On the Complexity of the Asymmetric VPN Problem

Thomas Rothvoß and Laura Sanità⋆

Institute of Mathematics, EPFL, Lausanne, Switzerland
{thomas.rothvoss,laura.sanita}@epfl.ch

Abstract We give the first constant factor approximation algorithm for the asymmetric Virtual Private Network (Vpn) problem with arbitrary concave costs. We even show the stronger result, that there is always a tree solution of cost at most $2 \cdot OPT$ and that a tree solution of (expected) cost at most $49.84 \cdot OPT$ can be determined in polynomial time.
For the case of linear cost we obtain a $(2+\varepsilon\frac{\mathcal{R}}{\mathcal{S}})$-approximation algorithm for any fixed $\varepsilon > 0$, where $\mathcal{S}$ and $\mathcal{R}$ ($\mathcal{R} \geq \mathcal{S}$) denote the outgoing and ingoing demand, respectively.
Furthermore, we answer an outstanding open question about the complexity status of the so called *balanced* Vpn problem by proving its **NP**-hardness.

1 Introduction

The asymmetric Virtual Private Network (Vpn) problem is defined on a communication network represented as an undirected connected graph $G = (V, E)$ with cost vector $c : E \to \mathbb{Q}_+$, where c_e indicates the cost of installing one unit of capacity on edge e. Within this network, there is a set of terminal nodes that want to communicate with each other, but the amount of traffic between pairs of terminals is not known exactly. Instead, each vertex v has two thresholds $b_v^+, b_v^- \in \mathbb{N}_0$, representing the cumulative amount of traffic that v can send and receive, respectively. The bounds implicitly describe a set of *valid* traffic matrices which the network has to support. In particular, a traffic matrix specifies for each *ordered* pairs of vertices (u, v), a non-negative amount of traffic that u wishes to send to v. Such a set of traffic demands corresponds to a valid traffic matrix if and only if the total amount of traffic entering and leaving each terminal v does not exceed its bounds b_v^- and b_v^+, respectively.

A solution to an instance of the asymmetric Vpn problem is given by a collection of paths $\mathcal{P}$ containing exactly one path for each ordered pair of terminals, and a capacity reservation $x : E \to \mathbb{Q}_+$. Such a solution $(\mathcal{P}, x)$ is *feasible* if every valid traffic matrix can be routed via the paths in $\mathcal{P}$ without exceeding the capacity reservation x. The aim is to find a feasible solution that minimizes the total cost of the installation.

⋆ Supported by Swiss National Science Foundation within the project "Robust Network Design"

A feasible solution is called a *tree solution* if the union of the selected paths induces a tree.

The VPN problem was introduced by Fingerhut et al. [1] and Gupta et al. [2], and it soon attracted a lot of attention in the network design community. In fact, the model is relevant for many practical applications where flexible communication scenarios are needed, e.g. to face phenomena like input data uncertainty, demands that are hard to forecast as well as traffic fluctuations, which are typical for instance in IP networks.

Such a high interest in the problem motivated several authors (see e.g. [3,4,5,6,7,8,9,10]) in the investigation of the model and its important variations. A recent survey on network design problems provided by Chekuri [11] reports a lot of interesting open questions concerning VPN models, some of them discussed below.

1.1 Related Work

The *asymmetric* VPN problem is **APX**-hard, even if we restrict to tree solutions [1,2]. The current best approximation algorithm gives a ratio of 3.55 [4]. Still, the best known upper bound on the ratio between an optimal solution and an optimal tree solution is 4.74 [9].

A quite natural variant of this problem is the so-called *balanced* VPN problem, that is, when the following condition holds: $\sum_v b_v^+ = \sum_v b_v^-$. Italiano et al. [5] show that, differently from the asymmetric version, an optimal tree solution in this case can be found in polynomial time, and Eisenbrand et al. [4] obtain that an optimal tree solution is in fact a 2-approximate solution for the general case. Unfortunately, it has been recently shown that the cheapest solution does not always have a tree structure [12]. Nevertheless, the complexity of the balanced VPN problem is still an open question [11,5].

Finally, an important variant of this problem is the *symmetric* VPN problem, where each vertex has one single integer bound b_v representing the total amount of traffic that v can exchange with the other nodes: in this case, a solution specifies an $u - v$ path for each *unordered* pair of nodes and a capacity reservation vector in such a way that every valid traffic matrix can be routed via the selected paths, where a valid traffic matrix now specifies an amount of flow that each unordered pair of nodes wishes to exchange, without exceeding the given threshold for each node. Both papers [1] and [2] show that an optimal tree solution can be computed in polynomial time. It has been conjectured in Erlebach et al. [3] and in Italiano et al. [5] that there always exists an optimal solution to the symmetric VPN problem that is a tree solution: this has become known as the *VPN tree routing conjecture*. The conjecture has first been proved for ring networks [6,7], and was finally settled for general graphs by Goyal et al. [8].

