_2 |V|)$-approximation algorithm using a scaling technique.

## 4. INTEGER LINEAR PROGRAMMING FORMULATION AND ASSOCIATED POLYHEDRA

The purpose of this article is to survey the polyhedral combinatorics-based results obtained, over the last decade, for the survivable network design problem where the survivability requirements are modeled using connectivity type vectors. Therefore, we now start with formulating the SNDP as an integer linear program.

For the sake of clarity, we recall the statement of the problems. Given an undirected graph $G = (V, E)$, a connectivity type vector $r \in \mathbb{Z}_{+}^{V}$ and an edge cost function $c \in \mathbb{R}_{+}^{E}$, the node-survivable (respectively, link-survivable) network design problem consists of finding a subgraph $H$ of $G$ such that for any pair of distinct nodes $s, t \in V$, $H$ contains at least $\min\{r(s), r(t)\}$ node-disjoint (respectively edge-disjoint) $st$-paths.

An important result in graph theory, relating disjoint paths between two given nodes of a graph and the cuts separating these two nodes, is the following, which is known as Menger's Theorem [101].

**Theorem 2 [101].** *Let $G = (V, E)$ be a graph with $s$, $t$ two distinct nodes of $V$. Then*

*i) the maximum number of edge-disjoint st-paths is equal to the minimum size of an st-cut, and*
*ii) if s and t are nonadjacent, the maximum number of node-disjoint st-paths is equal to the minimum size of a node cutset disconnecting s and t.*

From Theorem 2, it follows that the link-survivable network design problem is equivalent to the following integer linear program

$$\text{minimize} \sum_{e\in E} c(e)x(e)$$

subject to

$$x(e) \geq 0 \qquad \text{for all } e \in E, \tag{2}$$

$$x(e) \leq 1 \qquad \text{for all } e \in E, \tag{3}$$

$$x(\delta(W)) \geq \text{con}(W) \qquad \text{for all } W \subseteq V, \emptyset \neq W \neq V, \tag{4}$$

$$x(e) \in \{0, 1\} \qquad \text{for all } e \in E. \tag{5}$$

Here, for all $W \subseteq V, \emptyset \neq W \neq V$, $\text{con}(W) = \min\{r(W), r(V \setminus W)\}$ where $r(W) = \max\{r(u) \mid u \in W\}$. Inequalities (2) and (3) are called *trivial inequalities* and inequalities (4) are called *cut inequalities*.

It is not hard to see that the following inequalities are also satisfied by any solution to the node-survivable network design problem

$$\begin{aligned} x(\delta_{G\setminus U}(W)) &\geq r(s,t) - |U| \\ &\text{for all } s, t \in V,\ s \neq t, \text{ and} \\ &\text{for all } \emptyset \neq U \subseteq V \setminus \{s,t\} \text{ with } |U| < r(s,t), \\ &\text{for all } W \subseteq V \setminus U \text{ with } s \in W,\ t \in V \setminus W. \end{aligned} \tag{6}$$

Inequalities (6) are called *node cutset inequalities*. By adding these inequalities to the above integer linear program and using again Menger's theorem, we obtain an integer linear programming formulation for the NSNDP.

We remark that an equivalent form of inequalities (6) was considered in Grötschel et al. [76]. In fact, their node cutset inequalities only differ from (6) by their right-hand side, which is

$$\text{con}_{G\setminus U}(W) - |U|,$$

for $U \subseteq V, \emptyset \neq U \neq V$ and $|U| < \text{con}_{G\setminus U}(W)$. Since

$$\begin{aligned} \text{con}(W) &= \min\{r(W), r(V \setminus W)\} \\ &= \min\{\max\{r(s) \mid s \in W\}, \max\{r(t) \mid t \in V \setminus W\}\} \\ &= \max\{\min\{r(s,t) \mid s \in W,\ t \in V \setminus W\}\}, \end{aligned}$$

the equivalence is obvious. We decided to write the node cutset inequalities as in (6) to make the relation between Menger's theorem and the integer linear programming formulation for the NSNDP more straightforward.

The so-called polyhedral approach [110] has been successfully applied for many well-known NP-hard problems such as the traveling salesman problem and the max-cut problem. This approach, based on the description of the convex hull of the solutions of the problem, consists of reducing the problem to a sequence of linear programming problems by successively adding valid inequalities. More precisely, we start by considering a "selected" linear relaxation of the problem given by few inequalities. If the optimal solution of this relaxation, say $x_1$, is feasible, then it is optimal for the problem. Otherwise, we generate one or more valid inequalities that are violated by $x_1$, and add them to the linear relaxation. If the optimal solution of the new linear program is feasible, then we are done. Otherwise, we generate new violated inequalities, and so on. Unfortunately, this process does not guarantee any feasible optimal solution. If the last solution we thus obtained is not feasible, we use some branch-and-bound techniques combined with the inequality generation process until an integer optimal solution is obtained. Such an approach clearly depends on the search for an inequality system determining (or approximating) the polytopes associated with the solutions of the node-survivable network design problem and the link-survivable network design problem.

Let us denote by NSNDP($G, r$) [respectively, LSNDP($G$,$r$)] the convex hull of the solutions of (2)–(5) [respectively (2)–(6)]. The polytopes NSNDP($G$,$r$) and LSNDP($G$,$r$) are respectively called the *node-survivable network polytope* and the *link-survivable network polytope* of $G$. The node-survivable network design problem is then equivalent to the linear program

$$\text{minimize} \left\{ \sum_{e \in E} c(e)x(e) \mid x \in \text{NSNDP}(G, r) \right\},$$

while the link-survivable network design problem is the same as solving

$$\text{minimize} \left\{ \sum_{e \in E} c(e)x(e) \mid x \in \text{LSNDP}(G, r) \right\}.$$

The polytopes LSNDP($G, r$) and NSNDP($G, r$) have been extensively investigated in the past years. In [75], Grötschel and Monma thus established the dimension of both polytopes, and they also characterized which of the trivial inequalities (2) and (3) are facet-defining. In [76], Grötschel et al. considered the low connectivity case, that is, where $r \in \{0, 1, 2\}^V$, and they described when the cut inequalities (4) define facets for NSNDP($G, r$) and LSNDP($G, r$). Furthermore, they gave necessary conditions and sufficient conditions for the node cutset inequalities (6) to be facet-defining.

For a class of inequalities, the *separation problem* consists, given a vector $\bar{x}$, in finding a violated inequality in the class or proving that there is none. This problem is obviously one of the key ingredients in a polyhedral approach as we previously described. In fact, a fundamental result, based on the ellipsoid method and due to Grötschel et al. [74], states a polynomial equivalence between separation and optimization. More precisely, we can solve one of the two problems in polynomial time if and only if we can also solve the other problem in polynomial time. To solve with linear programming techniques the linear relaxations of both problems NSNDP and LSNDP, that is, when the constraints (5) are dropped, we cannot just list all inequalities (4) and (6) for the NSNDP, (4) for the LSNDP, because of their exponential number. However, the separation problems for both classes of inequalities (4) and (6) are polynomially solvable using polynomial-time maximum flow algorithms (e.g., the preflow-push algorithm of Goldberg and Tarjan [64] running in $O(n^3)$ time). Therefore, from the Grötschel, et al. result, the linear relaxations of both problems NSNDP and LSNDP can be solved in polynomial time.

Because the problems NSNDP and LSNDP are NP-hard, it is unlikely to obtain complete linear descriptions of the polytopes NSNDP($G, r$) and LSNDP($G, r$) on general graphs. The basic trivial, cut and node cutset inequalities, the latter only for NSNDP($G, r$), suffice to completely describe these two polytopes only in some special classes of graphs (see Section 6 for some examples). However, as it will turn out, partial descriptions of those polytopes may be sufficient to solve the problems to optimality. To this aim, further classes of valid inequalities are needed to get tighter linear relaxations. The following section presents some of these classes of inequalities and addresses their associated separation problem. We mention that those inequalities will be given for LSNDP($G, r$), because their validity for NSNDP($G, r$) comes directly from NSNDP($G, r$) $\subseteq$ LSNDP($G, r$). Moreover, we will restrict ourselves to the low connectivity case (i.e., $r \in \{0, 1, 2\}^V$); similar constraints can be easily extended to the general case (i.e., $r \in \mathbb{Z}_+^V$).

## 5. VALID INEQUALITIES AND THEIR SEPARATION PROBLEM

Throughout this section, we consider a graph $G = (V, E)$ and a connectivity type vector $r \in \{0, 1, 2\}^V$. The inequalities presented in this section have partitions of $V$ as underlying structures. However, some other classes of inequalities, based on more complicated structures, were introduced as well [112]. For instance, the widely studied traveling salesman problem is closely related to the 2-connected subgraph problem as mentioned in Section 2. Thus, Grötschel et al. [76] (see also Stoer [112]) extended to the polytopes NSNDP($G, r$) and LNSDP($G, r$) the comb inequalities, which are valid for the polytope associated with the solutions of the TSP. Boyd and Hao [20] introduced the same class of inequalities for the 2-edge connected network polytope, and gave necessary and sufficient conditions for these inequalities to be facet-defining.

### 5.1. *Multicut Inequalities*

Let $\{V_1, \ldots, V_p\}$ be a partition of $V$. If $\text{con}(V_i) = 1$ for $i = 1, \ldots, p$, the graph obtained from any solution to the LSNDP by contracting every subgraph $G(V_i)$, $i = 1, \ldots, p$, must then be connected. Therefore, the following inequality is valid for the polytope LSNDP($G, r$).

$$x(\delta(V_1, \ldots, V_p)) \geq p - 1 \quad \text{for all partition } \{V_1, \ldots, V_p\} \text{ such that } \text{con}(V_i) = 1, \text{ for } i = 1, \ldots, p. \quad (7)$$

Inequalities of type (7) are called *multicut inequalities*. In [75], Grötschel and Monma (see also [76]) showed that inequalities (7), together with trivial inequalities (2) and (3), suffice to describe the polytope LSNDP($G, r$) when $r(i) = 1$ for all $i \in V$.

Moreover, Nash-Williams [104] (see also Tutte [117]) proved that those inequalities (7), together with nonnegativity ones (2), characterize the dominant of the polytope LSNDP($G, r$) in this case of unit connectivity type vectors.

Cunningham [34] showed that, if $r(v) = 1$ for all $v \in V$, the separation problem associated with inequalities (7) can be reduced to $|E|$ minimum cut problems, and can then be solved in polynomial time. In [12], Barahona reduced the separation problem for those inequalities to a sequence of $|V|$ minimum cut problems, and then derived an $O(n^4)$ algorithm. Moreover, both algorithms provide the most violated inequality if there is any.

For the general case where $r \in \mathbb{Z}^V$ and there exists at least a node $u \in V$ such that $r(u) = 0$, Grötschel et al. [77] showed that the separation problem for inequalities (7) is NP-hard. Furthermore, if $r(u) \geq 1$ for all $u \in V$, as mentioned by Kerivin and Mahjoub [85], inequalities (7) can then be separated in polynomial time by applying the Cunningham or Barahona algorithms on the graph obtained from $G$ by contracting the set of nodes $\{u \in V \mid r(u) > 1\}$.

