THE VPN PROBLEM WITH CONCAVE COSTS

SAMUEL FIORINI, GIANPAOLO ORIOLO, LAURA SANITÀ, AND DIRK OLIVER THEIS

ABSTRACT. We consider the following network design problem. We are given an undirected network with costs on the edges, a set of terminals, and an upper bound for each terminal limiting the cumulative amount of traffic it can send or receive. The task is to select a path for each unordered pair of terminals and reserve minimum cost capacities so that all the sets of traffic demands that satisfy the bounds can be routed along the selected paths.

When the contribution of an edge to the total cost is proportional to the capacity reservation for that edge, this problem is referred to as the symmetric Virtual Private Network Design (sVPN) problem. Goyal, Olver and Shepherd (*Proc. STOC*, 2008) showed that there always exists an optimal solution to sVPN that is a tree solution, i.e., such that the support of the capacity reservation is a tree. Combining this with previous results by Fingerhut, Suri and Turner (*J. Alg.*, 1997) and Gupta, Kleinberg, Kumar, Rastogi and Yener (*Proc. STOC*, 2001), sVPN can be solved in polynomial time.

In this paper we investigate of the concave symmetric Virtual Private Network Design (csVPN) problem, where the contribution of each edge to the total cost is proportional to some concave, non-decreasing function of the capacity reservation. Note that csVPN is NP-hard, even if we restrict to tree solutions. We show that also the csVPN problem always admits an optimal solution that is a tree solution and we give a 24.92-approximation algorithm for the problem.

1. INTRODUCTION

The symmetric Virtual Private Network Design (sVPN) problem is defined as follows. We are given an undirected network with costs on the edges, a set of terminals, and an upper bound for each terminal limiting the cumulative amount of traffic it can send or receive. The bounds implicitly describe the set of traffic demands that the network should support: such sets of traffic demands are called valid. The task is to select a path for each unordered pair of terminals and reserve minimum cost capacities on the edges of the network so that all the valid set of traffic demands can be routed along the selected paths. The contribution of an edge to the total cost is proportional to the capacity reservation for that edge.

It was shown by Fingerhut, Suri and Turner [2] and Gupta, Kleinberg, Kumar, Rastogi and Yener [5] that sVPN can be solved in polynomial time if the sVPN tree routing conjecture holds. This conjecture states that each sVPN instance has an optimal solution whose support is a tree (in short, a *tree solution*), see, e.g., Erlebach and Rüegg [1], Italiano, Leonardi and Oriolo [8] and Hurkens, Keijsper and Stougie [7]. The sVPN tree routing conjecture was recently solved affirmatively by Goyal, Olver and Shepherd [3].

Goyal et al. solved the sVPN tree routing conjecture by settling an equivalent conjecture, the so-called PR conjecture due to Grandoni, Kaibel, Oriolo and Skutella [4]. The PR conjecture claims that each instance of the Pyramidal Routing (PR) problem has an optimal tree solution. In this problem, we are given an undirected graph with costs on the edges and a set of terminals. One of the terminals is marked as the root and some known amount of traffic is to be routed along paths from the root to the other terminals. The contribution of each edge to the total cost is proportional to a certain function of the number of paths in the routing using the edge. The name of the problem stems from the particular shape of the function used to compute the total cost (the "pyramidal" function $x \mapsto \max(x, B-x)$, where B is the total amount of traffic to be routed).

Département de Mathématique, Université Libre de Bruxelles, Brussels, Belgium. Email: sfiorini@ulb.ac.be.
Dipartimento di Ing. dell'Impresa, Università di Roma "Tor Vergata", Rome, Italy. Email:oriolo@disp.uniroma2.it.
Dipartimento di Ing. dell'Impresa, Università di Roma "Tor Vergata", Rome, Italy. Email: sanita@disp.uniroma2.it.
Département de Mathématique, Université Libre de Bruxelles, Brussels, Belgium. Email: dtheis@ulb.ac.be.

In this paper we investigate a natural generalization of the sVPN problem where the cost per unit capacity may decrease if a larger amount of capacity is reserved. More precisely, we define the concave symmetric Virtual Private Network Design (csVPN) problem as the sVPN, but the contribution of each edge to the total cost is now proportional to some arbitrary fixed concave, non-decreasing function f of the capacity reservation. For linear f one recovers the sVPN. However, for different choices of f, the csVPN is an NP-hard problem.

We show that the csVPN problem always admits an optimal solution that is a tree solution, extending therefore the result in [3]. We give a constant factor approximation algorithm for csVPN by reducing the problem to the Single Source Buy at Bulk (SSBB) problem.

1.1. **Detailed description of the problems.** In this paper, we consider four routing problems: the symmetric Virtual Private Network Design (sVPN) problem, the Pyramidal Routing (PR) problem and their generalizations with arbitrary concave costs: the concave symmetric Virtual Private Network Design (csVPN) problem and the Concave Routing (CR) problem.

We now describe the four problems in detail. All the problems involve an undirected, connected graph $G = (V, E)$ that represents a communication network. The graph comes with two vectors: a vector $c \in \mathbb{R}_+^E$ describing the edge costs and a vector $b \in \mathbb{Z}_+^V$ providing some information on the traffic that each vertex sends or receives (the exact interpretation depends on the problem). A vertex v with $b_v > 0$ is referred to as a *terminal*. We denote the set of terminals by W. Also, we let B be the sum of all components of b. In other words, we let $W := \{v \in V \mid b_v > 0\}$ and $B := \sum_{v \in V} b_v$.

sVPN. In the *symmetric Virtual Private Network design* (sVPN) problem, the vertices of G want to communicate with each other. However, the exact amount of traffic between pairs of vertices is not known in advance. Instead, for each vertex v the cumulative amount of traffic that it can send or receive is bounded from above by b_v. The general aim is to install minimum cost capacities on the edges of the graph supporting any possible communication scenario, where the cost for installing one unit of capacity on edge e equals its cost c_e.