Recently, Fiorini et al. [10] started the investigation of the symmetric VPN problem with concave costs. More precisely, the concave symmetric VPN problem is defined as the symmetric VPN problem, but the contribution of each edge to the total cost is proportional to some concave non-decreasing function of the capacity reservation. The motivation for studying this problem is due to the fact that buying capacity can often reflect an economy of scale principle: the more capacity is installed, the less is the per-unit reservation cost. They give a constant factor approximation algorithm for the problem, and show that also in this case there always exists an optimal solution that has a tree structure. An alternative subsequent proof of the latter result is also given by Goyal et al. [13]. The investigation of the concave asymmetric VPN problem has not been addressed so far.

The importance of tree solutions becomes more evident in the context of symmetric VPN and balanced VPN, where any tree solution has in fact a *central hub node*, as shown by [2] for the symmetric case and by [5] for the balanced case. More precisely, any tree solution in these cases has enough capacity such that *all* the terminal nodes could simultaneously route their traffic to some hub node r in network. Combining this with some simple observations, it follows that computing the cheapest tree solution reduces to computing the cheapest way to simultaneously send a given amount flow from the terminal nodes to some selected hub node r. In case of linear edge costs [2,5], the latter min-cost flow problem becomes simply a shortest path tree problem. In case that the edge costs are proportional to a non-decreasing concave cost function [10], the latter min-cost flow problem is known as *Single Sink Buy-At-Bulk* (SSBB) problem (a formal definition is given in the next section). Differently, the above property does not hold for tree solutions of asymmetric VPN instances.

We point out that in the literature there is another possible definition of SSBB that does not compute costs according to a concave cost function, but instead deals with an input set of possible cable types that may be installed on the edges, each with different capacity and cost. For this latter version of the problem, the first constant approximation (roughly 2000) is due to Guha et al. [14], subsequently reduced to 216 by Talwar [15] and to 76.8 by Gupta et al. [16], with an algorithm based on random sampling. Refining their approach, the approximation was later reduced to 65.49 by Jothi and Raghavachari [17], and eventually to 24.92 by Grandoni and Italiano [18]. In this paper, according to the first definition, we however refer to SSBB as the problem of routing a given amount of flow from some terminal nodes to a hub node minimizing a concave cost function on the capacity installed on the edges. It is shown in [10] that the (expected) 24.92-approximation algorithm of Grandoni and Italiano [18] can be used to obtain a tree solution with the same approximation factor for our version of SSBB.

1.2 Our Contribution

We give the first constant factor approximation algorithm for the asymmetric VPN problem with arbitrary concave costs, showing that a tree solution of expected cost at most $49.84 \cdot OPT$ can be computed in polynomial time. Moreover, in case of linear cost, we show that for any fixed $\varepsilon > 0$ a $(2 + \varepsilon\frac{\mathcal{R}}{\mathcal{S}})$-approximate solution can be obtained in polynomial time, with $\mathcal{R} := \sum_v b_v^-$, $\mathcal{S} := \sum_v b_v^+$, and without loss of generality $\mathcal{R} \geq \mathcal{S}$.

The key-point of our approximation results is showing that there always exists a cheap solution with a *capacitated* central hub node, which in particular has a cost of at most twice the optimum. More precisely, there exists a 2-approximate solution with enough capacity such that any subset of terminals could simultaneously send their flow to a hub node r up to a cumulative amount of $\mathcal{S}$. Then, we show how to approximate such a centralized solution by using known results on SSBB. Based on this, we can then state that there exists a VPN *tree solution*, with cost at most $2 \cdot OPT$. This substantially improves the previous known upper bound of 4.74 on the ratio between an optimal solution and an optimal tree solution, which only applies in case of linear costs. We remark that our result holds considering any non-decreasing concave cost function.

The technique used to prove our results is substantially different from the previous approaches known in literature. In fact, approximation algorithms developed in the past mostly relate on computing bounds on the *global cost* of an optimal solution, e.g. showing that an approximate solution constructed out of several matchings or Steiner trees, has a total cost that is not that far from the optimum [9,4,16].

In contrast, we focus locally on the *capacity* installed on an edge, and we show that, given any feasible solution, we can obtain a new solution with a capacitated central hub node, such that, on average the capacity on an edge is at most doubled. This result is independent on the cost function. Still, we reinterpret the known fact that, given a set of paths, the minimal amount of capacity to install on an edge can be computed by solving a bipartite matching problem on some auxiliary graph. Using duality, we look instead at minimal *vertex covers* on such graphs, and this reinterpretation allows us to develop a very simple analysis for our statement.

Eventually, we answer the open question regarding the complexity status of the balanced VPN problem with linear costs. We prove that it is **NP**-hard even with unit thresholds on each node.

2 Description of the Problem

In this section we describe in detail the problem addressed in this paper, and other related problems that we will use to state our results.

(Concave/Linear) Virtual Private Network. An instance $\mathcal{I}$ of the *concave Virtual Private Network* (cVPN) problem consists of an undirected connected graph $G = (V, E)$ with edge costs $c : E \rightarrow \mathbb{Q}^+$, two non-negative integer vectors $b^+ \in \mathbb{Z}^V, b^- \in \mathbb{Z}^V$, as well as a concave non-decreasing function $f : \mathbb{Q}_+ \rightarrow \mathbb{Q}_+$.