### 5.2. Partition Inequalities

In [76], Grötschel et al. introduced a class of valid inequalities for LSNDP$(G, r)$, called partition inequalities, that generalizes the cut inequalities (4). These inequalities are as follows. Let $\{V_1, \ldots, V_p\}$, $p \geq 3$, be a partition of $V$ such that $1 \leq \mathrm{con}(V_i) \leq 2$ for $i = 1, \ldots, p$. Denote $I_2 = \{i \mid \mathrm{con}(V_i) = 2, i = 1, \ldots, p\}$. The *partition inequality* induced by $\{V_1, \ldots, V_p\}$ is given by

$$x(\delta(V_1, \ldots, V_p)) \geq \begin{cases} p-1 & \text{if } I_2 = \emptyset, \\ p & \text{otherwise.} \end{cases} \tag{8}$$

Obviously, if all connectivity types are equal to 2, a partition inequality (8) is implied by sum of the cut constraints $x(\delta(V_i)) \geq 2$. [We remark that considering the case where $p = 2$ gives a cut inequality (4).]

The separation problem for the partition inequalities (8) is NP-hard in general [77]. Grötschel et al. [77] showed that, even in the restricted case where $r \in \{0, 1\}^V$, the separation problem remains NP-hard. If $r \in \{1, 2\}^V$, Kerivin and Mahjoub [85] proved that the separation problem associated with

$$x(\delta(V_1, \ldots, V_p)) \geq p \qquad \text{if } I_2 \neq \emptyset, \tag{9}$$

where $\{V_1, \ldots, V_p\}$ is a partition of $V$, reduces to minimizing a submodular function, and therefore, it can be solved in polynomial time (see Schrijver [110] for details on submodular functions). Recently, Barahona and Kerivin [13] devised a pure combinatorial algorithm, based on the submodular intersection problem, for separating inequalities (9). They showed that this problem reduces to a sequence of submodular flow problems, each of them having its complexity dominated by the one of solving $O(n^3)$ minimum cut problems. Combined with the algorithm of Section 5.1 for separating inequalities (8) with a right-hand side equal to $p - 1$, Barahona and Kerivin obtained an $O(n^7)$ algorithm for the separation problem associated with the partition inequalities (8).

### 5.3. $F$-Partition Inequalities

Suppose the connectivity type vector $r$ is such that $r(u) = 2$ for all $u \in V$. A class of valid inequalities for the polytope LSNDP$(G, r)$ in this case was introduced by Mahjoub [96] as follows. Consider a partition $\{V_1, \ldots, V_p\}$ of $V$ and let $F \subseteq \delta(V_1)$ with $|F|$ odd. By adding the inequalities

$$\begin{aligned} x(\delta(V_i)) &\geq 2 && \text{for } i = 2, \ldots, p, \\ -x(e) &\geq -1 && \text{for } e \in F, \\ x(e) &\geq 0 && \text{for } e \in \delta(V_1) \setminus F, \end{aligned}$$

we obtain

$$2x(\Delta) \geq 2(p-1) - |F|,$$

where $\Delta = \delta(V_1, \ldots, V_p) \setminus F$. Dividing by 2 and rounding up the right-hand side lead to

$$x(\Delta) \geq p - \left\lceil \frac{|F|}{2} \right\rceil. \tag{10}$$

Inequalities (10) are called *$F$-partition inequalities*. Note that if $|F|$ is even, the corresponding inequality (11) is then implied by inequalities (2), (3), and (4). It is straightforward to see that inequalities (10) remain valid for LSNDP$(G, r)$ when $r \in \{0, 1, 2\}^V$ and $\mathrm{con}(V_i) = 2$ for $i = 1, \ldots, p$.

The partition and $F$-partition inequalities are special cases of more general classes of inequalities given by Grötschel et al. [76] for LSNDP$(G, r)$. In [76], the authors also gave necessary conditions and sufficient conditions for these inequalities to be facet-defining. Furthermore, Kerivin et al. [87] considered a subclass of $F$-partition inequalities, called generalized odd-wheel inequalities, to give sufficient conditions for inequalities (10) to be facet-defining. They also introduced an extension of inequalities (10) to the case where the inducing partition $\{V_1, \ldots, V_p\}$ is such that $\mathrm{con}(V_i) \in \{1, 2\}$ for $i = 1, \ldots, p$ (see Section 7).

The separation problem for the $F$-partition inequalities is still an open question. However, if the sets $V_i$ of partitions are singletons, the corresponding $F$-partition inequalities are then blossom inequalities for $b$-matching, which can be separated in polynomial time with the algorithm of Padberg and Rao [106]. Moreover, when the edge subset $F$ is fixed, as pointed out by Baïou et al. [7], the separation problem for inequalities (10) can be solved in polynomial time. In fact, one can delete the set of edges $F$ from $G$ and consider the resulting graph $G' = (V', E')$, say. An $F$-partition in $G$ can be written in $G'$ as

$$x(\delta_{G'}(V_1, \ldots, V_p)) \geq p - \left\lceil \frac{|F|}{2} \right\rceil, \tag{11}$$

where $V_1$ contains exactly one node of each edge of $F$. There are $2^{|F|}$ possibilities to assign nodes of $F$ to $V_1$. For each one we can contract the nodes of $F$ in $V_1$ and solve the separation problem for inequalities (11). As Cunningham's algorithm and Barahona's algorithm provide a most violated multicut inequality, if there is any, this can then be done in polynomial time. As it will be shown in the sequel, $F$-partition inequalities play a central role for solving LSNDP and NSNDP, in the low connectivity case, within the framework of a cutting plane algorithm.

### 5.4. General Partition Inequalities

In [7], Baïou et al. studied a class of inequalities generalizing the multicut inequalities (7). Given a partition $\{V_1, \ldots, V_p\}$ of $V$ and two fixed scalars $a$ and $b$, they are of the form

$$x(\delta(V_1, \ldots, V_p)) \geq ap + b. \tag{12}$$

Inequalities (12) arise as valid inequalities for many variants of the survivable network design problem. For instance, we remark that the multicut inequalities (7) correspond to inequalities (12) where $a = 1$ and $b = -1$. Baïou et al. called these inequalities partition inequalities, however, to avoid confusion, we will here refer to inequalities (12) as *generalized partition inequalities*. Baïou et al. [7] showed that the separation problem for inequalities (12) can be reduced to minimizing a submodular function, and can then be solved in polynomial time.

Consider now the $k$-edge connected network problem, that is, the LSNDP where $r(u) = k$ for all $u \in V$. Grötschel et al. [76] introduced the following inequalities

$$x(\delta(V_1, \ldots, V_p)) \geq \left\lceil \frac{kp}{2} \right\rceil, \tag{13}$$

where $\{V_1, \ldots, V_p\}$ is a partition of $V$. Inequalities (13) are clearly redundant with respect to the cut inequalities (4) if $kp$ is even. To have an approximate separation routine, instead of separating inequalities (13), one can separate the inequalities

$$x(\delta(V_1, \ldots, V_p)) \geq \frac{kp}{2},$$

which are nothing but inequalities (12) where $a = \frac{k}{2}$ and $b = 0$.

Let $Z \subset V$ be a node set with $|Z| = t \leq k - 1$ and $\{V_1, \ldots, V_p\}$ a partition of $V \setminus Z$. For the $k$-node connected network problem (i.e., NSNDP where $r(u) = k$ for all $u \in V$), Grötschel and Monma [75] introduced the *node partition inequalities*, which are as follows

$$x(\delta_{G\setminus Z}(V_1, \ldots, V_p)) \geq \begin{cases} p - 1 & \text{if } k - t = 1, \\ \left\lceil \frac{p(k-t)}{2} \right\rceil & \text{if } k - t \geq 2. \end{cases} \tag{14}$$

Grötschel and Monma [75] also gave necessary and sufficient conditions for inequalities (14) to be facet-defining. If $k - t = 1$, inequalities (14) are then multicut inequalities, and therefore they can be separated in polynomial time. If $k - t$ is positive and even, they are nothing but inequalities (12) and their separation is also polynomially solvable. As we mentioned for inequalities (13), one can use Baïou et al.'s algorithm for separating inequalities (12) to approximate the separation problem for inequalities (14) where $k-t$ is positive and odd.

## 6. THE POLYTOPES LSNDP($G, r$) AND NSNDP($G, r$) ON SPECIAL GRAPHS

In this section, we discuss the polytopes NSNDP($G$,$r$) and LSNDP($G$,$r$) in some special classes of graphs. (See Subsection 3.1 for the definitions of the considered graphs.) In fact, these polytopes are known in many classes of graphs, and the inequality systems describing them are separable in polynomial time. Therefore, by the ellipsoid method [74], one gets polynomial-time cutting plane algorithms for solving the underlying optimization problems.

In [96], Mahjoub showed that when $G$ is series-parallel and $r(u) = 2$ for all $u \in V$, the polytope LSNDP($G, r$) is given by the trivial inequalities (2) and (3), and the cut inequalities (4). This linear description was generalized to the case where $r \in \{0, 2\}^V$ by Baïou and Mahjoub [8] as well as to the case where $r \in \{0, k\}^V$ and $k$ is even by Didi Biha and Mahjoub [47]. Recently, Kerivin and Mahjoub [86] extended those results to the more general case where the connectivity types are all even.

**Theorem 3 [86].** *If $G = (V, E)$ is series-parallel and $r(u)$ is even for all $u \in V$, the polytope LSNDP(G,r) is then completely described by the trivial inequalities* (2) *and* (3) *together with the cut inequalities* (4).

To our knowledge, the only linear description of LSNDP ($G$,$r$) where even and odd connectivity types are mixed is due to Didi Biha et al. [45]. For $r \in \{1, 2\}^V$ and in a subclass of series-parallel graphs containing all the outerplanar graphs, they showed that the link-survivable network polytope is completely described by the trivial inequalities (2) and (3), the cut inequalities (4) and the partition inequalities (8).

For connectivity type vectors $r$ such that $r(u) = 2$ for all $u \in V$, Barahona and Mahjoub [14] studied the polytopes NSNDP($G, r$) and LSNDP($G, r$) in the graphs that can be decomposed by 3-edge cutsets. (A *3-edge cutset* is a cut that consists of exactly three edges.) They showed that if a graph $G$ decomposes into $G_1$ and $G_2$ by a 3-edge cutset, the system describing LSNDP($G, r$) is then the union of both systems describing LSNDP($G_1, r$) and LSNDP($G_2, r$). As a consequence, they obtained that inequalities (10) together with the trivial and cut inequalities completely describe the link-survivable network polytope on Halin graphs for this case of connectivity type vectors. They also presented similar results for the polytope NSNDP($G, r$). Some extensions of this work to the case where $r \in \{0, 2\}^V$ were studied in [98].

In some practical situations, one may need to use more than one link between two given nodes of a link-survivable network. This case can be seen as a relaxation of the link-survivable network problem, and is usually easier to handle. Let $P(G, r)$ be the *dominant* of LSNDP($G, r$), that is, $P(G, r) = \text{LSNDP}(G, r) + \mathbb{R}^E_+$. The polyhedron $P(G, r)$ is nothing but the convex hull of the solutions of the relaxed LSNDP when multiple copies of edges are allowed. In [22], Chopra studied $P(G, r)$ when $r(u) = k$ for all $u \in V$ and $G$ is an outerplanar graph. For this case with $k$ odd, he showed that the following inequalities are valid for the polyhedron $P(G, r)$

$$x(\delta(V_1, \ldots, V_p)) \geq \left\lceil \frac{k}{2} \right\rceil p - 1 \quad \text{for all partitions } \{V_1, \ldots, V_p\} \text{ of } V. \tag{15}$$

Moreover, he proved the following.