Let $\binom{W}{2}$ denote the set of cardinality two subsets of W. A *set of traffic demands* $D = \{d_{uv} \mid \{u, v\} \in \binom{W}{2}\}$ specifies for each unordered pair of terminals $u, v \in W$ the amount $d_{uv} \in \mathbb{R}_+$ of traffic between u and v. A set D is *valid* if it respects the upper bounds on the traffic of the terminals. That is,

$$\sum_{u \in W} d_{uv} \le b_v \qquad \text{for all terminals } v \in W.$$

A solution to an instance of the sVPN problem, defined by the triple (G, b, c), consists of a collection of paths $\mathcal{P}$ containing exactly one u–v path P_{uv} in G for each unordered pair u, v of terminals, and edge capacities $\gamma_e \in \mathbb{R}_+$ ($e \in E$). Such a set of paths $\mathcal{P}$, together with edge capacities γ, is called a *virtual private network*. A virtual private network is *feasible* if all valid sets of traffic demands D can be routed without exceeding the installed capacities γ where all traffic between terminals u and v is routed along path P_{uv}, that is,

$$\gamma_e \ge \sum_{\{u,v\} \in \binom{W}{2} : e \in P_{uv}} d_{uv} \qquad \text{for all edges } e \in E.$$

Given $\mathcal{P}$, one may compute in polynomial time the minimum amount of capacity γ_e that has to be reserved on each edge e in order to obtain a feasible virtual private network $(\mathcal{P}, \gamma)$, see Gupta et al. [5] and Italiano et al. [8] for details.

PR[1] In the *Pyramidal Routing* (PR) problem, one of the terminals is marked as a *root*. We denote the root by r. Thus an instance of the PR problem is defined by a quadruple (G, r, b, c). For each vertex v, the number

[1] The definition of the PR problem given here differs from that of Grandoni et al. [4] and Goyal et al. [3]. Indeed, these authors assume that $b_v \in \{0, 1\}$ for each $v \in V$ and allow only unsplittable routings. We show later that this is not a restriction.

b_v describes the actual *demand* at the vertex. Thus B is the total demand. Note that the root has $b_r > 0$. The aim is to route b_v units of flow from r to each $v \in W$ at minimum cost.

A solution to the instance (G, r, b, c) of the PR problem consists of a "routing". Letting $\mathcal{A}$ denote the set of all simple paths contained in graph G, we define a *routing* as a vector in $\mathbb{R}_+^{\mathcal{A}}$. Thus, a routing $q \in \mathbb{R}_+^{\mathcal{A}}$ assigns a non-negative real number $q(P)$ to each path $P \in \mathcal{A}$. A routing q is said to be *feasible* if each path in its support links the root to some terminal, and

$$\sum_{P \in \mathcal{A}_{rv}} q(P) = b_v \qquad \text{for all vertices } v \in V,$$

where $\mathcal{A}_{rv}$ denotes the set of all paths having an end equal to r and the other equal to v. In particular, a feasible routing assigns a value of b_r to the trivial path starting and ending at the root (since it is simple, this path has no edge).

The name of the PR problem is due to its particular cost function: The cost of a feasible routing q is given by $z(q) := \sum_{e \in E} c_e \min\{\phi_e, B - \phi_e\}$, where ϕ is the "flow vector" of q. For any routing q and edge e, we let $\phi_e(q)$ denote the *total flow* on edge e for routing q. Thus, $\phi_e := \sum_{P \ni e} q(P)$. The vector $\phi(q) \in \mathbb{R}_+^E$ is referred to as the *flow vector* of q.

csVPN. The *concave symmetric Virtual Private Network Design* (csVPN) problem is defined similarly as the sVPN problem. The total cost of a capacity reservation γ is now $z(\gamma) := \sum_{e \in E} c_e\, f(\gamma_e)$, where $f : [0, B] \to \mathbb{R}_+$ is concave, non-decreasing and such that $f(0) = 0$. An instance of csVPN is described by a quadruple (G, b, c, f). (We assume we are given oracle access to the function f.)

CR, ndCR and asCR. The *Concave Routing* (CR) problem is defined as the PR problem. The total cost of a feasible routing q is $z(q) := \sum_{e \in E} c_e\, g(\phi_e)$, where $\phi = \phi(q)$ and $g : [0, B] \to \mathbb{R}_+$ is concave and such that $g(0) = 0$. An instance of CR is thus described by a quintuple (G, r, b, c, g). (As for csVPN, we assume we are given oracle access to g.)

We consider the following two restrictions of the CR problem. The instances of the *non-decreasing Concave Routing* (ndCR) problem are those for which g is non-decreasing. In this case, we use the letter f instead of g whenever possible. The instances of the *axis-symmetric Concave Routing* (asCR) problem are those for which g is *(axis)-symmetric*, that is, $g(B - x) = g(x)$ for all $x \in [0, B]$. In this case, we use the letter h instead of g whenever possible.

Tree solutions. A feasible solution to one of the problems described above is a *tree solution* if the capacity vector γ or the flow vector $\phi(q)$ has an acyclic support, in which case its support induces a tree in G.

1.2. **Previous work.** Many of the foundations of the sVPN problem appear in Fingerhut et al. [2] and Gupta et al. [5]. Both papers show that computing a tree solution of minimum cost gives a 2-approximation algorithm for the problem. Such a solution can be obtained in polynomial time by a single all-pair shortest paths computation. It has been discussed [6] and then conjectured in Erlebach et al. [1] and in Italiano et al. [8] that there always exists an optimal solution to the sVPN problem that is a tree solution: this has become known as the *VPN tree routing conjecture*. The conjecture has first been proved for the case of ring networks [7, 4], and then in general graphs [3]. Goyal et al. [3] prove the VPN tree routing conjecture by establishing another conjecture, the *PR conjecture*, which states that every instance of the PR problem admits an optimal tree solution.

The PR problem was proposed by Grandoni et al. [4]. The PR conjecture made its first apparition in their paper, together with a proof that the PR conjecture implies the VPN tree routing conjecture. Remarkably, besides establishing the PR conjecture, Goyal et al. [3] also show that the VPN tree routing and PR conjectures are *equivalent*, that is, one implies the other and vice versa.

1.3. **Our contribution / Paper outline.** In Section 2 we give a constant factor approximation algorithm for the csVPN problem[2]. We precede this result with a discussion, in Section 2.1, on the splittability of the solutions to both the csVPN and CR problems, and on when we can assume b to be a 0-1 vector.

