MATHEMATICS OF OPERATIONS RESEARCH
Vol. 00, No. 0, Xxxxxx 20xx, pp. xxx–xxx
ISSN 0364-765X | EISSN 1526-5471 |xx|0000|0xxx

DOI 10.1287/moor.xxxx.xxxx

From Uncertainty to Non-Linearity: Solving Virtual Private Network via Single-Sink Buy-at-Bulk

Fabrizio Grandoni
Computer Science Department, University of Rome Tor Vergata, Roma, Italy
email: grandoni@disp.uniroma2.it

Thomas Rothvoß
Institute of Mathematics, EPFL, Lausanne, Switzerland
email: thomas.rothvoss@epfl.ch

Laura Sanità
Institute of Mathematics, EPFL, Lausanne, Switzerland
email: laura.sanita@epfl.ch

The *virtual private network* problem (VPN) models scenarios where traffic is uncertain or rapidly changing. The goal is supporting at minimum cost a given family of traffic matrices, which are implicitly given by upper bounds on the ingoing and outgoing traffic at each node. Costs are classically defined by a linear function (*linear* VPN), but we consider here also the more general case of concave increasing costs (*concave* VPN).

In this paper we give the first constant factor approximation for concave VPN, and we improve the best known approximation factor for linear VPN. Our approximation results build upon a novel reduction, based on Kőnig's theorem, which allows us to turn uncertainty of traffic into non-linearity of the objective function. This way, we are able to reduce linear VPN and concave VPN to the *single-sink rent-or-buy* problem (SROB) and the *single-sink buy-at-bulk* problem (SSBB), respectively. Using the machinery developed for the latter two problems plus additional ideas, we are able to improve the approximation ratio for VPN.

Along the way we also obtain, among other results, an improved approximation algorithm for SSBB, and a tighter bound on the gap between the costs of arbitrary solutions and tree solutions for VPN. Furthermore, solving an open problem, we show that VPN remains **NP**-hard even in the *balanced* case, where the sum of ingoing and outgoing traffic bounds is equal.

Key words: approximation algorithms; randomized algorithms; virtual private network; buy-at-bulk; rent-or-buy.

MSC2000 Subject Classification: Primary: 68W25; Secondary: 68W20, 68W40

OR/MS subject classification: Primary: Analysis of algorithms; Secondary: Suboptimal algorithms, Computational complexity

History: Received: July 20, 2010; Revised: February 4, 2011.

1. Introduction. In a classical network design problem, we are given a graph and a traffic matrix, specifying the amount of flow that we need to route between any pair of nodes. The goal is reserving capacity on the edges of the graph at minimum cost, so that the traffic can be routed without exceeding the capacity. Different problems differ in the capacity cost function (linear, concave, etc.), in the nature of the traffic matrix (single-sink, multi-commodity, etc.), and in the constraints on the flow paths (splittable, unsplittable, etc.).

A common feature of the above problems is that the traffic matrix is fixed and known. However, real-world applications often involve uncertain and dynamic traffic. Motivated by that, Fingerhut et al. [13] and independently Duffield et al. [7] introduced the *Hose model*. In this model, the capacity reservation must support a family of traffic matrices rather than just one matrix. This family is implicitly given by imposing upper bounds on the total amount of flow that each node can send and receive. (Formal definitions are given in Section 2). The Hose model has been studied by many authors in several variants [8, 9, 11, 14, 18, 20, 23, 25, 27, 28, 38] (see also the survey by Chekuri [4]).

In the variant considered here, the so-called (asymmetric) Virtual Private Network problem (Vpn), we are given an undirected graph $G = (V, E)$, with edge costs (or weights) $w : E \to \mathbb{Q}_{\geq 0}$, and two integers b_v^+ and b_v^- for each node $v \in V$. A traffic matrix is feasible if and only if the total amount of flow entering and leaving each terminal v does not exceed its bounds b_v^- and b_v^+, respectively. Note that, in general, the bounds do not need to be symmetric (i.e., possibly $b_v^- \neq b_v^+$), and even the cumulative bound on the total ingoing demand $\mathcal{R} := \sum_v b_v^-$ can be different from the cumulative bound on the total outgoing demand $\mathcal{S} := \sum_v b_v^+$ (without loss of generality, we will always assume $\mathcal{R} \geq \mathcal{S}$). A solution to the problem consists of a capacity reservation x_e on each edge e, and a flow path P_{uv} for each ordered pair of nodes (u, v). This solution is feasible if any feasible traffic matrix can be routed according to paths P_{uv} without exceeding capacities x_e. The aim is to find a feasible solution that minimizes the total cost $\sum_{e\in E} w(e) \cdot x_e$ of the capacity reservation.

We also consider a generalization of the problem, that we call *concave* Vpn (cVpn), with objective function $\sum_{e\in E} w(e) \cdot \phi(x_e)$, where $\phi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a non-decreasing concave function, with $\phi(0) = 0$ (*capacity cost function*). This generalization combines traffic uncertainty with economies of scale.

1.1 Our Results. In this paper we give some new insights into Vpn and related problems[1]. Our main achievement is a technique to turn traffic uncertainty into non-linearity of the objective function. To that aim, we crucially exploit Kőnig's theorem (see, e.g., [6, 34]): *In a bipartite graph the maximum cardinality of a matching equals the minimum cardinality of a vertex cover.* This reduction allows us to turn the input Vpn instance into a concave-cost flow problem, where the traffic matrix is fixed and known in advance. The latter problem is then solved with existing approximation algorithms.

The concave-cost flow problems which we consider in this paper are the *single-sink rent-or-buy* problem (Srob) and the more general *single-sink buy-at-bulk* problem (Ssbb). Both problems are well-studied in the literature [10, 17, 19, 22, 23, 25, 29, 30, 31, 36, 37]. In Ssbb, we are given an undirected graph $G = (V, E)$, with weights $w(e)$ for each edge e, demands $d(v)$ for each node v, and a capacity cost function $\phi(\cdot)$. A feasible solution is a capacity reservation x_e on each edge e, which supports a simultaneous flow of $d(v)$ units from each node v to a fixed root $z \in V$. The goal is minimizing $\sum_{e\in E} w(e) \cdot \phi(x_e)$. Srob is the special case of Ssbb, where $\phi(x) = \min\{x, M\}$ for a given input parameter $M \geq 1$.

We next describe our results in more detail.

1.1.1 Concave Virtual Private Network. A $O(\log n)$-approximation for cVpn can be obtained by embedding the input metric into a tree metric with logarithmic average distortion [1, 12]. We give the first constant factor approximation for cVpn, showing that a solution of expected cost at most 40.82 times the optimum can be computed in polynomial time.

The key-idea of our approach is showing that there is a solution of cost at most twice the optimum, where all the traffic passes through a central hub node. Henceforth, for a proper choice of the hub and of the capacity-cost function, we can reduce the original problem to a Ssbb problem. A this point we can exploit the machinery developed in the Ssbb literature. In particular, any ρ-approximation for Ssbb leads to a 2ρ-approximation for cVpn.

The technique used to prove our results is substantially different from the previous approaches known in the literature. In fact, 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 and Kőnig's theorem, we rather focus on minimal *vertex covers* on such graphs. This reinterpretation allows us to prove stronger results in a simpler manner.

As a byproduct of our analysis, we also show that there is always a tree solution (that is, a solution whose support is a tree), with cost at most twice the optimum. This substantially improves the previous known upper bound of 4.74 on the ratio between optimal tree and graph solutions in [8], which only applies to the case of linear costs (i.e., to Vpn but not to cVpn).

1.1.2 Linear Virtual Private Network. We improve the approximation factor for the virtual private network problem with linear costs (Vpn) from 3.39 [3, 9] to 2.80.

The main insight in our proof is again a reduction, based on Kőnig's theorem, from Vpn to a concave-

[1] A preliminary version of these results appeared in APPROX'09 [32] and ICALP'10 [21].

cost flow problem, which this time is SROB. Observe that in VPN one has a linear cost function but several traffic matrices to take into account, while in SROB the cost function is non-linear but the traffic matrix is unique. A straightforward adaptation of the analysis for cVPN implies a $2\rho_{\text{SROB}}$ approximation for VPN, where ρ_{SROB} is the approximation factor for SROB. This gives a $2 \cdot 2.80 = 5.60$ approximation, which does not improve on the previous best result. In order to achieve the claimed approximation, we use a more tailored algorithm and analysis. In particular, we combine the random sampling based VPN algorithm in [9] with the *Core-Detouring Technique* developed in the context of SROB [10].

Along the way, we also obtain a $(2 + \varepsilon\frac{\mathcal{R}}{\mathcal{S}})$-approximation algorithm for any fixed $\varepsilon > 0$. Note that the latter approximation factor is smaller than 2.80 when $\mathcal{R} = O(\mathcal{S})$ (*quasi-balanced* VPN).

1.1.3 Balanced Virtual Private Network. The *balanced* VPN problem is the special case of VPN where $\mathcal{S} = \mathcal{R}$, i.e. the total ingoing demand equals the total outgoing demand. This subproblem attracted some attention in the literature, due to its similarities with the *symmetric* version of VPN (see Section 1.2), which was recently discovered [18] to be polynomial-time solvable. The authors of [4, 28] pose as an open problem whether balanced VPN is polynomial-time solvable as well. We show that this not true (unless **P** = **NP**), even with unit bounds on the nodes.

1.1.4 Single-Sink Buy-at-Bulk. We present an improved approximation algorithm for SSBB, with expected 20.41 approximation ratio. This improves over the previous best 23.93 approximation [3, 19]. Note that our reduction from cVPN to SSBB implies a $2 \cdot 20.41 = 40.82$ approximation for the first problem.

Our approximation algorithm in fact solves a variant CABSSBB of SSBB, which is better studied in the literature. In CABSSBB, economies of scale are modeled by defining a set of cable types, each one with a capacity and a cost. It is assumed that the ratio of cost to capacity decreases from smaller to larger cable types. The capacity is reserved by installing a proper number of each cable type on each edge. This can be seen as a discretization of the capacity cost function $\phi(\cdot)$ (or vice versa). Any approximation algorithm for CABSSBB provides an approximation algorithm for SSBB with essentially the same approximation factor (see Section 2).

Our improved approximation for CABSSBB is based on a generalization of the Core-Detouring Theorem in [10]: the *Multi-Core Detouring Theorem*. The goal of the core-detouring technique is to bound the cost of connecting a set of client nodes to a random subset of them. This is achieved by detouring the connection paths through a proper connected subgraph (the *core*). This technique was successfully applied in several network design problems, such as connected facility location and single-sink rent-or-buy, where the choice of a proper core is natural and obvious. Our Multi-Core Detouring Theorem applies also to more complex network design problems, such as CABSSBB, not exhibiting a convenient core.

Combining our approximation with a simple reduction, we also improve the approximation ratio of an unsplittable variant UNSSSBB of CABSSBB from 148.48 [3, 29] to 40.82.

Our main results are summarized in Table 1.

1.2 Related Work. All the problems considered in this paper are well-studied in the literature. We next describe some known results.