**Theorem 4 [22].** *If $G = (V, E)$ is outerplanar, $r(u) = k$ for all $u \in V$ with $k$ odd, the polyhedron $P(G, r)$ is then given by the nonnegativity inequalities* (2) *and inequalities* (15).

The polyhedron $P(G, r)$ was previously studied by Cornuéjols et al. [30]. They showed that on series-parallel graphs and for $r(u) = k$ for all $u \in V$ with $k$ even, the polyhedron $P(G, r)$ is completely described by the nonnegativity inequalities (2) and the cut inequalities (4). In [6], Baïou showed that this result also holds if $r \in \{0, 2\}^V$. For this class of graphs, Didi Biha and Mahjoub [46] (see also Didi Biha [43]) proved that inequalities (15) remain valid for the link-survivable network problem where $r(u) = k$ for all $u \in V$ with $k$ odd. They also showed that inequalities (15) together with the nonnegativity inequalities (2) completely describe the polyhedron $P(G, r)$ in that case. As a consequence, they obtained that Theorem 4 also holds on series-parallel graphs, as conjectured by Chopra [22]. This conjecture was also proved independently by Chopra and Stoer [25]. We remark that inequalities (15) are nothing but a particular case of generalized partition inequalities (12). Therefore, a direct consequence of the result of Baïou et al. [7] (see Subsection 5.4) is that inequalities (15) can be separated in polynomial time. (Recall that those inequalities are valid for the LSNDP only if the graph induced by the partition is series-parallel.)

As we mentioned in Section 2, the polytope LSNDP$(G, r)$ where $r \in \{0, 1\}^V$ is closely related to the Steiner tree polytope STP$(G, r)$, the extreme points of which are the incidence vectors of the Steiner trees of $G$. Over the last 2 decades, extensive research has been conducted on STP$(G, r)$ [23, 24, 44, 59, 99]. In [23, 24], Chopra and Rao described several classes of facet-defining inequalities for the dominant of the Steiner tree polytope in both directed and undirected cases. Didi Biha et al. [44] studied further facet-defining inequalities that generalize those introduced in [23, 24]. They also gave some linear descriptions of STP$(G, r)$ in some nontrivial subclasses of series-parallel graphs. In [99], Margot et al. considered an extended formulation of the Steiner tree problem, and they showed that it leads to a complete linear description of the associated polytope on 2-trees (i.e., maximal series-parallel graphs). Goemans [59] discussed another extended formulation of the Steiner tree problem and he characterized the associated polytope when the underlying graph is series-parallel. Moreover, he also described some classes of facet-defining inequalities for the Steiner tree polytope.

The node-survivable network polytope has also been investigated in some particular classes of graphs. In [14], Barahona and Mahjoub gave a complete description of the polytope NSNDP$(G, r)$ on Halin graphs when $r(u) = 2$ for all $u \in V$. Coullard et al. [32, 33] studied the Steiner 2-node connected subgraph problem, that is, the NSDNP where $r \in \{0, 2\}^V$. In [32], they gave a linear time algorithm for the Steiner 2-node connected subgraph problem on Halin graphs and on graphs noncontractible to $W_4$, the latter being the graphs that do not reduce to $W_4$ (i.e., the wheel on five nodes) by means of deletions and contractions of edges. They also described, in [33], the dominant of the polytope LNSDP$(G, r)$ where $G$ is a graph noncontractible to $W_4$ and $r \in \{0, 2\}^V$.

## 7. CRITICAL EXTREME POINTS

It is well known that the linear relaxation of a combinatorial optimization problem usually provides a near-optimal solution. To improve this solution, one has to add valid inequalities that are violated by fractional solutions. Many of these solutions may be extreme points of the linear relaxation, and therefore, characterizing the extreme points, among the ones of the linear relaxation, which may be separated in polynomial time, would be of great interest for solving the whole optimization problem. This question was first studied by Fonlupt and Mahjoub [49, 50] for the 2-edge connected network polytope, that is, the polytope LSNDP where $r(u) = 2$ for all $u \in V$. They introduced the concept of critical extreme points of the linear relaxation of the link-survivable network polytope. In this section, we discuss these extreme points.

Consider a graph $G = (V, E)$ and suppose $r(u) = 2$ for all $u \in V$. We denote by $P(G)$ the polytope given by the trivial inequalities (2) and (3) and the inequalities

$$x(\delta(W)) \geq 2 \qquad \text{for all } W \subset V,\ W \neq \emptyset. \tag{16}$$

We observe that inequalities (16) correspond to the cut inequalities (4) in the case where $r(u) = 2$ for all $u \in V$. Moreover, we also point out that the polytope $P(G)$ is the linear relaxation of LSNDP$(G, r)$ in this case.

Let $\bar{x}$ be a noninteger extreme point of $P(G)$. Let $\bar{x}'$ be a solution obtained by replacing some (but at least one) noninteger components of $\bar{x}$ by 0 or 1 (and keeping all the other components of $\bar{x}$ unchanged). If $\bar{x}'$ is a point of $P(G)$, then $\bar{x}'$ can be written as a strict convex combination of extreme points of $P(G)$. If $\bar{y}$ is such an extreme point, then $\bar{y}$ is said to be *dominated* by $\bar{x}$, and we write $\bar{x} \succ \bar{y}$. Note that an extreme point of $P(G)$ may dominate more than one extreme point of $P(G)$. Notice also that, if $\bar{x}$ dominates $\bar{y}$, that is, $\bar{x} \succ \bar{y}$, we then have

$$\begin{aligned}
&\{e \in E \mid 0 < \bar{y}(e) < 1\} \subset \{e \in E \mid 0 < \bar{x}(e) < 1\}\\
&\{e \in E \mid \bar{x}(e) = 0\} \subseteq \{e \in E \mid \bar{y}(e) = 0\}, \text{ and}\\
&\{e \in E \mid \bar{x}(e) = 1\} \subseteq \{e \in E \mid \bar{y}(e) = 1\}.
\end{aligned}$$

The relation $\succ$ defines a partial ordering on the extreme points of $P(G)$. The minimal elements of this ordering (i.e., the extreme points $x$ for which there is no extreme point $y$ such that $x \succ y$) correspond to the integer extreme points of $P(G)$. The minimal extreme points of $P(G)$ are called extreme points of *rank* 0. An extreme point $x$ of $P(G)$ is said to be of *rank* $k$, for a fixed $k$, if $x$ only dominates extreme points of rank less than or equal to $k - 1$ and if it dominates at least one extreme point of rank $k - 1$. We notice that if $\bar{x}$ is an extreme point of $P(G)$ of rank 1 and if we replace one fractional component of $\bar{x}$ by 1, keeping unchanged the other components, we obtain a feasible point $\bar{x}'$ of $P(G)$, which can be written as a convex combination of integer extreme points of $P(G)$. We also observe that the extreme points of $P(G)$ may have rank at most $|E|$.

Fonlupt and Mahjoub [49, 50] introduced the following reduction operations with respect to a solution $\bar{x}$ of $P(G)$.

- $\theta_1$: Delete an edge $e$ with $\bar{x}(e) = 0$.
- $\theta_2$: Contract an edge $e$ having one of its endnodes of degree 2.
- $\theta_3$: Contract a node subset $W$ such that $G(W)$ is 2-edge connected and $\bar{x}(e) = 1$ for all $e \in E(W)$.

Starting from a graph $G$ and a point $\bar{x}$ of $P(G)$, let $G'$ be a reduced graph and $\bar{x}'$ be a point of $P(G')$, both obtained by applying operations $\theta_1, \theta_2, \theta_3$. It is not hard to see that $\bar{x}$ is an extreme point of $P(G)$ if and only if $\bar{x}'$ is an extreme point of $P(G')$. Moreover, we have

**Lemma 5 [49, 50].** *$\bar{x}$ is an extreme point of $P(G)$ of rank 1 if and only if $\bar{x}'$ is an extreme point of $P(G')$ of rank 1.*

An extreme point of $P(G)$ is said to be *critical* [49, 50] if it is of rank 1 and if none of the operations $\theta_1, \theta_2, \theta_3$ can be applied to it. By Lemma 5, the characterization of the extreme points of rank 1 reduces to those of the critical extreme points of $P(G)$. In [49, 50], Mahjoub and Fonlupt gave the following necessary conditions for a fractional extreme point of $P(G)$ to be critical.

**Theorem 6 [49, 50].** *Let $G = (V, E)$ be a 2-edge connected graph and $\bar{x}$ a fractional extreme point of $P(G)$. If $\bar{x}$ is a critical extreme point of $P(G)$, then the following hold.*

*(i)* $V = V^1 \cup V^2$ *with* $V^1 \cap V^2 = \emptyset$,
$E = E^1 \cup E^2$ *with* $E^1 \cap E^2 = \emptyset$,
$(V^1, E^1)$ *is an odd cycle,*
$(V^1 \cup V^2, E^2)$ *is a forest whose set of leaves is* $V^1$ *and such that all the nodes in* $V^1$ *have degree 3,*
*(ii)* $\bar{x}(e) = \frac{1}{2}$ *for* $e \in E^1$,
$\bar{x}(e) = 1$ *for all* $e \in E^2$, *and*
*(iii)* $\bar{x}(\delta(W)) > 2$ *for all cuts* $\delta(W)$ *such that* $|W| \geq 2$ *and* $|\overline{W}| \geq 2$.

**Remark 2.1.** By (ii) and (iii) of Theorem 6, if $G$ supports a critical extreme point, then $G$ is 3-edge connected, and $|\delta(S)| \geq 4$ for every cut $\delta(S)$ such that $|S| \geq 2$ and $|\bar{S}| \geq 2$.

Theorem 6 has some interesting algorithmic and polyhedral consequences. We first note that operations $\theta_1, \theta_2, \theta_3$ can be performed in polynomial time and in any order. Consider now a graph $G = (V, E)$ and a critical extreme point $\bar{x}$. From Theorem 6, it follows that there exists an odd cycle $C$ of $G$ such that $\bar{x}(e) = \frac{1}{2}$ for $e \in C$ and $\bar{x}(e) = 1$ for $e \in E \setminus C$. Moreover, $E \setminus C$ induces a forest whose leaves are precisely the nodes of $V(C)$. So the inequality

$$\sum_{e \in C} x(e) \geq \frac{|C| + 1}{2}, \tag{17}$$

which is valid for the 2-edge connected network problem, is violated by $\bar{x}$. Actually, constraint (17) is an $F$-partition inequality (10) where $F$ is the set of leaves of the forest. Thus, by the remark above we have the following.

**Theorem 7 [49, 50].** *Critical extreme points can be separated from the 2-edge connected network polytope in polynomial time.*

Kerivin et al. [87] showed that inequality (17) is nothing but a special case of a more general class of facet-defining inequalities for the 2-edge connected network polytope. Consequently, by Theorem 7, critical extreme points may be separated by $F$-partition facets.

The concept of critical extreme points has also been studied by Mahjoub and Nocq [97] for the 2-node connected network polytope (i.e, NSNDP$(G, r)$ where $r(u) = 2$ for all $u \in V$) as well as by Kerivin et al. [87] for the (1,2)-link-survivable network polytope (i.e., LSNDP$(G, r)$ where $r \in \{1, 2\}^V$). The following inequalities

$$x(\delta_{G \setminus v}(W)) \geq 1 \qquad \text{for all } v \in V, W \subset V \setminus \{v\}, W \neq \emptyset. \tag{18}$$

are valid for the 2-node connected network polytope. We observe that these inequalities are a special case of the node partition inequalities (14). In [97], Mahjoub and Nocq studied the polytope $Q(G)$ given by inequalities (2), (3), (16), and (18). This polytope is nothing but the linear relaxation of the 2-node connected network polytope. They extended the concept of extreme points of rank 1 and critical extreme points to the polytope $Q(G)$. They also gave necessary and sufficient conditions for an extreme point of $Q(G)$ to be critical. In particular, they introduced the following operations defined with respect to a point $\bar{x}$ of $Q(G)$.