Our approximation algorithm works by reduction to the Single Source Buy at Bulk (SSBB) problem. The reduction is in two steps. We first observe in Section 2.2 that any approximation algorithm for SSBB gives an approximation algorithm for ndCR with the same approximation factor. We then show in Section 2.3 how to turn any approximation algorithm for ndCR into an approximation algorithm for csVPN with an approximation factor twice as large. Combining both steps, we obtain a 2ρ-approximation algorithm for csVPN from the ρ-approximation algorithm for SSBB due to Grandoni and Italiano [9], where $\rho = 24.92$. When restricted to csVPN instances admitting an optimal solution that is a tree solution, the approximation factor of our algorithm improves to ρ. This is because, in the analysis of our algorithm, we use the following key property of the csVPN problem: the cost of an optimal tree solution is never more than twice the cost of an optimal solution. (As pointed out above, a similar property was known for the sVPN problem.)

In Section 3 we prove our main result: every csVPN instance (G, b, c, f) has an optimal solution that is a tree solution. The proof builds upon an equivalence, stated in Section 3.1, between the csVPN problem and the asCR problem. We show that, when b is a 0-1 vector, solving an csVPN instance (G, b, c, f) is essentially the same as solving an asCR instance of the form (G, r, b, c, h) where h is obtained by symmetrizing f. Moreover, the csVPN instance (G, b, c, f) has an optimal solution that is a tree solution if and only if the asCR instance (G, r, b, c, h) has an optimal solution that is a tree solution. This allows us to focus only on asCR. By combining some simple polyhedral remarks with the fact that every PR instance has an optimal solution that is a tree solution, we show that every asCR instance has indeed an optimal solution that is a tree solution.

Although every csVPN (or asCR) instance has an optimal solution that is a tree solution, we show in Section 3.4 that this does *not* hold for every CR instance; even in case G is a cycle and some extra restrictions (other than being non-decreasing or symmetric) are put on the function g.

2. Approximation algorithms

2.1. **Preliminaries.** We start by discussing the integrality and splittability of the solutions to the problems. A routing is said to be *unsplittable* whenever its support contains at most one path between any two terminals.

In the *fractional relaxation* of the csVPN problem, for each pair of terminals u, v we are allowed to split the u–v flow along some set of u–v paths, but the fraction that we accommodate on any of these paths must be the same with respect to each valid set of traffic demands.

Note that the definition of the CR problem we have given in the introduction already allows fractional routings. Because the function $q \mapsto z(q)$ is concave and the set of feasible routings has a very simple structure (formally, it is a product of simplices), we can restrict our attention to unsplittable routings. This is stated in our first lemma whose proof can be found in the appendix. (Although Goyal et al.'s proof of same result for the PR problem [3, Lemma 2.2] also works for the more general CR problem, we include the proof here for completeness.)

Lemma 1. *Every* CR *instance has an unsplittable optimal solution. Moreover, given a fractional routing we can build an unsplittable routing for the same instance that does not cost more, in time polynomial in the size of the instance plus the size of the given fractional routing.*

For some instance I of the csVPN problem, we denote by $\mathrm{OPT}_{\text{tree}}(I)$ the cost of the optimal tree solution; by $\mathrm{OPT}(I)$ the cost of the optimal solution; by $\mathrm{OPT}_{\text{frac}}(I)$ the cost of the optimal solution to the fractional relaxation. Trivially, $\mathrm{OPT}_{\text{frac}}(I) \leq \mathrm{OPT}(I) \leq \mathrm{OPT}_{\text{tree}}(I)$. Analogously, we define $\mathrm{OPT}_{\text{tree}}(J)$ and $\mathrm{OPT}(J)$ for an instance J of the CR problem.

[2]Note that the csVPN problem is hard. In fact, the Steiner tree problem is a restriction of csVPN: let $b_v := 1$ for each terminal and $b_v := 0$ otherwise, and then let $f(x) := x$ for $x \in [0, 1]$ and $f(x) = 1$ for $x \in [1, B]$.

We now discuss whether for the csVPN problem or the CR problem we can assume without loss of generality that b is a 0-1 vector.

Given an instance $I = (G, b, c, f)$ of the csVPN problem (resp. an instance $J = (G, r, b, c, g)$ of the CR problem) such that b is not a 0-1 vector, we may define a new instance that we denote by $\tilde{I} = (\tilde{G}, \tilde{b}, \tilde{c}, f)$ (resp. $\tilde{J} = (\tilde{G}, \tilde{r}, \tilde{b}, \tilde{c}, g)$). To define $\tilde{I}$ from I, we proceed as follows. For each terminal v with $b_v > 1$, we add $k := b_v$ pendant edges $vu_1, \ldots, vu_k$ with cost zero, and let $\tilde{b}_v = 0$ and $\tilde{b}_{u_i} = 1$ for $i = 1, \ldots, k$. To define $\tilde{J}$ from J, we proceed similarly and let $\tilde{r}$ be one of the vertices pending from r except if $b_r = 1$ in which case we let $\tilde{r} = r$. We skip the proof of the following result.

Lemma 2. *Let I, $\tilde{I}$ be* csVPN *instances as above, and let J and $\tilde{J}$ be* CR *instances as above. Then the following statements hold.*

(i) $\mathrm{OPT}(\tilde{I}) \leq \mathrm{OPT}(I)$; $\mathrm{OPT}_{\text{frac}}(\tilde{I}) = \mathrm{OPT}_{\text{frac}}(I)$; $\mathrm{OPT}_{\text{tree}}(\tilde{I}) = \mathrm{OPT}_{\text{tree}}(I)$.

(ii) $\mathrm{OPT}(\tilde{J}) = \mathrm{OPT}(J)$; $\mathrm{OPT}_{\text{tree}}(\tilde{J}) = \mathrm{OPT}_{\text{tree}}(J)$.

It follows that for the CR problem (in *general* graphs) we can assume without loss of generality that b is a 0-1 vector. (Combining this remark with Lemma 1 it follows that our definition of the PR problem is consistent with that in Grandoni et al. [4] and Goyal et al. [3].)

2.2. **From SSBB to ndCR.** Our approximation algorithm for the csVPN problem, given in the next section, builds upon an approximation algorithm for the ndCR problem. Note that the the latter problem is also NP-hard. Nevertheless, as is easily seen by considering shortest paths trees, there always exists an optimal solution that is a tree solution. Recall that for an instance (G, r, b, c, f) of the ndCR problem, the function f is always assumed to be non-decreasing.

Lemma 3. *For any instance I of the* ndCR *problem we have* $\mathrm{OPT}(I) = \mathrm{OPT}_{\text{tree}}(I)$.

