The Multi-Budgeted and Weighted Bounded Degree Metric Steiner Network Problem

Georgios Stamoulis
Department of Data & Knowledge Engineering (DKE)
Maastricht University
P.O. Box 616, 6200 MD
Maastricht, The Netherlands
`georgios.stamoulis@maastrichtuniversity.nl`

Abstract

We study the multi-budgeted version of the metric *Survivable Network Design Problem* (SND) [8], also known as *Steiner Network* problem, where besides the usual connectivity requirements (i.e., lower bound on the number edge-disjoint paths) between pre-specified pairs of points, we are also asked to satisfy a set of extra linear constraints (the budgets). Based on combinatorial properties of extreme point solutions of the natural LP relaxation of the problem and using the iterative method, we design new approximation algorithms with the following performance guarantee: if any edge participates in at most $f \in \mathbb{N}$ budgets, then we show how we can devise an $(f + 2, f + 2)$ bi-criteria approximation algorithm in polynomial time i.e., return a solution with cost at most $(2 + f)$ times the optimal one with a potential budget overflow bounded from above by the same factor. In other words, the approximation and budget violation guarantee we achieve is a linear function of the maximum frequency among all the edges in the budgets, as opposed to a function of the *number of budgets* themselves and this constitutes a strict improvement over the previous approaches. Our approach is flexible enough and we demonstrate how it can be used to provide a $(4, 4)$ bi-criteria algorithm for the *Minimum Weighted Bounded Degree* version of the SND problem in which we have the standard SND connectivity constraints and, additionally, for every vertex v in the graph we have an upper bound on the sum of the weights of the edges incident to that edge in *any* feasible solution. These problems, besides their obvious theoretical interest, have numerous real life applications from Cloud Computing to Optical Networking systems.

1 Introduction

One of the very basic questions in the parallel/distributed computing community is the following: *How can we distribute/parallelize the flow of information to disjoint channels such that certain properties are guaranteed?* This falls very well under the Fault-Tolerant setting and to settings where we would like to exchange messages in a system (memory) with non overlapping channels. Also to settings where we split the computational task in autonomous chunks. We would like to route these “packets” individually through disjoint/autonomous channels to avoid possible interference between them or to maximize the probability that not all will be lost during a fault-sensitive situation.

Moreover, in distributed and network design settings one of the most basic requirement is to design suitable networks with connectivity properties that allow the network (or a sub-network of it) to survive a number of link failures. On top of that, such networks should be cost efficient. This falls under the very broad scheme of designing fault-tolerant networks, also known as survivable networks.

From the point of view of a theoretical modelling, such problems can be described by a simple, non-directed graph $G = (V, E)$ together with a weight (cost) function $w : E \rightarrow \mathbb{Q}^+$ on the set of edges (links). In other words, each link has a particular cost in case we choose to select (install it) in our final network. Together with this, we are given a set of *connectivity requirements* $\mathcal{R} = \{\{s_i, t_i\}\}$, $s_i, t_i \in V(G)$ such that $s_i \neq t_i$ i.e., the two nodes that constitute a pair should be distinct. Let $|\mathcal{R}| = q \in \mathbb{Z}^+$ be the number of all pairs. With each pair of source-sink vertices $\{s_i, t_i\} \in \mathcal{R}$ is associated a positive integer r_i . The goal is to find a minimum weight (cost) *Steiner network* $S \subseteq G$ such that in S there are *at least* r_i disjoint paths connecting the pair $\{s_i, t_i\} \in \mathcal{R}$. Note that, since the graph is non-directed, we do not require that the source-sink pairs to be ordered pairs of nodes.

The theoretical formalization of the problem resembles the usual *Steiner tree* problem: in this problem we are given a weighted graph $G = (V, E)$ and a set of required nodes $R \subseteq V$. The goal is to find a minimum cost subgraph T of G such that T spans all the required vertices in R . In other words, in T there exists a path between each distinct pair of vertices $\{\rho_1, \rho_2\} \in R \times R$. The fact that Steiner tree problem can be cast in the Steiner Network problem immediately proves that Steiner Network not only is **NP**-hard [6], but also **APX**-hard [24]. This resemblance is the main reason why the problem is called in the literature as Steiner Network (or as Survivable Network Design problem-SND). Moreover, the classical *Spanning tree* problem is also a special case of Steiner Network: define $\mathcal{R} = \{v, u\}$ for each distinct pair of vertices $u, v \in V(G)$ and let $r_i = 1$ for each pair $i \in \mathcal{R}$.

Naturally, the Steiner Network problem is a very important combinatorial problem, both from a theoretical and from a more practical perspective: this problem is motivated by distributed telecommunication applications where we want to find a *sparse* and cost-efficient network of an initially given dense one that can *tolerate failures and delays* on its links. In other words, even if any $r_i - 1$ edges fail in the new network, pair $i \in \mathcal{R}$ will still be connected in the new network. Highly important links (pairs $\{s_i, t_i\}$) have high connectivity requirement.

The problem probably is even more interesting from a *theoretical* point of view. Since the problem is **NP**-hard, we cannot hope to find the minimum cost Steiner network that satisfies the connectivity requirements in polynomial time (in the size of the input). So, a very natural and well studied approach is to settle for an *approximate* solution which, informally, would output a good approximate solution of the optimal one in polynomial time. Since the focus on this problem, and the focus of this paper as well, is on approximation algorithms, we define them formally in the following:

Definition 1 (Approximation Algorithm). *Let Π be a combinatorial optimization problem and x be an instance for Π . Let opt be the optimal solution for x and $value(opt)$ be its value. We assume that Π is a minimization problem. An algorithm A for Π is called a ρ -**approximation algorithm** for Π , for some $\rho > 1$, iff the following is true:*

1. *A runs in polynomial time on the input instance size of x , for any instance x .*
2. *A produces a feasible solution s for x .*

3. $\text{value}(s) \leq \rho \cdot \text{value}(\text{opt})$, for any instance x .

Prior to the seminal work of Kamal Jain [8] who derived a 2-approximation for the above problem, the best approximation guarantee was $2 \cdot \max_{i \in q} \{r_i\}$ [25]. A faster implementation of this result was given in [5]. This was improved to $2 \log \max_{i \in q} \{r_i\}$ in [7]. As we said, in the seminal work of Jain [8] a 2-approximation algorithm as given. Again, this is very interesting for two reasons: the obvious reason is that it significantly improved the approximation guarantee (providing the *first* constant approximation ratio for this problem) for a very well known and well studied combinatorial problem. The second reason, is that this result was based on the concept of *iterated* rounding of linear programs. The novel ideas introduced in this paper triggered a host of other very related results and the approach was so successful that even a whole book is dedicated to the study of iterative methods in combinatorial optimization [12].

The approach of Jain is the following: First, by introducing binary variables x_e for every edge $e \in E$ (which will model the decision to choose or not the corresponding edge in the final solution), we formulate the problem as an integer linear program where we want to minimize the sum of the costs of the selected edges $w^T x \equiv \sum_{e \in E} w_e x_e$

$$\text{subject to} \quad \sum_{e \in \delta(S)} x_e \geq \max_{s_i \in S, t_i \notin S} \{r_i\}, \quad \forall S \subseteq V \quad (1)$$

$$x_e \in \{0, 1\}, \quad \forall e \in E(G) \quad (2)$$

where, as always, by $\delta(v)$ for any vertex v , we mean the edges that have one endpoint in v i.e., $\delta(v) = \{e \in E(G) : e \cap \{v\} \neq \emptyset\}$. This generalizes in a straightforward way for subsets of vertices S such as $\delta(S)$ contains the edges that cross S (i.e., with one endpoint in S and the other in $V \setminus S$).

The intuition of the first set of constraints is fairly simple: in every feasible solution we require that for every subset of vertices S , at least $\max\{r_i\}$ (for $s_i \in S, t_i \notin S$) edges *cross* this set (i.e., have one endpoint in S and the other not in S). This would allow at least $\max\{r_i\}$ paths to cross S . Since we do this for every subset S , by the max-flow/min-cut theorem we guarantee that the paths are disjoint.

As usual, the integrality constraints $x_e \in \{0, 1\}$ are replaced by the $x_e \in [0, 1]$ and the above linear program is solved to optimality in polynomial time using the Ellipsoid method and a separation oracle based on a very simple max flow computation. This allow us to solve the above LP even if it has an exponential number of constraints (one per set $S \subseteq V$, to produce an optimal *fractional* vector $x^* \in [0, 1]^E$. The cost of this vector, i.e. $\text{opt} = \sum_{e \in E} x_e^* w_e$ is a *lower bound* on the cost of the true integral optimal solution. Then, a polyhedral characterization of *any* extreme point solution¹ is given i.e., that every extreme point solution of the bounded polyhedron defined by the above linear program (after we drop the integrality constraints) satisfies the following property: $\exists e \in E$ such that $x_e \geq \frac{1}{2}$. In other words, any basic (extreme point) feasible solution x is always *sparse* (a few number of non-zero entries) and thus must always have at least one variable with relatively large fractional value (in particular greater than $\frac{1}{2}$).

If, for every subset $S \subseteq V$, we define $\max_{s_i \in S, t_i \notin S} \{r_i\} = f(S)$, then the constraint (1) of the above LP becomes $\sum_{e \in \delta(S)} x_e \geq f(S)$. The above characterization of the extreme

¹Extreme point, or basic feasible solution is any point y belonging in the polyhedron defined by the corresponding inequalities such that y *cannot* be written as a convex combination of two other feasible points.

point solutions holds also whenever the function $f(S)$ is *weakly supermodular*: A function $f : 2^V \rightarrow \mathbb{Z}$ is weakly supermodular if $f(\emptyset) = f(V) = 0$ and, $\forall A, B \in 2^V$ one of the following holds:

1. $f(A) + f(B) \leq f(A \cap B) + f(A \cup B)$,
2. $f(A) + f(B) \leq f(A - B) + f(B - A)$.

It can be shown using standard arguments that $f(S) = \max_{s_i \in S, t_i \notin S} \{r_i\}$ is a weakly supermodular function [24]. So, in fact, the result of Jain concerns a more general class of problems, of which one special case happens to be the SND problem when $f(S)$ has the previous form.

Given the characterization of all extreme point solutions x of the polyhedron defined by the relaxed LP introduced above, a 2-approximation algorithm is-almost-immediate: solve the LP to get a fractional optimal vector x^* . Let I be the set of indexes (edges) whose fractional value is “high”: $I = \{e \in E : x_e^* \geq \frac{1}{2}\}$. Include all these edges into the solution (i.e., round their fractional value to 1). Re-define the connectivity requirements of all source-sink pairs $\mathcal{R}$ and resolve the linear program to obtain a new optimal basic feasible solution for the reduced instance. Since each time we round up variables by a factor of at most 2, the cost of the final solution would be at most twice the cost of the optimal fractional LP solution.

1.1 Generalizations of the SND problem

Since the Steiner (Survivable) Network Design problem is so well studied, several generalizations of the SND problem have also been studied in the literature recently. Most common among these is the *Minimum Bounded degree Steiner Network* which is defined as follows: The problem is identical with the standard SND but now we have an additional set of degree constraints for a subset of vertices $W \subseteq V$. In particular, for every vertex $w \in W$, we are given a positive integer number b_w and we require that every feasible solution must have *at most* b_w edges incident on vertex w . Vertices $v \in V \setminus W$ are not constraint. If $x \in \{0, 1\}^E$ is a solution to the LP then, in order x to be feasible, it must satisfy the following constraint: for every vertex $w \in W$, $\sum_{e \in \delta(w)} x_e \leq b_w$, where $b_w \in \mathbb{Z}^+$ is the degree bound of vertex w .