1.2.1 Virtual Private Network. The Hose model and VPN were independently defined by Fingerhut, Suri and Turner [13] and by Duffield, Goyal and Greenberg [7]. Since then, this problem was studied by various authors in several variants. In particular, due to technological reasons, the solution is constrained to induce a tree in some variants.

The version of VPN that we refer to in this paper is also called *asymmetric* VPN. VPN is **APX**-hard even when restricted to tree solutions [23]. On the positive side, a $O(\log n)$ approximation can be obtained by applying tree embedding techniques [1, 12]. The same approximation bound holds in the more general polyhedral model, where one needs to support all the traffic matrices in a given polyhedron [4]. Constant approximation algorithms are presented in [8, 9, 23, 25, 38]. The current best 3.55 approximation is due to Eisenbrand, Grandoni, Oriolo, and Skutella [9]. Using the recent improvement of the Steiner tree approximation factor from 1.55 to 1.39 [3], this approximation bound can be refined to 3.39. It is known that the optimum solution is not always a tree. Curiously, the algorithms in [23, 25] construct a tree

Table 1 Comparison between prior work and our results. Expected approximations are marked with $\star$.

Problem	This paper	Prior work
SSBB and CABSSBB approximation	$20.41^\star$	$23.93^\star$ [3, 19]
UNSSSBB approximation	$2\rho_{\text{CABSSBB}} \leq 40.82^\star$	$148.48^\star$ [3, 29]
VPN approximation	$2.80^\star$ $2+\varepsilon\mathcal{R}/\mathcal{S}$ $2\rho_{\text{SROB}}$	$3.39^\star$ [3, 9] $1+\mathcal{R}/\mathcal{S}$ [9]
cVPN approximation	$2\rho_{\text{SSBB}} \leq 40.82^\star$	$O(\log n)$ [12]
gap tree/graph solution for VPN	2	4.74 [8]
gap tree/graph solution for cVPN	2	$O(\log n)$ [12]
complexity of balanced VPN	**NP**-hard	–

solution, while the current best algorithm in [9] does not. We will use a variant of the latter algorithm to achieve our improved approximation bound.

In [9] a $(1+\mathcal{R}/\mathcal{S})$-approximation is presented. This gives a 2-approximation for the *balanced* case $\mathcal{S}=\mathcal{R}$, which improves on the 3-approximation by Italiano, Leonardi, and Oriolo [28]. Here we improve the result in [9], by presenting a $2+\varepsilon$ approximation whenever $\mathcal{R}=O(\mathcal{S})$, for an arbitrary constant $\varepsilon>0$. In [28] it is proved that, differently from the asymmetric version, an optimal tree solution for the balanced case can be computed in polynomial time, and raise the question whether or not the problem is polynomial-time solvable. Although it has been recently shown that the cheapest solution does not always have a tree structure [33], the complexity of the balanced VPN problem is still an open question [4, 28]. We settle this question by showing that the problem remains **NP**-hard in that special case as well.

An important well-studied variant is the *symmetric* version of VPN, where $b_v^+ = b_v^-$ for each v, and one needs to use the same path to route the flow from u to v and from v to u. In [13, 23] a 2-approximation is given for this problem. In the same papers the authors show that an optimal tree solution can be computed in polynomial time. The famous *VPN tree routing conjecture* states that symmetric VPN always has an optimal tree solution, and hence can be solved in polynomial time. In a breakthrough paper [18], Goyal, Olver and Shepherd recently proved that this conjecture is true (see also [20, 27] for former proofs of the conjecture on ring networks, which introduce part of the ideas used in [18]).

In case of concave costs, only the symmetric version of the problem has been investigated so far. Fiorini, Oriolo, Sanità, and Theis [14] show that the generalization of symmetric VPN with concave costs is **APX**-hard, and give a constant approximation for this problem. More precisely, they show that the 24.92-approximation algorithm for SSBB in [19] can be turned into a 24.92-approximation algorithm for symmetric VPN with concave costs. However, their reduction crucially relies on the fact that the bounds are symmetric and that there is an optimal solution with a tree structure, which is not true in our setting. Nevertheless, our results show that a constant approximation exists for the asymmetric case as well.

1.2.2 Buy-at-Bulk and Rent-or-Buy. SSBB has been extensively studied in the literature, in the mentioned variant called CABSSBB. This problem is **NP**-hard, e.g., by reduction from the Steiner tree problem. Meyerson, Munagala, and Plotkin [31] gave an $O(\log n)$ approximation for this problem. Garg, Khandekar, Konjevod, Ravi, Salman, and Sinha [17] described an $O(k)$ approximation, where k is the number of cable types. The first constant approximation is due to Guha, Meyerson, and Munagala [22]: the approximation ratio of their algorithm is roughly 2000. This approximation was reduced to 216 by Talwar [37]. Gupta, Kumar, and Roughgarden [25] described an improved 76.8 approximation algorithm, based on random sampling. Refining their approach, the approximation was later reduced to 65.49 by Jothi and Raghavachari [29], and eventually to 24.92 by Grandoni and Italiano [19].

One can consider an unsplittable version UnsSsbb of CabSsbb, where the flow from each source to the sink must be routed along a unique path. The algorithm by Talwar is a 216-approximation for UnsSsbb as well. Unfortunately, this is not the case for the following improved random-sampling algorithms (i.e., those algorithms do not guarantee that the flow is unsplittable). Jothi and Raghavachari [29] show how to transform the 76.8 approximation algorithm for CabSsbb by Gupta et al. [25] into a $2 \cdot 76.8 = 153.6$ approximation algorithm for UnsSsbb. Their approach is algorithm-specific: it would not work with an (even slightly) different algorithm. (For example, it cannot be applied to the improved CabSsbb algorithm in the same paper). In Section 2, we will describe a procedure which exploits (as a black box) any ρ-approximation algorithm for CabSsbb to obtain a 2ρ-approximation algorithm for UnsSsbb. In particular, this implies a 49.84-approximation for UnsSsbb using the CabSsbb algorithm in [19], and a 40.82-approximation using our refined approximation for CabSsbb.

Srob [23, 30, 36] can be interpreted as the special case of CabSsbb, with only two cable types, one of very small capacity and cost per unit capacity = 1, and the other of cost $M \geq 1$ and very large capacity. The current best approximation ratio for Srob is 2.92 [10].

The improved Steiner tree approximation algorithm in [3], trivially implies improved approximation factors 2.80, 23.93, and 148.48 for Srob, CabSsbb (and hence Ssbb), and UnsSsbb, respectively.

1.2.3 Core Detouring. In a seminal work, Gupta, Kumar, and Roughgarden [25] introduced a random-sampling-based framework to design and analyze approximation algorithms for network design. This way, they achieved improved approximation algorithms for Vpn, Ssbb and Srob (see also [8, 9, 10, 19, 29]). Generalizations and adaptations of their approach were later successfully applied to several other problems, including *multi-commodity rent-or buy* [2, 15, 24], *connected facility location* [10], *stochastic (online) Steiner tree* [15, 16, 26], *universal TSP* [16, 35] and many others.

One of the key ingredients in Gupta et al.'s approach is connecting a set C of client nodes to a randomly and independently sampled subset of them. The shortest-path distances from the client set to the sampled subset are then bounded against the cost of an optimum Steiner tree over the sampled nodes. In a recent work, Eisenbrand, Grandoni, Rothvoß, and Schäfer gave an improved analytical tool, *core detouring*, to bound the connection cost above [10]. The crux of their method is designing a sub-optimal connection scheme, and bounding its cost. In their scheme connection paths are detoured through a proper connected subgraph (*core*). This technique is summarized in their Core Detouring Theorem.

This theorem is existential in flavor: it is sufficient to show the existence of a convenient core G', of small cost and sufficiently close to the clients C. For some network design problems, a natural candidate core is provided by the structure of the optimum solution. For example, the optimum solution for connected facility location and single-sink rent-or-buy contains a Steiner tree T. Applying the Core Detouring Theorem to T leads to improved approximation algorithms for those two problems [10]. In this paper we further extend this framework, by showing that core-detouring can be successfully applied to other network design problems, where the optimum solution does not exhibit any convenient core. In particular, this holds for Vpn and Ssbb. As we will see, the construction of a good core for the considered problems involves a few non-trivial ideas.

1.3 Organization. The rest of this paper is organized as follows. In Section 2 we introduce some preliminary notions. In Sections 3 and 4 we give our results on cVpn and Vpn, respectively. In Section 5 we show that balanced Vpn is **NP**-hard. In Section 6 we give our improved algorithm for Ssbb.

2. Preliminaries.

Problems cVpn and Vpn are formally defined as follows. A *capacity cost function* is any non-decreasing concave function $\phi : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$, with $\phi(0) = 0$.

> Concave Virtual Private Network (cVpn). The input consists of an undirected graph $G = (V, E)$, with edge weights $w : E \to \mathbb{Q}_{\geq 0}$, ingoing and outgoing traffic upper bounds $b^- \in \mathbb{Q}^V_{\geq 0}$ and $b^+ \in \mathbb{Q}^V_{\geq 0}$, and a capacity cost function $\phi(\cdot)$. A traffic matrix $T \in \mathbb{Q}^{V\times V}_{\geq 0}$ is feasible if, for every $v \in V$, $\sum_u T_{u,v} \leq b^-_v$ and $\sum_u T_{v,u} \leq b^+_v$. The goal is to compute a set of paths $P = \{P_{uv}\}_{(u,v)\in V\times V}$ and a capacity reservation $x \in \mathbb{Q}^E_{\geq 0}$ such

that: (1) the cost $\sum_{e\in E} w(e)\cdot\phi(x_e)$ is minimized and (2) for any feasible traffic matrix T, $\sum_{P_{uv}\ni e} T_{u,v} \le x_e$ for every $e \in E$.

VIRTUAL PRIVATE NETWORK (VPN). The special case of CVPN with $\phi(x_e) = x_e$.

By duplicating nodes, we can assume without loss of generality that, for each $v \in V$, (b_v^+, b_v^-) is either $(0,0)$, $(1,0)$ (*sender* node), or $(0,1)$ (*receiver* node). Of course, this is not a polynomial reduction. However, our algorithms and their analysis can be easily adapted to perform this node duplication implicitly (see also [25]). We next denote as S and R the set of senders and receivers, respectively. That is, the total ingoing demand is $\mathcal{R} = |R|$ and the total outgoing demand is $\mathcal{S} = |S|$. By symmetry reasons, we can assume $|R| \ge |S|$ without loss of generality.

Given a collection of paths P, the optimal choice of the capacities x_e can be computed in polynomial time with the following approach [9, 23, 28]. Consider the bipartite graph $G_e = (S\cup R, E_e)$, where $E_e = \{(s,r)\,|\,e \in P_{sr}\}$. Then the value of x_e is the maximum cardinality of a matching in G_e.

We next define SSBB and SROB.

SINGLE-SINK BUY-AT-BULK (SSBB). The input consists of an undirected graph $G = (V,E)$, with edge weights $w : E \to \mathbb{Q}_{\ge 0}$, a set of *sources* $D \subseteq V$, with demands $d : D \to \mathbb{N}$, a *sink* node $z \in V$, and a capacity cost function $\phi(\cdot)$. The goal is to compute a capacity reservation $x \in \mathbb{Q}^E_{\ge 0}$ such that: (1) the cost $\sum_{e\in E} w(e)\cdot\phi(x_e)$ is minimized and (2) it is possible to route simultaneously $d(v)$ units of flow from each $v \in D$ to z.