As we show below, there exists an approximation factor preserving reduction from the ndCR problem to the *Single Source Buy at Bulk* (SSBB) problem. The SSBB problem is defined as follows: we are given an undirected graph $G = (V, E)$ with edge costs $c \in \mathbb{R}_+^E$, where each vertex $v \in V$ wants to exchange an amount of flow $b_v \in \mathbb{Z}_+$ with a common source vertex r. In order to support the traffic, we can install cables on edges. Specifically we can choose among k different cables: each cable $i \in \{1, \ldots, k\}$ provides $\mu(i)$ units of capacity at price $p(i)$. For each $i \in \{1, \ldots, k-1\}$, it is assumed that $\mu(i) < \mu(i+1)$ and $\frac{p(i)}{\mu(i)} \geq \frac{p(i+1)}{\mu(i+1)}$. The latter inequality is referred to as the *economy of scale principle*. The goal is to find a minimum cost installation of cables such that a flow of value b_v can be routed simultaneously from r to each vertex $v \in V$. An instance of the SSBB problem is therefore defined by a quintuple (G, r, b, c, K), where $K = \{(\mu(i), p(i)) \mid i = 1, \ldots, k\}$ describes the different cable types.

A solution to the SSBB problem specifies a routing $q \in \mathbb{R}_+^{\mathcal{A}}$ and, for each edge e, a multiset κ_e of cables to install (repetitions are allowed, that is, we can install several cables of the same type). A solution (q, κ) is feasible if q sends b_v units of flow from r to each vertex v (exactly as in the ndCR problem) and $\sum_{i \in \kappa_e} \mu(i) \geq \phi_e$ for all edges e, where $\phi = \phi(q)$ denotes the flow vector of q. The cost of the feasible solution (q, κ) is $\sum_{e \in E} \sum_{i \in \kappa_e} c_e \, p(i)$. As for the other problems, we let $\mathrm{OPT}(I)$ denote the cost of the optimal solution for a SSBB instance I.

We point out that there is some confusion in the literature in the definition of the SSBB problem, because there are two different definitions of the problem: In some papers the SSBB problem is defined above, and in some other papers the SSBB problem is defined as the problem we call ndCR. It is stated (see, e.g., Gupta et al. [10]) that from an approximation viewpoint, the two formulations are equivalent up to a factor of 2, but we could not find a proof of this statement to refer to. Therefore, in the appendix we provide a proof for the following lemma, that is enough for our purpose.

Lemma 4. *There exists an approximation factor preserving reduction from* ndCR *problem to the* SSBB *problem.*

We refer to the appendix for the proof. By combining Lemmas 1 and 4, we obtain the following corollary.

Corollary 5. *There exists a ρ-approximation algorithm for* `ndCR` *problem returning a routing that is unsplittable.*

The best approximation algorithm for the `SSBB` problem is currently due to Grandoni and Italiano [9], with $\rho = 24.92$. Therefore, there exists a 24.92-approximation algorithm for the `ndCR` problem.

2.3. **An approximation algorithm for the `csVPN` problem.** The following theorem, which will be proved in Section 3.3, shows that there always exists an optimal solution to the `csVPN` problem that is a tree solution.

Theorem 6. *There always exists an optimal solution to the* `csVPN` *problem that is a tree solution.*

In order to state our approximation algorithm for the `csVPN` problem we need two other results from the literature.

First, let $I = (G, b, c, f)$ be an instance of the `csVPN` problem and let T be a tree that connects all the terminals. As noted above in Section 1.1, it is straightforward to compute the minimum amount of capacity we have to reserve on each edge of T in order to get a feasible virtual private network using for each pair of terminals the unique path in T connecting them. We denote the cost of the resulting feasible virtual private network by $z(T)$.

For any choice of root $r \in V(T)$, one can derive from T a tree solution to the `ndCR` instance $I(r) = (G, r, b(r), c, f)$, where we let $b_v(r) := b_v$ for all vertices $v \neq r$, and $b_r(r) := \max\{b_r, 1\}$. Indeed, we can simply route the b_v units of demand to each terminal v using the unique r–v path contained in T. We denote the resulting routing by $q_{T,r}$ and its cost by $z(q_{T,r})$. The next lemma easily follows from results of Gupta et al. [5, Lemma 2.1] (see also Italiano et al. [8, Lemma 2.4]).

Lemma 7. *There exists a vertex r of T such that $z(T) = z(q_{T,r})$.*

Next, suppose that we are given a feasible solution $q(r)$ to an instance $J(r) = (G, r, b(r), c, f)$ of the `ndCR` problem. By Lemma 1, we may assume that $q(r)$ is unsplittable. As observed by Goyal et al. [3], we can build a feasible solution to the instance $J = (G, b(r), c, f)$ of the `csVPN` problem as follows: for each pair of terminals u, v, choose the path P_{uv} to be any path in $P_u \Delta P_v$ from u to v, where P_u and P_v respectively denote the unique r–u and r–v paths in the support of $q(r)$. Let $\mathcal{P}$ be the union of the selected P_{uv} paths. Recall that we may efficiently deduce from $\mathcal{P}$ the minimum capacity reservation γ such that $(\mathcal{P}, \gamma)$ is a feasible virtual private network. Let $z(\mathcal{P}) := \sum_{e \in E} c_e \gamma_e$. The following lemma is straightforward:

Lemma 8. $z(\mathcal{P}) \leq z(q(r))$.

We are now ready to describe our approximation algorithm for the `csVPN` problem. The input to the algorithm is a `csVPN` instance (G, b, c, f).

Algorithm 1 Approximation algorithm for `csVPN`

(1) For each $r \in V$, compute a ρ-approximate unsplittable solution $q(r)$ to the `ndCR` instance $(G, r, b(r), c, f)$.
(2) Let r^* be such that $z(q(r^*)) = \min_{r \in V} z(q(r))$.
(3) From $q(r^*)$ build a solution $(\mathcal{P}, \gamma)$ to the `csVPN` instance (G, b, c, f) as for Lemma 8.
(4) Output $(\mathcal{P}, \gamma)$.

Theorem 9. *Algorithm 1 is a ρ-approximation algorithm for* `csVPN`.