A vertex v such that $b_v^+ + b_v^- > 0$ is referred to as a *terminal*: by duplicating nodes, we can assume without loss of generality that each terminal is either a *sender* s, with $b_s^+ > 0, b_s^- = 0$, or a *receiver* r, with $b_r^+ = 0, b_r^- > 0$. Let S and R be set of senders and receivers, respectively.

The vectors b^+ and b^- specify a set of valid traffic matrices that can be interpreted as follows. Let $K_{S,R}$ be the complete bipartite graph with nodes partitioned into senders and receivers: each valid traffic matrix corresponds to a *fractional b-matching* on $K_{S,R}$ and vice versa.

A solution to an instance of the problem is a pair $(\mathcal{P}, x)$, where $\mathcal{P}$ is a collection of paths $\mathcal{P} := \{P_{sr} \mid \forall r \in R, s \in S\}$, and $x \in \mathbb{Q}_+^E$ specifies the capacity to install on each edge of the network. A solution is *feasible* if the installed capacities suffice to route each valid traffic matrix via the selected paths $\mathcal{P}$. A feasible solution is *optimal* if it minimizes the emerging cost $\sum_{e \in E} c_e \cdot f(x_e)$.

If $f(x_e) = x_e$, that means we have *linear* costs on the edges, we term this problem just *Virtual Private Network* (VPN) problem. We call an instance of the problem *balanced* whenever $\mathcal{S} := \sum_{s \in S} b_s^+$ equals $\mathcal{R} := \sum_{r \in R} b_r^-$.

Given a collection of paths $\mathcal{P}$, the minimum amount of capacity x_e that has to be install on $e \in E$ to turn $(\mathcal{P}, x)$ into a feasible solution can be computed in polynomial time as follows (see [2,5,4] for details):

$$x_e = \text{maximal cardinality of a } b\text{-matching in } G_e = (S \cup R, E_e), \\ \text{with } (s, r) \in E_e \Leftrightarrow e \in P_{sr}$$

Notice that, since the graph G_e is bipartite, an optimum capacity reservation vector x will always be integer.

Single Sink Buy-At-Bulk. An instance of the *Single Sink Buy-At-Bulk* (SSBB) problem consists of an undirected connected graph $G = (V, E)$ with edge costs $c : E \to \mathbb{Q}_+$, a demand function $d : V \to \mathbb{N}$, a root $r \in V$ and a concave non-decreasing function $f : \mathbb{Q}_+ \to \mathbb{Q}_+$.

The aim is to find capacities $x_e \in \mathbb{Q}_+$ for the edges, sufficient to simultaneously route a demand of $d(v)$ from each node v to the root, such that the emerging cost $\sum_{e \in E} c_e \cdot f(x_e)$ is minimized.

Sometimes it is assumed that $d(v) \in \{0, 1\}$, and in this case the vertices $D = \{v \in V \mid d(v) = 1\}$ are called *clients*.

Single Sink Rent-or-Buy. An instance of the *Single Sink Rent-or-Buy* (SROB) problem consists of an undirected connected graph $G = (V, E)$ with edge costs $c : E \to \mathbb{Q}_+$, a demand function $d : V \to \mathbb{N}$, a root $r \in V$ and a parameter $M \geq 1$.

The aim is to find capacities $x_e \in \mathbb{Q}_+$ for the edges, sufficient to simultaneously route a demand of $d(v)$ from each node v to the root, such that the emerging cost $\sum_{e \in E} c_e \cdot \min\{x_e, M\}$ is minimized. Note that this problem is a special case of Single Sink Buy-At-Bulk.

Steiner Tree. An instance of the *Steiner tree* problem consists of an undirected connected graph $G = (V, E)$ with edge costs $c : E \to \mathbb{Q}_+$ and a set of terminals $K \subseteq V$.

The aim is to find the cheapest tree $T \subseteq E$ spanning the terminals.

3 Approximation Results

We now state the first constant factor approximation algorithm for cVPN, starting with some simplifying assumptions that we can make on a cVPN instance without loss of generality.

First, by duplicating nodes, we may assume b^+, b^- to be 0/1 vectors, that means, $b^+_s = 1, b^-_s = 0$ for a sender s, and $b^+_r = 0, b^-_r = 1$ for a receiver r. The latter assumption is correct if we can guarantee that the paths in a solution between copies of a terminal v and copies of a terminal u are all the same. Our algorithm developed below can be easily adapted in such a way that it satisfies the latter consistence property, and that it runs in polynomial time even if the thresholds are not polynomially bounded. Note that, under these assumptions, $\mathcal{S} = |S|$ and $\mathcal{R} = |R|$. Then by symmetry, suppose that $|R| \geq |S|$.

We propose the following algorithm.

Algorithm 1 cVPN algorithm

1. Choose a sender $s^* \in S$ uniformly at random as the *hub*
2. Compute a ρ_{SSBB}-approximate SSBB tree solution $(x_e)_{e \in E}$ for graph G with clients $S \cup R$, root s^* and cost function $c_e \cdot f(\min\{x_e, |S|\})$
3. Return $((P_{sr})_{s \in S, r \in R}, x')$ with path P_{sr} being the unique path in the tree defined by the support of x_e, and $x'_e = \min\{x_e, |S|\}$

Note that $f(\min\{x_e, |S|\})$ indeed is concave and non-decreasing in x_e. Let $OPT := OPT_{\text{VPN}}(\mathcal{I})$ be the optimum cost for the cVPN instance $\mathcal{I}$.