- $\theta'_1$: Replace a set of parallel edges by only one edge.
- $\theta'_2$: Contract $W \subset V$ such that $\bar{x}(e) = 1$ for all $e \in E(W)$ and $|\delta(W)| \leq 3$.

Moreover they proved the following.

**Lemma 8 [97].** *Let $\bar{x}$ be an extreme point of $Q(G)$ with $\bar{x}'$ and $G'$ being the solution and the graph obtained from $\bar{x}$ and $G$ by repeated applications of the operations $\theta_1$, $\theta_2$, $\theta_3$, $\theta'_1$ and $\theta'_2$. Then $\bar{x}$ is an extreme point of $Q(G)$ of rank 1 if and only if $\bar{x}'$ is an extreme point of $Q(G')$ of rank 1.*

We now look at the case where $r \in \{1, 2\}^V$. The $F$-partition inequalities (10) can straightforwardly be extended to the case $r \in \{1, 2\}^V$ as follows

$$x(\Delta) \geq p - 1 - \left\lfloor \frac{p_1 + |F|}{2} \right\rfloor, \tag{19}$$

where $p_1 = |\{i \mid \mathrm{con}(V_i) = 1,\ i = 2, \ldots, p\}|$. We remark here that $|F|$ is not necessarily odd. In fact, inequalities (19) are dominated by the cut and trivial inequalities if and only if $p_1$ and $|F|$ have the same parity.

Let $R(G, r)$ be the polytope described by the trivial inequalities (2) and (3), the cut inequalities (4) and the partition inequalities (8). The interest in considering the partition inequalities (8) for $R(G, r)$ is because they can be separated in polynomial time as proved in [85] (see also Section 8). Given a solution $\bar{x}$ of $R(G, r)$, the following operations, described in [87] and given with respect to $\bar{x}$, extend in a straightforward way the operation $\theta_2$, introduced above, to the case where $r \in \{1, 2\}^V$.

$\theta_1''$: Contract an edge $uv$ such that $\bar{x}(uv) = 1$, $r(u) = 1$ and $\bar{x}(\delta(u)) \leq 2$.

$\theta_2''$: Contract an edge $uv$ such that $r(u) = 2$, $|\delta(u)| = \{uv, uw\}$ and $r(w) = 2$.

Note that these reduction operations, as well as $\theta_1', \theta_2'$, can also be realized in polynomial time. We also notice that operation $\theta_3$, previously given for the case where $r(u) = 2$ for all $u \in v$, can be extended to the (1,2)-link-survivable network problem by considering node sets $W \subset V$ with $r(u) = 2$ for all $u \in W$.

With a graph obtained from $G$ by contracting an edge $e = uv \in E$, we associate the connectivity type vector $r_e \in \{1, 2\}^{|V|-1}$ such that $r_e(w) = \text{con}(\{u, v\})$ and $r_e(u) = r(u)$ if $u \in V \setminus \{u, v\}$, where $w$ is the node that arises from the contraction of $e$. Let $G' = (V', E')$ be a graph obtained by repeated applications of operations $\theta_1, \theta_2, \theta_3, \theta_1'', \theta_2''$. Denote by $r' \in \{1, 2\}^{V'}$ the connectivity type vector corresponding to the graph $G'$ and by $\bar{x}'$ the restriction of $\bar{x}$ on $E'$. If $\bar{x}$ is an extreme point of $R(G, r)$, then $\bar{x}'$ is also an extreme point of $R(G', r')$. Moreover, we have the following.

**Lemma 9 [87].** *(i) If $a'x \geq \alpha'$ is a valid inequality of the (1,2)-link survivable network polytope on $G'$ of type either* (4), (8) *or* (19)*, then the inequality $ax \geq \alpha$ where $a(e) = a'(e)$ if $e \in E'$, $a(e) = 1$ if e has its endnodes in different classes of the partition and $\alpha = \alpha'$, is valid for the (1,2)-link-survivable network polytope on G. Moreover, if $a'x \geq \alpha'$ is violated by $\bar{x}'$, then $ax \geq \alpha$ is also violated by $\bar{x}$.*

*(ii) If $ax \geq \alpha$ is a valid inequality of the (1,2)-link-survivable network polytope on G of type* (4) *[respectively* (8)*] [respectively* (19)*], which is violated by $\bar{x}$, then there is an inequality, valid for the (1,2)-link-survivable network polytope on $G'$ of type* (4) *[respectively* (8)*] [respectively* (19)*], which is violated by $\bar{x}'$.*

Lemma 9 shows that looking for inequalities of type (4), (8), or (19), which are violated by $\bar{x}$, reduces to looking for such inequalities which are violated by $\bar{x}'$ on $G'$. We observe that this procedure can be applied for any solution of $R(G, r)$, and in consequence, it may permit us to separate fractional solutions, which are even not extreme points of $R(G, r)$. Moreover, if $r(u) = 2$ for all $u \in V$ and $\bar{x}$ is an extreme point of $P(G)$ of rank 1, then, as mentioned above, there is a $F$-partition that cuts off this solution and that can be found in polynomial time. In addition, this $F$-partition inequality may be facet-defining.

Lemma 9 also holds for the 2-node connected network polytope when we consider the operations $\theta_1, \theta_2, \theta_1', \theta_2'$ and the inequalities (16), (14), (10). Actually, in this case, the graph $G'$ is obtained by applications of the operations $\theta_1, \theta_2, \theta_1', \theta_2'$. And if there is one of those inequalities that is violated by $\bar{x}'$ in $G'$, then there is also one that is violated by $\bar{x}$ in $G$. Thus, as for the (1,2)-link-survivable network polytope, the separation of $\bar{x}$ by inequalities of type (16), (14), (10) in $G$ reduces to the separation of $\bar{x}'$ by these inequalities in $G'$.

Operations $\theta_1, \theta_2, \theta_3, \theta_1'', \theta_2''$ (respectively $\theta_1, \theta_2, \theta_3, \theta_1', \theta_2'$) have been used by Kerivin et al. [87] in a preprocessing phase of a cutting plane algorithm for the (1,2)-link-survivable network design problem (respectively, the 2-node connected network problem). As will be seen in Section 8, these operations are very effective for solving these problems.

## 8. A BRANCH-AND-CUT ALGORITHM

Branch-and-cut is the most successful paradigm for solving NP-hard combinatorial optimization problems to optimality. This approach has been used for devising efficient algorithms for the survivable network design problem, starting from the work of Grötschel et al. [78]. In fact, Grötschel et al. developed a branch-and-cut algorithm for solving the low-survivability case on real-world instances that have up to 108 nodes and sparse graphs induced by all the possible links. The theoretical results presented in the previous sections have some interesting algorithmic applications as shown hereafter by the description of a branch-and-cut algorithm devised by Kerivin et al. [87] for the (1,2)-link survivable network design problem and the 2-node connected network problem, denoted later by (1,2)LSNDP and 2NSNDP, respectively.

### *8.1. The Overall Algorithm*

To this aim, we consider a graph $G = (V, E)$, a connectivity type vector $r \in \{1, 2\}^V$ and a cost vector $c \in \mathbb{R}_+^E$ on the edges of $G$. We first describe the framework of the algorithm where we use the survivable network design problem to refer to either the (1,2)-link survivable network design problem or the 2-node connected network problem. The initial linear program is given by the so-called degree inequalities [i.e., cut inequalities (4) induced by single nodes] and the trivial inequalities (2) and (3) as follows

$$\begin{aligned} &\text{minimize } \sum_{e \in E} c(e)x(e) \\ &\text{subject to} \\ &x(\delta(u)) \geq r(u) \qquad \text{for all } u \in V, \\ &0 \leq x(e) \leq 1 \qquad \text{for all } e \in E. \end{aligned}$$

The optimal solution $\bar{x} \in \mathbb{R}^E$ of a relaxation of the survivable network design problem is feasible if $\bar{x}$ is an integer vector that satisfies all the cut inequalities (4), as well as the node cutset inequalities (6) for the 2NSNDP. Usually, the solution $\bar{x}$ is not feasible for the SNDP, and thus, at each iteration of the branch-and-cut algorithm, it is necessary to generate valid inequalities for the SNDP, which are violated by $\bar{x}$. These inequalities are picked from a pool formed by the cut inequalities (4), the partition inequalities (8) and the $F$-partition inequalities (19) for the (1,2)-link survivable network design problem. For the 2-node connected network problem, the node cutset inequalities (6), the node partition inequalities (14) and the $F$-partition inequalities (10) make up the pools. We remark that all these inequalities are global (i.e., valid throughout the branch-and-cut tree) and several inequalities may be added at each iteration. Moreover, the separation of those inequalities is performed according to the orders specified above, to apply the right separation routine to a class of inequalities.

The separation routines used in the branch-and-cut algorithm are either exact algorithms or heuristics depending on the associated class of inequalities. Frequently, these separation routines are based on maximum flow computations that can be done in polynomial time using the efficient preflow-push algorithm of Goldberg and Tarjan [64], which runs in $O(n^3)$ time. Let $G' = (V', E')$ be the graph obtained from $G$ and $\bar{x}$ by repeated applications of the operations $\theta_1$, $\theta_2$, $\theta_3$, $\theta_1''$ and $\theta_2''$ (respectively, $\theta_1$, $\theta_2$, $\theta_1'$ and $\theta_2'$) for the (1,2)LSNDP (respectively, 2NSNDP) as described in Section 7. We denote by $r' \in \{1, 2\}^{V'}$ the connectivity type vector corresponding to $G'$, and by $\bar{x}'$ the restriction of $\bar{x}$ on $E'$. To speed up the separation routines, the latter are applied to the graph $G'$ with an edge weight vector given by $\bar{x}'$. In fact, from Lemma 9, looking for valid inequalities for the survivable network design problem on $G$ violated by $\bar{x}$ reduces to looking for such inequalities in $G'$ violated by $\bar{x}'$.

The separation of both cut inequalities (4) and node cutset inequalities (6) can be performed by computing Gomory-Hu trees on $G'$ [65]. (A Gomory-Hu tree $T$ on $G'$ has the property that for any pair of nodes $s$, $t \in V'$, the minimum $st$-cut in $T$ is also a minimum $st$-cut in $G'$, and using Gusfield's algorithm [79], finding $T$ requires $|V'| - 1$ maximum flow computations.) To deal with the two possible right-hand sides of the inequalities (4), only the nodes $u \in V'$ with $r'(u) = 2$ are first considered in a Gomory-Hu tree and then these nodes are shrunk in $G'$ to provide a graph on which another Gomory-Hu tree is computed. For the node cutset inequalities (6), their separation reduces to a sequence of $|V'|$ Gomory-Hu tree computations. In consequence, an exact algorithm that permits us to separate the cut inequalities (4) [respectively, node cutset inequalities (6)] can be implemented to run in $O(n^4)$ time [respectively, $O(n^5)$ time].