Proof of Theorem 9. Consider an optimal solution to an instance $I = (G, b, c, f)$ of the `csVPN` problem with cost $\mathrm{OPT} = \mathrm{OPT}(I)$. Let T be an optimal tree for I, thus, $\mathrm{OPT}_{\text{tree}}(I) = z(T)$. From Theorem 6 we know that $\mathrm{OPT} = z(T)$. By Lemma 7, $z(T) \geq \min_{r \in V(T)} z(q_{T,r})$. Since $q_{T,r}$ is a solution to the `ndCR` instance $(G, r, b(r), c, f)$, it follows that $\min_{r \in V(T)} z(q_{T,r}) \geq \min_{r \in V} \mathrm{OPT}((G, r, b(r), c, f)) =$

$\mathrm{OPT}(G, \tilde{r}, b(\tilde{r}), c, f)$, for some $\tilde{r} \in V$. By construction, $\mathrm{OPT}(G, \tilde{r}, b(\tilde{r}), c, f) \geq \frac{1}{\rho} z(q(\tilde{r})) \geq \frac{1}{\rho} z(q(r^*))$. From Lemma 8, we have $z(q(r^*)) \geq z(\mathcal{P})$. Putting everything together, we obtain $\rho \mathrm{OPT}(G, b, c, f) \geq z(\mathcal{P})$. □

By Corollary 5 and the results by Grandoni and Italiano [9], we conclude that there exists a 49.84-approximation algorithm for the csVPN problem.

3. Tree routings

3.1. **From csVPN to asCR.** We show here that the csVPN problem is equivalent to the asCR problem with h symmetric, when b is a 0-1 vector. The proof of the next lemma builds upon a generalization of results in [5], [4] and [3]. For $f\colon [0, B] \to \mathbb{R}_+$ concave and non-decreasing, we define

(1) $$h\colon [0, B] \to \mathbb{R}_+ : x \mapsto \begin{cases} f(x) & \text{if } x \leq B/2, \\ f(B - x) & \text{if } x \geq B/2. \end{cases}$$

Then h is concave and (axis-)symmetric.

Lemma 10. *Let (G, b, c, f) be an instance of the* csVPN *problem with $b \in \{0, 1\}^V$, and (G, r, b, c, h) an instance of the* asCR *problem with h as in* (1). *The value of the optimal solution is the same for both problems. Moreover, there exists an optimal solution to (G, b, c, f) that is a tree solution if and only if there exists an optimal solution to (G, r, b, c, h) that is a tree solution.*

We refer the reader to the appendix for the proof. It follows from Lemmas 2 and 10 that if one shows that every asCR instance with $b \in \{0, 1\}^V$ has a tree solution that is optimal, then it follows that every csVPN instance admits a tree solution that is optimal. The following definition is central to this section.

Definition (*Tree property*). An instance (G, r, b, c, h) of the asCR problem has the *tree property (w.r.t. h)* if there exists an optimal routing that is a tree routing. A graph has the *tree property (w.r.t h)* if for every choice of r, b and c, the instance (G, r, b, c, h) has the tree property.

3.2. **Some tools.** In the remainder of the paper, $h\colon [0, B] \to \mathbb{R}_+$ will be a fixed concave symmetric function.

We now develop a few tools for the asCR problem. We start with some notations. Let $\mathcal{P}$ be a routing for an instance (G, r, b, c, h) and let e be an edge. We let:

$$y_e(\mathcal{P}) := h(\phi_e(\mathcal{P}))$$

Note that $y(\mathcal{P})$ is a vector in $\mathbb{Z}^E$ and the cost of the routing $\mathcal{P}$ is then $\sum_{e \in E} c_e y_e(\mathcal{P})$. When there is no risk of confusion, we simply write y for $y(\mathcal{P})$.

Given a graph $G = (V, E)$, root $r \in V$ and demands $b \in \mathbb{Z}_+^V$ we define the *Concave Routing polyhedron* (or asCR *polyhedron*) $Q = Q(G, r, b, h)$ as the dominant of the convex hull of the y-vectors of routings in G (it is a polyhedron, since the number of routings is finite). Thus we have

$$Q := \mathrm{conv}\{y(\mathcal{P}) \in \mathbb{R}^E : \mathcal{P} \text{ is a routing in } G\} + \mathbb{R}_+^E.$$

Solving an instance of the asCR problem of the form (G, r, b, c, h) amounts to minimizing the linear function $y \mapsto c^T y$ over the corresponding asCR polyhedron $Q(G, r, b, h)$. This is used in the next lemma which provides a way to state the tree property without referring to edge costs.

Lemma 11. *The tree property holds for a certain graph G if and only if, for each extreme point $y(\mathcal{P})$ of the* asCR *polyhedron, there exists a tree routing $\mathcal{T}$ such that $y(\mathcal{P}) = y(\mathcal{T})$. In other words, the tree property holds for G if and only if for any routing $\mathcal{P}$ in G there exists a collection of tree routings $\mathcal{T}_1, \ldots, \mathcal{T}_\ell$ and non-negative coefficients $\lambda_1, \ldots, \lambda_\ell$ summing up to* 1 *such that*

(2) $$\sum_{i=1}^{\ell} \lambda_i \, y(\mathcal{T}_i) \leq y(\mathcal{P}).$$

If y and y' are two vectors in $\mathbb{R}_+^E$ such that $y' \leq y$ we say that y is *dominated* by y'. So if a routing $\mathcal{P}$ satisfies (3) for some choice of tree routings $\mathcal{T}_i$ and non-negative coefficients λ_i summing up to 1, then the y-vector of $\mathcal{P}$ is dominated by the corresponding convex combination of y-vectors of tree routings. So proving the tree property amounts to proving that the y-vector of any routing is dominated by a convex combination of y-vectors of tree routings.

3.3. **Tree Property.**

Theorem 12. *Given any concave and symmetric function h, every instance (G, r, b, c, h) of the* asCR *problem has the tree property.*

Proof. Let $\mathcal{P}$ be a routing for an instance (G, r, b, c, h) of the asCR problem. In the following, we simply write $p_e(\mathcal{P})$ instead of $p(\phi_e(\mathcal{P}))$.

Let $p : [0, B] \to \mathbb{R}_+$ be the pyramidal function $p(x) = \min\{x, B - x\}$, $\forall x \in [0, B]$. By definition, $p(x) \leq B/2$.