Let us first argue, that the capacity reservation x'_e in fact suffices. Consider an edge e, which is used by k paths in the SSBB solution. Then the capacity reservation is $x'_e \geq \min\{k, |S|\}$. It is easy to see that this is sufficient for the constructed cVPN solution. Clearly the cost of this solution is equal to the cost of the SSBB-solution.

We will now show that indeed, there is a SSBB-solution of cost at most $2 \cdot OPT$ for the instance defined in Step (2) of the algorithm. As it was pointed out in [10], any solution for SSBB can then be turned into a tree solution of at most the same cost[1].

To prove this, we first define $\mathcal{I}'$ as a modified cVPN instance, which differs from $\mathcal{I}$ in such a way that there is a *single* sender with *non-unit* threshold, and in particular:

$$b^+_v(\mathcal{I}') = \begin{cases} |S| & \text{if } v = s^* \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad b^-_v(\mathcal{I}') = \begin{cases} 1 & \text{if } v \in S \cup R \\ 0 & \text{otherwise} \end{cases}$$

[1] This is not true anymore, if the function f is not concave, but defined by a set of cables. In that case one might loose a factor of 2 in the approximation.

Intuitively we reroute all flow through the hub s^*. We will now prove that this new cVPN instance coincides with the SSBB problem, i.e. their optimum values are identical.

Let OPT_{SSBB} be the cost of an optimum SSBB solution for the instance defined in Step (2) of the algorithm.

Lemma 1. $OPT_{\text{VPN}}(\mathcal{I}') = OPT_{\text{SSBB}}$.

Proof. Let P_{s^*v} be the paths in a cVPN solution for $\mathcal{I}'$. Consider an edge $e \in E$ and let $v_1, \ldots, v_k \in S \cup R$ be the nodes, such that $e \in P_{s^*v_i}$. If $k \leq |S|$ we can define a traffic matrix in which s^* sends 1 unit of flow to all v_i. If $k > |S|$, we may send 1 unit of flow from s^* to each node in $v_1, \ldots, v_{|S|}$. Anyway the needed capacity of e is $x_e = \min\{k, |S|\}$, which costs $c_e \cdot f(\min\{k, |S|\})$. This is the same amount, which an SSBB solution pays for capacity k on $e \in E$. Thus both problems are equal. $\square$

The critical point is to show that:

Lemma 2. $E[OPT_{\text{VPN}}(\mathcal{I}')] \leq 2 \cdot OPT_{\text{VPN}}(\mathcal{I})$.

Proof. Let $\mathcal{P} = \{P_{sr} \mid s \in S, r \in R\}$ be the set of paths in the optimum cVPN solution for $\mathcal{I}$ and x_e be the induced capacities. We need to construct a cVPN solution of $\mathcal{I}'$, consisting of s^*-v paths P'_{s^*v} for $v \in S \cup R$.

The solution is surprisingly simple: Choose a receiver $r^* \in R$ uniformly at random as a second hub. Take $P'_{s^*r} := P_{s^*r}$ as s^*-r path. Furthermore concatenate $P'_{s^*s} := P_{s^*r^*} + P_{r^*s}$ to obtain a s^*-s path. To be more precise we can shortcut the latter paths, such that they do not contain any edge twice.

We define a sufficient capacity reservation x'_e as follows: Install $|S|$ units of capacity on the path $P_{s^*r^*}$. Then for each sender $s \in S$ (receiver $r \in R$) install in a cumulative manner one unit of capacity on P_{sr^*} (on P_{s^*r}, respectively). Note that x'_e is a random variable, depending on the choice of s^* and r^*. We show that $E[x'_e] \leq 2x_e$. Once we have done this, the claim easily follows from Jensen's inequality and concavity of f:

$$E[OPT_{\text{VPN}}(\mathcal{I}')] \leq E[\sum_{e \in E} c_e f(x'_e)] \leq \sum_{e \in E} c_e f(E[x'_e])] \leq 2 \cdot OPT_{\text{VPN}}(\mathcal{I})$$

Now consider an edge $e \in E$. Since we want to bound the quantity $E[x'_e]$ in terms of the original capacity x_e, let us inspect, how this capacity is determined. Define the bipartite graph $G_e = (S \cup R, E_e)$ containing an edge $(s, r) \in E_e$ if and only if $e \in P_{sr}$. Then x_e must be the cardinality of a maximal matching in G_e. Kőnig's theorem (see e.g. [19,20]) says that there is a vertex cover $C \subseteq S \cup R$ with $x_e = |C|$ (see Figure 1 for a visualization).

We now distinguish two cases and account their expected contribution to $E[x'_e]$.