For this generalized problem, a $(2, 2b_v + 3)$ *bi-criteria* algorithm (i.e., a polynomial time algorithm that returns a solution at most twice as costly as the optimal one while the degree of each vertex $v \in W$ is at most $2b_v + 3$) is known for the problem [11] which was further improved to $(2, 2b_v + 2)$ in [15]. In [13] an *additive* (in the second criterion) $(2, b_v + 6r_{\max} + 3)$ ($r_{\max} = \max_{j \in \mathcal{R}} \{r_j\}$) bi-criteria algorithm was given which, although additive in nature, does not directly compare with the previous bi-criteria algorithms. The previous approaches were put under a common unified framework in [14]. In [9], [16], [4] the problem was studied in the case where the input graph G is directed (as opposed to all previous results concerning non-directed graphs).

In [10] the author defined and studied the *budgeted* version of the SND problem which is the standard SND problem complicated with an extra linear constraint $\text{lin} = \ell^T x = \sum_{e \in E} x_e \ell_e \leq L \in \mathbb{Q}^+$. In other words, any feasible solution $x \in \{0, 1\}^E$ should, in addition, satisfy lin . The author notices that a $(4, 4)$ bi-criteria algorithm (4-approximation on the objective function value plus a violation of the packing constraint lin by a factor of at most 4) can be easily derived by a direct application of the famous Caratheodory’s theorem on convex polytopes. Solve the LP defined by (1) and (2) + lin and get a basic fractionally feasible solution x^*

that does not violate the linear budget. By Caratheodory's theorem, x^* can be written as a convex combination of at most two points of the SND polyhedron: $x^* = \lambda y + (1 - \lambda)z$ such that $\lambda \in [0, 1]$ and $y, z \in [0, 1]^E$ are extreme point solutions of the fractional SND polyhedron defined by (1) plus (2). By Jain's result [8] each of y, z has a variable with fractional value greater than half. This means that at least one variable in one of the λy or $(1 - \lambda)z$ has value $\geq \frac{1}{4}$. Apply Jain's algorithm and a (4, 4) bi-criteria algorithm is immediate.

Moreover, in that paper it was shown that for every basic feasible solution x of the generalized LP (that spells out the connectivity requirements plus the linear budget), there is always a coordinate with fractional value greater than $\frac{1}{3}$. This is proved by the following way: a basic feasible solution is obtained for the relaxed LP + lin. Then, the author notices that among all *tight* sets (sets corresponding to the connectivity constraints spelled in (1) that are satisfied with equality by the corresponding bfs) of vertices that define the basic feasible solution, there must be at least one such set S with degree ≤ 3 (i.e., at least three edges cross the corresponding set S). Since the sum of these edges must sum at least to $f(S) = \max_{s_i \in S, t_i \notin S} \{r_i\}$ which is at least 1, we have that there must be an edge crossing S with fractional value $\geq \frac{1}{3}$. This is done by considering only the tight constraints of subsets of vertices defining the obtained basic feasible solution x (by dropping the linear budget). Since there is only one linear budget, this drops the dimension of the vector space defined by these sets by 1 and then a counting argument on the remaining tight sets is done to produce the desired result.

A slight issue regarding the previous approach is the following: what if $f(S) \geq 4, \forall \{s_i, t_i\} \in \mathcal{R}$? The previous results claim that always exists a set of vertices S which is crossed by at most three edges, but if $\exists v \in S$ and $u \notin S$ such that $\{u, v\} \in \mathcal{R}$ then we should have at least 4 edges crossing S (since in any case $f(S) \geq 4$). It turns out that the result of [10] holds when we are allowed to take in our solution several copies of an edge x . i.e $\forall e \in E$ we have an upper bound $u_e \in \mathbb{Z}^+$ indicating how many copies of e we are allowed to use. On the other hand, our results hold even for the case where $u_e = 1, \forall e \in E$.

1.2 Our Contribution

In this article we study the SND problem with an *unbounded* number of extra linear budgets. In particular, we assume that each edge $e \in E$ has a (possibly more than one) color. Each of the colors represents the budget constraints and let $\ell_j(e) \in \mathbb{Q}^+$ be the length that e contributes to the j th budget if e is selected. Let $\mathcal{C} = \{C_j\}$ be the collection of all colors (budgets). For each edge $e \in E$, let $c(e) \subseteq \mathcal{C}$ be the colors of e and let E_j be the edges of color C_j (an edge can belong to multiple E_j s i.e., the E_j s do *not* form a partition of the edge set). Finally, let $f = \max_{e \in E} \{|c(e)|\}$ i.e., f equals to the maximum *frequency* among the edges (how many times an edge appears to the extra budgets). In other words, f is the maximum number of ones in the rows of the constraint matrix modelling the linear budgets. Now, for each color class $C_j \in \mathcal{C}$ we have a linear budget:

$$\ell_j^T x = \sum_{e \in E_j} \ell_j(e)x(e) \leq L_j$$

where $L_j \in \mathbb{Q}^+$ and our task is to find a minimum cost Steiner network that satisfies (1) plus all the linear budgets as above. We also assume that all the costs and lengths satisfy the *triangle inequality*: The input graph instance G is a complete graph and for every three distinct vertices v_1, v_2, v_3 we have that $c(\{v_1, v_2\}) + c(\{v_2, v_3\}) \geq c(\{v_1, v_3\})$. Given an

instance G we define the *metric closure* of G to be a new complete graph G' where the new cost or length of every edge $\{u, v\} \in E(G')$ is the cost of the minimum cost/length $u - v$ path on G . This transformation does not increase the cost of the optimal solution since no edge cost/length increases in G' .

In the following we will use the notation x_e or $x(e)$ interchangeably, to mean the same thing.

We note that the usual token redistribution argument is very unlikely to work for this case, due to the nature of the constraints. Nevertheless, we provide an adaptation of the *fractional* token argument (first introduced in [17]) which seems to better facilitate the structure of the problem and we prove that even in this generalization of the problem considered in [10], a similar claim can be made: for any basic feasible solution to the relaxed LP defined by (2) plus all the linear budgets, there always exists a fractional entry with value $\geq \frac{1}{2+f}$. This gives us the following

Theorem 1. *There exists a polynomial time $(2 + f, 2 + f)$ polynomial time bi-criteria approximation algorithm for the multi-budgeted Survivable Network Design problem where each edge $e \in E$ is part in at most f budgets.*

Of particular interest are two special cases: First, the above result gives us a $(3, 3)$ -approximation algorithm for the case of an unlimited number of budgets where each edge e can be part in at most one constraints (i.e., it has a unique color $c(e) \in \mathcal{C}$). This extends the result of [10] where the same claim is true, but only for a *single* budget. Secondly, the above result gives us a $(4, 4)$ bi-criteria approximation algorithm for the *weighted* Minimum Bounded Degree Survivable Network Design Problem:

Definition 2. *We are given a simple, non-directed graph $G = (V, E)$, a set of connectivity requirements $\mathcal{R}$ as above, a cost function on the edges $w : E \rightarrow \mathbb{Q}^+$, such that each edge $e = \{u, v\}$ has additionally two extra **lengths** $\ell(e, u)$ and $\ell(e, v)$ both in $\mathbb{Q}^+$. We are also given a subset $W \subseteq V$ of the vertices and for each $v \in W$ a positive number $\beta_v \in \mathbb{Q}^+$. The goal is to find a minimum cost subgraph H of G that satisfies all connectivity requirements in $\mathcal{R}$ such that the weighted degree $\sum_{e \in E(H) \cap \delta(v)} \ell(e, v)$ of each vertex v in W is at most β_v .*

If $\ell(e, u) = \ell(e, v) = 1 \ \forall e = \{u, v\}$, then the problem is the classical minimum bounded degree SND which was mentioned above. The problem where $\ell(e, u) = \ell(e, v) = w(e)$ was studied in [3, 2] where a $(2, 7)$ bi-criteria approximation algorithm was proposed. We show how we can obtain a $(4, 4)$ bi-criteria algorithm for the more general case where $\ell(e, u) \neq \ell(e, v) \neq w(e)$ by noticing that this is a special case of the multi-budgeted SND for $f = 2$. Our result is incomparable to the result of Fugunaga & Nagamochi, but it would be probably preferable in scenarios where a larger overflow on the vertex budget constraints cannot be tolerated (on the expense of course on producing potentially costlier networks). The problem was studied in the directed graph setting in [20, 19]. See also [1, 18] for some *on-line* special cases/variants of the problem and see also [22] for other potential case in relevant scenarios.

As it is the case with all generalization of SND problem [11, 15, 13, 10, 3], our algorithm relies on existing tools developed originally by Jain, although the technical details provide extra challenge. On the other hand, since the usual approach (employed for example in [10]) seems that it cannot give us any meaningful result (because with k budgets the dimension of the vector space defined by the remaining sets after dropping these budgets could drop

by as much as k), our contribution is to show how we can employ a particular *fractional* charging scheme that let us prove our result for a generalization of the problem defines in [10] and which the author leaves as open problem. Also, we note that an application of Caratheodory's theorem in our case does not yield any meaningful insight on the fractional values of the extreme point solutions.

1.3 Applications

In this subsection we will try to highlight few potential applications of the proposed studied case to some real world settings.

Cloud Computing: Imagine a collection of services $S = \{s_1, \dots, s_n\}$ and a multi-cloud system consisting of a collection of independent terminals $T = \{t_1, \dots, t_q\}$. Services in S want to connect to terminals in T through a backbone network consisting of intermediate nodes. Each service $s_i \in S$ can be internally split into p_i independent computational units. For example consider the following CAD application $s_i \in S$ for movie animation: The application is split into three autonomous computational tasks: one task for rendering and requires intensive computational GPU use, the second task requires large amount of storage to store intermediate CAD documents and the third task requires typical user interface and visualization resources. This application would like to connect these 3 tasks to the multi-cloud. Due to heavy computational power, the service would like to minimize delays due to possible failures or congestions to some links/channels. So, preferably, the service would like to establish 3 separate and independent channels to 3 different clouds $t_{i_1}, t_{i_2}, t_{i_3}$ all in T . In other words, the service would like to compute 3 edge disjoint paths from the source to three different clouds.

On the other hand, the service would like to guarantee a minimum level of Quality of Service with respect to the cost and the delay (latency). For every link e service s_i uses, it has to pay $w(e) \in \mathbb{Q}^+$ units of currency. Service s_i has a *budget* $\beta_i \in \mathbb{Q}^+$ on how much currency is able to spend. So, s_i imposes the following constraint on the edges (links) of the underlying backbone system, let's call it G , it's able to use to connect to the cloud:

$$\sum_{e \in E(G)} w(e) \cdot x_{ie} \leq \beta_i,$$

with the intentional meaning of x_{ie} be the following: $x_{ie} = 1$ if service $s_i \in S$ "rents" link $e \in E(G)$ for use, otherwise $x_{ie} = 0$. Similar situation applies to the potential delay: each service s_i has a deadline $\gamma_i \in \mathbb{Q}^+$ i.e., it would like it's computational task to be finished after γ_i units of time. Using link $e \in E(G)$ imposes a delay of $d(e) \in \mathbb{Q}^+$. Similar to the previous, s_i would like the links used to connect its tasks to the cloud not to impose delay more than γ_i units thus the linear constraint:

$$\sum_{e \in E(G)} d(e) \cdot x_{ie} \leq \gamma_i.$$

To make the situation even more complicated, each *internal backbone proxy node* $v \in V(G)$ has an upper bound on connections it can accept and forward. Each connection has a priority and a workload and depends on the channel it uses (if the channel for example is slow or fast), i.e., time it needs to be processed and go through a node v . Let $l(e)$ be the workload of any task coming through edge e incident to v . Node v can process at most $\zeta(v)$ workload of

connections at a given snapshot of time, otherwise it fails (crashes from overload). So, node v imposes the following linear budget constraint to the system:

$$\sum_{e \in \delta(v)} \sum_{s_i \in S} l(e) \cdot x_{ie} \leq \zeta(v), \quad \forall v \in V(G),$$

where as usual, $\delta(v)$ is the set of edges incident to vertex v . Observe that the double sum is in fact a linear constraint.