We now turn our attention to the separation of the partition inequalities (8), which are exclusively considered when $r \in \{1, 2\}^V$. An exact algorithm to separate those inequalities was given by Barahona and Kerivin [13] and its time complexity is $O(n^7)$. However, in the context of a branch-and-cut algorithm, this time complexity may seem a little too high, and some heuristics were devised for separating those inequalities [78, 87, 112]. Kerivin et al. [87] thus devised a heuristic where they consider two cases depending on the value of the right-hand side. In fact, using Barahona's algorithm [12], one can solve exactly the separation problem of the multicut inequalities (7) and the partition inequalities (9) violated by more than 1. Kerivin et al. then developed a heuristic to separate the partition inequalities (9) violated by less than 1. The latter is based on the transformation of cuts given by a Gomory-Hu tree on $G'$ into partitions by applying Barahona's algorithm to both shores of the cuts. Their whole approach leads to a heuristic for separating the inequalities (8), which can be implemented to run in $O(n^5)$ time. The separation of the node partition inequalities (14) can be reduced to a sequence of $|V'|$ separation problem of multicut inequalities (7) in the graph $G'$ with one less node. (We recall that these inequalities are only considered for the 2NSNDP.) Therefore, using Barahona's algorithm, separating the inequalities (14) can be done in $O(n^5)$ time.

We finally discuss the separation routines for the $F$-partition inequalities (10) and (19), the separation problem of which has not been established yet. Therefore, two heuristics were devised by Kerivin et al. [87] for separating both inequalities (10) and (19). The idea of the first one comes directly from the study of the critical extreme points (see Section 7), and especially from Theorems 6 and 7. This heuristic consists of looking for cycles in $G'$ formed by fractional valued edges and for any of these cycles $(v_1, \ldots, v_p)$, trying to generate a violated $F$-partition inequality induced by the partition $\{V' \setminus \{v_1, \ldots, v_p\}, \{v_1\}, \ldots, \{v_p\}\}$ and an edge subset among the edges having exactly one extremity in the cycle. This first heuristic can then be implemented using a recursive algorithm that determines the 2-connected components in a graph, leading to an $O(n^2)$ time complexity. Another heuristic was devised in [87], which transforms cuts containing as many edges $e \in E'$ with $\bar{x}'(e) = 1$ as possible into $F$-partitions. To determine such cuts, one can compute a Gomory-Hu tree on $G'$ with the edge weight vector $(1 - \bar{x}'(e), e \in E')$. Given a cut $\delta(W)$ obtained from the Gomory-Hu tree, a $F$-partition inequality is then generated by considering the partition induced by $W$ and the nodes in $\overline{W}$ and by picking an edge subset $F \in \delta(W)$. (The same process can be applied for the partition induced by $\overline{W}$ and the nodes in $W$.) This second heuristic requires the solution of $O(n)$ minimum cut problems and runs in $O(n^4)$ time. Another way to generate a $F$-partition from a cut would be to apply Barahona's algorithm [12] to each shore of the cut; the complexity of the second heuristic would then increase to $O(n^5)$ time.

### *8.2. Computational Experiments*

We now briefly discuss some of the computational results obtained by Kerivin et al. [87] for both (1,2)LSNDP and 2NSNDP using the branch-and-cut algorithm previously described. The test problems, which consist of complete graphs, came from the TSPLIB library [109]. The number of nodes ranged up to 574 when node types are all equal to 2, and up to 101 when $r \in \{1, 2\}^V$. Moreover, if the (1,2)-link survivable network design problem was considered, the connectivity type vector was randomly generated.

Their first series of experiments concerns the SNDP with $r(u) = 2$ for all $u \in V$, that is, the 2-link connected network problem and the 2-node connected network problem. It appeared that the linear relaxations given by the trivial and cut inequalities (together with the node-partition inequalities for the 2NSNDP) provided good lower bounds. The average relative error between these lower bounds and the optimal values was actually less than 1%. Furthermore, for both problems, the $F$-partition inequalities (10) appeared to be very efficient to solve those problems without any need of branching, or at least, to considerably improve the lower bound given at the root node of the branch-and-cut tree. This remark confirms the one from Baïou [5] for the 2LSNDP where the $F$-partition inequalities (10) are separated in polynomial time by first fixing the edge subset $F$. Moreover, their experiments also showed that the two separation heuristics for the $F$-partition inequalities (10) detected a large enough number of such violated inequalities, and therefore were very useful. Furthermore, they noticed that the solution obtained for the 2LSNDP and the 2NSNDP are also optimal for the traveling salesman problem in the majority of the cases, showing thus that considering those inequalities in a branch-and-cut algorithm may be useful for solving the TSP.

Kerivin et al. then considered the (1,2)-link survivable network design problem to estimate the importance of the partition inequalities (8) and the $F$-partition inequalities (19) in the solution of that problem. They first noticed that the partition inequalities (8) played a central role for solving the (1,2)LSNDP to optimality. In fact, by considering them together with the $F$-partition inequalities (19), the relative error between the optimal value and the lower bound achieved at the root node considerably decreased, and several problems could be solved without any branching. A direct outcome of that remark is the efficiency of the separation routines, presented in Section 8.1, to detect violated partition inequalities (8). Actually, almost three-quarters of the violated partition inequalities were detected by the heuristic transforming cuts into partitions, even though violated multicut inequalities (7) and partition inequalities (9) violated by more than 1 were first sought. The $F$-partition inequalities (19) appeared in a smaller proportion than the $F$-partition (10) for the SNDP with $r(u) = 2$ for all $u \in V$. Nevertheless, combined with the cut and partition inequalities, they speeded up the solution of the (1,2)LSNDP in some cases, and eventually solved it to optimality at the root node of the branch-and-cut tree. This implies that the heuristics for separating the $F$-partition inequalities may be less efficient for the (1,2)LSNDP, yet these inequalities seemed to be useful.

Finally, the interest of the reduction operations introduced in Section 7 was also evaluated in [87] (see also Kerivin [84]) by making the same experiments with and without them. Kerivin et al. [87] then reported that for both cases [i.e., $r(u) = 2$ for all $u \in V$ and $r \in \{1, 2\}^V$] the solutions of the problems consumed much more CPU time when the reduction operations were not considered. In fact, getting an optimal solution might need a few seconds with those operations, and several hours without them. Moreover, using the reduction operations seemed to make the separation routines more efficient, the number of detected violated inequalities being higher in that case. It was also mentioned in [87] that many of the $F$-partitions (10) that cut off fractional solutions of the 2LSNDP and the 2NSNDP in the experiments were facet-defining in $G$, because of the application of the reduction operations.

## 9. SURVIVABILITY WITH LENGTH CONSTRAINTS

In general, the survivability requirement is not sufficient to guarantee a cost effective routing. Indeed, the alternative routing paths may be too long and then too costly to be suitable. For instance, an optimal 2-connected network may be a Hamiltonian cycle (i.e., a cycle going through all the nodes of the network exactly once). In consequence, further technical constraints have to be added; in particular, one can impose a limit on the length of the rerouting paths. Actually, there are two types of rerouting strategies in telecommunications. The first one, called *local rerouting*, consists of rerouting the traffic between the extremities of the failed link. This link, together with the rerouting path, thus form a cycle. A network suitable for this strategy would then be one where each link belongs to a cycle (also called ring) not exceeding a certain length. Such a network is called a *self-healing ring network* [123]. This is, for instance, the case of the SDH/SONET networks. The second strategy is the *end-to-end rerouting*. In that case, if a link fails, the traffic must be rerouted between its origin-destination nodes. To limit the rerouting, one thus must have at least two edge (node)-disjoint paths with bounded length between each origin-destination pair, so that if one of the paths fails, the traffic may be rerouted (in minimum time) on the second one. This corresponds, for instance, to the ATM networks and the Internet. In many practical situations, the length of the routing path is considered as the number of links (also called hops) in the path, and then we talk about a *hop-constrained path*. In this section, we discuss some variants of these two length constrained survivable network design problems.

### 9.1. Survivability with Bounded Rings

In [53], Fortz et al. considered the problem of designing a minimum cost 2-node connected network such that each edge belongs to a cycle of a bounded length. This problem can be presented as follows: given a graph $G = (V, E)$ such that each edge $e \in E$ has a cost $c(e)$ and a length $d(e)$, and a positive integer $K$, the problem consists of finding a minimum cost 2-node connected subgraph $(W, F)$ such that each edge of $F$ belongs to a cycle of length less than or equal to $K$. Fortz et al. called this problem the *2-connected subgraph with bounded rings problem*. This problem is a generalization of the 2-node connected subgraph problem, that is the NSNDP with $r(v) = 2$ for all $v \in V$. In fact, the latter is nothing but the 2-connected subgraph with bounded rings problem when $K = \infty$. Fortz et al. [53] derived valid and facet-defining inequalities for the associated polytope, and devised separation procedures. They also presented a cutting plane algorithm and discussed experimental results. In [52], Fortz and Labbé gave a formulation for the problem based on a set covering approach. They provided further classes of facets and discussed the associated separation problems.

They also reported computational results obtained with a cutting plane algorithm. For a complete survey of this problem, see Fortz [51].

In [54], Fortz et al. studied the edge version of the above problem, the *2-edge connected subgraph with bounded rings problem (2ECSBR)*. They considered the case where the length of each edge is 1. So the problem here is to find a minimum cost 2-edge connected subgraph such that each edge belongs to a cycle with no more than $K$ edges. Fortz et al. [54] introduced a class of valid inequalities, and, using this, they gave an integer programming formulation for the problem in the space of the design variables. In what follows we describe these inequalities.

Let $G = (V, E)$ be a graph and $K \geq 3$. If $\pi = \{V_0, \ldots, V_p\}$ is a partition of $V$, then we let $C_\pi = \cup_{i=0}^{p-1}[V_i, V_{i+1}] \cup [V_0, V_p]$ and $T_\pi = \delta(V_0, \ldots, V_p) \setminus C_\pi$. Suppose now that the partition $\pi$ is such that $p \geq K$ and let $e \in [V_0, V_p]$. Consider the inequality

$$x(T_\pi^e) \geq x_e \tag{20}$$

where

$$T_\pi^e = T_\pi \cup ([V_0, V_p] \setminus \{e\}). \tag{21}$$

Fortz et al. [54] showed that inequalities (20) are valid for the polytope associated with the 2ECSBR. Inequalities (20) are called *cycle inequalities*. Moreover, they proved the following.

**Theorem 10 [54].** *Let $G = (V, E)$ be a graph and $K \geq 3$. The 2ECSBR is equivalent to the following integer linear programming problem*

$$\begin{aligned}
\min \quad & \sum_{e \in E} c_e x_e \\
s.t. \quad & x(\delta(W)) \geq 2 \quad W \subset V, \emptyset \neq W \neq V, && (22)\\
& x(T_\pi^e) \geq x_e \quad e \in [V_0, V_K],\ \pi = (V_0, \ldots, V_K) \\
& \qquad \textit{is a partition of } V, \textit{and } T_\pi^e \\
& \qquad \textit{is defined by (21)}, && (23)\\
& 0 \leq x_e \leq 1 \quad e \in E, && (24)\\
& x_e \in \{0, 1\} \quad e \in E. && (25)
\end{aligned}$$

By adding to the formulation given by Theorem 10 the constraints

$$x(\delta_{G-v}(W)) \geq 1, \quad W \subset V \setminus \{v\},\ v \in V,$$

we obtain a formulation for the 2-node connected subgraph with bounded rings problem (when the lengths are equal to 1).