1. Case: $s^* \in S \cap C$ or $r^* \in R \cap C$. We account the worst case of $|S|$ units of capacity. The expected contribution is then

$$\Pr[(s^* \in S \cap C) \vee (r^* \in R \cap C)] \cdot |S| \leq \frac{|S \cap C|}{|S|} \cdot |S| + \frac{|R \cap C|}{|R|} \cdot |S| \leq |C|$$

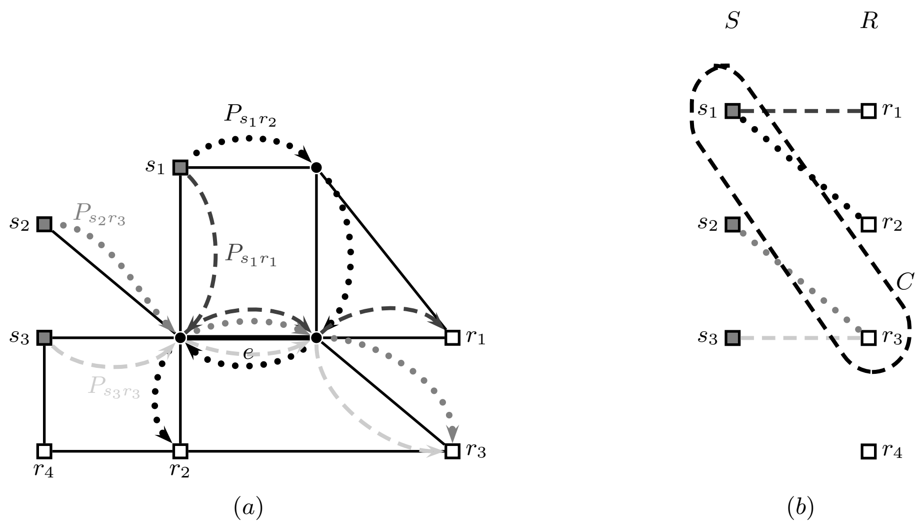

Figure 1. Example of a cVPN instance in (a), where terminals are depicted as rectangles, senders are drawn solid. Only paths, crossing edge e are shown. In (b) the graph G_e with vertex cover C is visualized, implying that $x_e = 2$.

using $|R| \geq |S|$.

2. Case: $s^* \in S \backslash C, r^* \in R \backslash C$. We bound the probability of this case by 1. We know that edge (s^*, r^*) cannot exist in G_e since all edges need to be incident to C. Consequently e does not lie on the path $P_{s^*r^*}$. Thus we just have to install 1 unit of capacity for each sender s, such that $(s, r^*) \in E_e$. But only sender in $S \cap C$ may be adjacent to r^* in G_e, thus this number is at most $|S \cap C|$. A similar argument holds for the receivers. The expected contribution of this case is consequently upperbounded by $|S \cap C| + |R \cap C| = |C|$.

Combining the expected capacities for both cases we derive that $E[x'_e] \leq 2|C| = 2x_e$, which implies the claim. $\square$

As a consequence, our algorithm yields a $2\rho_{\text{SSBB}}$-approximation. Using the expected 24.92-approximation of [18], we conclude

Theorem 1. *There is an expected* 49.84*-approximation algorithm for* cVPN *which even yields a tree solution.*

Using the derandomized SSBB algorithm of van Zuylen [21] with an approximation factor of 27.72 and the fact that all choices for $s^* \in S$ can be easily tried out, one obtains

Corollary 1. *There is a deterministic factor* 55.44*-approximation algorithm for* cVPN*, which even yields a tree solution.*

Corollary 2. *Given any* cVPN *solution of cost* α*, one can find deterministically and in polynomial time a tree solution of cost at most* 2α*.*

Until now the best upper bound on the ratio of optimum solution by optimum tree solution was $3 + \sqrt{3} \approx 4.74$ due to [9] which only worked in case of linear cost.

3.1 Linear Costs

Next suppose that $f(x_e) = x_e$, meaning that we have linear costs on the edges. The 3.55-approximation algorithm of [4] still yields the best known ratio for VPN.

Observe that the cost function $c_e \cdot f(\min\{x_e, |S|\}) = c_e \cdot \min\{x_e, |S|\}$ for the SSBB instance constructed in the algorithm, matches the definition for the Single Sink Rent-or-Buy problem (SROB) with parameter $M = |S|$, root s^* and clients $S \cup R$, thus any ρ_{SROB}-approximate SROB algorithm can be turned into a $2\rho_{\text{SROB}}$-algorithm for VPN.

In general $\rho_{\text{SROB}} \leq 2.92$ is the best known bound due to [22], but in a special case we can do better. In [22] it was proved, that for any constant $\delta > 0$, there is a $1 + \delta\frac{|D|}{M}$-approximation algorithm for SROB. Since $D = S \cup R$ is the set of clients and $M = |S|$, this directly yields

Corollary 3. *For any fixed* $\varepsilon > 0$*, there is a polynomial time* $(2 + \varepsilon\frac{\mathcal{R}}{\mathcal{S}})$*-approximation algorithm for* VPN*.*

Recall that this result also holds in case of non-unit demands.

4 Hardness of Balanced VPN

We here consider the balanced VPN problem with linear costs. Recall that, while the asymmetric VPN is **NP**-hard even restricted to tree solutions, an optimal tree solution for this case can be computed in polynomial time as in the symmetric version [5]. So far, the complexity of the balance VPN was an open question [5,11]: we now show that the problem is **NP**-hard even with unit thresholds on the nodes, by reduction from the Steiner Tree problem.