The above scenario can be in fact modelled in a straightforward way and solved as the Multi-Budgeted Survivable Network Design problem, where we have all the standard connectivity requirements plus the extra linear budgets defined above for every service and internal node of the system.

Optical Computing: In modern optical fiber network systems, we encode the information as an electromagnetic signal and we transfer it through the optical fiber as a beam of light in a specified frequency. Typically, at most one beam of light is allowed to travel through the fiber at any given time. In WDM² optical networks we allow *multiplexing* of a number of different light beams to travel simultaneously through the optical fiber as follows: We partition the electromagnetic spectrum into a number of k non-overlapping intervals. For each interval f_i we have an upper bound w on how many different beams of light that have frequencies within this interval can travel at the same time through the optical carrier. This constraint is imposed since, otherwise, we would have quantum phenomena as *interference* of the light beams of a given interval. Naturally, if we allow a large number of beams of light with frequencies within a small interval to travel through the optical fiber, we can expect with very high probability two or more beams to be interfered.

In the *Optical* version of the SND problem, the underlying backbone network is in fact an optical network. We can easily model the above requirement on how many different beams of light can travel through a channel but simply giving a unique color to every link (the unique color corresponds to a single budget constraint) and add, for each color class C_i a single budget:

$$\sum_{e \text{ has color } C} x_e \leq w$$

and this guarantees that any feasible solution should respect this bound imposed by optical computing technologies. the intentional meaning of the indicator variable x_e is the following: if we choose to use link e then $x_e = 1$ otherwise $x_e = 0$.

2 Characterization of Extreme Point Solutions

We consider the polyhedron that includes all the feasible points (corresponding to feasible solutions) for the multi-budgeted version of SND problem:

$$\mathcal{P}^I = \left\{ x \in \{0, 1\}^E : \sum_{e \in \delta(S)} x_e \geq f(S), \forall S \subseteq V \text{ and } \sum_{e \in E_j} \ell_j(e) x_e \leq L_j, \forall C_j \in \mathcal{C} \right\}$$

²WDM stands for Wavelength-Division Multiplexing

where as usual for all $S \subseteq V$, $f(S) = \max_{s_i \in S, t_i \notin S} \{r_i\}$. We drop the integrality constraints on the variables in the above formulation and this defines the *fractional* polyhedron $\mathcal{P}_{SND-b}^f$ which from now on we will call it simply $\mathcal{P}_f$ to emphasize we are talking about the fractional polyhedron of the budgeted Survivable Network Design problem. This polyhedron contains all the fractionally feasible vectors.

Before we continue, we start with some essential definitions:

Definition 3. Let $E' \subseteq E$ be a subset of the edges of a graph. Then, we define the **characteristic vector** of E' to be the binary vector $\chi_{E'} \in \{0, 1\}^E$ such that $\chi_{E'}(e) = 1 \Leftrightarrow e \in E'$ i.e. the i -th component of $\chi_{E'}$ is 1, if the i -th edge belongs to E' and zero otherwise.

Definition 4. Let y be a real-valued vector in a n -dimensional space. Define the **support** of y to be the set of indices of all the non-zero components of y i.e. $\text{supp}(y) = \{i \in [n] : y_i \neq 0\}$.

Definition 5. A family $\mathcal{L}$ of subsets of some universe of elements U is called **laminar** if it is not intersecting i.e. for any two subsets $L_1, L_2 \in \mathcal{L}$ it is not the case that all of the $L_1 \setminus L_2$, $L_2 \setminus L_1$ and $L_1 \cap L_2$ are non empty.

Also we say that a given linear constraint is *tight* if it is satisfied by a particular solution vector with equality. Our study begins with a characterization of all extreme point (or basic) solutions of $\mathcal{P}_f$ (which carry over to the integral polytope as well):

Lemma 1. Let the connectivity requirement function $f(S) = \max_{s_i \in S, t_i \notin S} \{r_i\}$ be weakly super-modular and let x be an extreme point solution of $\mathcal{P}_f$ such that $0 < x_e < 1$ for all $e \in E$. Then, there exists a laminar family $\mathcal{L} \subseteq 2^V$, and a set $\mathcal{Q} \subseteq \mathcal{C}$ such that:

1. $\sum_{e \in \delta(S)} x_e = f(S)$, $\forall S \in \mathcal{L}$,
2. $\sum_{e \in E_j} \ell_j(e) x_e = L_j$, $\forall C_j \in \mathcal{Q}$,
3. the characteristic vectors of $\chi_{\delta(S)}$ for $S \in \mathcal{L}$ and χ_{C_j} for $C_j \in \mathcal{Q}$ are all linearly independent, and
4. $|\mathcal{L}| + |\mathcal{Q}| = |E| = |\text{supp}(x)|$.

This Lemma is proved by standard properties of basic feasible solutions [21] of polytopes defined by linear (in)equalities: indeed we can form a basic feasible solution by selecting $|E|$ linearly independent constraints, set them to equality and solve the corresponding linear system. The last item of the lemma simply says that the number of non-zero variables is the number of linearly independent constraints set to equality. The fact that we can take the family of tight constraints of subsets of vertices $\mathcal{L}$ to be laminar, follows from standard un-crossing arguments [8], [12].

Armed with Lemma 1, we now prove the main theorem of this paper that asserts that for any given bfs of $\mathcal{P}_f$ there exists variable with “high” fractional value:

Theorem 2. Take any basic feasible solution (a vertex) of the polyhedron $\mathcal{P}_f$ such that the connectivity requirement function is weakly super-modular and such that each edge appears in at most f budget constraints. Then there always exists a variable e such that $x_e \geq \frac{1}{2+f}$.

Proof. The proof will be given by contradiction. In particular, assume we have a basic feasible solution $x \in (0, 1)^E$ of $\mathcal{P}_f$ and for all $e \in \text{supp}(x)$ we have that $x_e < \frac{1}{2+f}$. We will employ a fractional charging scheme which works as follows: Initially each non-zero edge e is given a positive charge of value at most 1. Then, this edge will redistribute fractionally this charge to sets $S \in \mathcal{L}$ and $C_j \in \mathcal{Q}$, where $C_j = \{c(e)\}$, in such a way that at the end each tight set $S \in \mathcal{L}$ and $C_j \in \mathcal{Q}$ will receive charge at least one for a total receiving charge of at least $|\mathcal{L}| + |\mathcal{Q}| = |E|$. On the other hand, we will show that the total charge distributed must be *strictly* less than $|E|$, which will give the desired contradiction.

For each edge $e \in \text{supp}(x)$, with endpoints u, v , the distribution scheme we carry out is the following:

1. Edge e gives charge of x_e to each of the color classes $C_j \in \{c(e)\}$ only if $C_j \in \mathcal{Q}$.
2. For each endpoint u of e , edge e assigns charge of x_e to the *minimal* set (inclusion-wise) $S \in \mathcal{L}$ that contain u , if such set exists and
3. Assigns charge $1 - (2+f)x_e$ to the minimal set $S \in \mathcal{L}$ that contains *both* of its endpoints, if such a set exists.

Note that by the hypothesis, $x_e < \frac{1}{2+f} \Rightarrow 1 - (2+f)x_e > 0$ so, for every edge, each charge distributed is strictly positive. The total charge distributed per edge is at most $1 - (2+f) \cdot x_e + 2x_e + f \cdot x_e = 1$. We first show that it is not the case that all edges can spend charge of exactly 1:

Claim 1. *There exists at least one edge $e \in \text{supp}(x)$ such that the charge that e distributes according to the charging scheme is strictly less than 1.*

Proof of Claim: There exists a natural way to represent the laminar family $\mathcal{L}$ as a *forest* which we will also use in the following: we say that $S_i \in \mathcal{L}$ is *child* of $S \in \mathcal{L}$ if S is the smallest (minimal) set in $\mathcal{L}$ containing S_i . In other words S_i is a child of S if there is no other set $S' \subseteq S$ such that $S_i \in S'$. Such S is unique. Symmetrically, S is the *parent* of S_i . Any parentless set $S \in \mathcal{S}$ is called *root* of the subtree rooted at S . And childless sets S are called *leaves*. The subtree rooted at S contains all the sets that are descendants (the transitive, reflexive closure of the relation *child*) of S . Any root $R_i \in \mathcal{L}$ must contain, by definition, edges that cross R_i , i.e., $\delta(R_i) \neq \emptyset$ because $\sum_{e \in \delta(R_i)} x_e = f(R_i)$ and if $f(R_i)$ was equal to zero, then the vector $\chi_{R_i} = 0^E$ is not linearly independent, a contradiction. So, there must be edges crossing any root R_i . Each such edge $e \in \delta_{R_i}$ wastes a charge of $1 - (2+f)x_e > 0$ because it is not the case that both endpoints of such edge belong to the same set $S \in \mathcal{L}$. $\square$

So, there are edges that give charge strictly less than 1, for a total assigned charge of strictly less than $|E| = |\text{supp}(x)|$. In the following we will show how each tight object $S \in \mathcal{L}$ and $C_j \in \mathcal{Q}$ receives charge of at least one for a total receiving charge of at least $|\mathcal{L}| + |\mathcal{Q}| = |E|$, which will be a contradiction of the hypothesis of the lemma that such a charging scheme exists, which is implied by the hypothesis that $\exists e \in \text{supp}(x) : x_e \geq 1/(2+f)$.

We begin with the case of the color tight classes $C_j \in \mathcal{Q}$, which is (relatively) easier: each such class C_j will receive a charge of x_e from every edge e such that $C_j \in c(e) (\Leftrightarrow e \in E_j)$, for a total charge of $\sum_{e \in E_j} x_e$. We must show that this quantity is ≥ 1 , which is not immediately obvious:

Claim 2. Let $C_j \in \mathcal{Q}$ be a color budget constraint that is met with equality by the basic feasible solution $x \in \mathcal{P}_f$ i.e., $\sum_{e \in E_j} x_e \ell_j(e) = L_j$. Then $\ell_j(e) \leq L_j \forall e \in E_j$.