Observe that (G, r, b, c, p) is an instance of both the PR and the asCR problem. From Goyal et al. [3], we know that (G, r, b, c, p) has the tree property. Therefore, by Lemma 11, there exists a collection of tree routings $\mathcal{T}_1, \ldots, \mathcal{T}_\ell$ and non-negative coefficients $\lambda_1, \ldots, \lambda_\ell$ summing up to 1 such that for each edge $e \in E$ the following holds:

$$\sum_{i=1}^{\ell} \lambda_i \, p(\phi_e(\mathcal{T}_i)) \leq p(\phi_e(\mathcal{P})). \tag{3}$$

Since h is symmetric, for each edge e we have: $h(\phi_e(\mathcal{P})) = h(B - \phi_e(\mathcal{P})) = h(p(\phi_e(\mathcal{P})))$. Moreover, by Lemma 13.(d) we know that h is not decreasing in the interval $[0, B/2]$. Therefore, since $p(\phi_e(\mathcal{P})) \leq B/2$, it follows that:

$$h(\phi_e(\mathcal{P})) = h(p(\phi_e(\mathcal{P}))) \geq h(\sum_{i=1}^{\ell} \lambda_i \, p(\phi_e(\mathcal{T}_i))).$$

Since h is concave and symmetric an, for each $i = 1..\ell$, $h(\phi_e(\mathcal{T}_i)) = h(B - \phi_e(\mathcal{T}_i)) = h(p(\phi_e(\mathcal{T}_i)))$:

$$h(\sum_{i=1}^{\ell} \lambda_i \, p(\phi_e(\mathcal{T}_i))) \geq \sum_{i=1}^{\ell} \lambda_i h(p(\phi_e(\mathcal{T}_i))) = \sum_{i=1}^{\ell} \lambda_i h(\phi_e(\mathcal{T}_i)).$$

Therefore, $h(\phi_e(\mathcal{P})) \geq \sum_{i=1}^{\ell} \lambda_i h(\phi_e(\mathcal{T}_i))$ and the statement follows from Lemma 11.

□

We are now ready to show that also the csVPN problem always amits an optimal solution tothat is a tree solution.

Proof of Theorem 6. First, consider an instance (G, b, c, f) of the csVPN with $b_v \in \{0, 1\}$ for each $v \in V$. Here the statement follows from Lemma 10 and Theorem 12. Now suppose some vertices have demand greater than 1. We can reduce to the previous case by adding, for each vertex v with $b_v > 1$, b_v pendant edges $vu_1, \ldots, vu_{b_v}$ with cost zero to the graph and letting $b_v = 0$ and $b_{u_i} = 1$ for $i = 1, \ldots, b_v$. The new instance that is a tree solution. Trivially, it follows that also the original instance has an optimal solution that is a tree solution.

3.4. **A remark on non-symmetric concave funtions.** It follows from the results of the previous section that the tree property holds for the CR problem when g is non-decreasing, g is pyramidal and, for outerplanar graphs, when g is symmetric. Moreover, we are not aware of any instance with g symmetric where it does not hold.

An example in the appendix shows that, in general, the tree property does *not* hold when g is not symmetric, even if $g(x) \leq g(B - x)$, for each $x \in [0, B/2]$, and G is a ring network. It is also possible to slightly modify the example as to show that the tree property does not hold when $g(x) \geq g(B-x)$, for each $x \in [0, B/2]$.

REFERENCES

[1] T. Erlebach and M. Rüegg. Optimal Bandwidth Reservation in Hose-Model VPNs with Multi-Path Routing. In *Proceedings of the Twenty-Third Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM*, 4:2275-2282, 2004.

[2] J.A. Fingerhut, S. Suri, J.S. Turner. Designing least-cost nonblocking broadband networks. In *Journal of Algorithms*, volume 24(2): 287–309, 1997.

[3] N. Goyal, N. Olver, and B. Shepherd. The VPN conjecture is true. To appear in *Proceedings of the ACM Symposium on Theory of Computing*, 2008.

[4] F. Grandoni, V. Kaibel, G. Oriolo, and M. Skutella. A short proof of the VPN tree routing conjecture on ring networks. To appear in *Operation Research Letters*.

[5] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, and B. Yener. Provisioning a virtual private network: A network design problem for multicommodity flow. In *Proceedings of the 33rd ACM Symposium on Theory of Computing*, 389–398, ACM Press, 2001.

[6] A. Gupta, and A. Kumar. Personal communication, 2002.

[7] C. A. J. Hurkens, J. C. M. Keijsper, and L. Stougie. Virtual private network design: A proof of the tree routing conjecture on ring networks. *SIAM Journal on Discrete Mathematics*, 21:482–503, 2007.

[8] G. Italiano, S. Leonardi, and G. Oriolo. Design of trees in the hose model: the balanced case. *Operation Research Letters*, 34(6): 601–606, 2006.

[9] F. Grandoni, and G. Italiano. Improved Approximation for Single-Sink Buy-at-Bulk. *International Symposium on Algorithms and Computation (ISAAC)*, 111–120, 2006.

[10] A. Gupta, A. Kumar, M. Pàl and T. Roughgarden. Approximation via Cost Sharing: Simpler and Better Approximation Algorithms for Network Design. *Journal of the ACM (JACM)*, 54(3):11, 2007.

APPENDIX

Basics on concave functions. We give a few simple facts concerning concave functions. Consider a function $f : C \to \mathbb{R}$ defined over a convex subset C of $\mathbb{R}^d$. The function f is *concave* if $f(\lambda x + \mu y) \geq \lambda f(x) + \mu f(y)$ holds for all $x, y \in C$ and $\lambda, \mu \in \mathbb{R}_+$ such that $\lambda + \mu = 1$. In other words, concave functions are those for which the image of a convex combination is greater than or equal to the corresponding convex combination of images. The definition states this for convex combinations involving two points. The general case follows easily by induction.

When $d = 1$ and f is defined over the interval $[0, B]$ for some nonnegative number B, we say that it is *(axis-)symmetric* if $f(B - x) = f(x)$ for all $x \in [0, B]$.