Given an instance $\mathcal{I}$ for Steiner Tree consisting of a graph $G = (V, E)$ with cost function $c : E \to \mathbb{Q}_+$, and set of $k + 1$ terminals $\{v_1, \ldots, v_k, v_{k+1}\}$, we construct an instance $\mathcal{I}'$ of the balanced VPN problem on a graph $G' = (V', E')$ as follows.

First, introduce two large numbers: $C := \sum_{e \in E} c_e + 1$, and $M \gg (k + 1)C$. To construct G' from G, add a vertex a_4 and make it adjacent to the vertices $v_1, v_2, \ldots, v_k$ by edges of cost C. Then, add a path $v_{k+1}, a_1, a_2, a_3, a_4$, where the first two edges of the path have cost M, while the last two edges have cost kM. Finally, add k vertices $w_1, w_2, \ldots, w_k$, each of them adjacent to a_2 with a zero cost edge, and add $2k - 1$ vertices $u_1, u_2, \ldots, u_{2k-1}$, each of them adjacent to a_3 with a zero cost edge. Figure 2 shows the resulting graph G'.

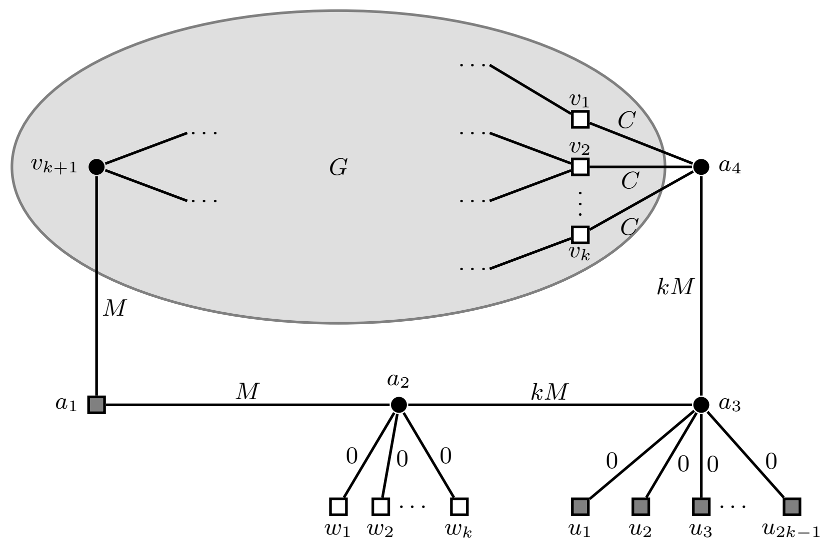

Figure 2. VPN instance $\mathcal{I}'$. Edges are labeled with their cost. Terminals are depicted as rectangles, senders are drawn in gray.

Define the set of senders as $S := \{a_1\} \cup \{u_1, u_2, \ldots, u_{2k-1}\}$ and the set of receivers as $R := \{v_1, v_2, \ldots, v_k\} \cup \{w_1, w_2, \ldots, w_k\}$. Note that indeed $|S| = |R|$.

Lemma 3. *There exists a solution to the Steiner tree instance $\mathcal{I}$ of cost at most C^* if and only if there exists a solution to the balanced* VPN *instance $\mathcal{I}'$ with cost at most $Z = 2k^2M + 2M + kC + C^*$.*

Proof. ($\Rightarrow$) The only if part is trivial. Suppose there exists a solution T to the Steiner tree instance $\mathcal{I}$ of cost C^*. We construct a solution to $\mathcal{I}'$ by defining the following paths:

- $P_{a_1 w_i} = \{a_1, a_2\} \cup \{a_2, w_i\}$, for $i = 1, \ldots, k$;
- $P_{a_1 v_i} = \{a_1, v_{k+1}\} \cup \{$the edges of the unique $(v_{k+1} - v_i)$-path induced by $T\}$, for $i = 1, \ldots, k$;
- $P_{u_j w_i} = \{u_j, a_3\} \cup \{a_3, a_2\} \cup \{a_2, w_i\}$, for $i = 1, \ldots, k$, $j = 1, \ldots, 2k-1$;
- $P_{u_j v_i} = \{u_j, a_3\} \cup \{a_3, a_4\} \cup \{a_4, v_i\}$, for $i = 1, \ldots, k$, $j = 1, \ldots, 2k-1$.

Finally, install the following amount of capacity on the edges of the graph: $x_e = k$ for $e = \{a_2, a_3\}$ and $e = \{a_3, a_4\}$, $x_e = 0$ for $e \in E \setminus T$ and $x_e = 1$ otherwise. It is easy to see that the resulting set of paths and the capacity vector x define a solution to $\mathcal{I}'$ of cost at most Z.

($\Leftarrow$) For the reverse direction, suppose we have a VPN solution $(\mathcal{P}, x)$ to $\mathcal{I}'$ with cost at most Z. Recall that we may assume x to be an integer vector. We now have to argue that in fact this solution must be of the same structure as suggested in the ($\Rightarrow$) part.