Proof of Claim: We can easily see that it is not the case that $L_j = 0$, since otherwise we would have that $\chi_{C_j} = 0^E$ which is not linearly independent. So, $L_j > 0$ and since the constraint is met with equality we know that $\exists e \in E_j$ such that $x_e > 0$. On the other hand, we see that $\ell_j(e) \leq L_j, \forall e \in E_j$. Indeed, assume not, i.e., $\exists e \in E$ such that $\ell_j(e) > L_j$: then any feasible *integral* solution must have $x_e = 0, \forall e \in E_j$, and in such a case we can drop the constraint after we delete all such edges of color C_j . In other words, if $\ell_j(e) > L_j$ for some edge of color C_j , delete the edge from the graph without any loss. All the remaining edges satisfy the desired property that $\ell_j(e) \leq L_j \forall e \in E_j$, for all color classes $C_j \in \mathcal{Q}$. Take such a constraint:

$$\sum_{e \in E_j} x_e \ell_j(e) = L_j \implies \sum_{e \in E_j} \underbrace{\left\lfloor \frac{\ell_j(e)}{L_j} \right\rfloor}_{\leq 1} x_e = 1 \implies \sum_{e \in E_j} x_e \geq 1$$

where the last inequality follows by an immediate averaging argument and because all quantities involved are positive. $\square$

The previous claim is enough to prove that the charge received by any color class $C_j \in \mathcal{Q}$ is at least 1.

We now continue to prove that also any set $S \in \mathcal{L}$ receives charge at least 1. As a first case, consider the scenario where $S \in \mathcal{L}$ is a leaf of some tree in the tree-representation of $\mathcal{L}$. Since $S \in \mathcal{L}$, it is the case that $\sum_{e \in \delta(S)} x_e = f(S)$ and by definition $f(S)$ is an integer at least 1. This set will receive charge from all the crossing edges $e \in \delta(S)$ for a total charge $\sum_{e \in \delta(S)} x_e = f(S) \geq 1$ (remember that edge e with one endpoint on S will give S charge x_e if S is the smallest set containing one endpoint of e . Since S is a leaf, it is in fact the smallest set that contains one endpoint of all the crossing edges).

Now assume that $S \in \mathcal{L}$ is not a leaf and that it has q children $Z_1, Z_2, \dots, Z_q$ all of which are in $\mathcal{L}$ (by definition). This means that $\sum_{e \in \delta(Z_i)} x_e = f(Z_i)$, a positive integer and the same is of course true for S . The charge received by S comes from different sets of edges (whose indices are in the support of x):

1. Let A be the set of edges in $\text{supp}(x)$ that cross between two children of S , i.e., $A = \{e = \{u, v\} \in E : u \in Z_i \wedge v \in Z_j\}$ for two distinct $i, j \in [q]$. Since S is the smallest (minimal) set $S \in \mathcal{L}$ that contains *both* endpoints of all these edges in A , each such edge will assign charge of $1 - (2 + f)x_e$ to S , for a total charge of

$$\sum_{e \in A} (1 - (2 + f)x_e) = |A| - (2 + f) \sum_{e \in A} x_e.$$

2. Let $B = \{e = \{u, v\} \in E : u \in Z_i \wedge v \in S\}$, i.e., B contains all edges with one endpoint on one of S 's children and the other endpoint in S (not in any other child of S). By the charging scheme, these edges will give to S charge of x_e (one endpoint of e is "owned"

by S and thus charge of x_e is received) plus a charge of $1 - (2 + f)x_e$ because S is the minimal set containing both of e 's endpoints for a total charge of

$$\sum_{e \in B} \left(1 - (1 + f)x_e\right) = |B| - (1 + f) \sum_{e \in B} x_e.$$

3. Let $\Gamma = \{e \in \delta(S) : e \cap \{S \setminus \cup_{i \in q} Z_i\} \neq \emptyset\}$. In other words, Γ contains all edges that cross S with one endpoint in $S \setminus \cup_{i \in q} Z_i$ (not in any of its children) and the other endpoint outside S (since it is a crossing edge). Each of these edges $e \in \Gamma$ give S a charge of x_e (because S is the smallest set that contains exactly one endpoint of e) for a total charge of $\sum_{e \in \Gamma} x_e$.
4. Let Δ be the rest of edges, i.e., $\Delta = \{\cup_{i \in [q]} \{\delta(Z_i)\}\}$ such that $e = \{u, v\} \cap \{S \setminus \cup_{i \in q} Z_i\} = \emptyset$, i.e., Δ contains all edges that cross Z_j and S at same time. These edges do not assign any charge to S .

We note that if $|A| = |B| = |\Gamma| = 0$, then $\chi_S = \sum_{i \in [q]} \chi_{Z_i}$, i.e., the characteristic vectors $\chi_S, \chi_{Z_1}, \dots, \chi_{Z_q}$ of $S, Z_1, \dots, Z_q$ respectively are *not* linearly independent, a contradiction. So at least one of A, B, Γ must be nonempty and so the charge received from S is strictly greater than zero.

By the previous four cases, we see that S receives a total charge of

$$\begin{aligned} \Omega &= |A| - (2 + f) \sum_{e \in A} x_e + |B| - (1 + f) \sum_{e \in B} x_e + \sum_{e \in \Gamma} x_e \\ &= |A| + |B| - f \cdot \left(\sum_{e \in A} x_e + \sum_{e \in B} x_e \right) + \underbrace{\left(\sum_{e \in \Gamma} x_e - 2 \sum_{e \in A} x_e - \sum_{e \in B} x_e \right)}_{f(S) - \sum_{i \in [q]} f(Z_i) = \Psi \in \mathbb{Z}} > 0. \end{aligned}$$

We want to show that the above expression (which we already know is positive) is in fact at least 1. Lets call it Ω and lets call the quantity $f(S) - \sum_{i \in [q]} f(Z_i)$ as Ψ . Note that Ψ , although obviously an integer as addition and subtraction of integer quantities, might not always be positive. Finally, let $\Xi = \Omega - \Psi = |A| + |B| - f(\sum_{e \in A} x_e + \sum_{e \in B} x_e)$.

Claim 3. *If $|A| + |B| = 0$, then the charge received by S is at least 1.*

Proof. The fact that we have no edges in A or in B means that either (1) S has no children and so the charge received is $\sum_{e \in \Gamma} x_e = f(S) \geq 1$ or (2) it has $q \geq 1$ children $Z_1, Z_q \in \mathcal{L}$ but all the edges in $\delta(Z_j), \forall j \in [q]$ are in fact in Δ (i.e., crossing at the same time S and Z_j). So,

$$\sum_{Z_j} f(Z_j) = \sum_{e \in \Delta} x_e < f(S)$$

which means that

$$f(S) = \sum_{e \in \Gamma} x_e + \sum_{e \in \Delta} x_e \tag{3}$$

$$\Rightarrow \sum_{e \in \Gamma} x_e = f(S) - \sum_{Z_j} f(Z_j). \tag{4}$$

But $\Gamma \neq 0$ since, as we have already discussed above, the characteristic vectors of χ_S and χ_{Z_j} are not linearly independent i.e., $\chi_S = \sum_{j \in [q]} \chi_{Z_j}$. So, $\sum_{e \in \Gamma} x_e = f(S) - \sum_{Z_j} f(Z_j) > 0$ but the last expression involves only integers and thus it is in fact ≥ 1 . The fact that in this case $\Omega = \sum_{e \in \Gamma} x_e$ concludes the case that the charge received by S is at least one. $\square$

So we assume that $|A| + |B| \geq 1$. There are several subcases to consider regarding the value of $f(S) - \sum_{i \in [q]} f(Z_i) = \Psi$.

Case 1: $\Psi > 0$

This actually means (since Ψ is an integer) that $\Psi \geq 1 \implies \Omega \geq 1$ (in fact $\Omega > 1$, but for our purposes the inequality is enough). This follows also from the fact that

$$\begin{aligned} |A| + |B| - f \cdot \left(\sum_{e \in A} x_e + \sum_{e \in B} x_e \right) &> |A| + |B| - f \cdot \left(|A| \cdot \frac{1}{2+f} + |B| \cdot \frac{1}{2+f} \right) \\ &= |A| + |B| - \frac{f}{2+f} (|A| + |B|) \\ &= \frac{2}{2+f} (|A| + |B|) \\ &> 0 \end{aligned}$$

where in the first inequality we used the fact that $x_e < \frac{1}{2+f}$, $\forall e \in A, B$ and the last inequality the fact that $|A| + |B| > 0$ and this proves that the charge received by S in this case is at least 1, and so we are done.

Case 2: $\Psi = 0$

By the definition of Ψ this means that $f(S) = \sum_{j \in [q]} f(Z_j)$, where $Z_j, j \in [q]$ are the q children of S in the tree representation of the laminar family $\mathcal{L}$. Obviously we must have that $q \geq 1$.

We will first handle the case where the number of children of S in $\mathcal{L}$ is at least 2 i.e., $q \geq 2$.

Lemma 2. *Assume that $f(S) = \sum_{j \in [q]} f(Z_j)$ and $q \geq 2$. Then the charge received by S by our charging scheme is at least one.*

Proof. Observe that in this case we immediately conclude that $f(Z_j) \leq f(S) - 1$ for all $j \in [q]$, since $f(S) > 0$ for all $S \in \mathcal{L}$. From the definition of the connectivity function $f(S)$ we have that $f(S) = \max_{s_i \in S, t_i \notin S} \{r_i\}$ from which we conclude that s_i cannot belong to any of the children Z_j of S . Assume that this is not the case, i.e., $\exists j \in [q]$ such that $s_i \in Z_j$. Then, since $t_i \notin S$ and $Z_i \subset S$, we have that $f(Z_j) = f(S)$ a contradiction. So $s_i \in S \setminus \cup_{j \in [q]} Z_j$. This further means that

$$\sum_{e \in \delta(S) \setminus \cup_{j \in [q]} \delta(Z_j)} x_e = f(s) \geq 1.$$

The edges in $\delta(S) \setminus \cup_{j \in [q]} \delta(Z_j)$ are precisely the edges in Γ , and these edges distribute a charge of $\sum_{e \in \Gamma} x_e$ to S which, from the above, is greater or equal than 1 and so in this case we are done. $\square$

We will now handle the more complicated case where $q = 1$. Let Z be the unique child of S in the laminar family $\mathcal{L}$. Given that $\Psi = 0$ we have that $f(S) = f(Z)$. As before let $\{s, t\} : s \in S$ and $t \notin S$ the pair of vertices responsible for the connectivity requirement $f(S)$ and similarly let $\{s_Z, t_Z\} : s_Z \in Z$ and $t_Z \notin Z$ the pair of vertices responsible for the connectivity requirement $f(Z)$ of the child Z . We will make a series of observations regarding the structure of this case:

Case 1: $s_i \in S \setminus Z$. Observe that in this case, in direct analogy with the previous discussion, we have that

$$\sum_{e \in \delta(S) \setminus \delta(Z)} x_e = \sum_{e \in \Gamma} x_e = f(S) \geq 1.$$

And so the charge distributed to S in this case is at least 1, so we are done.

Case 2: $s_i \in Z$. In that case, observe that if t_Z is in S (in fact in $S \setminus Z$, due to the definition of $f(Z)$), then the charge received by S is at least $f(Z) \geq 1$ because each edge e that crosses Z has the other endpoint in $S \setminus Z$ and it contributes charge at least x_e to S (they are edges in B) for a total charge at least $f(Z) \geq 1$ and we are done.