SINGLE-SINK RENT-OR-BUY (SROB). The special case of SSBB where $\phi(x_e) = \min\{x_e, M\}$, for an input parameter $M \in \mathbb{Q}_{\ge 1}$.

Also in this case, by duplicating nodes, without loss of generality we can assume that $d(v) = 1$ for each source v (see, e.g., [25]). Observe that SROB (and hence SSBB) is also a generalization of the classical *Steiner tree* problem (ST): in that case $D \cup \{z\}$ gives the set of terminals, and $M = 1$. We remark that an optimal solution to SROB consists of a Steiner tree containing the root, whose edges support at least M paths each, and a shortest path from each source to the Steiner tree.

It is known that SSBB always admits an optimal tree solution, see e.g. the proof given by Karger and Minkoff [30]. (A *tree solution* is a solution which reserves positive capacity on a subtree of the input graph). Adapting the proof in [30], it is possible to transform any given solution into a tree solution of the same or smaller cost in polynomial time[2]: this will be needed in our CVPN algorithm to guarantee that any SSBB approximation algorithm can be used as a black box.

LEMMA 2.1 *Given any solution to* SSBB*, there is a polynomial-time deterministic algorithm to obtain a tree solution which costs no more.*

It remains to define CABSSBB and UNSSSBB.

CABLE SINGLE-SINK BUY-AT-BULK (CABSSBB). The input consists of an undirected graph $G = (V,E)$, with edge weights $w : E \to \mathbb{Q}_{\ge 0}$, a set of *sources* $D \subseteq V$, with demands $d : D \to \mathbb{N}$, a *sink* $z \in V$, and a set of *cable types* $1, 2, \ldots, k$, with capacities $\mu_1 \le \mu_2 \le \ldots \le \mu_k$ and costs (per unit length) $\sigma_1 \le \sigma_2 \le \ldots \le \sigma_k$, where $\delta_i := \frac{\sigma_i}{\mu_i}$ is a decreasing function of i (*economies of scale*). The goal is to find a cable installation $\{n_{i,e}\}_{1\le i\le k, e\in E}$, with $n_{i,e} \in \mathbb{N}$, minimizing $\sum_{i,e} w(e)\,\sigma_i\, n_{i,e}$ and such that $d(v)$ units of flow can be routed simultaneously from each source node $v \in D$ to the sink without exceeding the capacity $\sum_i \mu_i\, n_{i,e}$ on each edge e.

UNSPLITTABLE SINGLE-SINK BUY-AT-BULK (UNSSSBB). The same problem as CABSSBB, with the extra constraint that the flow from each source to the sink must be routed along a unique path.

Once again, by duplicating nodes, without loss of generality we can assume unit demands [25]. In order to obtain a good approximation for SSBB, it is sufficient to obtain a good approximation for CABSSBB.

[2] All the lemmas of this Section are proved in the Appendix for the sake of completeness.

Algorithm 1 cVPN algorithm

(i) For any possible choice of a *hub* sender $s' \in S$:

(a) Compute a ρ_{SSBB}-approximate SSBB solution for the input graph, with sources $D = S \cup R$, sink $z = s'$, and capacity cost function $\phi'(\cdot) = \phi(\min\{\cdot, |S|\})$. Turn it into a tree solution $x \in \mathbb{Q}^E_{\geq 0}$ which costs no more.

(b) Compute paths $P = \{P_{sr}\}_{s\in S, r\in R}$ and capacities x', with path P_{sr} being the unique simple path between s and r in the tree defined by the support of x_e, and $x'_e = \min\{x_e, |S|\}$.

(ii) Return the pair (P, x') of smallest cost.

Lemma 2.2 *For any $\varepsilon > 0$, given a ρ-approximation algorithm for* CabSsbb*, there is a $(1+\varepsilon)\rho$-approximation algorithm for* Ssbb*.*

Remark 2.1 *As noted in [14], the $(1+\varepsilon)$ factor can be avoided with the current best algorithms for* CabSsbb*. This is because they implement a preliminary cable-selection step, where consecutive selected cable types differ in cost and/or cost per unit capacity at least by some multiplicative factor $\alpha > 1$. This selection step can be mimicked by performing $O(\log|D|)$ binary searches over $\phi(\cdot)$.*

The proof of the following simple reduction is implicitly given, e.g., in [17, 37]. We explicitly prove it in the Appendix, for the sake of self-containedness and clarity.

Lemma 2.3 *Given a ρ-approximation algorithm for* CabSsbb*, there is a 2ρ-approximation algorithm for* UnsSsbb*.*

For a given instance $\mathcal{I}$ of an optimization problem $\mathcal{P}$, we denote the corresponding optimal cost as $OPT_{\mathcal{P}}(\mathcal{I})$. We sometimes use $OPT_{\mathcal{P}}(\mathcal{I})$ also to denote any fixed solution of that cost. When no confusion is possible, we will sometimes omit $\mathcal{I}$ or $\mathcal{P}$ (or both). All the problems that we are considering are **NP**-hard minimization problems. In this context a ρ-approximation algorithm, $\rho > 1$, is an algorithm which produces in polynomial time a feasible solution of cost at most $\rho \cdot OPT_{\mathcal{P}}(\mathcal{I})$. The best known approximation factor for the Steiner tree problem is denoted by ρ_{ST}. Currently $\rho_{\text{ST}} < 1.39$ [3]. For a given undirected graph $G = (V, E)$, with edge *weights* (or *costs*) $w : E \to \mathbb{Q}_{\geq 0}$, we let $w(u, v)$ denote the shortest path distance between nodes u and v. We also define $w(u, V') := \min_{v\in V'}\{w(u, v)\}$ and $w(E') = \sum_{e\in E'} w(e)$, for any $V' \subseteq V$ and $E' \subseteq E$. For notational convenience, we sometimes identify a subgraph G' of G with its set of nodes $V(G')$ or its set of edges $E(G')$. In particular, we use $w(u, G')$ and $w(G')$ as shortcuts for $w(u, V(G'))$ and $w(E(G'))$, respectively. We also use $v \in G'$ and $e \in G'$ in place of $v \in V(G')$ and $e \in E(G')$.

3. Concave Virtual Private Network. In this section we present the first constant factor approximation algorithm for cVPN.

Let $\mathcal{I} = (G, w, S, R, \phi)$ be a cVPN instance. Recall that S and R denote senders and receivers respectively, and $|R| \geq |S|$ by assumption. Consider Algorithm 1. Note that $\phi'(x) = \phi(\min\{x, |S|\})$ is a non-decreasing concave function, with $\phi'(0) = 0$: hence the Ssbb instance constructed in Step (i)a is well-defined. In Step (i)a we also exploit the fact that given any solution to Ssbb, it is possible to compute a tree solution of the same or smaller cost in polynomial time (by Lemma 2.1).

Lemma 3.1 *Algorithm 1 computes a solution to* cVPN *in polynomial time.*

Proof. The claim on the running time is trivial. Let us show 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. □

It remains to bound the approximation factor of the algorithm. To that aim, we let s^* be a sender chosen uniformly at random, and focus on the iteration of the algorithm with $s' = s^*$. Then, it is sufficient to show that the solution computed in that iteration is cheap in expectation.

Consider the SSBB instance $\mathcal{I}'_{s^*}$ computed in Step (i)a for sender s^*.

LEMMA 3.2 *Algorithm 1 computes a solution of cost at most* $\rho_{\text{SSBB}} \cdot E[OPT_{\text{SSBB}}(\mathcal{I}'_{s^*})]$.

PROOF. For any choice of s^*, the SSBB solution computed in Step (i)a equals the cost of the cVPN solution computed in Step (i)b. The claim follows by an averaging argument. □

We will now prove that $E[OPT_{\text{SSBB}}(\mathcal{I}'_{s^*})]$ is at most $2 \cdot OPT_{\text{cVPN}}(\mathcal{I})$. To do that, it is useful to define a modified cVPN instance $\mathcal{I}_{s^*}$ with the following traffic bounds:

$$b_v^+ = \begin{cases} |S| & \text{if } v = s^*; \\ 0 & \text{otherwise,} \end{cases} \qquad \text{and} \qquad b_v^- = \begin{cases} 1 & \text{if } v \in S \cup R; \\ 0 & \text{otherwise.} \end{cases}$$

In other terms, s^* can send up to $|S|$ units of flow, and each node in $S \cup R$ can receive one unit of flow.

LEMMA 3.3 $OPT_{\text{SSBB}}(\mathcal{I}'_{s^*}) = OPT_{\text{cVPN}}(\mathcal{I}_{s^*})$.

PROOF. Let P_{s^*v} be the paths in a cVPN solution for $\mathcal{I}_{s^*}$. 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's. If $k > |S|$, we may send 1 unit of flow from s^* to each node in $v_1, \ldots, v_{|S|}$. In any case, the needed capacity of e is $x_e = \min\{k, |S|\}$, which costs $w(e) \cdot \phi(\min\{k, |S|\})$. This is the same amount, which an SSBB solution pays for capacity k on $e \in E$. The claim follows. □

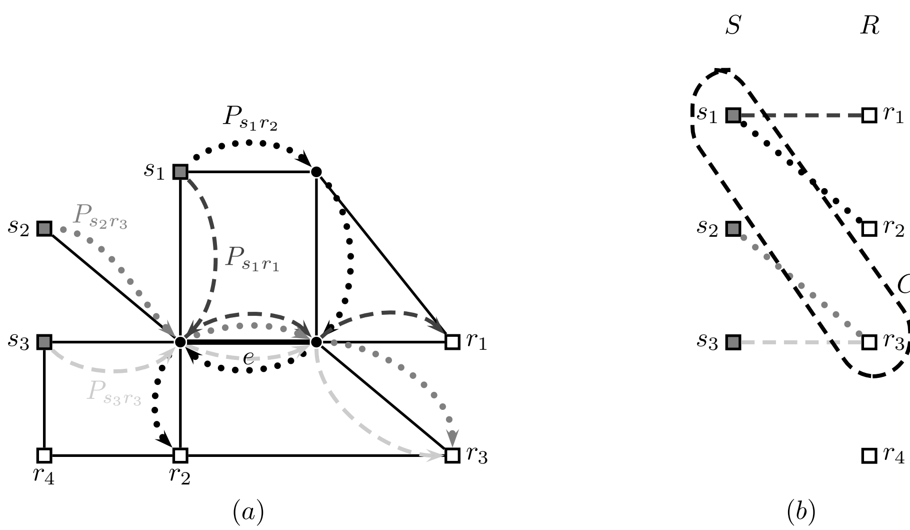


Figure 1: Example of a cVPN instance in (a), where terminals are depicted as rectangles, senders are drawn solid. Only paths, crossing edge e are shown. In (b) the graph G_e with vertex cover C_e is visualized, implying that $x_e = 2$.

The next lemma is the heart of our analysis: here we crucially exploit Kőnig's theorem.

LEMMA 3.4 $E[OPT_{\text{cVPN}}(\mathcal{I}_{s^*})] \leq 2 \cdot OPT_{\text{cVPN}}(\mathcal{I})$.