First, we show that the paths in $\mathcal{P}$ from a_1 to v_i ($i = 1, \ldots, k$) contain the edge $\{a_1, v_{k+1}\}$, while the paths from a_1 to w_i ($i = 1, \ldots, k$) contain the edge $\{a_1, a_2\}$. Similarly, we show that the paths from u_j to v_i ($i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$) contain the edge $\{a_3, a_4\}$, while the paths from u_j to w_i ($i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$) contain the edge $\{a_2, a_3\}$. Then, under the above assumptions, we show that the support of the solution contains a set of edges $T \subseteq E$ that span the vertices $\{v_1, \ldots, v_{k+1}\}$ and whose total cost does not exceed C^*. The result then follows.

Claim 1. *For $i = 1, \ldots, k$, we have $\{a_1, v_{k+1}\} \in P_{a_1 v_i}$ and $\{a_1, a_2\} \in P_{a_1 w_i}$.*

Proof. Suppose there is a path from a_1 to some node v_i that does not contain the edge $\{a_1, v_{k+1}\}$. Necessarily, it must contain the edges $\{a_1, a_2\}$, $\{a_2, a_3\}$ and $\{a_3, a_4\}$. This means, that the capacity to be installed on the latter edges fulfills $x_{a_1 a_2} \geq 1$ and $x_{a_2 a_3} + x_{a_3 a_4} \geq (2k-1) + 2$, where the last inequality easily follows considering the valid traffic matrix, in which a_1 sends 1 unit of flow to v_i and the remaining senders send $2k-1$ units of flow to the remaining receivers. Therefore, the cost of the emerging solution is at least $2k^2M + kM + M > Z$ for every $k \geq 2$, yielding a contradiction. We can prove in a similar manner that $\{a_1, a_2\} \in P_{a_1 w_i}$ for all $i = 1, \ldots, k$. □

Claim 2. *For $i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$, we have $\{a_3, a_4\} \in P_{u_j v_i}$ and $\{a_2, a_3\} \in P_{u_j w_i}$.*

Proof. First, we focus on the capacity installed on the edges $e = \{a_2, a_3\}$ and $e' = \{a_3, a_4\}$. Clearly, $x_e + x_{e'} \geq 2k-1$. We now prove that in fact the inequality is strict.

Suppose it holds with equality. We inspect the bipartite graphs G_e and $G_{e'}$ (this time without a_1, since we already proved that it uses neither e nor e') and let $C_e, C_{e'}$ be the minimum vertex covers on $G_e, G_{e'}$, respectively. By hypothesis, $|C_e| + |C_{e'}| = 2k-1$. That means that there is at least one node $r \in R$ that does not belong to any of the two covers. Now notice that G_e and $G_{e'}$ are complementary bipartite graphs, since the union of their edges gives the complete bipartite graph $K_{2k-1,2k}$. It follows than that $C_e \cup C_{e'} = S \setminus \{a_1\}$: otherwise, there would be a node $s \in S \setminus (\{a_1\} \cup C_e \cup C_{e'})$ with an incident edge (s, r), that is neither covered by C_e nor by $C_{e'}$, a contradiction.

As a conclusion all senders in C_e route to the $2k$ receivers on paths containing the edge e, while all senders in $C_{e'}$ route to the $2k$ receivers on paths containing the edge e'. Then it is easy to see, that the installed capacities satisfy $x_{v_{k+1} a_1} \geq k$ and $x_{a_1 a_2} \geq k$. Therefore, the cost of the emerging solution is at least $2k^2M - kM + 2kM > Z$, for $k \geq 3$, a contradiction.

It follows that $x_e + x_{e'} \geq 2k$. Suppose now, there is a path from some u_j to some v_i that does not contain the edge $e = \{a_3, a_4\}$. Necessarily, it must contain the edges $\{a_1, a_2\}$ and $\{v_{k+1}, a_1\}$. Using the previous claim, it is easy to see that the installed capacities satisfy $x_{v_{k+1} a_1} \geq 2$ and similarly $x_{a_1 a_2} \geq 2$. Therefore, the cost of the emerging solution is at least $2k^2M + 4M > Z$, again a contradiction.

We can prove in a similar manner that there is no path from some u_j to some w_i that does not contain the edge $e' = \{a_2, a_3\}$. □

Putting all together, it follows that $x_{v_{k+1}a_1} \geq 1$, $x_{a_1a_2} \geq 1$, $x_{a_2a_3} \geq k$, $x_{a_3a_4} \geq k$, and the cost of the capacity installed on the latter edges is at least $2k^2M + 2kM$.

Now, consider the edges $\{a_4, v_i\}$, $i = 1, \dots, k$: clearly, $\sum_{i=1,\dots,k} x_{a_4v_i} \geq k$, since we can define a traffic matrix where k senders in $S \setminus \{a_1\}$ simultaneously send k units of flow to $v_1, \dots, v_k$. It follows that $\sum_{i=1,\dots,k} c_{a_4v_i} x_{a_4v_i} \geq k \cdot C$.