So we have that t_Z should be outside S . In the following we will assume that the sets of edges A, B, Γ are restricted in the sub-instance induced by S and its unique child Z . Let $\bar{S} \subseteq S$ be the set of *vertices* in $S \setminus Z$ which are adjacent to vertices in Z through edges in B . First, we observe that $\bar{S}$ cannot be the empty set. If so,.... So, $\bar{S} \neq \emptyset$. We can immediately see that if $\sum_{e \in B} x_e \geq 1$ then we are done: the charge received by S is at least 1. Also, it is not hard to observe that

$$\sum_{e \in B} x_e = \sum_{e \in \Gamma} x_e < 1.$$

In other words, the sum of the fractional values of the edges that cross Z but not S should be equal with the sum of the fractional value of the edges that cross S but not Z . Indeed, we have that

$$\begin{aligned} f(Z) &= \sum_{e \in \Delta} x_e + \sum_{e \in B} x_e, \text{ and} \\ f(S) &= \sum_{e \in \Gamma} x_e + \sum_{e \in \Delta} x_e \\ \Rightarrow \sum_{e \in B} x_e &= \sum_{e \in \Gamma} x_e. \end{aligned}$$

In the following, we will describe a procedure that basically “shortcuts” and avoids all the vertices in $\bar{S}$. The main purpose of the process is to eliminate vertices in $\bar{S}$ through short-cutting taking advantage of the metric properties of the weight and length functions. The idea of the procedure is the following: Let B_Z be the set of vertices in Z that are adjacent to at least one vertex in $\bar{S}$ through edges in B . In other words, let

$$B_Z = \left\{ v \in Z : \exists e = \{u, v\} \in B \text{ such that } v \in e \right\}.$$

The other endpoint u of e is in $\bar{S}$. Similarly, let Γ_S be the set of vertices *outside* S that are adjacent to vertices in $\bar{S}$ through edges in Γ :

$$\Gamma_S = \left\{ v \in V \setminus \bar{S} = S \setminus Z : \exists e = \{u, v\} \in \Gamma \text{ such that } v \in e, u \in V \bar{S} \right\}.$$

We start with any vertex, let's say v , in $\bar{S}$. This vertex is adjacent, through edges in B , to some vertices in Z , through some edges in A to some other vertices in $\bar{S}$ and through some edges in Γ to some vertices in Γ_S . Let $B(v), A(v), \Gamma(v)$ be the corresponding set of edges. In more detail, the procedure is the following:

We have a bin $\beta(v)$ of all edges incident to v . We partition this set of edges to two sets: $B(v)$ and $A(v) \cup \Gamma(v)$. From the above discussion we know that

$$\sum_{e \in B(v)} x_e = \sum_{e \in A(v) \cup \Gamma(v)} x_e.$$

For each 2-edge path (e_1, e_2) such that $e_1 = \{v, u_1\}$ with $u_1 \in Z$ (i.e., $e_1 \in B(v)$) and $e_2 = \{v, u_2\}$ such that $u_2 \in \{\bar{S}, V \setminus S\}$ (i.e., $e_2 \in \{A(v), \Gamma(v)\}$) we insert the new edge $e_{1,2} = \{u_1, u_2\}$ with fractional value $x_{e_{1,2}} = \min\{x_{e_1}, x_{e_2}\}$ and we subtract $x_{e_{1,2}}$ from x_{e_1} and x_{e_2} . If edge $e_{1,2}$ already exists then we simply update its fractional value by adding $\min\{x_{e_1}, x_{e_2}\}$. We will show that $x_{e_{1,2}}$ never becomes overloaded. By this “short-cutting” at least one of the edges e_1, e_2 adjacent to v will be removed. We iterate as long as $\beta(v)$ is not empty. We repeat for all $v \in \bar{S}$.

Figure 1: An example of the process of eliminating the vertices in $S \setminus Z$: In the first step we eliminate v by short-cutting the edges $\{u_1, v\}, \{v, u_2\}$ with the inclusion of the edge $\{u_1, u_2\}$. In other words, without any loss, we can remove from our solution the two corresponding edges and add the edge $\{u_1, u_2\}$, thus avoiding vertex v . After that, we do the same on the vertex u_2 with the inclusion of the edge $\{u_1, u_3\}$ i.e., delete edges $\{u_1, u_2\}$ and $\{u_2, u_3\}$ and include the edge $\{u_1, u_3\}$. If this edge exists, we update its fractional value appropriately. We show that under no circumstances $x_{\{u_1, u_3\}}$ becomes overloaded (value greater than 1). Observe that even if there is a new set $S' \in \mathcal{L}$ that has Z as its unique child, and even if we need to short-cut $\{u_1, u_3\}$, the invariant property guarantees that still new edge won't be overloaded.

Throughout the process we preserve the following invariant:

Invariant: The property we want to preserve is that *the sum of the fractional values of the edges adjacent to vertices $u_1 \in B_Z$ and to vertices $u_2 \in \Gamma_S$ is preserved throughout the process.*

Since at each step we delete an edge incident to v (for all $v \in \bar{S}$), the procedure will terminate after $|\delta(v)|$ steps, for each such v . At the end of this procedure v will not have any edges incident to it, so $\sum_{e \in \delta(v)} x_e = 0$, for each $v \in \bar{S}$. To show that the procedure works, we need to show that it preserves the invariant property.

Claim 4. *Take any vertex $v \in \bar{S}$. Then for v we have that*

$$\sum_{e \in \delta(v) \cap B} x_e = \sum_{e \in \delta(v) \setminus B} x_e$$

In other words, for each $v \in \bar{S}$, the sum of the fractional values of the edges that go from v to vertices in Z is the same with the sum of fractional values of the variable corresponding to edges from v to either other vertices in $\bar{S}$ or to vertices in Γ_S .

Proof. This is a very easy, but equally important, property of the vertices of $\bar{S}$. To prove the statement of the lemma, let $x_B(v) = \sum_{e \in \delta(v) \cap B} x_e$ and simply consider $Z' := Z \cup \{v\}$ and noticing that $f(Z) = f(Z')$. Indeed, take $Z \cup \{v\}$ for such $v \in \bar{S}$. Since $f(Z) = f(Z \cup \{v\})$ we have that

$$\sum_{e \in \delta(Z \cup \{v\})} x_e = \sum_{e \in \delta(Z)} x_e.$$

The two previous sums are identical except on the two terms $\sum_{e \in B \cap \delta(v)} x_e$ and $\sum_{e \in \delta(v) \setminus B} x_e$. Since they appear on the two different sums they should be same since the other terms cancel each other out $\square$

The previous claim has the following important implication: given $v \in \bar{S}$, the procedure of eliminating v will end. In other words, we cannot end up in a situation that v has a single lonely edge adjacent to it that we cannot eliminate. At the final step of our procedure (for v), there would be adjacent to it two edges with identical fractional values: one with the other endpoint in Z and the other edge with the other endpoint in $\bar{S} \cup \Gamma_S$ and the final short-cutting step would eliminate both of them at same time.

Claim 5. *The sum of the values of the fractional variables/edges incident to any $u \in \{B_Z, \Gamma_S\}$ is preserved.*

Proof. Take such a vertex $u \in \{B_Z, \Gamma_S\}$. Since the two cases are identical, we will show the claim for $u \in B_Z$. We need to show that $\sum_{e \in \delta(u)} x_e$ does not change after the termination of the procedure with respect to its initial value. First of all, we notice that in the case that $\min\{x_{e_1}, x_{e_2}\} = x_{e_1}$ (remember that e_1 is the edge incident to u) then the sum $\sum_{e \in \delta(u)} x_e = x(v)$ after the deletion of e_1 and the addition of e_{12} remains unchanged, so we are fine.

The slightly more complicated case arises when $\min\{x_{e_1}, x_{e_2}\} = x_{e_2}$. In that case, the edge e_1 will remain in the solution with new updated value $x_{e_1} - x_{e_2}$. But, incident to vertex $u \in B_Z$ will be the new edge e_{12} with value $x_{e_{12}} = x_{e_2}$ (or, in case e_{12} already exists, with updated value $x_{e_{12}} := x_{e_{12}} + x_{e_2}$). So, since at this step the values of the other variables

of the incident edges to u remain unchanged, we have that the new value of $x(v)$ would be $x_{e_1} - x_{e_2} + x_{e_2} = x_{e_1}$ i.e., same as the previous step.

With identical argumentation we have that $\sum_{e \in \delta(u)} x_e = x(v)$ for $v \in \Gamma_S$ is preserved throughout the process. $\square$

The same is true if, after a single update step, $u_2 \in \bar{S}$: after the update step $x(u_2)$ is preserved. In other words, after the step of elimination of a vertex v , $x(u)$ is preserved for all u adjacent to v through edges in A, B or Γ .

Claim 6. *The new solution is feasible with respect to both the connectivity requirement function $f(\cdot)$ and the additional budget constraints. Moreover, the cost of the new solution produced is not greater than the original one.*

Proof. The fact that the new solution respects the connectivity requirement function f comes from the previous claim since all vertices $v \in \{B_Z, \Gamma_S\}$ affected by the procedure have their $x(v)$ value ($= \sum_{e \in \delta(v)} x_e$) unchanged. The fact that all vertices in $\bar{S}$ are eliminated does not affect any connectivity requirement function since, as we have already seen, no vertex in $\bar{S}$ is part of any such requirement pair (otherwise we would be done).

For the budget and cost functions, the claim follows directly from the metric properties of costs/lengths: each time we short-cut (i.e., replace) e_1, e_2 with e_{12} we know that $c(e_{12}) \leq c(e_1) + c(e_2)$ and the same for each length. So, since we have packing constraints, none of the lengths increase and the same is of course true for the cost. So, the new solution respects all the length budgets and has cost at most the cost of the initial solution, so we are done. $\square$

Claim 7. *After the termination of the above procedure, no edge is “overloaded” i.e., no edge such that its corresponding variable has value ≥ 1 .*

Proof. In order an edge $e = \{u_1, u_2\}$ to get overloaded, it must be the case that due the course of our procedure we repeatedly add the corresponding edge through different short-cuts. Observe that any such edge should be in Δ i.e., one of its endpoints should be in Z and the other outside S . The worst case scenario is when edge e is already in the solution with fractional value $x_e \leq \frac{1}{2+f}$ and all possible shortcuts add to that edge as well. In other words, the worst possible case is when the sum of *all* fractional edges in B sums up to a single edge e and that edge already exists. We have already observed that $\sum_{e' \in B} x_{e'} < 1$, otherwise we are done (charge received by S would be greater or equal to 1).

Now assume that $\sum_{e' \in B} x_{e'} > \frac{f+1}{f+2}$ such that $x_e + \sum_{e' \in B} x_{e'} > 1$. Then observe that the charge that S would receive is

$$\sum_{e \in B} \left(1 - (1+f)x_e\right) = |B| - (1+f) \sum_{e \in B} x_e > |B| - (1+f).$$

So, the charge is strictly greater than $|B| - 1 - f$. Assume that this quantity is strictly less than one such that $|B| < 2+f \Rightarrow |B| \leq 1+f$. Then, since we have at most $1+f$ edges summing up to at least $\frac{1+f}{2+f}$ (and strictly less than 1), then, by an immediate averaging argument, there must be an edge with fractional value at least $\frac{1+f}{(2+f)(1+f)} = \frac{1}{2+f}$, a contradiction by our hypothesis that all fractional variables have value less than $\frac{1}{2+f}$.