PROOF. Let $P = \{P_{sr} \mid s \in S, r \in R\}$ be the set of paths in the optimum cVPN solution for $\mathcal{I}$ and x_e be the induced capacities. We next construct a cVPN solution for $\mathcal{I}_{s^*}$, consisting of s^*-v paths P'_{s^*v} for $v \in S \cup R$. Choose a receiver $r^* \in R$ uniformly at random as a second hub. Take $P'_{s^*r} := P_{s^*r}$ as the s^*-r path. Furthermore concatenate $P'_{s^*s} := P_{s^*r^*} + P_{r^*s}$ to obtain a s^*-s path. More precisely, we 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 $s \in S$ (resp., receiver $r \in R$) install in a cumulative manner one unit of capacity on P_{sr^*} (resp., on P_{s^*r}). 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 the concavity of ϕ:

$$E[OPT_{\text{cVpn}}(\mathcal{I}_{s^*})] \leq E\Big[\sum_{e\in E} w(e)\phi(x'_e)\Big] \leq \sum_{e\in E} w(e)\phi(E[x'_e]) \leq 2\cdot OPT_{\text{cVpn}}(\mathcal{I}).$$

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 says that there is a vertex cover $C_e \subseteq S \cup R$ with $x_e = |C_e|$ (see Figure 1 for an illustration).

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

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

$$\Pr[(s^* \in S\cap C_e) \vee (r^* \in R \cap C_e)] \cdot |S| \leq \left(\frac{|S\cap C_e|}{|S|} + \frac{|R \cap C_e|}{|R|}\right)\cdot |S| \overset{|R|\geq|S|}{\leq} |C_e|.$$

Case: $s^* \in S\backslash C_e$ and $r^* \in R\backslash C_e$. 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_e. Consequently e does not lie on the path $P_{s^*r^*}$. Thus we have to install 1 unit of capacity for each sender s, such that $(s, r^*) \in E_e$. But only senders in $S \cap C_e$ may be adjacent to r^* in G_e, thus this number is at most $|S \cap C_e|$. A similar argument holds for the receivers. The expected contribution of this case is consequently upperbounded by

$$\Pr[(s^* \in S \setminus C_e) \wedge (r^* \in R \setminus C_e)] \cdot (|S\cap C_e| + |R \cap C_e|) \leq 1\cdot(|S\cap C_e| + |R\cap C_e|) = |C_e|.$$

Combining the two bounds above we obtain $E[x'_e] \leq 2|C_e| = 2x_e$, which implies the claim. □

Lemmas 3.1, 3.2, 3.3 and 3.4 imply the following theorem.

Theorem 3.1 *Given an (expected) ρ_{Ssbb} approximation algorithm for* Ssbb, *there is an (expected) $2\rho_{\text{Ssbb}}$ approximation algorithm for* cVpn.

We will prove in Section 6 that there is a randomized approximation algorithm for Ssbb with $\rho_{\text{Ssbb}} = 20.41$. Together with Theorem 3.1, this immediately gives the following corollary.

Corollary 3.1 *There is an expected* 40.82*-approximation algorithm for* cVpn.

Observe that the solution computed by Algorithm 1 is a tree solution. In several frameworks, the computed solution is constrained to be a tree due to technological reasons. Hence, an interesting question is how large is the gap between the best tree solution and the best (graph) solution. As a byproduct of our analysis, we obtain that this gap is at most 2, improving on the previous best bound $3+\sqrt{3} \approx 4.74$ due to [8] (which only worked in the special case of linear costs).

Corollary 3.2 *Any* cVpn *instance admits a tree solution of cost at most twice the optimal cost.*

4. Linear Virtual Private Network. In this section we present an expected 2.80 approximation for Vpn (in its classical formulation with linear costs). This improves on the previous best 3.39 approximation for this problem [3, 9]. We start by presenting in Section 4.1 a $(2+\varepsilon)$-approximation for the *quasi-balanced* case, where $|R| = O(|S|)$ (recall that $|R| \geq |S|$ by assumption). In Section 4.2 we give a 2.80-approximation for the case $|R| \gg |S|$.

4.1 The Quasi-Balanced Case. The next lemma is a specialization of Theorem 3.1.

Lemma 4.1 *Given a ρ_{Srob} approximation algorithm for* Srob, *there is a $2\rho_{\text{Srob}}$ approximation algorithm for* Vpn.

Algorithm 2 VPN algorithm.

(i) Choose a receiver $r^* \in R$ uniformly at random.

(ii) Mark each receiver uniformly at random with probability $\frac{\alpha}{|S|}$. Let R' be the marked receivers.

(iii) For each $s \in S$, compute a ρ_{ST}-approximative Steiner tree T_s spanning $\{s, r^*\} \cup R'$ and install cumulatively 1 unit of capacity on T_s.

(iv) Install 1 unit of capacity cumulatively on the shortest path from each receiver r to the closest node in $R' \cup \{r^*\}$.

Proof. Consider Algorithm 1. Observe that $\phi(\min\{x_e, |S|\}) = \min\{x_e, |S|\}$ since $\phi(x) = x$ for VPN. Hence the SSBB instance considered in Step (i)a is indeed an SROB instance with $M = |S|$, sink s', and sources $D = S \cup R$. Hence in that step we can use any ρ_{SROB} approximation algorithm for SROB. The claim follows. □

The above lemma does not directly imply an improvement of the approximation ratio for VPN, since the best known approximation factor for SROB is currently 2.80 [3, 10]. However, when $|R| = O(|S|)$, we can exploit a better approximation algorithm for SROB given in [10].

Theorem 4.1 *For any $\varepsilon > 0$, there is a $(2 + \varepsilon|R|/|S|)$-approximation algorithm for* VPN.

Proof. In [10] it is shown that, for any $\delta > 0$, there is a $1 + \delta\frac{|D|}{M}$-approximation algorithm for SROB. Combining this result with the reduction from Lemma 4.1, and choosing $\delta = \varepsilon/4$, we obtain the claimed approximation factor:

$$2\Big(1 + \delta\frac{|D|}{|M|}\Big) = 2 + 2\delta\frac{|R| + |S|}{|S|} \le 2 + 4\delta\frac{|R|}{|S|} = 2 + \varepsilon\frac{|R|}{|S|}.$$

□

4.2 The Unbalanced Case. Given the result from Section 4.1, we next assume that $|S| \le \varepsilon|R|$ for an arbitrarily small constant $\varepsilon > 0$.

Suppose to be given a VPN instance $\mathcal{I}$. We consider Algorithm 2, which is a slight adaptation of the VPN algorithm in [9]. The quantity α is a positive constant to be fixed later.

From an intuitive point, the analysis proceeds as follows. Replace all the senders with a unique sender s, chosen randomly, with $b_s^+ = |S|$. We will prove that, on average, the optimum cost of the VPN instance does not increase. The optimum solution to the new instance will be a tree, consisting of a core C_s (a tree) with capacity $|S|$, and paths $U_{s,r}$ from any $r \in R$ to C_s, each one contributing with one unit of capacity to the corresponding set of edges. Let Σ_S be the total cost of the cores C_s and let Σ_C be the average of $\sum_{r\in R} w(U_{s,r})$ (over all s). Then $\Sigma_S + \Sigma_C \le OPT_{\mathrm{VPN}}$. It is not difficult to bound the expected cost of the Steiner trees T_s in terms of Σ_S and Σ_C. For any sender s, we can bound the expected distances of the receivers in R to the nearest marked receiver by applying the Core Detouring Theorem with core C_s and paths $U_{s,r}$. By averaging over all s, we again obtain a bound in terms of Σ_S and Σ_C.

For any sender $s \in S$, let $\mathcal{I}'_s$ be the SROB instance with sources $D = R$, root $z = s$ and parameter $M = |S|$. We will first bound the cost of the solution output by the algorithm with respect to $OPT_{\mathrm{SROB}}(\mathcal{I}'_s)$, and then compare $OPT_{\mathrm{SROB}}(\mathcal{I}'_s)$ with $OPT_{\mathrm{VPN}}(\mathcal{I})$.

Let C_s be the Steiner tree in $OPT_{\mathrm{SROB}}(\mathcal{I}'_s)$, and $U_{s,r}$ the shortest path from $r \in R$ to C_s. Define $\Sigma_S := \sum_{s\in S} w(C_s)$ and $\Sigma_C := \frac{1}{M}\sum_{s\in S}\sum_{r\in R} w(U_{s,r})$. First we upper bound the cost of the $M = |S|$ Steiner trees computed by Algorithm 2.

Lemma 4.2 $E[\sum_{s\in S} w(T_s)] \le \rho_{\mathrm{ST}} \cdot \Sigma_S + \rho_{\mathrm{ST}}(\alpha + \varepsilon) \cdot \Sigma_C$.

Proof. Recall that $M/|R| = |S|/|R| \le \varepsilon$. For each s take the core C_s and attach the path $U_{s,r}$ for all $r \in R'$. Each $U_{s,r}$, $r \in R$, is used with probability at most $\frac{\alpha}{M} + \frac{1}{|R|}$, thus there exists a Steiner tree over $\{s, r^*\} \cup R'$ of expected cost $w(C_s) + (\frac{\alpha}{M} + \frac{1}{|R|})\sum_{r\in R} w(U_{s,r})$. Multiplying this quantity by the

Steiner tree approximation factor, and summing over all s we obtain

$$\sum_{s\in S} E[w(T_s)] \le \rho_{\mathrm{St}}\left(\sum_{s\in S} w(C_s) + \left(\frac{\alpha}{M} + \frac{1}{|R|}\right)\sum_{s\in S}\sum_{r\in R} w(U_{s,r})\right) \le \rho_{\mathrm{St}}\Sigma_S + \rho_{\mathrm{St}}(\alpha+\varepsilon)\Sigma_C.$$

□

We next bound the cost of connecting each receiver to the closest node in $R'\cup\{r^*\}$. To that aim, we exploit the following Core Detouring Theorem developed in the context of Srob [10][3].

Theorem 4.2 (Core-Detouring) *[10] Given an undirected graph $G=(V,E)$, with edge weights $w: E\to\mathbb{Q}_{\ge 0}$, clients $C\subseteq V$, a connected subgraph G', a root $z\in V(G')$, and a probability $p\in(0,1]$, mark each client independently with probability p, and denote the marked clients by C'. Then*

$$E\Big[\sum_{v\in C} w(v, C'\cup\{z\})\Big] \le \frac{0.8067}{p} w(G') + 2\sum_{v\in C} w(v,G').$$

With this tool at hand, the next lemma is easy to prove.

Lemma 4.3 $E[\sum_{r\in R} w(r, R'\cup\{r^*\})] \le \frac{0.8067}{\alpha}\Sigma_S + 2\Sigma_C$.