It is not hard to see that the separation problem for inequalities (20) associated with an edge $e = st$ reduces to finding a minimum weight edge subset that intersects all $st$-paths of length $\leq K - 1$. Fortz et al. [54] showed that, when $K \leq 4$, this problem reduces to a max-flow problem in an appropriate directed graph, and hence, can be solved in polynomial time. As a consequence, they obtained a polynomial-time separation algorithm for inequalities (20) when $K \leq 4$. Unfortunately, McCormick [100] showed that the above constrained min-cut problem is NP-hard if $K \geq 13$. A question that is still open is whether or not this problem is polynomially solvable for $K = 5$. Fortz et al. [54] described further valid inequalities. Using these inequalities as well as the cycle inequalities they developed a branch-and-cut algorithm for the 2ECSBR and presented computational results.

### *9.2. Hop-Constrained Paths*

**9.2.1. The Hop-Constrained Spanning Tree Problem.** Hop-constraints have been considered by Gouveia [66, 67] for the minimum spanning tree problem. The problem is then, given a graph $G = (V, E)$ with weights on the links and a *root* node, to find a minimum spanning tree such that the (unique) path between the root and any other node in the graph has no more than $L$ links (hops), where $L$ is a fixed positive integer. This restriction guarantees a specified level of service with respect to certain performance measures. This problem is NP-hard even for $L = 2$ (see for instance [37]). Gouveia [66] gave a multicommodity flow formulation for that problem and discussed a Lagrangian relaxation to improve the LP bound. Gouveia [67] proposed a hop-indexed reformulation of a multicommodity flow formulation, which is based on an extended description of the $L$-walk polyhedron. The reported computational results show that the new formulation is attractive to use when $L$ is small. Unfortunately, as the number of variables of the model grows up with $L$, the size of the corresponding linear programming relaxation may lead to excessive computer storage requirements or to excessive computational time when more dense instances, or instances with a bigger value of $L$ or a larger number of nodes, are considered. Gouveia and Requejo [73] proposed a Lagrangian relaxation for the problem that dualizes the hop-indexed flow conservation constraints. Reported results show that this relaxation is a good alternative to directly solving the corresponding linear programming relaxation. In [37], Dahl studied the problem for $L = 2$ from a polyhedral point of view and gave a complete description of the associated polytope when the graph is a wheel. Gouveia and Janssen [68] discussed a generalization of the previous problem where two different cable technologies with different reliabilities are available. They formulated the problem as a directed multicommodity flow model and used Lagrangian relaxation together with subgradient optimization to derive lower bounds. Gouveia and Magnanti [69] considered the problem that consists of finding a minimum spanning tree such that the number of edges between any pair of nodes in the tree is limited to a given bound $D$ (i.e., the diameter). This problem is polynomially solvable if $D \leq 3$ and NP-hard if $D \geq 4$. Gouveia and Magnanti [69] derived single source formulations for the problem based on the concept of tree centers along with some computational experiments. They also pointed out that the case when $D$ is odd is harder to solve than the even one. In [70], Gouveia et al. introduced a new modeling approach for the case when $D$ is odd and showed that this approach performs better than the one in Gouveia and Magnanti [69].

**9.2.2. The Hop-Constrained Path Problem.** The closely related and basic routing *hop-constrained path problem* has also seen increased attention recently. This problem consists of finding between two distinguished nodes $s$ and $t$ a minimum cost path with no more than $L$ edges when $L$ is fixed. This problem can be solved efficiently using dynamic programming. In fact, it was this approach that motivated the extended description of the $L$-walk polyhedron described in [67]. In what follows we briefly discuss this problem.

The *L-path polytope*, denoted by $L$PP($G$) is the convex hull of the incidence vectors of the $st$-paths having no more than $L$ edges. Clearly, the following inequalities are valid for $L$PP($G$).

$$x(\delta(W)) \geq 1, \quad \text{for all } st\text{-cuts } \delta(W), \tag{26}$$

and are called *st-cut inequalities*. In [38], Dahl considered the dominant of the $L$-path polytope, that is the polyhedron $L\mathrm{PP}(G)+\mathbb{R}^E_+$. He described a class of valid inequalities for the problem and gave a complete description of that polyhedron when $L \leq 3$. In particular, he introduced a class of valid inequalities as follows.

Let $\{V_0, V_1, \ldots, V_{L+1}\}$ be a partition of $V$ such that $s \in V_0$, $t \in V_{L+1}$ and $V_i \neq \emptyset$ for all $i = 1, \ldots, L$. Let $T$ be the set of edges $e = uv$ where $u \in V_i$, $v \in V_j$ and $|i - j| > 1$. Then the inequality

$$x(T) \geq 1, \tag{27}$$

is valid for the $L$-path polyhedron. Using the same partition, this inequality can be generalized in a straightforward way, as follows to the case when $K$ edge-disjoint paths are required between $s$ and $t$

$$x(T) \geq K. \tag{28}$$

Inequalities (27) and (28) are called *L-path cut inequalities* (or *jump inequalities* [40]). The separation problem for these inequalities can be solved in polynomial time, if $L \leq 3$. In fact, it is easily seen that this problem reduces to finding a minimum edge set that intersects all the $st$-paths with no more than $L$ edges. Because $L \leq 3$, as shown by Fortz et al. [54], this can be done in polynomial time. Dahl [38] showed that inequalities (27), together with inequalities (26) and the nonnegativity inequalities, completely describe the $L$-path polyhedron when $K \leq 3$. This implies that for nonnegative costs $c(e)$, $e \in E$, the hop-constrained path problem when $L \leq 3$ is equivalent to minimizing $\sum_{e \in E} c(e)x(e)$ subject to (26), (27), and $x(e) \geq 0$ for all $e \in E$.

In [105], Nguyen described a general class of valid inequalities for the $L$-path polyhedron, and, using LP-duality, he showed that these inequalities together with the $st$-cut inequalities (26) characterize this polyhedron for every $L$. He also gave an efficient algorithm that enables separation from this polyhedron.

In [40], Dahl and Gouveia considered the directed hop-constrained path problem. Note that the $st$-cut inequalities (26), and the $L$-path-cut inequalities (27), (28), can be easily extended to that problem. Dahl and Gouveia [40] described a class of valid inequalities obtained by lifting from the directed $L$-path-cut inequalities and showed that these inequalities together with the flow conservation constraints and the trivial inequalities characterize the directed $L$-path polytope when $L \leq 3$. They also identified valid inequalities and addressed some polyhedral issues for the case when $L \geq 4$.

In [31], Coullard et al. investigated the structure of the polyhedron associated with the directed $st$-walks having exactly $L$ arcs of a directed graph, where a directed walk is a directed path that may go through the same node more than once. They presented an extended formulation of the problem and, using projection, they gave a linear description of the associated polyhedron. They also discussed classes of facets of that polyhedron. In [39], Dahl et al. considered the polytope of the directed $st$-walks having no more than $L$ arcs. They presented an extended formulation for the underlying $L$-walk problem when $L = 4$, and used projection to obtain a complete linear description of that polytope for the same value of $L$. They also described generalized valid inequalities that define facets for the dominant of that polytope, which, quite surprisingly, shows that obtaining a complete description for the dominant of the $st$-walk polytope when $L = 4$ is much harder than obtaining such a description for the polytope itself. (Note that if $L \leq 3$, a walk is also a path, and then the polyhedral investigation of Dahl and Gouveia [40] also holds for the 3-walk polytope.)

**9.2.3. The Hop-Constrained Network Design Problem.** A more general network design problem with hop-constraints that has also been investigated is *the hop-constrained network design problem (HCNDP)*. This can be presented as follows: given a graph $G = (V, E)$ with weights on the links, a set of pairs of terminals and two positive integers $K$ and $L$, find a minimum weight subgraph such that between each pair of terminals there are at least $K$ edge-disjoint paths with no more than $L$ links. This problem is NP-hard even when $K = 1$ and $L = 2$ [41]. In Balakrishnan and Altinkemer [9], the HCNDP was studied when $K = 1$ within the framework of a more general model for backbone networks. The authors gave a mixed-integer programming formulation and developed a Lagrangian-based algorithm to generate lower bounds and feasible solutions. In a recent work, for the same case, Pirkul and Soni [107] introduced multicommodity flow-based formulations and developed heuristics based on the linear relaxations. Also, extensive computational results are reported.

The HCNDP was considered in Dahl and Johannessen [41] for $K = 1$ and $L = 2$. The authors gave an integer programming formulation and described classes of valid inequalities. Using this, they developed a cutting plane algorithm and presented computational results. In [81], Huygens et al. studied the HCNDP in the case when there is only one pair of terminals, say $s$ and $t$, $K = 2$ and $L = 3$. They gave an integer programming formulation for the problem in this case in the space of the design variables. They showed that the *st*-cut inequalities (inequalities (26) with right-hand side 2) and the $L$-path inequalities (28) (with $K = 2$) together with the 0–1 integrality constraints represent this problem, and they gave an extension of this formulation to the case where $K \geq 2$. They also discussed the polytope $P(G, L)$ given by the constraints of the linear relaxation of this formulation. In particular, they proved the following.

**Theorem 11 [81].** *P(G,L) is integral, if* $L \leq 3$.

Theorem 11 implies that the associated polytope is equal to $P(G, L)$. In addition, because the separation problem for the *st*-cut and $L$-path cut inequalities can be solved in polynomial time when $L \leq 3$, from Theorem 11, it follows that the HCNDP when $L \leq 3,\ K = 2$ and only one pair of terminals is considered can be solved in polynomial time using a cutting plane algorithm. As pointed out in [81], the formulation given above (for the HCNDP when $L \leq 3$, $K = 2$ and only one pair of terminal is considered) is no longer valid for the problem if $L \geq 4$. However, for $L \leq 3$, one can see that this formulation can be easily extended to the HCNDP with an arbitrary number of pairs of terminals.

**9.2.4. Related Hop-Constrained Problems.** Hop-constraints have also been considered for related network design problems. In [15], Ben-Ameur defined some classes of 2-connected graphs satisfying path (and cycle)-length constraints. He introduced some parameters and established properties and relationships between these graphs. Moreover, he investigated the hop-constrained flow problem and gave lower bounds on the number of edges of these graphs. As a consequence, he obtained some valid inequalities for the underlying survivable network design problem. Gouveia et al. [71] considered an MPLS (Multi-Protocol Label Switching) network design model with hop constraints. They gave mixed-integer programming formulations and discussed computational results. In [72], Gouveia et al. studied the design of MPLS over optical networks. They also used hop constraints to guarantee maximum delay quality of service.

Itai et al. [82] studied the complexity of several variants of the maximum disjoint hop-constrained path problem. This consists of finding the maximum number of disjoint paths between two nodes $s$ and $t$ of length equal to (or bounded by) $K$ where $K$ is a positive integer. They showed that the problem is NP-complete for $K \geq 5$ and polynomially solvable for some of the variants for $K \leq 4$. In particular, they devised a polynomial-time algorithm for the problem when the paths must be node-disjoint (respectively, edge-disjoint) and of length bounded by $K$, with $K \leq 4$ (respectively $K \leq 3$). Bley [19] addressed approximation and computational issues for the node-disjoint and edge-disjoint hop-constrained path problems. In particular, he showed that the problem of computing the maximum number of edge-disjoint paths between two given nodes of length equal to 3 is polynomial. This result answered an open question in [82].