So, $\sum_{e' \in B} x_{e'} < \frac{1+f}{2+f}$ from which we have that the maximum value an edge can get after the termination of our procedure for all $v \in \bar{S}$ is

$$\sum_{e' \in B} x_{e'} + x_e < \frac{1+f}{2+f} + \frac{1}{2+f} = 1$$

and we are done with the proof of the claim. $\square$

Another interesting property of the above simple short-cutting procedure is that it is “invertible”. By this we mean the following: Let $\mathcal{S}$ is the subset of edges added through the course of the procedure for a particular set $S \in \mathcal{L}$ with the above mentioned properties. Observe that an edge e from B_Z to Γ_S might be added several times in $\mathcal{S}$. We keep each copy separately. Each time we “short-cut” 2 edges $\{u_1, v\}, \{v, u_2\}$, where $u_1 \in B_Z$ and $u_2 \in \bar{S} \cup \Gamma_S$, we update $\mathcal{S}$ by inserting a new copy of $\{u_1, u_2\}$ to it together with a record of its “middle-point” $v \in \bar{S}$ and a record of its fractional value. It might help to visualize $\mathcal{S}$ as a stack and we push $e = \{u_1, u_2\}$ in $\mathcal{S}$ with a record of its middle-point and its value. In this way, anytime we have an edge $\{v_1, v_2\} \in \mathcal{S}$ with $v_1 \in B_Z$ and $v_2 \in \Gamma_S$ we know exactly how we ended up with this edge. Multiple copies of e have different mid-points, so this will avoid any confusion.

So, given the termination of the procedure, we have a bunch of edges in $\mathcal{S}$ for each of which we have kept a record of its mid-point. Now the inverting procedure should be clear: For each $e = \{u_1, u_2\} \in \mathcal{S}$, pop e and add in the solution the edges $\{u_1, v\}$ and $\{u_2, v\}$ where v is the mid-point in the record of e . Give to both of these edges fractional value x_e whose record we have kept together with the midpoint. Clearly, by doing that as long as $\mathcal{S}$ is not empty, we will end up with our initial instance/solution characterized by x .

Another equivalent way to see the above procedure is the following: As before, let $\mathcal{S}$ be the set of edges added through the shortcut procedure for $S \in \mathcal{S}$. Each such edge $e \in \mathcal{S}$ has one of its endpoints in B_Z and the other in Γ_S . An easy but important observation is that each distinct $e \in \mathcal{S}$ (we treat each copy of e separately) corresponds to a unique *path* i.e., a sequences of vertices $u_1, u_2, \dots, u_x$ such that $u_1 \in Z, u_x \in \Gamma_S$ and $u_2, \dots, u_{x-1} \in \bar{S}$. This is immediate by just looking on the short-cutting procedure as an inductive process: For the base case, let $\{u_1, v\}$ and $\{v, u_2\}$ be the two edges to be short-cut with $u_1 \in B_Z, v \in \bar{S}, u_2 \in \Gamma_S$. Then, the new edge $\{u_1, u_2\} \in \mathcal{S}$ is indeed a path with the desired property. Now, if $\{u_1, v\}$ corresponds in fact to a series of short-cuts then, by induction, $\{u_1, v\}$ corresponds to a path with endpoints u_1, v and some intermediate vertices which belong in $\bar{S}$. We simply add v to this set of vertices and we have a path with endpoints u_1, u_2 and all intermediate vertices are in $\bar{S}$. This should also simplify the inversion process: Take any edge $e \in \mathcal{S}$ (in fact each separate copy of e) that corresponds to a unique path and “extract” its intermediate vertices. Add the corresponding edges along the path with the appropriate fractional value (fractional value of edge e in $\mathcal{S}$). For each edge $e \in \mathcal{S}$ we just need to keep track of its intermediate vertices.

We need to prove that after the short-cutting/elimination of all vertices in $\bar{S}$, for a single S that has only once child (with all observations we made above), the new solution x' we obtain is still a basic feasible solution. We have already proved it is feasible, so we need to see if it is basic solution. So assume that x' is not a basic solution. This means that there are other basic feasible solutions $x_1, \dots, x_p$ and p positive numbers $\lambda_i, i \in [p]$ that sum up to 1 such that $x' = \sum_{i \in [p]} \lambda_i x_i$. Observe that $c^T x_i = c^T x'$ since by construction $c^T x' \leq c^T x$ and x is the initial optimal solution for the LP. In other words, x' lies in the facet that has all points of cost $c^T x$.

Let $F \subseteq [n]$ be a subset of edges/indices of variables. We will first show that if we restrict the initial (or any) basic feasible solution x to the sub-instance induced after the removal of variables in F and the appropriate update of the budget/connectivity requirement bounds, we are still left with a basic feasible solution:

Lemma 3. *Let I be an instance of the budgeted Survivable Steiner Network problem and x be a basic feasible solution to the LP for I . Let $F \subseteq [n]$ a subset of edges in the support of x and let $\bar{x}$ a new solution where we have made zero every variable in F . Let $\bar{I}$ the new sub-instance of I after the removal of all edges in F with updated bounds on $f()$ and L_j appropriately (i.e., if $e \in F$ and $e \in C_j$ for some budget constraint j , then decrease L_j by x_e). Then $\bar{x}$ is a basic feasible solution for $\bar{I}$ on the reduced LP.*

Proof. The fact that $\bar{x}$ is a feasible solution for $\bar{I}$ is trivial from the feasibility of x for I . Assume that $\bar{x}$ is not basic solution. This means that there are two feasible solutions x^1 and x^2 such that $\bar{x} = \lambda x^1 + (1 - \lambda)x^2$, $\lambda \in (0, 1)$. But then if we expand x^j for $j \in \{1, 2\}$ by including all the entries of x that correspond to edges not included in the sub-instance (i.e., expand x_j by including all edges from F with same fractional value as in x), we have a feasible solution for I . So, x can be written as a convex combination of the two “expanded” solutions created x^1 and x^2 with the same multiplier λ contradicting the fact that x is basic. $\square$

We will use the above claim to prove that x' is basic solution.

First of all, we see that we can partition x , our initial optimal basic solution, into three restricted solutions x_1, x_2 and x_3 which are basic feasible solutions for the corresponding induced and restricted instances, according to the previous lemma. As a first step, let x_1 be the restriction of x to the sub-instance induced by $V \setminus S$ i.e., x_1 includes only those indices of variable corresponding to edges that are *not* incident to any vertex on S , with identical fractional values as in x . For the previous lemma, x_1 is basic feasible solution for the reduced instance after the reduction of the connectivity bounds and budget bounds appropriately. Identically, let x_2 be the restriction of x to the sub-instance induced by the vertices in $Z \cup \{V \setminus \bar{S}\}$ and has all the edges in $E(Z) \setminus \{A \cup B \cup \Gamma\}$ (including the edges in Δ). Finally, let x_3 be the restriction of x to the edges in $A \cup B \cup \Gamma$. Clearly, $x = x_1 + x_2 + x_3$ and the support sets of these three solutions are disjoint. All these solutions are basic feasible solutions according to Lemma 3.

With identical argumentation, we can partition x' into three feasible sub-solutions, each one corresponding to a particular sub-instance: x'_1 is the restriction of x' to the sub-instance induced by $S \setminus Z$ (i.e., all the edges of x' that have both of their endpoints outside Z), x'_2 corresponds to the sub-solution induced by the set of edges in $E(Z) \cup \Delta$ and x'_3 be the set of edges added during the course of the process to eliminate all vertices from $\bar{S}$. Let E_3 be this set of edges (and these edges go from $B_Z \subseteq Z$ to $V \setminus S$). As before, $x' = x'_1 + x'_2 + x'_3$ and the corresponding set of edges are disjoint (the indexes form a partition of $\text{supp}(x')$).

From the description of our procedure, it should be clear that $x_1 = x'_1$ and $x_2 = x'_2$ since the only set of edges touched/modified by the procedure are the edges in $A \cup B \cup \Gamma$ so the two other sets of edges of our partition remain unchanged. So, the difference between x and x' lies only on the indices of the sub-solutions x_3 and x'_3 . Since x_1 and x_2 are both basic feasible solutions for the corresponding reduced instances, the reason that x' is not a basic solution is because x'_3 is not a basic solution for the corresponding reduced instance. So, x'_3 is the only part of x' that can be written as a convex combination of other feasible solutions

(for the same reduced instance) and us assume that $x'_3 = \lambda_1 x_3^1 + \lambda_2 x_3^2 + \dots + \lambda_w x_3^w$ (where $w \leq |\text{supp}(x'_3)|$ by Caratheodory's Theorem and all multipliers are positive and sum up to 1).

First we observe that $x_3^j(i) = 0$ if $i \notin \text{supp}(x'_3)$ (i.e., the i -th index of x_3^j should be zero, otherwise the corresponding index of x'_3 would be non-zero). Observe also that, in fact, the subgraph induced by x'_3 (and also by x_3 and by all x_3^j) is a *bipartite* graph: all edges go from some vertices in Z and in particular from vertices in B_Z to vertices outside of S . There is no edge between the vertices in B_Z since these edges are included in x'_2 (and also x_2) and no edges between the vertices outside S that are adjacent to those vertices in B_Z since these edges have been included in x'_1 (also x_1) and similarly for the other side of the bipartition.

As we said, since x' is not a basic solution, it must be the case that x'_3 is not basic solution for the corresponding sub-instance. Now consider the following solutions y^j for each of which we have that $y_1 = x_1, y_2 = x_2$. We will describe now the solutions y_3^j : As mentioned, each edge in x_3^j has one endpoint in B_Z and the other in Γ_S . So, we will apply the “inversion” procedure on the set of edges implied by each x_3^j . Take an edge $\{u_1, u_2\}$ corresponding to a variable $x_3^j(e)$ with non-zero value. This edge should be present in (the support of) x' as well. Such an edge in x' corresponds to a set of paths, all of which have same endpoints $u_1 \in B_Z$ and $u_2 \in \Gamma_S$. So, in y_3^j , we insert all these paths (intermediate vertices and corresponding edges). We need to appropriately assign fractional values to all these edges, by appropriately assigning fractional values along the edges of each path and this can be easily done as follows: let $\vartheta(e)$ be the sum of the fractional values of all copies of e in x' . Add in y_3^j each path, and give to all edges along the path fractional value equal to $x_3^j(e) \cdot \frac{x'(e)}{\vartheta(e)}$. In that way, the sum of all the fractional value of each separate copy of e corresponding to a unique path would be $x_3^j(e)$.

Since $x'_3 = \sum_{j=1}^w \lambda_j x_3^j$, an immediate application of the invariant property gives us that $\sum_{i=1}^w \lambda_i y_3^i = x_3$, which contradicts the fact that x is a basic solution, also showing that x' cannot be written as convex combination of other feasible solutions, thus it is a basic solution, and we are done.

A short recap of the ideas above: We have started with particular basic feasible solution x . The tight, linearly independent connectivity constraints can be represented by a laminar family $\mathcal{L}$. If there does not exist any $L \in \mathcal{L}$ that has exactly one child in the tree representation of $\mathcal{L}$ then we are done: our charging scheme works. The problem appears when there exist two sets $S, Z \in \mathcal{L}$ with $f(S) = f(Z)$ (together with all particular properties discussed already). We have shown a procedure that short-cuts all vertices in $S \setminus Z$ and produces a new feasible solution x' . In this solution, $S \notin \mathcal{L}$. If x' is not basic, we can use the constructive version of Caratheodory's theorem to replace the part which does not correspond to a basic sub-solution with a particular basic feasible sub-solution of cost no more than the initial one and we are done. After the elimination of such a troubling sub-instance S we can apply our algorithm if after this step there exists a variable with “high” fractional value, or continue eliminating other such subcases.