Proof. Let $s\in S$. Applying the Core Detouring Theorem 4.2 with $C=R$, $G'=C_s$, $z=r^*$, and $p=\alpha/M$,

$$E\Big[\sum_{r\in R} w(r, R'\cup\{r^*\})\Big] \le \frac{0.8067}{\alpha/M} w(C_s) + 2\sum_{r\in R} w(U_{s,r}).$$

Averaging this bound over all s, we obtain

$$E\Big[\sum_{r\in R} w(r, R'\cup\{r^*\})\Big] \le \frac{0.8067M}{\alpha}\sum_{s\in S}\frac{1}{M} w(C_s) + 2\frac{1}{M}\sum_{s\in S}\sum_{r\in R} w(U_{s,r}) = \frac{0.8067}{\alpha}\Sigma_S + 2\Sigma_C.$$

□

The crucial part is now to relate the cost of $OPT_{\mathrm{Srob}}(\mathcal{I}'_s)$ to the optimum cost $OPT_{\mathrm{Vpn}}(\mathcal{I})$. Similarly to Lemma 3.4, the next lemma relies on Kőnig's Theorem.

Lemma 4.4 $\sum_{s\in S} OPT_{\mathrm{Srob}}(\mathcal{I}'_s) \le |S|\cdot OPT_{\mathrm{Vpn}}(\mathcal{I})$.

Proof. In order to prove the lemma, it is convenient to define a Vpn instance $\mathcal{I}_s$ with the same receiver set as $\mathcal{I}$, and with sender set $\{s\}$, where the out-traffic bound for s is $b^+_s = |S|$. (Recall that $b^+_s = 1$ for the original problem). Then using Lemma 3.3, we obtain $OPT_{\mathrm{Srob}}(\mathcal{I}'_s) = OPT_{\mathrm{Vpn}}(\mathcal{I}_s)$. We now prove that $\sum_{s\in S} OPT_{\mathrm{Vpn}}(\mathcal{I}_s) \le |S|\cdot OPT_{\mathrm{Vpn}}(\mathcal{I})$ by showing that, for a random sender s^*, $E[OPT_{\mathrm{Vpn}}(\mathcal{I}_{s^*})] \le OPT_{\mathrm{Vpn}}(\mathcal{I})$. The claim follows by an averaging argument.

Let $P=\{P_{sr}\}_{(s,r)\in S\times R}$ be the optimal paths for $\mathcal{I}$ and let $x=\{x_e\}_{e\in E}$ be the induced capacities. Consider the bipartite graph $G_e=(S\cup R, E_e)$, with $(s,r)\in E_e$ iff $e\in P_{sr}$. Let $C_e\subseteq S\cup R$ be a vertex cover for G_e of size x_e (which exists by Kőnig's Theorem). Consider the solution to $\mathcal{I}_{s^*}$ induced by paths $\{P_{s^*r}\}_{r\in R}$ and let $x'=\{x'_e\}_{e\in E}$ be the corresponding capacity reservation. Clearly $x'_e \le \min\{|N_e(s^*)|, |S|\}$, whereby $N_e(s^*)$ are the nodes adjacent to s^* in G_e. Let us show that $E[x'_e]\le x_e$. The event $\{s^*\in S\cap C_e\}$ happens with probability $\frac{|S\cap C_e|}{|S|}$. In this case we can upper bound x'_e with $|S|$. In the complementary case we can upper bound x'_e with $|N_e(s^*)| \le |R\cap C_e|$, where we exploit the fact that s^* can be only adjacent to nodes of $R\cap C_e$ (otherwise C_e would not be a vertex cover). Altogether

$$E[x'_e] \le \frac{|S\cap C_e|}{|S|}\cdot|S| + \left(1-\frac{|S\cap C_e|}{|S|}\right)\cdot|R\cap C_e| \le |S\cap C_e| + |R\cap C_e| = |C_e| = x_e.$$

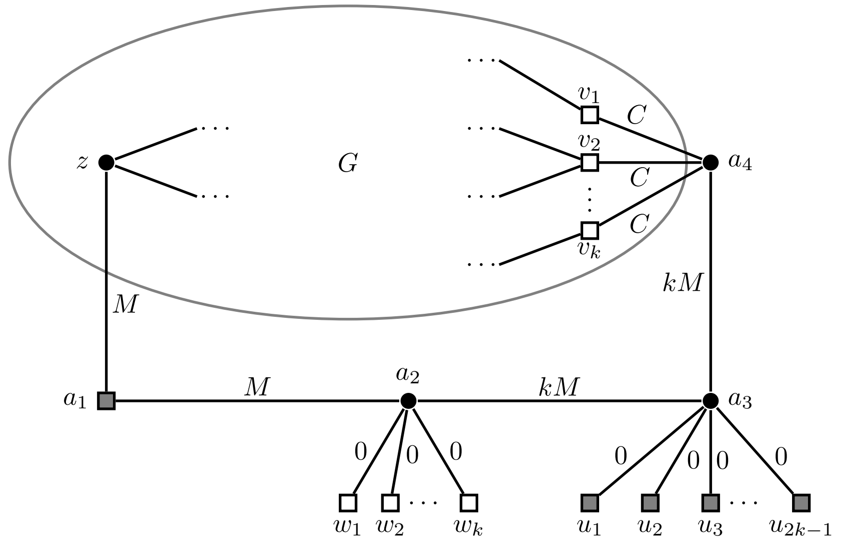
The claim follows since

$$E[OPT_{\mathrm{Vpn}}(\mathcal{I}_{s^*})] \le E\Big[\sum_{e\in E} w(e)x'_e\Big] \le \sum_{e\in E} w(e)x_e = OPT_{\mathrm{Vpn}}(\mathcal{I}).$$

□

[3]In [10], the right hand side of the inequality contains an extra term $e^{-p|C|}\cdot|C|\cdot w(G')$. Since adding dummy clients in the root does not change the connection costs, our simplified version of the formula follows by a limit argument.

Figure 1 VPN instance $\mathcal{I}_{\text{VPN}}$. Edges are labeled with their cost. Senders and receivers are depicted as gray and white squares, respectively.



Lemma 4.5 *For a suitable choice of* α *and* $|R|/|S|$ *large enough, Algorithm 2 gives an expected* 2.80 *approximation for* VPN.

Proof. From Lemma 4.4,

$$\Sigma_S + \Sigma_C = \frac{1}{M}\sum_{s\in S}\left(M\, w(C_s) + \sum_{r\in R} w(U_{s,r})\right) \le \frac{1}{M}\sum_{s\in S} OPT_{\text{SROB}}(\mathcal{I}'_s)$$
$$\le \frac{1}{M}|S|\cdot OPT_{\text{VPN}}(\mathcal{I}) = OPT_{\text{VPN}}(\mathcal{I}). \qquad (1)$$

By Lemmas 4.2 and 4.3, the expected cost of the solution computed by the algorithm is at most

$$(\rho_{\text{ST}}\cdot\Sigma_S + \rho_{\text{ST}}(\alpha+\varepsilon)\Sigma_C) + \left(\frac{0.8067}{\alpha}\Sigma_S + 2\Sigma_C\right) \overset{\alpha=0.5748}{\le} 2.80(\Sigma_C+\Sigma_S) \overset{(1)}{\le} 2.80\cdot OPT_{\text{VPN}}.$$

□

Theorem 4.3 *There is an expected* 2.80*-approximation algorithm for* VPN.

Proof. The claim follows from Theorem 4.1 and Lemma 4.5. □

5. Hardness of Balanced Virtual Private Network. In this section we consider the balanced case of the VPN problem, i.e. $|S| = |R|$. Solving the open problem in [4, 28], we show that even this special case if **NP**-hard via a reduction from the Steiner tree problem.

We start by describing the reduction from Steiner tree. Consider a Steiner tree instance $\mathcal{I}_{\text{ST}}$, consisting of a graph $G = (V, E)$, with edge weights $w: E \to \mathbb{Q}_{\ge 0}$, and $k+1 \ge 4$ terminals $\{v_1, v_2, \ldots, v_k, z\}$. We construct an instance $\mathcal{I}_{\text{VPN}}$ of the balanced VPN problem on a graph $G' = (V', E')$ as follows. First, introduce two large numbers: $C := \sum_{e\in E} w(e) + 1$, and $M \gg (k+1)C$. To construct G' from G, add a node a_4 and make it adjacent to the nodes $v_1, v_2, \ldots, v_k$ by edges of cost C. Then, add a path z, 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 nodes $W = \{w_1, w_2, \ldots, w_k\}$, each of them adjacent to a_2 with a zero cost edge, and add $2k-1$ nodes $U = \{u_1, u_2, \ldots, u_{2k-1}\}$, each of them adjacent to a_3 with a zero cost edge. Define the set of senders as $S := \{a_1\} \cup U$ and the set of receivers as $R := \{v_1, v_2, \ldots, v_k\} \cup W$. Note that indeed $|S| = |R|$. Figure 1 illustrates this reduction.

We next show that the optimal solution to the Steiner tree instance $\mathcal{I}_{\text{ST}}$ has cost W^* if and only if the optimal solution for the balanced VPN instance $\mathcal{I}_{\text{VPN}}$ has cost $Z^* = 2k^2M + 2M + kC + W^*$. We split the proof in the if (Lemma 5.2) and only if (Lemma 5.1) parts.

LEMMA 5.1 *Given a solution to $\mathcal{I}_{\mathrm{ST}}$ of cost W^*, there is a solution to $\mathcal{I}_{\mathrm{VPN}}$ of cost at most $Z^* = 2k^2M + 2M + kC + W^*$.*

PROOF. Let T^* be the considered Steiner tree. We denote as T^*_{uv} the unique simple path between nodes u and v in T^*. We construct a solution to $\mathcal{I}_{\mathrm{VPN}}$ by defining the following paths:

- $P_{a_1 w_i} = \{a_1, a_2\} \cup \{a_2, w_i\}$, for $i = 1, \ldots, k$;
- $P_{a_1 v_i} = \{a_1, z\} \cup T^*_{zv_i}$, 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 $i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$;
- $P_{u_j v_i} = \{u_j, a_3\} \cup \{a_3, a_4\} \cup \{a_4, v_i\}$, for $i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$.

Finally, we install the following amount of capacity on the edges of the graph G':

$$x_e = \begin{cases} k & \text{if } e = \{a_2, a_3\}, \{a_3, a_4\}; \\ 0 & \text{if } e \in E \setminus E(T^*); \\ 1 & \text{otherwise.} \end{cases}$$

The cost of this capacity reservation is $W^* + 2k^2M + 2M + kC = Z^*$. To see feasibility, observe that no path uses the edges $E \setminus E(T^*)$, while edges $\{a_1, a_2\}$, $\{a_1, z\}$, and $E(T^*)$ are used by a_1 only (hence one unit of capacity is sufficient). Edges $\{a_2, w_i\}$, $\{a_4, v_i\}$, and $\{a_3, u_j\}$, $i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$, are used by at most one sender/receiver in every feasible traffic matrix. Edges $\{a_2, a_3\}$ and $\{a_3, a_4\}$ are used only by receivers w_i and v_i, $i = 1, \ldots, k$, respectively. Hence k units of capacity are sufficient on those edges. □

LEMMA 5.2 *Given a solution to $\mathcal{I}_{\mathrm{VPN}}$ of cost $Z^* = 2k^2M + 2M + kC + W^*$, there is a solution to $\mathcal{I}_{\mathrm{ST}}$ of cost at most W^*.*

PROOF. Let (P, x) be the considered solution to $\mathcal{I}_{\mathrm{VPN}}$ with cost Z^*. Recall that we may assume x to be an integer vector. We next argue that in fact this solution must be of the same structure as suggested in Lemma 5.1.