## 10. CONCLUDING REMARKS

In this article, we surveyed some optimization techniques for the survivable network design problem. We focused on the undirected network case. Nevertheless, survivability has also been considered in directed networks. In that case, the survivability conditions are the same as in the undirected case, except that "path" is replaced by "directed path." So the formulation given for the undirected case can straightforwardly be extended to this one. Although it has applications to many practical situations, the directed survivable network design problem has not seen as much attention as the undirected case. In [36] (see also [35]) Dahl studied Steiner problems in directed graphs. He investigated the polyhedral structure and developed cutting plane algorithms. Ball et al. [11] (see also Liu [93]) studied the two terminal Steiner tree problem in directed graphs. (Notice that this problem is solvable in polynomial time.) They proposed an extended formulation for the problem and used projection to obtain facet-inducing inequalities for the associated polyhedron in the natural space.

Moreover, several researchers investigated directed formulations for variants of the link-survivable network design problem. In [21], Chopra introduced a directed formulation for the 2-edge connected subgraph problem. Using projection, he also showed that some classes of valid inequalities can be obtained from the directed cut inequalities. Magnanti and Raghavan [94] introduced a multicommodity flow formulation for the generalized Steiner problem and edge connectivity. They showed that this formulation is stronger than the undirected cut formulation, and they projected out some known classes of valid inequalities.

The *capacitated survivable network design problem* has also been investigated. For this problem, in addition to the connectivity requirements, we are given a set of demands between some pairs of node and a set of discrete capacities associated with the edges of the graph, each with an associated building cost. The problem is to construct a survivable network and find which capacities to install on the edges so that each demand (or some prescribed fraction of the demand) can be routed in the event of a failure, so that the overall cost is mininum. Stoer and Dahl [113] were the first who considered this problem. They studied the 2-connected case where at most one edge (node) fails at a time. In [42] (see also [113]), they devised a branch-and-cut algorithm for the problem and solved a large number of real-world instances. A more general model including length and routing constraints has been considered in [4, 16, 88]. For that model, survivability is provided using traffic rerouting strategies (e.g., local rerouting, end-to-end rerouting).

Valid inequalities induced by cutsets usually appear as subsystems in formulations for network design problems. Therefore, a deep knowledge of the polyhedra yielded by these subsystems, in particular concerning their facial structure, may be of great interest for solving these problems by cutting plane algorithms. Bienstock and Muratore [18] studied the subsystem induced by a cutset in the capacitated survivable model. Let $\mathcal{C}$ be a cutset of the graph and $L$ and $D$ two positive integers. Bienstock and Muratore [18] considered the polyhedron $P_{\mathcal{C}}(L, D)$, convex hull of the solutions of the following inequality system

$$\sum_{e \in \mathcal{C} \setminus \{e\}} x(e) \geq L \qquad \text{for all } e \in \mathcal{C}, \tag{29}$$

$$\sum_{e \in \mathcal{C}} x(e) \geq D \tag{30}$$

$$x(e) \in \mathbb{Z}_+ \qquad \text{for all } e \in \mathcal{C}.$$

Here the variable $x(e)$ corresponds to the capacity to install on edge $e$, inequalities (29) express the fact that in the event of a failure at least $L$ units of capacity are required in the surviving links of $\mathcal{C}$ and inequality (30) is the demand inequality with $D$ equal to the amount of demand that must go through $\mathcal{C}$. Bienstock and Muratore [18] described valid inequalities and structural properties of the extreme points of this polyhedron. They also used this in a cutting plane algorithm for the capacitated survivable network design problem. Magnanti and Wang [95] studied a similar poyhedron without the demand constraint (30) and with different right-hand sides in (29).

These investigations are certainly a first step toward the development of a complete and efficient cutting plane approach for the general capacitated model. Such an approach may also be combined with other tools like column generation techniques and approximation algorithms.

## ACKNOWLEDGMENTS

We would like to thank the anonymous referees for their valuable comments. We would also like to thank Douglas Shier for his constructive remarks. Part of this work was done when the second author was visiting the service G.O.M. of the Université Libre de Bruxelles, Belgium. The financial support is greatly appreciated.

## REFERENCES

[1] A. Aggarwal and N. Garg, A scaling technique for better network design, Proc 5th Annual ACM-SIAM Symposium on Discrete Algorithms, 1994, pp. 233–240.

[2] A. Aggarwal, P. Klein, and R. Ravi, When trees collapse: An approximation algorithm for the generalized Steiner problem in networks, SIAM J Comput 24 (1995), 440–456.

[3] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, Network flows: Theory, algorithms, and applications, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[4] D. Alevras, M. Grötschel, and R. Wassäly, Capacity and survivability models for telecommunication networks, Technical Report SC 97–24, ZIP, Berlin, Germany, 1997.

[5] M. Baïou, Le problème du sous-graphe Steiner 2-arête connexe: Approche polyédrale, Ph.D. Dissertation, Université de Rennes 1, Rennes, France, 1996.

[6] M. Baïou, On the dominant of the Steiner 2-edge connected subgraph polytope, Discr Appl Math 112 (2001), 3–10.

[7] M. Baïou, F. Barahona, and A.R. Mahjoub, Separation of partition inequalities, Math Oper Res 25 (2000), 243–254.

[8] M. Baïou and A.R. Mahjoub, Steiner 2-edge connected subgraph polytopes on series-parallel graphs, SIAM J Discr Math 10 (1997), 505–514.

[9] A. Balakrishnan and K. Altinkemer, Using a hop-constrained model to generate alternative communication network design, ORSA J Comput 4 (1992), 192–205.

[10] A. Balakrishnan, T.L. Magnanti, and P. Mirchandani, Connectivity-splitting models for survivable network design, Networks 43 (2004), 10–27.

[11] M.O. Ball, W.G. Liu, and W.R. Pulleyblank, Two terminal Steiner tree polyhedra, Technical Report CORR 87-33, University of Waterloo, Dept. of Combinatorics and Optimization, Waterloo, CA, 1987.

[12] F. Barahona, Separating from the dominant of the spanning tree polytope, Oper Res Lett 12 (1992), 201–203.

[13] F. Barahona and H. Kerivin, Separation of partition inequalities with terminals, Discr Optim 1 (2004), 129–140.

[14] F. Barahona and A.R. Mahjoub, On two-connected subgraph polytopes, Discr Math 147 (1995), 19–34.

[15] W. Ben-Ameur, Constrained length connectivity and survivable networks, Networks 36 (2000), 17–33.

[16] W. Ben-Ameur and T.-T.-L. Pham, Design of survivable network based on end-to-end rerouting, Proc third international workshop of Design of Reliable Communication Networks, Budapest, Hungary, 2001, pp. 258–265.

[17] D. Bienstock, E.F. Brickell, and C.L. Monma, On the structure of minimum-weight $k$-connected spanning networks, SIAM J Discr Math 3 (1990), 320–329.

[18] D. Bienstock and G. Muratore, Strong inequalities for capacitated survivable network design problems, Math Program 89 (2001), 127–147.

[19] A. Bley, On the complexity of vertex-disjoint length-restricted path problems, Computat Complexity 12(3–4) (2003), 131–149.

[20] S.C. Boyd and T. Hao, An integer polytope related to the design of survivable communication networks, SIAM J Discr Math 6 (1993), 612–630.

[21] S. Chopra, Polyhedra of the equivalent subgraph problem and some edge connectivity problems, SIAM J Discr Math 5 (1992), 321–337.

[22] S. Chopra, The k-edge connected spanning subgraph polyhedron, SIAM J Discr Math 7 (1994), 245–259.

[23] S. Chopra and M.R. Rao, The Steiner tree problem I: Formulations, compositions and extension of facets, Math Program 64 (1994), 209–229.

[24] S. Chopra and M.R. Rao, The Steiner tree problem II: Properties and classes of facets, Math Program 64 (1994), 231–246.

[25] S. Chopra and M. Stoer, Private communication.

[26] W. Chou and H. Frank, Survivable communication networks and the terminal capacity matrix, IEEE Trans Circuit Theory 17 (1970), 183–192.

[27] W. Chou and H. Frank, Connectivity communication networks and the terminal capacity matrix, IEEE Trans Circuit Theory 17 (1970), 486–490.

[28] N. Christofides and C.A. Whitlock, "Network synthesis with connectivity constraints—A survey," Operational

Research '81, J.P. Brans (Editor), North Holland, Amsterdam, 1981, pp. 705–723.

[29] L.W. Clark and G. Anandalingam, A bootstrap heuristic for designing minimum cost survivable networks, Comput Oper Res 22 (1995), 921–924.

[30] G. Cornuéjols, J. Fonlupt, and D. Naddef, The traveling salesman problem on a graph and some related integer polyhedra, Math Program 33 (1985), 1–27.

[31] C.R. Coullard, A.B. Gamble, and J. Liu, "The $K$-walk polyhedron," Advances in optimization and approximation, D.-Z. Du and J. Sun (Editors), Kluwer Scientific Publishers, Norwell, 1994, pp. 9–29.

[32] C.R. Coullard, A. Rais, R.L. Rardin, and D.K. Wagner, Linear-time algorithm for the 2-connected Steiner subgraph problem on special classes of graphs, Networks 23 (1993), 195–206.

[33] C.R. Coullard, A. Rais, R.L. Rardin, and D.K. Wagner, The dominant of the 2-connected-Steiner subgraph polytope for $W_4$-free graphs, Discr Appl Math 66 (1996), 195–206.

[34] W.H. Cunningham, Optimal attack and reinforcement of a network, J ACM 32 (1985), 549–561.

[35] G. Dahl, Contribution to the design of survivable directed networks, Ph.D. Dissertation, University of Oslo, Technical Report TF R 48/91, Norwegian Telecom, Research Dept, 1991.

[36] G. Dahl, Directed Steiner problems with connectivity constraints, Discr Appl Math 47 (1993), 109–128.

[37] G. Dahl, The 2-hop spanning tree problem, Oper Res Lett 23 (1998), 21–26.

[38] G. Dahl, Notes on polyhedra associated with hop-constrained paths, Oper Res Lett 25 (1999), 97–100.

[39] G. Dahl, N. Foldnes, and L. Gouveia, A note on hop-constrained walk polytopes, Oper Res Lett 32 (2004), 345–349.

[40] G. Dahl and L. Gouveia, On the directed hop-constrained shortest path problem, Oper Res Lett 32 (2004), 15–22.

[41] G. Dahl and B. Johannessen, The 2-path network design problem, Networks 43 (2004), 190–199.

[42] G. Dahl and M. Stoer, A cutting plane algorithm for multicommodity survivable network design problems, INFORMS J Comput 10 (1998), 1–11.

[43] M. Didi Biha, Graphes k-arête connexes et polyèdres, Ph.D. Dissertation, Université de Rennes 1, Rennes, France, 1998.

[44] M. Didi Biha, H. Kerivin, and A.R. Mahjoub, Steiner trees and polyhedra, Discr Appl Math 112 (2001), 101–120.

[45] M. Didi Biha, H. Kerivin, and A.R. Mahjoub, On the polytope of the (1,2)-survivable network design problem, Preprint, 2004.

[46] M. Didi Biha and A.R. Mahjoub, $k$-Edge connected polyhedra on series-parallel graphs, Oper Res Lett 19 (1996), 71–78.

[47] M. Didi Biha and A.R. Mahjoub, Steiner $k$-edge connected subgraph polyhedra, J Comb Optim 4 (2000), 131–144.

[48] K.P. Eswaran and R.E. Tarjan, Augmentation problems, SIAM J Comput 5 (1976), 653–665.

[49] J. Fonlupt and A.R. Mahjoub, Critical extreme points of the 2-edge connected spanning subgraph polytope, Proc IPCO'99, LNCS 1610, 1999, pp. 166–183.