Lemma 13. *Let $f : [0, B] \to \mathbb{R}_+$ be a concave function. Then the following assertions hold.*
(a) For all $\alpha \in [0, 1]$ and $a \in [0, B]$ we have $\alpha f(a) \leq f(\alpha a)$.
(b) We have $f(y) - f(y - a) \leq f(x) - f(x - a)$ for all $a, x, y \in [0, B]$ with $a \leq x \leq y$.
(c) If f is symmetric and not identically 0 then we have $f(x) > 0$ for all $x \in (0, B)$.
(d) If f is symmetric then f is non-decreasing in the interval $[0, B/2]$.
(e) If f is symmetric then for all $0 \leq x \leq y \leq B/2$ we have $f(y + x) \geq f(y - x)$.

Proof. (a) Since f is a non-negative concave function, we have

$$f(\alpha a) = f(\alpha a + (1 - \alpha)0) \geq \alpha f(a) + (1 - \alpha)f(0) \geq \alpha f(a).$$

(b) If $x = y$ the assertion trivially holds. Thus we may assume $x < y$. We may also assume $x \leq y - a$. Indeed, otherwise we redefine a, x and y as $y-x$, $y-a$ and y respectively. Letting $\alpha = (y-x-a)/(y-x)$, we have $x = \alpha(x - a) + (1 - \alpha)(y - a)$ and $y - a = (1 - \alpha)x + \alpha y$. By the concavity of f, we get $f(x) \geq \alpha f(x-a)+(1-\alpha)f(y-a)$ and $f(y-a) \geq (1-\alpha)f(x)+\alpha f(y)$. Adding the two last inequalities, we obtain $\alpha(f(y) - f(y - a)) \leq \alpha(f(x) - f(x - a))$. The assertion follows whenever $\alpha > 0$.

Now assume $\alpha = 0$, that is, $x = y - a$. In this case, we have $x = \frac{1}{2}(x - a) + \frac{1}{2}y$ and $f(x) \geq \frac{1}{2}f(x-a)+\frac{1}{2}f(y)$. Since the last inequality is equivalent to $f(y)-f(x) \leq f(x)-f(x-a)$ and $x = y-a$, the assertion follows.

(c) Suppose, by contradiction, that $f(b) = 0$ for some $b \in (0, B)$. Using symmetry, we can suppose that $b \leq B/2$. Using (a) with $b = \alpha a$ and the non-negativity of f, we have $f(a) = 0$ for $a \geq b$. By symmetry, we have $f(a) = 0$ for $a \leq b$. Therefore, f is identically 0, a contradiction.

(d) We again argue by contradiction: Suppose that $f(a) > f(b)$ for some a, b such that $0 \leq a < b \leq B/2$. Then $b = \lambda a + \mu(B - a)$ for some $\lambda, \mu \in \mathbb{R}_+$ with $\lambda + \mu = 1$. Because f is concave and symmetric, we have $f(b) = f(\lambda a + \mu(B - a)) \geq \lambda f(a) + \mu f(B - a) = \lambda f(a) + \mu f(a) = f(a)$, a contradiction.

(e) Take x and y such that $0 \leq x \leq y \leq B/2$. Notice that $B - (y - x) \geq y + x$ since $B/2 \geq y$. Then, $y + x = \lambda(y - x) + \mu(B - (y - x))$ for some $\lambda, \mu \in \mathbb{R}_+$ with $\lambda + \mu = 1$. Recalling that f is concave and symmetric, it follows that $f(y+x) \geq \lambda f(y-x)+\mu f(B-(y-x)) = \lambda f(y-x)+\mu f(y-x) = f(y-x)$. □

Proofs for Section 2.1.

Proof of Lemma 1. Let $I = (G, r, b, c, g)$ be an instance of CR, and let q be a feasible routing for I.

Now consider some terminal v. Let $\mathcal{P}_{rv} = \{P_1, \dots, P_t\}$ denote the set of all r–v paths contained in the support of q. From q, we define t routings $q_1, \dots, q_t$, as follows. For $i \in \{1, \dots, t\}$, we let $q_i(P) = b_v$ if $P = P_i$, $q_i(P) = 0$ if $P = P_j$ with $j \neq i$, and $q_i(P) = q(P)$ otherwise. In other words, q_i routes all b_v units of demand to v on the single path P_i and otherwise behaves as q.

The key observation is that q is a convex combination of $q_1, \dots, q_t$. More precisely, we have

$$q = \sum_{i=1}^{t} \frac{q(P_i)}{b_v} q_i.$$

By the concavity of the cost function $q \mapsto z(q)$, there exists $i \in \{1, \ldots, t\}$ such that q_i does not cost more than q. The result then follows by induction. □

Proofs for Section 2.2.

Proof of Lemma 4. Let $I = (G, r, b, c, f)$ be an instance of `ndCR` (thus, f is non-decreasing). Consider the instance $J = (G, r, b, c, K)$ of `SSBB` obtained by setting

$$K := \{(1, f(1)), (2, f(2)) \ldots, (B, f(B))\}.$$

Thus, in instance J, the cable $i \in \{1, \ldots, B\}$ has capacity $\mu(i) := i$ and price $p(i) := f(i)$. The capacity of the cables are clearly increasing. By Lemma 13.(a), the economy of scale principle is satisfied.

In order to prove the result it suffices to show the following: (i) From a solution to I one may build, in polynomial time, a solution to J of the same cost; (ii) From a solution to J one may build, in polynomial time, a solution to I that does not cost more.

(i) Each solution to I yields a solution to J of the same cost. In virtue of Lemma 1, we may assume that the solution to I is unsplittable. In particular, it is integral. Now take the same routing and install on each edge e a single cable of capacity $\phi_e \in \mathbb{Z}_+$, where ϕ_e denotes the total flow routed on edge e.

(ii) Conversely, a solution to J yields a solution to I: take the same routing. We now compare the costs of the two solutions. The cost of the latter solution is $\sum_{e \in E} c_e f(\phi_e)$ where, as above, ϕ_e denotes the amount of flow routed on e. (Notice that this time ϕ_e is not necessarily integral.) The cost of the former solution is $\sum_{e \in E} c_e \sum_{i \in \kappa_e} p(i) = \sum_{e \in E} c_e \sum_{i \in \kappa_e} f(i)$.

Consider some edge e and let $\gamma_e := \sum_{i \in \kappa_e} \mu(i) = \sum_{i \in \kappa_e} i$. Without loss of generality, we may assume that $\gamma_e \leq B$. Indeed, if this is not the case we can repeatedly replace some cable of capacity $\mu(j) = j$ by a cable of capacity $\mu(j-1) = j - 1$. This does not increase the cost of the solution.