From the above Claims, Lemmata and discussion we have:

Corollary 1. *Given any basic feasible solution x of the natural LP relaxation for the Metric Survivable Network Design problem, characterized by a set of tight budget constraints $\mathcal{Q}$ and a laminar family $\mathcal{S}$ of tight constraints of connectivity requirements we can, without any loss, produce a new basic feasible solution x'' such that it is not the case that there exists an $S \in \mathcal{S}$ with only one child in the tree-representation of $\mathcal{S}$.*

Case 3: $\Psi < 0$

As usual, let us assume that $f(S) - \sum_{j \in [q]} f(Z_j) = -k$ for some positive integer k . As a first step we will make some, relatively easy, observations about the structure of the sub-instance that will greatly help us deal with this case.

First of all, as in the previous subsection, we assume that $\sum_{e \in B} x_e < \frac{1+f}{2+f}$, and $\sum_{e \in \Gamma} x_e < 1$, otherwise the charge received by S is at least 1 and so we are done.

Observation 1. As before, let $\bar{S} = S \setminus \bigcup_{j \in [q]} Z_j \subseteq V$ to be the set of vertices belonging to S but not to any of its children. By definition of the case, S must have at least one child. Let $r_S = \{s, t\}$ be the pair of vertices of connectivity requirement that defines $f(S)$, i.e.,

$$r_S = \max_{s_i \in S, t_i \notin S} \{r_i\}.$$

As a first observation, we notice that $s \notin \bar{S}$. In other words, s must belong to a child Z_j of S , otherwise we are done since the charge that S would receive would be at least one. Indeed, if $s \in \bar{S}$ then all the edges $e \in \delta(s)$ would assign charge to S at least x_e , since at least one of their endpoints is $s \in \bar{S}$ and $\bar{S}$ is the smallest set in $\mathcal{L}$ that contains s by definition. Since $f(S) \geq 1 \Rightarrow \sum_{e \in \delta(s)} x_e \geq 1$, and we are done.

So, we assume that $s_i \notin \bar{S}$.

Observation 2. Second, we see that it is not the case that S has a lonely child, otherwise we are done since S would receive charge of at least 1. Indeed, if $q = 1$, let Z be the lonely child of S . From the previous observation, s_i belongs in Z . But we also know, since $\Psi = -k < 0$, that $f(Z) > f(S)$. Let $\{s_Z, t_Z\}$ be the pair of vertices such that $s_Z \in Z$ and $t_Z \notin Z$ that are responsible for the connectivity requirement $r_Z = f(Z)$.

1. If $t_Z \notin S$, i.e., t_Z is outside both Z (by definition) and S , then $f(S)$ should be equal to r_Z by definition of the function f and so $f(S) = f(Z)$, but we know this is not the case.
2. So, $t_Z \in S$. But in that case, as with the case with Observation 1, we see that S would receive charge of at least 1, and we are done in this case as well.

Observation 3. Take any vertex $v \in \bar{S} = S \setminus \bigcup_{j \in [q]} Z_j$, i.e., v belongs in S but not in any other of the children of S (in the tree representation of the laminar family $\mathcal{L}$). Then we have that

$$\sum_{e \in B(Z_j) \cap \delta(v)} x_e = \sum_{e \in \delta(v) \setminus B(Z_j)} x_e,$$

where, in analogy with the previous sub-section, $B(Z_j) = \{e \in \delta(Z_j) \setminus \Delta\}$ i.e., the set of edges with one endpoint in Z_j and the other in $\bar{S}$. In other words, the sum of the fractional values of all edges going from a child Z_j of S to a single vertex $v \in \bar{S}$ is equal to the sum of the fractional values of edges “leaving” v either through edges in another $B(Z_{j'})$ or through edges in Γ . Indeed, this is true by considering $Z'_j := Z_j \cup \{v\}$ and noticing that $f(Z'_j) = f(Z_j)$.

Observation 4. Assume that $B = \emptyset$. Then $\Gamma = \emptyset$ as well. Indeed, assume not and take any vertex $v \in \bar{S}$ such that v has an edge from Γ incident to it. Take any child Z_j of S and consider the set of vertices $Z_j \cup \{v\}$ which has connectivity requirement equal to $f(Z_j)$. But then

$$\sum_{e \in \delta(Z_j \cup \{v\})} x_e = \sum_{e \in \delta(Z_j)} x_e + \sum_{e \in \delta(v)} x_e > f(Z_j),$$

a contradiction. So $B = \emptyset \Rightarrow \Gamma = \emptyset$.

In direct analogy with the case when $\Psi = 0$, we will again describe a procedure that “shortcuts” all vertices in $\bar{S} = S \setminus \cup_{j \in [q]} Z_j$. Although the idea is identical as in the previous section, we will describe it here for the sake of completeness.

For each child $Z_j \subseteq S$ of S in the laminar family of tight sets $\mathcal{L}$, let $B(Z_j)$ be the subset of vertices of Z_j that are adjacent through some edges in B to vertices in $\bar{S}$ and let Γ_S the vertices outside S that are adjacent to vertices in $\bar{S}$ through some edges in Γ i.e.,

$$\begin{aligned} B(Z_j) &= \left\{ v \in Z_j \text{ such that } \exists e = \{u, v\} \in B \text{ with } u \in \bar{S} \right\} \\ \Gamma_S &= \left\{ v \in V \setminus S \text{ such that } \exists e = \{u, v\} \in \Gamma \text{ with } u \in \bar{S} \right\}. \end{aligned}$$

We start with any vertex $v \in \bar{S}$. We know that this vertex is adjacent to some vertices in some Z_j s, to some other vertices in $\bar{S}$ and, finally, to some vertices outside S . As in the previous subsection, we will crucially use Observation 3 in our procedure to guarantee its termination. We choose any child of Z_j such that $v \cap B(Z_j) \neq \emptyset$. We pick any vertex u_1 from any such Z_j that is adjacent to v and, from there, we search for a maximal path containing consecutive vertices from $\bar{S}$ until we either end up in some $Z_{j'}, j' \neq j$ or in some vertex $u_2 \in \Gamma_S$ i.e., outside S . Let $[u_1, v, v_1, \dots, v_\rho, u_2]$ be that path such that $u_1 \in B(Z_j), u_2 \in \{\Gamma_S, Z_{j'}\}$ for some $j' \neq j$, $v, v_1, \dots, v_\rho$ are all in $\bar{S}$ and, moreover, $\{u_1, v\}, \{v, v_1\}, \dots, \{v_{\rho-1}, v_\rho\}, \{v_\rho, u_2\}$ are all edges in the support of x . We subtract from all edges along the path $[u_1, v, v_1, \dots, v_\rho, u_2]$ the amount equal to the minimum fractional variable corresponding to an edge along this path and we add a distinct copy of the edge $\{u_1, u_2\}$. In this way, at least one edge incident to at least one of the vertices in $\bar{S}$ along the path, will be eliminated. We repeat this procedure until all edges adjacent to all vertices in $\bar{S}$ are eliminated and thus all vertices in $\bar{S}$ disappear.

After the termination of the procedure we will end up with a set of edges $\mathcal{S}$ added during the course of the process. We keep each copy of a particular edge separately and each such edge corresponds to a particular path. We also keep track to the set of all paths (along with the corresponding edges) $\mathcal{P}$. All the preservation properties hold in this case as well with identical reasonings, so we will avoid the duplicate details. In such a way, a new *feasible* solution x' is being produced. In order to argue that x' is a basic solution, we proceed as before. We assume that x' is not a basic solution so in fact it can be written as a convex combination of two other feasible solutions x_1, x_2 i.e., $x' = \lambda x_1 + (1 - \lambda)x_2$ for some $\lambda \in (0, 1)$. From x_1, x_2 we construct two new feasible solutions y_1, y_2 by taking edges in $\mathcal{S}$ (that are present in x_1, x_2) and unfold them in direct analogy with the previous subsection. This way, we introduce all the vertices of $\bar{S}$ in both solutions y_1, y_2 and the corresponding edges along the paths with appropriate fractional value. In this way, the restricted sub-instance of the paths $\mathcal{P}$, and they can be written as a convex combination of the restricted sub-instances of y_1, y_2 , contradicting that x is a basic feasible solution. So, x' must be basic, and we are done. $\square$

Figure 2: A snapshot of the process of eliminating the vertices in $S \setminus Z$: Starting from u_1 and v , we find a maximal path following consecutive vertices in $\bar{S}$ until we hit $u_2 \in Z_2$. Let p be the path and let $\mu = \min_{e \in p} \{x_e\}$ be the minimum fractional value of edges along the path. We subtract μ from all fractional values along the path and we add a new copy of $\{u_1, u_2\}$ with fractional value μ in our solution. This way at least one edge along the path will be eliminated (for example the gray edge incident to vertex v_2). At the end of the procedure we will have edges going from only Z_1 to Z_2 or from either of these two sets to the “outside” world (outside S). Observation 3 guarantees that the procedure will terminate successfully eliminating all edges incident to vertices in $\bar{S}$.

This concludes the proof that every tight object $S \in \mathcal{L}$ and $C_j \in \mathcal{Q}$ receives a fractional token value at least one, for a total fractional token distributed of $|\mathcal{L}| + |\mathcal{Q}| = |E|$. But from the previous claim, we know that the total fractional token distributed is strictly less than $|E|$ (each edge distributes at least one but there must exist at least one edge distributing strictly less than one fractional token). Contradiction. So the hypothesis must be false, and so there must exist an edge $e \in E : x_e > \frac{1}{(2+f)}$.

3 A $((2 + f), (2 + f))$ Bi-criteria Algorithm

Given Theorem 2, the algorithm is relatively simple and is depicted in Algorithm 1. Although the proof that the algorithm is a $(2 + f, 2 + f)$ bi-criteria algorithm differs only slightly with other similar proofs, we will include it here for completeness.

Step 2.(α) can be solved in polynomial time using the fact that the new connectivity requirement function f' is again *weakly* super-modular and using the Ellipsoid method with the usual separation oracle used to solve the initial LP, see [8, 24] for implementation details. Step 2.(ε) can be easily implemented to run in polynomial time: we do not need to update all $S \in 2^V$, but rather all $S \in \mathcal{L}$ such that $S \cap X \neq \emptyset$, where $\mathcal{L}$ is the laminar family which, together with $\mathcal{Q}$, defines the current extreme point solution x , see Lemma 2. In step 2.(β) we apply the shortcut procedure only to the “pathological” cases described in the corresponding subsections: (1) A tight set S with a single child Z such that $f(S) = f(Z)$ such that the “charge” is not enough for S and (2) Tight sets S in $\mathcal{L}$ with q children $Z_1, \dots, Z_q$ in $\mathcal{L}$ and $f(S) < \sum_{i \in [q]} f(Z_i)$, with again not enough charge for S . So, in fact, we apply the rounding procedure to the new solution x' derived from x after the short-cut procedure to appropriate

Algorithm 1 Iterative Algorithm for the multi-budgeted SND problem

Input: An instance of the multi-budgeted Survivable Network Design problem.

Output: A subgraph of G in which all the connectivity requirements are satisfied.