Claim 5.1 *For $i = 1, \ldots, k$, one has $\{a_1, z\} \in P_{a_1 v_i}$ and $\{a_1, a_2\} \in P_{a_1 w_i}$.*

PROOF. Assume for a contradiction that $\{a_1, z\} \notin P_{a_1 v_i}$ for some v_i. Then $P_{a_1 v_i}$ must contain $\{a_1, a_2\}$, $\{a_2, a_3\}$, and $\{a_3, a_4\}$. We now investigate how much capacity must be installed on such edges. First, $x_{a_1 a_2} \geq 1$ (since $\{a_1, a_2\}$ is used by at least one path). Now, consider the feasible 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. All senders different from a_1 must use a path containing either $\{a_2, a_3\}$ or $\{a_3, a_4\}$, while the path from a_1 to v_i uses both. This means, that the capacity to be installed on the latter edges fulfills $x_{a_2 a_3} + x_{a_3 a_4} \geq (2k-1) + 2 = 2k+1$. Therefore, for $k \geq 2$, the cost of the emerging solution is at least $2k^2M + kM + M > Z^*$, yielding a contradiction.

A symmetric argument proves the second part of the claim. □

Claim 5.2 *One has $x_{a_2 a_3} + x_{a_3 a_4} \geq 2k$.*

PROOF. Let $e = \{a_2, a_3\}$ and $e' = \{a_3, a_4\}$. All $2k-1$ senders U must use a path containing either e or e', therefore $x_e + x_{e'} \geq 2k-1$. Let us prove that in fact the inequality is strict. Assume by contradiction that $x_e + x_{e'} = 2k-1$. Define the bipartite graphs G_e and $G_{e'}$ as usual, and remove node a_1 from them. Let C_e and $C_{e'}$ be corresponding minimum vertex covers. Claim 5.1 implies that the paths of a_1 use neither e nor e'. Hence the edge set of G_e and $G_{e'}$ with or without node a_1 is the same. We can conclude by Kőnig's theorem and the feasibility of the solution that

$$2k - 1 = x_e + x_{e'} \geq |C_e| + |C_{e'}|.$$

As a consequence, there is at least one receiver $r' \in R$ that does not belong to any of the two covers.

Now observe that G_e and $G_{e'}$ are edge disjoint, since no path can use both e and e'. Furthermore, $G_e \cup G_{e'}$ is a complete bipartite graph on $U \times R$. This is because any sender u_i is forced to use either e

or e'. Suppose that there exists a sender $u' \in U \setminus \{C_e \cup C_{e'}\}$. Then edge $\{u', r'\}$ is neither covered by C_e nor by $C_{e'}$, a contradiction. As a consequence, $C_e \cup C_{e'} \supseteq U$. Since $|C_e| + |C_{e'}| \leq 2k - 1 = |U|$, we can conclude that $C_e \cup C_{e'} = U$ and $C_e \cap C_{e'} = \emptyset$.

The disjointness of C_e and $C_{e'}$ implies that all senders in C_e (resp., $C_{e'}$) route to the $2k$ receivers on paths containing e (resp., e'). Consider any choice of k senders $U' \subseteq U$. Define a traffic matrix as follows: if $u_i \in C_e \cap U'$, send one unit from u_i to some v_j, and otherwise from u_i to some w_j. This can be done in a feasible way, since there are enough receivers v_j and w_j. This way each sender in U' uses both edges $\{a_1, z\}$ and $\{a_1, a_2\}$. Hence $x_{a_1 z} \geq k$ and $x_{a_1 a_2} \geq k$.

Therefore, for $k \geq 3$, the cost of the emerging solution is at least $kM \cdot (x_e + x_{e'}) + M \cdot (x_{a_1 z} + x_{a_1 a_2}) \geq kM \cdot (2k - 1) + 2kM > Z^*$. This gives the desired contradiction. □

Claim 5.3 *For $i = 1, \ldots, k$ and $j = 1, \ldots, 2k-1$, one has $\{a_3, a_4\} \in P_{u_j v_i}$ and $\{a_2, a_3\} \in P_{u_j w_i}$.*

Proof. Assume by contradiction that there is a path from some u_j to some v_i that does not contain $e = \{a_3, a_4\}$. Necessarily, it must contain the edges $\{a_1, a_2\}$ and $\{z, a_1\}$. Consider the feasible traffic matrix in which u_j sends 1 unit of flow to v_i and a_1 sends 1 unit of flow to some v_h, $h \neq i$. From Claim 5.1, $\{z, a_1\}$ is used by two paths and therefore $x_{za_1} \geq 2$. Similarly, one can show $x_{a_1 a_2} \geq 2$. Therefore, by Claim 5.2, the cost of the emerging solution is at least $kM \cdot (x_{a_2 a_3} + x_{a_3 a_4}) + M \cdot (x_{za_1} + x_{a_1 a_2}) \geq 2k^2 M + 4M > Z^*$, a contradiction.

A symmetric argument shows that there is no path from some u_j to some w_i that does not contain $e' = \{a_2, a_3\}$. □

Claim 5.4 *One has $\sum_{i=1,\ldots,k} x_{a_4 v_i} \geq k$.*

Proof. Consider a feasible traffic matrix where k senders in U simultaneously send k units of flow to $v_1, \ldots, v_k$. From Claim 5.3, all these senders use paths containing $\{a_3, a_4\}$, and therefore each of those paths contains at least one edge $\{a_4, v_i\}$. □

From Claims 5.1, 5.2, and 5.4, the cost of edges not in E is at least

$$M \cdot (x_{za_1} + x_{a_1 a_2}) + kM \cdot (x_{a_2 a_3} + x_{a_3 a_4}) + C \cdot \sum_{i=1,\ldots,k} x_{a_4 v_i} \geq 2k^2 M + 2M + kC.$$

Finally, let T be the subset of edges of E that are in the support of the solution. From the discussion above,

$$w(T) \leq Z^* - (2k^2 M + 2M + kC) = W^*.$$

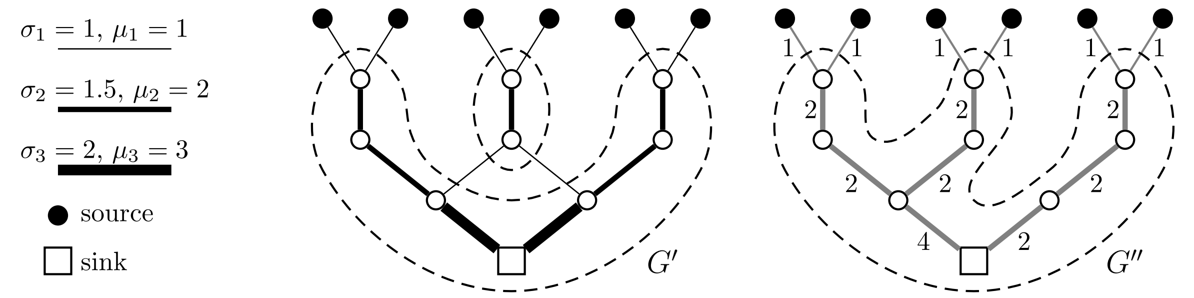
Assume by contradiction that T does not contain a Steiner tree on nodes $v_1, \ldots, v_k, z$. Then there exists at least one node v_i such that the path from a_1 to v_i contains at least 2 edges with cost C. But in this case, we would have $\sum_{i=1,\ldots,k} x_{a_4 v_i} \geq k + 1$: In fact, we can define a traffic matrix where a_1 sends one unit of flow to v_i and the remaining v_j's receive one unit of flow from $k - 1$ senders in U. Thus the cost of the solution would be at least $2k^2 M + 2M + (k+1)C > Z^*$, a contradiction. We conclude that indeed T contains a Steiner tree T'. Its cost is $w(T') \leq w(T) \leq W^*$. The claim follows. □

Theorem 5.5 *The balanced* Vpn *problem is* **NP**-*hard.*

Proof. The claim follows from Lemma 5.1, Lemma 5.2, and the **NP**-hardness of the Steiner tree problem [5]. □

Remark 5.1 *Note that the above reduction is not approximation preserving. In particular, the* **APX**-*hardness of Steiner tree is not conveyed to balanced* Vpn. *Hence, our reduction does not exclude the existence of a PTAS for balanced* Vpn*: This is an interesting open problem.*

Figure 2 All the edges have cost 1, besides the edges incident to the root, which have cost $M \gg 1$. In the optimum CabSsbb solution (left side) exactly one cable on each edge is installed, as shown in the picture. The graph $G' = G_1$ is disconnected. On the right, edges of T, with edge labels m_e. Edges of the connected subgraph $G'' = \{e \in T \mid m_e \geq \gamma/2\}$ (given that $\gamma = 3$) are drawn bold.



6. Single-Sink Buy-at-Bulk. In this section we present our improved 20.41 approximation algorithm for CabSsbb (and hence for Ssbb).

The main novelty in our approach is the following generalization of the Core Detouring Theorem. For a given edge-weight function $w(\cdot)$, and a given (possibly disconnected) subgraph G', we let $w_{G'}(v,u)$ be the distance from v to u in the graph resulting from the contraction of the connected components of G'. In other words, $w_{G'}(v,u)$ is the cost of the shortest v-u path, if edges in G' are for free.

Theorem 6.1 (Multi-Core Detouring) *Given an undirected graph $G = (V, E)$, with edge weights $w : E \to \mathbb{Q}_{\geq 0}$, clients $C \subseteq V$, a subgraph G', a root $z \in V$, and a probability $p \in (0, 1]$, mark each client independently with probability p, and denote the marked clients by C'. Then*

$$E\Big[\sum_{v\in C} w(v, C' \cup \{z\})\Big] \leq \frac{0.8067}{p} w(G') + 2\sum_{v\in C} w_{G'}(v, z).$$

Proof. Let $\theta_p(C, G') := 2\sum_{v\in C} w_{G'}(v, z) + \gamma\, w(G')$ with $\gamma := \frac{0.8067}{p}$. We will find a connected subgraph G'' of G with $z \in V(G'')$, having $\theta_p(C, G'') \leq \theta_p(C, G')$. The claim then follows by applying the Core Detouring Theorem 4.2 to G''. Let P_{vz} be the path, attaining the length $w_{G'}(v, z)$, i.e. it is a shortest v-z path in G, where edges in G' have cost 0. Since these paths are shortest paths, we may assume that $\bigcup_{v\in C} P_{vz}$ induces a tree T, rooted at z (see Figure 2). For $e \in T$, let $m_e := |\{v \in C \mid e \in P_{vz}\}|$ be the number of v-z paths that contain e. Let G'' be the graph, induced by the edges $e \in T$ with $m_e \geq \gamma/2$ ($G'' := \{z\}$ if no such edge exists). Moving from a leaf of T to the root z, the quantity m_e can only increase, hence the subgraph G'' is connected and $z \in V(G'')$. To upper bound $\theta_p(C, G'')$, we still use P_{vz} as the v-z path, even if $w_{G''}(v, z)$ is attained by a different path. Any edge $e \in T$ contributes with a term $\gamma w(e)$ to $\theta_p(C, G')$ if $e \in G'$, and with a term $2m_e w(e)$ otherwise. Note that $\gamma w(e) \leq 2m_e w(e)$ if and only if $m_e \geq \gamma/2$. By the definition of G'', the contribution of e to $\theta_p(C, G'')$ is $\min\{2m_e w(e), \gamma w(e)\}$, which is never larger than the contribution of e to $\theta_p(C, G')$. Applying the Core Detouring Theorem to the core G'' yields

$$\begin{aligned} E\Big[\sum_{v\in C} w(v, C' \cup \{z\})\Big] &\leq \theta_p(C, G'') \leq \sum_{e\in T\setminus G''} 2m_e w(e) + \sum_{e\in G''} \gamma w(e) \\ &= \sum_{e\in T} \min\{2m_e w(e), \gamma w(e)\} \leq \sum_{e\in T\setminus G'} m_e w(e) + \sum_{e\in G'} \gamma w(e) = \theta_p(C, G'), \end{aligned}$$

and the claim follows. □

Next consider the input CabSsbb instance. By adding dummy sources in the sink, we can assume that $|D|$ is a multiple of all the capacities μ_i. We will use the *aggregation algorithm* in [25]. Given a set of demands $x(v) \in [0, U)$, whose total value is a multiple of $U > 0$, and a tree T, this algorithm computes a random flow along T to aggregate the demands such that:

(1) The amount of flow along each edge of T is at most U;
(2) The final demand $x'(v)$ at each node is either 0 or U;

Algorithm 3 CabSsbb algorithm.