[50] J. Fonlupt and A.R. Mahjoub, Critical extreme points of the 2-edge connected spanning subgraph polytope, Math Program, to appear [also available as Technical Report RR-04-17, LIMOS, Université Blaise Pascal, Clermont-Ferrand, France].

[51] B. Fortz, Design of survivable networks with bounded rings, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.

[52] B. Fortz and M. Labbé, Polyhedral results for two-connected networks with bounded rings, Math Program 93 (2002), 27–54.

[53] B. Fortz, M. Labbé, and F. Maffioli, Solving the two-connected network with bounded meshes problem, Oper Res 48 (2000), 866–877.

[54] B. Fortz, A.R. Mahjoub, S.T. McCormick, and P. Pesneau, The 2-edge connected subgraph problem with bounded rings, Math Program, to appear [also available as Technical Report RR-04-18, LIMOS, Université Blaise Pascal, Clermont-Ferrand, France].

[55] A. Frank, Augmenting graphs to meet edge-connectivity requirements, SIAM J Discr Math 5 (1992), 25–53.

[56] G.N. Frederickson and J. Jájá, On the relationship between the biconnectivity augmentation problem and traveling salesman problem, Theoret Comput Sci 19 (1982), 189–201.

[57] H.N. Gabow, M.X. Goemans, and D.P. Williamson, An efficient approximation algorithm for the survivable network design problem, Math Program 82 (1998), 13–40.

[58] M.R. Garey and D.S. Johnson, Computers and intractability: A guide to the theory of NP-completeness, W. H. Freeman and Company, New York, 1979.

[59] M.X. Goemans, The Steiner tree polytope and related polyhedra, Math Program 63 (1994), 157–182.

[60] M.X. Goemans and D.J. Bertsimas, Survivable networks, linear-programming relaxations and the parsimonious property, Math Program 60 (1993), 145–166.

[61] M.X. Goemans, A.V. Goldberg, S. Plotkin, D.B. Shmoys, É. Tardos, and D.P. Williamson, Approximation algorithms for network design problems, Proc SODA, 1994, pp. 223–232.

[62] M.X. Goemans and D.P. Williamson, A general approximation technique for constrained forest problems, SIAM J Comput 24 (1995), 296–317.

[63] M.X. Goemans and D.P. Williamson, "The primal-dual method for approximation algorithms and its application to network design problems," Approximation algorithms for NP-hard problems, D.S. Hochbaum (Editor), PWS Publishing Company, Boston, MA, 1997, pp. 144–191.

[64] A.V. Goldberg and R.E. Tarjan, A new approach to the maximum-flow problem, J ACM 35 (1988), 921–940.

[65] R.E. Gomory and T.C. Hu, Multi-terminal network flows, J Soc Indust Appl Math 9 (1961), 551–570.

[66] L. Gouveia, Multicommodity flow models for spanning trees with hop constraints, Eur J Oper Res 95 (1996), 178–190.

[67] L. Gouveia, Using variable redefinition for computing lower bounds for minimum spanning and Steiner trees with hop constraints, INFORMS J Comput 10 (1998), 180–188.

[68] L. Gouveia and E. Janssen, Designing reliable tree networks with two cable technologies, Eur J Oper Res 105 (1998), 552–568.

[69] L. Gouveia and T.L. Magnanti, Network flow models for designing diameter constrained spanning and Steiner trees, Networks 41 (2003), 159–173.

[70] L. Gouveia, T.L. Magnanti, and C. Requejo, A 2-path approach for odd-diameter-constrained minimum spanning and Steiner trees, Networks 44 (2004), 254–265.

[71] L. Gouveia, P. Patrício, and A. de Sousa, MPLS network design with fault-tolerance and hop constraints, Preprint, 2004.

[72] L. Gouveia, P. Patrício, A. de Sousa, and R. Valadas, MPLS over WDM network design with packet level QoS constraints based on ILP models, Proc IEEE INFOCOM 2003, Vol. 1, 2003, pp. 576–586.

[73] L. Gouveia and C. Requejo, A new Lagrangian relaxation approach for the hop-constrained minimum spanning tree problem, Eur J Oper Res 132 (2001), 539–552.

[74] M. Grötschel, L. Lovász, and A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981), 70–89.

[75] M. Grötschel and C.L. Monma, Integer polyhedra associated with certain network design problem with connectivity constraints, SIAM J Discr Math 3 (1990), 502–523.

[76] M. Grötschel, C.L. Monma, and M. Stoer, Facets for polyhedra arising in the design of communication networks with low-connectivity constraints, SIAM J Optim 2(3) (1992), 474–504.

[77] M. Grötschel, C.L. Monma, and M. Stoer, Computational results with a cutting plane algorithm for designing communication networks with low-connectivity constraints, Oper Res 40 (1992), 309–330.

[78] M. Grötschel, C.L. Monma, and M. Stoer, "Design of survivable networks," Handbook in Operations Research and Management Science 7, M.O. Ball, C.L. Monma, and G. Nemhauser (Editors), North-Holland, Amsterdam, 1995, pp. 617–671.

[79] D. Gusfield, Very simple algorithms and programs for all pairs network flow analysis, Computer Science Division, University of California, Davis, 1987.

[80] T.-S. Hsu and M.-Y. Kao, A unifying augmentation algorithm for two-edge connectivity and biconnectivity, J Comb Optim 2 (1998), 237–256.

[81] D. Huygens, A.R. Mahjoub, and P. Pesneau, Two edge-disjoint hop-constrained paths and polyhedra, SIAM J Discr Math 8(2) (2004), 287–312.

[82] A. Itai, Y. Perl, and Y. Shiloach, The complexity of finding maximum disjoint paths with length constraints, Networks 12 (1982), 277–286.

[83] K. Jain, A factor 2 approximation algorithm for the generalized Steiner network problem, Combinatorica 21 (2001), 39–60.

[84] H. Kerivin, Réseaux fiables et polyèdres, Ph.D. Dissertation, Université Blaise Pascal, Clermont-Ferrand, France, 2000.

[85] H. Kerivin and A.R. Mahjoub, Separation of the partition inequalities for the (1,2)-survivable network design problem, Oper Res Lett 30 (2002), 265–268.

[86] H. Kerivin and A.R. Mahjoub, On survivable network polyhedra, Discr Math 290 (2005), 183–210.

[87] H. Kerivin, A.R. Mahjoub, and C. Nocq, "(1,2)-survivable networks: Facets and branch-and-cut," The Sharpest Cut, MPS-SIAM Series in Optimization, M. Grötschel (Editor), MPS/SIAM, Philadelphia, 2004, pp. 121–152.

[88] H. Kerivin, D. Nace, and T.-T.-L. Pham, Design of capacitated survivable networks with a single facility, IEEE/ACM Trans Networking, 13(2) (2005), 248–261.

[89] S. Khuller and B. Raghavachari, Improved approximation algorithms for uniform connectivity problems, J Algorithms 21 (1996), 434–450.

[90] S. Khuller and U. Vishkin, Biconnectivity approximation and graph carvings, J ACM 41 (1994), 214–235.

[91] C-W. Ko and C.L. Monma, Heuristic methods for designing highly survivable communication networks, Technical Report, Bellcore, 1989.

[92] E. Lawler, Combinatorial optimization: Networks and matroids, Holt, Rinehart and Winston, New York, 1976.

[93] W.G. Liu, Extended formulations and polyhedral projection, Ph.D. Thesis, Dept. of Combinatorics and Optimization, University of Waterloo, 1988.

[94] T.L. Magnanti and S. Raghavan, Strong formulations for network design problems with connectivity requirements, Networks 45(2) (2005), 61–79.

[95] T. Magnanti and Y. Wang, Polyhedral properties of the network restoration problem—With the convex hull of a special case, MIT ORC Working Paper OR 323-97, 1997.

[96] A.R. Mahjoub, Two-edge connected spanning subgraphs and polyhedra, Math Program 64 (1994), 199–208.

[97] A.R. Mahjoub and C. Nocq, On the linear relaxation of the 2-node connected subgraph polytope, Discr Appl Math 95 (1999), 389–416.

[98] A.R. Mahjoub and P. Pesneau, On the Steiner 2-edge connected polytope, Preprint, 2002.

[99] F. Margot, A. Prodon, and T.M. Liebling, Tree polytope on 2-trees, Math Program 63 (1994), 183–191.

[100] S.T. McCormick, Private communication.

[101] K. Menger, Zur allgemeinen kurventheorie, Fundam Math 10 (1927), 96–115.

[102] C.L. Monma, B.S. Munson, and W.R. Pulleyblank, Minimum-weight two connected spanning networks, Math Program 46 (1990), 153–171.

[103] C.L. Monma and D.F. Shallcross, Methods for designing communication networks with certain two-connected survivability constraints, Oper Res 37 (1989), 531–541.

[104] N.St.J.A. Nash-Williams, Edge-disjoint spanning trees of finite graphs, J Lond Math Soci 36 (1961), 445–450.

[105] V.H. Nguyen, A complete linear description for the $k$-path polyhedron, Preprint, 2003.

[106] M. Padberg and R. Rao, Odd minimum cutsets and $b$-matching, Math Oper Res 7 (1982), 67–80.

[107] H. Pirkul and S. Soni, New formulations and solution procedures for the hop constrained network design problem, Eur J Oper Res 148 (2003), 126–140.

[108] R. Ravi and D.P. Williamson, An approximation algorithm for minimum-cost vertex connectivity problems, Proc 6th

Annual ACM-SIAM Symposium on Discrete Algorithms, 1995, pp. 332–341.

[109] G. Reinelt, TSPLIB—A traveling salesman problem library, ORSA J Comput 3 (1991), 376–384.

[110] A. Schrijver, Combinatorial optimization: Polyhedra and efficiency, Springer-Verlag, Berlin, 2003.

[111] K. Steiglitz, P. Weigner, and D.J. Kleitman, The design of minimum-cost survivable networks, IEEE Trans Circuit Theory 16 (1969), 455–460.

[112] M. Stoer, Design of survivable networks, Lecture notes in mathematics 1531, Springer-Verlag, Berlin, 1992.

[113] M. Stoer and G. Dahl, A polyhedral approach to multicommodity survivable network design, Num Math 68 (1994), 149–167.

[114] J.W. Suurballe, Disjoint paths in a network, Networks 4 (1974), 125–145.

[115] J.W. Suurballe and R.E. Tarjan, A quick method for finding shortest pairs of disjoint paths, Networks 14 (1984), 325–336.

[116] K. Takamizawa, T. Nishizeki, and N. Saito, Linear-time computability of combinatorial problems on series-parallel graphs, J ACM 29 (1982), 623–641.

[117] W.T. Tutte, On the problem of decomposing a graph into $n$ connected factors, J Lond Math Soci 36 (1961), 221–230.

[118] D.P. Williamson, M.X. Goemans, M. Mihail, and V.V. Vazirani, A primal-dual approximation algorithm for generalized Steiner network problems, Combinatorica 15 (1995), 435–454.

[119] P. Winter, Generalized Steiner problem in outerplanar networks, BIT 25 (1985), 485–496.

[120] P. Winter, Generalized Steiner problem in series-parallel networks, J Algorithms 7 (1986), 549–566.

[121] P. Winter, Steiner problem in Halin networks, Discr Appl Math (1987), 281–294.

[122] P. Winter, Steiner problem in networks: a survey, Networks 17 (1987), 129–167.

[123] T.H. Wu, Fiber networks service survivability, Artech House, Norwood, MA, 1992.