1. **Initialize:** $f' = f$ and $H = \emptyset$.
 2. **while** $f' \neq 0$ (i.e., there are still unsatisfied requirements):
 - (α) Solve the relaxed LP as defined by (2) with f' as connectivity requirement function plus the remaining budgets. Let x be the bfs obtained.
 - (β) Apply the shortcut procedure to all relevant instances as explained in the previous section.
 - (γ) Let $X = \{e \in E : x_e \geq \frac{1}{(2+f)}\}$. Include all $e \in X$ in H .
 - (δ) **Update Budgets:** If $e \in X \cap C_j \Rightarrow L_j := L_j - x_e$.
 - (ε) **Update Connectivity Requirements:** $f'(S) := f(S) - |\delta(S) \cap X|$, $\forall S \in \mathcal{S}$.
 3. **Return:** H .
-

sets. The fact that x' is a basic feasible solution guarantees that we can apply Theorem 2.

Theorem 3. *Algorithm 1 returns a feasible solution $\bar{x} \in \{0, 1\}^E$ in polynomial time for which the following holds:*

1. $c^T \bar{x} = \sum_{e \in E} c_e \bar{x}_e \leq (2 + f) \cdot \text{opt}$ where opt is the optimal fractional solution cost of the initial LP, and
2. $\ell_j^T \bar{x} = \sum_{e \in E_j} \bar{x}(e) \ell_j(e) \leq (2 + f) \cdot L_j$.

Proof. This is a relatively standard inductive argument, but for completeness we provide it here in full details. The correctness of the algorithm is implied by Theorem 2.

First of all, to prove the violation of the budgets, it suffices to notice that in each iteration for every edge we round to 1, we decrease its budget by at least $1/(2 + f)$. More formally, take a budget $C_j : \sum_{e \in E_j} x(e) \ell_j(e) \leq L_j$ and let T be the set of edges which are greater than $1/(2 + f)$ and we rounded up to 1 at some point in the algorithm. At the end, the budget will be

$$\begin{aligned} \sum_{e \in T} \ell_j(e) &\leq \sum_{e \in T} (2 + f) \bar{x}(e) \ell_j(e) \\ &\leq (2 + f) L_j. \end{aligned}$$

To prove that the algorithm approximates within a factor of $(2 + f)$ the (lower bound of the) optimal objective function value, we employ an inductive argument: for the base case assume that the algorithm terminates only after one iteration. This means that X defines a feasible set of edges. Since $X = \{e \in E : \bar{x}_e \geq 1/(2 + f)\} \Rightarrow \sum_{e \in X} c_e \leq \sum_{e \in X} (2 + f) \bar{x}_e c_e \leq (2 + f) \text{opt}$, where as usual $\text{opt} = \sum_e \bar{x}_e c_e$. Now assume that the claim is true for up to k iterations. We will show that the claim remains true if the algorithm takes one more, i.e., for $k + 1$ iterations.

Let X' be the current set of edges we round to 1 in the current iteration and let f' be the residual connectivity requirement after we round all edges in X' to 1. Let X'' be all the set of edges picked in subsequent iterations to satisfy the new connectivity requirement function f' . It is not hard to observe that the current solution $\bar{x}$ at the k -th iteration is also a feasible solution for f' if we restrict $\bar{x}$ to the edges in $E \setminus \{X'\}$. By inductive hypothesis the cost of X'' is at most $(2 + f) \sum_{e \in E \setminus X'} x_e c_e$ and set $X''' = X'' \cup X'$. The cost of X''' is

$$\begin{aligned} \sum_{e \in X'''} c_e x_e &= \sum_{e \in X''} c_e x_e + \sum_{e \in X'} c_e x_e \\ &\leq (2 + f) \sum_{e \in E \setminus X'} x_e c_e + c(X') \\ &\leq (2 + f) \sum_{e \in E} c_e x_e, \end{aligned}$$

where the last inequality follows since all edges in X' have fractional value $\geq 1/(2 + f)$ and this concludes that the cost of the solution is at most thrice the cost of the optimal solution cost. $\square$

Corollary 2. *There exist a $(4, 4)$ polynomial time bi-criteria approximation algorithm for the Weighted Bounded Degree Survivable Network Design problem.*

Proof. Given an instance I for the Weighted Bounded Degree Survivable Network Design problem, run Algorithm 1. This is a special case of the budgeted SND problem with $f = 2$, giving us the desired result. $\square$

4 Conclusions

In this paper, we have studied the multi-budgeted version of the classical SND problem. Using Linear Programming techniques in conjunction with iterative rounding approach, we have developed bi-criteria algorithms and proven their worst case guarantees. In particular, we proved that if the maximum *frequency* of an edge $e \in E(G)$ (i.e., number of extra budget linear constraints e is part of) is f , then we can approximate the optimal objective function value within a worst case factor of $(2 + f)$ and the solution returned might violate the bounds imposed by budgets by the same factor. One might wonder if there can be better approximability guarantees. Unfortunately this is highly unlikely: a better than $(f + 2)$ approximation guarantee would imply immediately a better than 2 approximation algorithm for the un-budgeted Steiner Network problem (by simply taking $f = 0$ as this is true in the case without budgets) improving the seminal Jain's algorithm [8] and this is considered at the moment of this writing a very challenging research direction.

5 Acknowledgements

The author want to express his gratitude to the anonymous referees for their suggestions and very constructive comments for a preliminary version of this manuscript. The current work of the author is supported by a NWO TOP grant. Part of this work was done while the author was within the LAMSADE team in University of Paris-Dauphine, partially supported by the

Swiss National Science Foundation Early Post-Doc mobility fellowship P1TIP2_152282. A very preliminary version of this work was presented in the 22nd International Colloquium Structural Information and Communication Complexity SIROCCO 2015 [23].

References

- [1] Sina Dehghani, Soheil Ehsani, MohammadTaghi Hajiaghayi, and Vahid Liaghat. Online degree-bounded steiner network design. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 164–175. SIAM, 2016.
- [2] Takuro Fukunaga and Hiroshi Nagamochi. Network design with weighted degree constraints. In Sandip Das and Ryuhei Uehara, editors, *WALCOM: Algorithms and Computation, Third International Workshop, WALCOM 2009, Kolkata, India, February 18-20, 2009. Proceedings*, volume 5431 of *Lecture Notes in Computer Science*, pages 214–225. Springer, 2009.
- [3] Takuro Fukunaga and Hiroshi Nagamochi. Network design with weighted degree constraints. *Discrete Optimization*, 7(4):246–255, 2010.
- [4] Harold N. Gabow. On the l_∞ -norm of extreme points for crossing supermodular directed network lps. In Michael Jünger and Volker Kaibel, editors, *Integer Programming and Combinatorial Optimization, 11th International IPCO Conference, Berlin, Germany, June 8-10, 2005, Proceedings*, volume 3509 of *Lecture Notes in Computer Science*, pages 392–406. Springer, 2005.
- [5] Harold N. Gabow, Michel X. Goemans, and David P. Williamson. An efficient approximation algorithm for the survivable network design problem. In Giovanni Rinaldi and Laurence A. Wolsey, editors, *Proceedings of the 3rd Integer Programming and Combinatorial Optimization Conference, Erice, Italy, April 29 - May 1, 1993*, pages 57–74. CIACO, 1993.
- [6] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
- [7] Michel X. Goemans, Andrew V. Goldberg, Serge A. Plotkin, David B. Shmoys, Éva Tardos, and David P. Williamson. Improved approximation algorithms for network design problems. In Daniel Dominic Sleator, editor, *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms. 23-25 January 1994, Arlington, Virginia.*, pages 223–232. ACM/SIAM, 1994.
- [8] Kamal Jain. A factor 2 approximation algorithm for the generalized steiner network problem. *Combinatorica*, 21(1):39–60, 2001.
- [9] Sanjeev Khanna, Joseph Naor, and F. Bruce Shepherd. Directed network design with orientation constraints. In David B. Shmoys, editor, *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, January 9-11, 2000, San Francisco, CA, USA.*, pages 663–671. ACM/SIAM, 2000.

- [10] Piotr Krysta. Bicriteria network design via iterative rounding. In Lusheng Wang, editor, *Computing and Combinatorics, 11th Annual International Conference, COCOON 2005, Kunming, China, August 16-29, 2005, Proceedings*, volume 3595 of *Lecture Notes in Computer Science*, pages 179–187. Springer, 2005.
- [11] Lap Chi Lau, Joseph Naor, Mohammad R. Salavatipour, and Mohit Singh. Survivable network design with degree or order constraints. In David S. Johnson and Uriel Feige, editors, *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, pages 651–660. ACM, 2007.
- [12] Lap Chi Lau, R. Ravi, and Mohit Singh. *Iterative Methods in Combinatorial Optimization*. Cambridge University Press, 2011.
- [13] Lap Chi Lau and Mohit Singh. Additive approximation for bounded degree survivable network design. In Cynthia Dwork, editor, *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 759–768. ACM, 2008.
- [14] Lap Chi Lau and Hong Zhou. A unified algorithm for degree bounded survivable network design. *Math. Program.*, 154(1-2):515–532, 2015.
- [15] Anand Louis and Nisheeth K. Vishnoi. Improved algorithm for degree bounded survivable network design problem. In Haim Kaplan, editor, *Algorithm Theory - SWAT 2010, 12th Scandinavian Symposium and Workshops on Algorithm Theory, Bergen, Norway, June 21-23, 2010. Proceedings*, volume 6139 of *Lecture Notes in Computer Science*, pages 408–419. Springer, 2010.
- [16] Vardges Melkonian and Éva Tardos. Algorithms for a network design problem with crossing supermodular demands. *Networks*, 43(4):256–265, 2004.
- [17] Viswanath Nagarajan, R. Ravi, and Mohit Singh. Simpler analysis of LP extreme points for traveling salesman and survivable network design problems. *Oper. Res. Lett.*, 38(3):156–160, 2010.
- [18] Joseph Naor, Debmalya Panigrahi, and Mohit Singh. Online node-weighted steiner tree and related problems. In Rafail Ostrovsky, editor, *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 210–219. IEEE Computer Society, 2011.
- [19] Zeev Nutov. Approximating directed weighted-degree constrained networks. In Ashish Goel, Klaus Jansen, José D. P. Rolim, and Ronitt Rubinfeld, editors, *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques, 11th International Workshop, APPROX 2008, and 12th International Workshop, RANDOM 2008, Boston, MA, USA, August 25-27, 2008. Proceedings*, volume 5171 of *Lecture Notes in Computer Science*, pages 219–232. Springer, 2008.
- [20] Zeev Nutov. Approximating directed weighted-degree constrained networks. *Theor. Comput. Sci.*, 412(8-10):901–912, 2011.
- [21] Alexander Schrijver. *Theory of Linear and Integer Programming*. John Wiley & sons, 1998.

- [22] Mohammad Shojafar, Nicola Cordeschi, and Enzo Baccarelli. Energy-efficient adaptive resource management for real-time vehicular cloud services. *IEEE Transactions on Cloud Computing*, PP.99:1–22, 2016.
- [23] Georgios Stamoulis. Approximation algorithms for multi-budgeted network design problems. In Christian Scheideler, editor, *Structural Information and Communication Complexity - 22nd International Colloquium, SIROCCO 2015, Montserrat, Spain, July 14-16, 2015, Post-Proceedings*, volume 9439 of *Lecture Notes in Computer Science*, pages 135–148. Springer, 2015.
- [24] Vijay V. Vazirani. *Approximation Algorithms*. Springer, 2004.
- [25] David P. Williamson, Michel X. Goemans, Milena Mihail, and Vijay V. Vazirani. A primal-dual approximation algorithm for generalized steiner network problems. *Combinatorica*, 15(3):435–454, 1995.