By Lemma 13.(a), we have that, for $i \in \kappa_e$, $\frac{i}{\gamma_e} f(\gamma_e) \leq f(i)$, thus $\sum_{i \in \kappa_e} \frac{i}{\gamma_e} f(\gamma_e) \leq \sum_{i \in \kappa_e} f(i)$, that is, $f(\gamma_e) \leq \sum_{i \in \kappa_e} f(i)$. On the other hand, we have $f(\phi_e) \leq f(\gamma_e)$ because f is non-decreasing and $\phi_e \leq \gamma_e$. Hence, we have $f(\phi_e) \leq \sum_{i \in \kappa_e} f(i)$. Because this holds for all edges e, the cost of the `ndCR` solution does not exceed that of the `SSBB` solution. The result follows. □

Proofs for Section 3.1.

Proof of Lemma 10. For a generic set $\mathcal{S}$ of paths, let $n_e(\mathcal{S})$ be the number of paths in $\mathcal{S}$ using the edge e. Let $(\mathcal{P}, \gamma)$ be a feasible virtual private network for (G, b, c, f), with $\mathcal{P} = \{P_{ij} : i \neq j \in W\}$. For each terminal i, let $\mathcal{P}_i = \{P_{ij} : j \in W \setminus \{i\}\}$. It is shown in [5] (Theorem 3.2) and [4] (Lemma 3) that the following holds:

$$\gamma_e \geq \frac{1}{B} \sum_{i \in W} \min\{n_e(\mathcal{P}_i), B - n_e(\mathcal{P}_i)\}.$$

Notice that from $\mathcal{P}_i$ we can build an unsplittable routing q_i for the instance of `ndCR` (G, i, b, c, h), simply letting $q_i(P) = b_v$ for each $P = P_{iv} \in \mathcal{P}_i$, and $q_i(P) = 0$ otherwise. Moreover, notice that in this case $n(\mathcal{P}_i) = \phi(q_i)$.

Since f is concave and non-decreasing we have:

$$\sum_{e \in E} c_e f(\gamma_e) \geq \sum_{e \in E} c_e f(\frac{1}{B} \sum_{i \in W} \min\{n_e(\mathcal{P}_i), B - n_e(\mathcal{P}_i)\})$$

$$(*) \qquad \geq \frac{1}{B} \sum_{e \in E} c_e \sum_{i \in W} f(\min\{n_e(\mathcal{P}_i), B - n_e(\mathcal{P}_i)\})$$

Suppose vice versa that we are given a routing q_r for (G, r, b, c, h). From Lemma 1 we can assume q_r to be an unsplittable routing. Let $\mathcal{Q}_r := \{Q_i, i \in W\}$ be the set of path from r to i defined by q_r. Once again, notice that $\phi(q_r) = n(\mathcal{Q}_r)$. Following [3] (Lemma 2.3), we define a collection of paths $\tilde{\mathcal{Q}} = \{\tilde{Q}_{ij} : i \neq j \in W\}$, where $\tilde{Q}_{ij}$ is any $i - j$ path in the component of $Q_i \Delta Q_j$. Let $z(\tilde{\mathcal{Q}})$ be the minimum amount of capacity that we must install on edge e so that $(\tilde{\mathcal{Q}}, z(\tilde{\mathcal{Q}}))$ is a feasible virtual private network for (G, b, c, f). It is shown in [3] that the following holds:

$$\delta_e \leq \min\{n_e(\mathcal{Q}_r), B - n_e(\mathcal{Q}_r)\}.$$

Since f is concave and non-decreasing and the previous inequality holds for each $r \in W$:

$$(**) \qquad \sum_{e \in E} c_e f(\delta_e) \leq \min_{i \in W} \sum_{e \in E} c_e f(\min\{n_e(\mathcal{Q}_i), B - n_e(\mathcal{Q}_i)\})$$

Then the statement easily follows from inequalities $(*)$ and $(**)$. □

Proofs for Section 3.2.

Proof of Lemma 11. For the backward implication, let $\mathcal{P}$ be an optimal solution to an instance of the asCR Problem with respect to some cost vector $c \in \mathbb{R}^E_+$. Then (3) implies $\sum_{i=1}^{\ell} \lambda_i \, c^T y(\mathcal{T}_i) \leq c^T y(\mathcal{P})$. So at least one of the tree routings $\mathcal{T}_1, \dots, \mathcal{T}_\ell$ has a cost which does not exceed the cost of $\mathcal{P}$. That is, at least one of the tree routings is optimal.

Let us now prove the forward implication by contradiction. Suppose that the tree property holds for G but the asCR polyhedron has an extreme point $y(\mathcal{P})$ such that there is no tree routing $\mathcal{T} : y(\mathcal{P}) = y(\mathcal{T})$. Then we can separate $y(\mathcal{P})$ from the other points of the asCR polyhedron by a hyperplane. Because dominants are upper-monotone, it follows that there exists a non-negative cost vector c such that $c^T y(\mathcal{P}) < c^T y(\mathcal{Q})$ for all routings $\mathcal{Q}$ such that $y(\mathcal{Q}) \neq y(\mathcal{P})$. In particular, we have $c^T y(\mathcal{P}) < c^T y(\mathcal{T})$ for all tree routings $\mathcal{T}$, a contradiction. The result follows. □

Example for Section 3.4.

Example 14. Consider an instance (G, r, b, c, g) of the CR problem, where G is a ring with vertices $V(G) = \{0, 1, 2, 3, 4\}$ (the vertices of the cycle are numbered consecutively clockwise).

Let $r := 0$; $b_i := 1, i = 0, \dots, 4$; $c_e := M$ for $e = \{3, 4\}$, $c_e := M + \epsilon$ for $e = \{0, 1\}$, $c_e := 0$ otherwise. Finally, let g be defined by linear interpolation of the following points: $g(0) = 0, g(2) = 2, g(3) = 2 + 2\epsilon, g(5) = 0$. It is easy to check that g is concave, non-negative, non-symmetric and $g(x) \leq g(B - x)$, for each $x \in [0, B/2]$.

Consider the routing q where the paths from 0 to i go counterclockwise for $i = 1, 2, 3$, while the path from 0 to 4 goes clockwise. The cost of this solution is $(2 + \epsilon)M + \epsilon$, and it is easy to check that taking ϵ and M respectively small and big enough, there is no cheaper feasible tree routing.