(i) Select a subset of cable types $i(1), \ldots, i(k')$ in increasing order of capacity, where $i(1) = 1$ and $i(k') = k$.

(ii) For $t = 0, 1, \ldots, k'$:

(a) **(Collection)** Let D_t be the set of nodes with positive demand. Each node in D_t is marked with probability $p_t = \alpha\sigma_{i(t)}/\sigma_{i(t+1)}$ (probability 1 if $t = 0$). Let D'_t be the set of marked nodes. Each node sends its demand to the closest node in $D'_t \cup \{r\}$ along a shortest path, using cables of type $i(t)$ (type 1 for $t = 0$). Let $d'_t(w)$ be the new demand collected at each $w \in D'_t \cup \{r\}$.

(b) **(Aggregation)** If $t < k'$, compute a ρ_{ST}-approximate Steiner tree T_t on $D'_t \cup \{r\}$. Apply the aggregation algorithm to T_t with $U = \mu_{i(t+1)}$ and $x(w) = d'_t(w) \pmod{\mu_{i(t+1)}}$ for each terminal node w. The corresponding flow is supported by installing cables of type $i(t + 1)$ (at most one for each edge of T_t). Let $d''_t(w)$ be the new demand aggregated at each node w.

(c) **(Redistribution)** If $t < k'$, for each node $w \in D'_t \cup \{r\}$, consider the subset of nodes $D_t(w) \subseteq D_t$ that sent their demand to w during the collection step (including w itself, if $w \neq r$). Uniformly select a random subset $\widetilde{D}_t(w)$ of $D_t(w)$ of cardinality $d''_t(w)/\mu_{i(t+1)}$. Send $\mu_{i(t+1)}$ units of flow back from w to each node in $\widetilde{D}_t(v)$ along shortest paths, installing cables of type $i(t + 1)$.

(3) The expected demand at each node is preserved, that is: $\Pr[x'(v) = U] = x(v)/U$.

We consider the CabSsbb Algorithm 3, which is a slight variant of the algorithms in [19]. For notational convenience, we assume $\sigma_{i(k'+1)} = \infty$ and $i(0) = 0$. The algorithm initially selects a subset of k' cable types $i(1), i(2), \ldots, i(k')$: this step is explained in more details below. Then there is a sequence of $k' + 1$ rounds. In each round the demand of the sources (which is initially 1 for each source) is aggregated in a smaller and smaller subset of sources. At the beginning of round $t \geq 1$, the demand at each source is in $\{0, \mu_{i(t)}\}$. Each round t consists of three steps. Initially the demand is collected at a random subset of aggregation points (Collection Step). The quantity α in this step is a proper constant to be fixed later. Then a Steiner tree is computed on the aggregation points, and the demand is aggregated along such tree via the aggregation algorithm in multiples of $\mu_{i(t+1)}$ (Aggregation Step). This is possible since the sum of the $d'_t(w)$'s, and hence of the $x(w)$'s, is a multiple of $\mu_{i(t+1)}$. Eventually, the aggregated demand is redistributed back to the source nodes (Redistribution Step). Only cables of type $i(t)$ and $i(t + 1)$ are used in round t. At the end of the round the demand at each source is in $\{0, \mu_{i(t+1)}\}$.

It remains to specify how the cable types $i(1), \ldots, i(k')$ are chosen. Let $\beta > 1$ be a constant to be fixed later. Differently from prior work on the topic, we use a randomized cable selection rule.

> Randomized Cable Selection Rule. Let $i(1) = 1$. Given $i(t)$, $1 < i(t) < k$, $i(t + 1)$ is chosen as follows. Let $i'(t) > i(t)$ and $i''(t) > i(t)$ be the smallest indexes such that $\frac{\delta_{i'(t)}}{\delta_{i(t)}} \leq \frac{1}{\beta}$ and $\frac{\sigma_{i''(t)}}{\sigma_{i(t)}} \geq \beta$, respectively. If no proper $i'(t)$ (resp., $i''(t)$) exists, we let $i'(t) = k$ (resp., $i''(t) = k$). If $i'(t) \geq i''(t)$, $i(t+1) = i'(t)$. Otherwise, $i(t+1) = i''(t) - 1$ with probability $\frac{\sigma_{i''(t)} - \beta\sigma_{i(t)}}{\sigma_{i''(t)} - \sigma_{i''(t)-1}}$, and $i(t + 1) = i''(t)$ otherwise.

Note that, as required $i(1) = 1$ and $i(k') = k$.

Lemma 6.1 *For any $t \in \{1, 2, \ldots, k' - 2\}$ and $h \in \{0, 1, \ldots, k' - t - 1\}$, and for any $s \in \{1, 2, \ldots, k\}$, $i(t') < s \leq i(t' + 1)$:*

$$\textbf{(a)}\ \delta_{i(t+h)} \leq \frac{1}{\beta^h}\delta_{i(t)}; \qquad \textbf{(b)}\ E[\sigma_{i(t+h)}] \geq \beta^h E[\sigma_{i(t)}]; \qquad \textbf{(c)}\ E\left[\min\left\{\frac{\sigma_{i(t'+1)}}{\sigma_s}, \frac{\delta_{i(t')}}{\delta_s}\right\}\right] \leq \beta.$$

Proof.

(a) By economies of scale, $\delta_{i(t+1)} \leq \delta_{i'(t)} \leq \delta_{i(t)}/\beta$. A simple induction implies (a).

(b) If $i'(t) \geq i''(t)$, $\sigma_{i(t+1)} \geq \beta\,\sigma_{i(t)}$ deterministically for any value of $\sigma_{i(t)}$. Otherwise

$$E[\sigma_{i(t+1)}|i(t)=\eta] = \frac{\sigma_{i''(t)} - \beta\,\sigma_\eta}{\sigma_{i''(t)} - \sigma_{i''(t)-1}}\sigma_{i''(t)-1} + \left(1 - \frac{\sigma_{i''(t)} - \beta\,\sigma_\eta}{\sigma_{i''(t)} - \sigma_{i''(t)-1}}\right)\sigma_{i''(t)} = \beta\,\sigma_\eta.$$

Summing over η, we obtain $E[\sigma_{i(t+1)}] = \beta\,E[\sigma_{i(t)}]$ for $i'(t) < i''(t)$. Altogether, $E[\sigma_{i(t+1)}] \geq \beta\,E[\sigma_{i(t)}]$. Property (b) follows by induction.

(c) Property (c) is trivially true for $s = 1$, so assume $s > 1$. Let us condition over $t' = \tau \geq 1$. For $i'(\tau) \geq i''(\tau)$, either $s = i(\tau+1)$ and hence $\frac{\sigma_{i(\tau+1)}}{\sigma_s} = 1$, or $\frac{\delta_{i(\tau)}}{\delta_s} \leq \frac{\delta_{i(\tau)}}{\delta_{i(\tau+1)-1}} < \beta$ by the minimality of $i'(\tau)$. Otherwise ($i'(\tau) < i''(\tau)$), being $\sigma_s \geq \sigma_{i(\tau)}$ and by the same argument as above,

$$E\left[\frac{\sigma_{i(\tau+1)}}{\sigma_s}\right] \leq E\left[\frac{\sigma_{i(\tau+1)}}{\sigma_{i(\tau)}}\right] = \frac{\beta\sigma_{i(\tau)}}{\sigma_{i(\tau)}} = \beta.$$

The claim follows by summing over all the possible values of τ. □

Let A_t be the cost of the t-th round, $t \in \{0,1,\ldots,k'\}$. Let moreover A^c_t, A^a_t, and A^r_t denote the collection, aggregation, and redistribution costs of the t-th round, $t \in \{1,\ldots,k'-1\}$ respectively. By $OPT(s)$ we denote the cost paid by the optimum solution for cables of type s. The following lemma is an adaptation of a similar result in [19] (proof in the appendix).

Lemma 6.2 *For $t' \in \{1,\ldots,k'\}$ and $t \in \{1,\ldots,k'-1\}$,*

$$(\mathbf{1})\Pr[d \in D_{t'}|v \in D_0] = \frac{1}{\mu_{i(t')}};\quad (\mathbf{2})A_0 \leq \rho_{\mathrm{ST}}\sum_s \frac{\sigma_{i(1)}}{\sigma_s}OPT(s);\quad (\mathbf{3})E[A_{k'}] \leq \sum_s \frac{\delta_{i(k')}}{\delta_s}OPT(s);$$

$$(\mathbf{4})E[A^a_t] \leq E\left[\sum_s \min\left\{\rho_{\mathrm{ST}}\alpha\frac{\delta_{i(t)}}{\delta_s}, \rho_{\mathrm{ST}}\frac{\sigma_{i(t+1)}}{\sigma_s}\right\}OPT(s)\right];\qquad (\mathbf{5})E[A^r_t] \leq E\left[\frac{\delta_{i(t+1)}}{\delta_{i(t)}}\,A^c_t\right].$$

Hence it remains to bound $E[A^c_t]$. Following [19, 25], it is not hard to show that $E[A^c_t] \leq \frac{2}{\alpha}E[A^a_t]$. We next present an improved bound based on the Multi-Core Detouring Theorem 6.1.