Finally, let T be the subset of edges of E that are in the support of the solution. Suppose that T does not span the nodes $v_1, \dots, v_{k+1}$. Then there exists at least one node v_i such that the path from a_1 to v_i contains at least 2 edges with cost C. But in this case, we would have $\sum_{i=1,\dots,k} x_{a_4v_i} \geq k + 1$ and the cost of the solution exceeds Z. We conclude that indeed T contains a Steiner tree and $c(T) \leq Z - (2k^2M + 2M + kC) = C^*$. □

From the discussions above, it follows:

Theorem 2. *The balanced* VPN *problem is* **NP***-hard.*

Note that the above reduction is not approximation preserving, i.e. in contrast to **NP**-hardness, the **APX**-hardness of Steiner tree [23] is not conveyed to balanced VPN. In other words, our reduction does not exclude the possible existence of a PTAS for balanced VPN.

References

1. Fingerhut, J., Suri, S., Turner, J.: Designing least-cost nonblocking broadband networks. In: Journal of Algorithms, 24(2). (1997) 287–309
2. Gupta, A., Kleinberg, J., Kumar, A., Rastogi, R., Yener, B.: Provisioning a virtual private network: A network design problem for multicommodity flow. In: Proc. of the 33rd ACM Symposium on Theory of Computing (STOC). (2001) 389–398
3. Erlebach, T., Rüegg, M.: Optimal bandwidth reservation in hose-model vpns with multi-path routing. In: Proc. of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM, 4. (2004) 2275–2282
4. Eisenbrand, F., Grandoni, F., Oriolo, G., Skutella, M.: New approaches for virtual private network design. In: SIAM Journal on Computing, 37(3). (2007) 706–721
5. Italiano, G., Leonardi, S., Oriolo, G.: Design of trees in the hose model: the balanced case. In: Operation Research Letters, 34(6). (2006) 601–606
6. Hurkens, C., Keijsper, J., Stougie, L.: Virtual private network design: A proof of the tree routing conjecture on ring networks. In: SIAM Journal on Discrete Mathematics, 21. (2007) 482–503
7. Grandoni, F., Kaibel, V., Oriolo, G., Skutella, M.: A short proof of the vpn tree routing conjecture on ring networks. In: Operation Research Letters, 36(3). (2008) 361–365
8. Goyal, N., Olver, N., Shepherd, B.: The vpn conjecture is true. In: Proc. of the 40th annual ACM symposium on Theory of computing (STOC). (2008) 443–450
9. Eisenbrand, F., Grandoni, F.: An improved approximation algorithm for virtual private network design. In: Proc. of the 16th ACM-SIAM Symposium on Discrete Algorithms (SODA). (2005)
10. Fiorini, S., Oriolo, G., Sanità, L., Theis, D.: The vpn problem with concave costs. Submitted manuscript (2008)

11. Chekuri, C.: Routing and network design with robustness to changing or uncertain traffic demands. In: SIGACT News, 38(3). (2007) 106–128
12. Sanità, L.: Robust Network Design. PhD thesis, Università Sapienza di Roma (2008)
13. Goyal, N., Olver, N., Shepherd, B.: Personal communication. (2008)
14. Guha, S., Meyerson, A., Munagala, K.: A constant factor approximation for the single sink edge installation problems. In: Proc. of the 33rd Annual ACM Symposium on Theory of Computing (STOC), New York (2001) 383–388
15. Talwar, K.: The single-sink buy-at-bulk LP has constant integrality gap. In: Proc. of the 9th International Conference on Integer programming and combinatorial optimization (IPCO). (2002) 475–486
16. Gupta, A., Kumar, A., Roughgarden, T.: Simpler and better approximation algorithms for network design. In: Proc. of the 35th Annual ACM Symposium on Theory of Computing (STOC), New York (2003) 365–372
17. Jothi, R., Raghavachari, B.: Improved approximation algorithms for the single-sink buy-at-bulk network design problems. In: Proc. of the 9th Scandinavian Workshop on Algorithm Theory (SWAT). (2004) 336–348
18. Grandoni, F., Italiano, G.: Improved approximation for single-sink buy-at-bulk. In: Proc. of the 17th International Symposium on Algorithms and Computation (ISAAC). (2006) 111–120
19. Cook, W., Cunningham, W., Pulleyblank, W., Schrijver, A.: Combinatorial optimization. John Wiley & Sons Inc., New York (1998)
20. Schrijver, A.: Combinatorial optimization. Polyhedra and efficiency. Vol. A. Volume 24 of Algorithms and Combinatorics. Springer-Verlag, Berlin (2003)
21. van Zuylen, A.: Deterministic sampling algorithms for network design. In: Proc. of the 16th annual European symposium on Algorithms (ESA). (2008) 830–841
22. Eisenbrand, F., Grandoni, F., Rothvoß, T., Schäfer, G.: Approximating connected facility location problems via random facility sampling and core detouring. In: Proc. of the 19th ACM-SIAM Symposium on Discrete Algorithms (SODA). (2008) 1174–1183
23. Chlebik, M., Chlebikova, J.: Approximation hardness of the steiner tree problem on graphs. In: Proc. of 8th Scandinavian Workshop on Algorithm Theory (SWAT). (2002)