Lemma 6.3 *For all $t \in \{1,2,\ldots,k'-1\}$, $E[A^c_t] \leq E\left[\sum_s \min\left\{2\frac{\delta_{i(t)}}{\delta_s}, \frac{0.8067}{\alpha}\frac{\sigma_{i(t+1)}}{\sigma_s}\right\}OPT(s)\right]$.*

Proof. Let $j \in \{1,2,\ldots,k\}$ be an integer value to be fixed later. We denote by G_j be the graph induced by the edges where OPT installs at least one cable of type $s > j$. Note that this graph might be disconnected. (See Figure 2). By the Multi-Core Detouring Theorem 6.1 applied with $C = D_t$, $z = r$, $p = p_t$ and $G' = G_j$,

$$E[A^c_t] := E\left[\sigma_{i(t)}\sum_{d\in D_t} w(d, D'_t \cup \{r\})\right] \leq E\left[\sigma_{i(t)}\left(2\sum_{d\in D_t} w_{G_j}(d,r) + \frac{0.8067}{p_t}c(G_j)\right)\right].$$

By definition,

$$E\left[\frac{0.8067}{p_t}\sigma_{i(t)}c(G_j)\right] = E\left[\frac{0.8067\,\sigma_{i(t+1)}}{\alpha}c(G_j)\right] \leq E\left[\frac{0.8067}{\alpha}\sum_{s>j}\frac{\sigma_{i(t+1)}}{\sigma_s}OPT(s)\right].$$

By Lemma 6.2(1), $\Pr[d \in D_t|d \in D] = \frac{1}{\mu_{i(t)}}$. Therefore

$$E\left[2\sigma_{i(t)}\sum_{d\in D_t} w_{G_j}(d,r)\right] = E\left[2\frac{\sigma_{i(t)}}{\mu_{i(t)}}\sum_{d\in D} w_{G_j}(d,r)\right] = E\left[2\delta_{i(t)}\sum_{d\in D} w_{G_j}(d,r)\right].$$

Let $L_{t,j}$ be the cost of routing the flow as in OPT, but paying zero on the edges of G_j, and $\delta_{i(t)}$ per unit of flow on the remaining edges. Then trivially $\delta_{i(t)}\sum_{d\in D} w_{G_j}(d,r) \leq L_{t,j}$. In turn, OPT pays at least δ_s per unit flow on each cable of type $s \leq j$, which implies $L_{t,j} \leq \sum_{s\leq j}\frac{\delta_{i(t)}}{\delta_s}OPT(s)$. We can conclude that

$$E\left[2\sigma_{i(t)}\sum_{d\in D_t} w_{G_j}(d,r)\right] \leq E\left[2\sum_{s\leq j}\frac{\delta_{i(t)}}{\delta_s}OPT(s)\right].$$

Altogether

$$E[A_t^c] \le E\left[2\sum_{s\le j}\frac{\delta_{i(t)}}{\delta_s}OPT(s) + \frac{0.8067}{\alpha}\sum_{s>j}\frac{\sigma_{i(t+1)}}{\sigma_s}OPT(s)\right].$$

Since, deterministically, $\delta_{i(t)}/\delta_s$ is decreasing in t while $\sigma_{i(t+1)}/\sigma_s$ is increasing in t, the claim follows by choosing j properly. □

We can now state the main result of this section.

Theorem 6.2 *There is an expected* 20.41 *approximation algorithm for* CabSsbb.

Proof. From Lemmas 6.2 and 6.3, the cost A of the approximate solution computed by Algorithm 3 satisfies:

$$\begin{aligned}E[A] &= A_0 + E[A_{k'}] + \sum_{t=1}^{k'-1}(E[A_t^c] + E[A_t^a] + E[A_t^r])\\ &\le \rho_{\mathrm{ST}}\sum_s \frac{\sigma_{i(1)}}{\sigma_s}OPT(s) + \sum_s\frac{\delta_{i(k')}}{\delta_s}OPT(s) + \sum_{t=1}^{k'-1}E\left[\sum_s \min\left\{\rho_{\mathrm{ST}}\alpha\frac{\delta_{i(t)}}{\delta_s}, \rho_{\mathrm{ST}}\frac{\sigma_{i(t+1)}}{\sigma_s}\right\}OPT(s)\right]\\ &\quad+\sum_{t=1}^{k'-1}E\left[\left(1+\frac{\delta_{i(t+1)}}{\delta_{i(t)}}\right)\sum_s\min\left\{2\frac{\delta_{i(t)}}{\delta_s}, \frac{0.8067}{\alpha}\frac{\sigma_{i(t+1)}}{\sigma_s}\right\}OPT(s)\right].\end{aligned}$$

Define

$$\begin{aligned}apx(s) :=& \rho_{\mathrm{ST}}\frac{\sigma_{i(1)}}{\sigma_s} + \frac{\delta_{i(k')}}{\delta_s} + \\ &+\sum_{t=1}^{k'-1}\left(\left(1+\frac{\delta_{i(t+1)}}{\delta_{i(t)}}\right)\min\left\{2\frac{\delta_{i(t)}}{\delta_s}, \frac{0.8067}{\alpha}\frac{\sigma_{i(t+1)}}{\sigma_s}\right\} + \min\left\{\rho_{\mathrm{ST}}\alpha\frac{\delta_{i(t)}}{\delta_s}, \rho_{\mathrm{ST}}\frac{\sigma_{i(t+1)}}{\sigma_s}\right\}\right),\end{aligned}$$

so that $E[A] \le \sum_s E[apx(s)]OPT(s)$. By Lemma 6.1(a),

$$\begin{aligned}apx(s) &\le \rho_{\mathrm{ST}}\frac{\sigma_{i(1)}}{\sigma_s} + \frac{\delta_{i(k')}}{\delta_s} + \sum_{t=1}^{k'-1}\min\left\{\left(2\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\alpha\right)\frac{\delta_{i(t)}}{\delta_s}, \left(\frac{0.8067}{\alpha}\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\right)\frac{\sigma_{i(t+1)}}{\sigma_s}\right\}\\ &\le \rho_{\mathrm{ST}}\frac{\sigma_{i(1)}}{\sigma_s} + \frac{\delta_{i(k')}}{\delta_s} + \max\left\{2\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\alpha, \frac{0.8067}{\alpha}\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\right\}\sum_{t=1}^{k'-1}\min\left\{\frac{\delta_{i(t)}}{\delta_s}, \frac{\sigma_{i(t+1)}}{\sigma_s}\right\}\\ &\le \frac{\delta_{i(k')}}{\delta_s} + \max\left\{2\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\alpha, \frac{0.8067}{\alpha}\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\right\}\sum_{t=0}^{k'-1}\min\left\{\frac{\delta_{i(t)}}{\delta_s}, \frac{\sigma_{i(t+1)}}{\sigma_s}\right\}.\end{aligned}$$

Consider any cable type s, and let $i(t') < s \le i(t'+1)$. Assume $t' \le k'-2$ (the analysis is analogous for $t' = k'-1$). By Lemma 6.1,

$$\begin{aligned}E\left[\sum_{t=0}^{k'-1}\min\left\{\frac{\delta_{i(t)}}{\delta_s}, \frac{\sigma_{i(t+1)}}{\sigma_s}\right\}\right] &\le \frac{1}{\delta_s}\sum_{t=t'+1}^{k'-1}E[\delta_{i(t)}] + E\left[\min\left\{\frac{\delta_{i(t')}}{\delta_s}, \frac{\sigma_{i(t'+1)}}{\sigma_s}\right\}\right] + \frac{1}{\sigma_s}\sum_{t=0}^{t'-1}E[\sigma_{i(t+1)}]\\ &\le \frac{E[\delta_{i(t'+1)}]}{\delta_s}\sum_{j=0}^{k'-t'-2}\frac{1}{\beta^j} + \beta + \frac{E[\sigma_{i(t')}]}{\sigma_s}\sum_{j\ge 0}\frac{1}{\beta^j} \le \frac{2\beta}{\beta-1} + \beta - \frac{E[\delta_{i(t'+1)}]}{\delta_s(\beta-1)\beta^{k'-t'-2}}.\end{aligned}$$

Let us set $\alpha = 0.531$ and $\beta = 2.80$. Observe that

$$\frac{\delta_{i(k')}}{\delta_s} \le \frac{E[\delta_{i(k'-1)}]}{\delta_s} \le \frac{E[\delta_{i(t'+1)}]}{\delta_s\beta^{k'-t'-2}} \le \max\left\{2\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\alpha, \frac{0.8067}{\alpha}\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\right\}\frac{E[\delta_{i(t'+1)}]}{\delta_s(\beta-1)\beta^{k'-t'-2}}.$$

Hence

$$E[apx(s)] \le \max\left\{2\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\alpha, \frac{0.8067}{\alpha}\left(1+\frac{1}{\beta}\right)+\rho_{\mathrm{ST}}\right\}\cdot\left(\frac{2\beta}{\beta-1}+\beta\right) \overset{\alpha=0.531,\beta=2.80}{<} 20.41.$$

The claim follows. □

COROLLARY 6.1 *There is an expected* 20.41 *approximation algorithm for* SSBB.

PROOF. The claim follows from Lemma 2.2 and Theorem 6.2 (the rounding mistakes in the proof of the theorem absorb the extra factor $1+\varepsilon$). □

COROLLARY 6.2 *There is a* $2 \cdot 20.41 = 40.82$ *approximation algorithm for* UNSSSBB.

PROOF. The claim follows from Theorem 6.2 and Lemma 2.3. □

Acknowledgments. The third author is supported by Swiss National Science Foundation within the project "Robust Network Design".

References

[1] Y. Bartal, *On approximating arbitrary metrics by tree metrics*, Symposium on Theory of Computing (STOC), ACM, 1998, pp. 161–168.

[2] L. Becchetti, J. Könemann, S. Leonardi, and M. Pál, *Sharing the cost more efficiently: improved approximation for multicommodity rent-or-buy*, ACM Transactions on Algorithms **3** (2007), no. 2.

[3] J. Byrka, F. Grandoni, T. Rothvoß, and L. Sanità, *An improved LP-based approximation for Steiner tree*, Symposium on Theory of Computing (STOC), ACM, 2010, pp. 583–592.

[4] C. Chekuri, *Routing and network design with robustness to changing or uncertain traffic demands*, SIGACT News **38** (2007), no. 3, 106–128.

[5] M. Chlebik and J. Chlebikova, *Approximation hardness of the Steiner tree problem on graphs*, Scandinavian Workshop on Algorithm Theory (SWAT), Springer, 2002, pp. 170–179.

[6] W. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver, *Combinatorial optimization*, John Wiley & Sons Inc., 1998.

[7] N. G. Duffield, P. Goyal, A. Greenberg, P. Mishra, K. K. Ramakrishnan, and J. E. van der Merwe, *A flexible model for resource management in virtual private networks*, SIGCOMM, ACM, 1999, pp. 95–108.

[8] F. Eisenbrand and F. Grandoni, *An improved approximation algorithm for virtual private network design*, Symposium on Discrete Algorithms (SODA), SIAM, 2005, pp. 928–932.

[9] F. Eisenbrand, F. Grandoni, G. Oriolo, and M. Skutella, *New approaches for virtual private network design*, SIAM Journal on Computing **37** (2007), no. 3, 706–721.

[10] F. Eisenbrand, F. Grandoni, T. Rothvoß, and G. Schäfer, *Connected facility location via random facility sampling and core detouring*, Journal of Computer and System Sciences **76** (2010), 709–726.

[11] T. Erlebach and M. Rüegg, *Optimal bandwidth reservation in hose-model VPNs with multi-path routing*, Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), IEEE, 2004, pp. 2275–2282.

[12] J. Fakcharoenphol, S. Rao, and K. Talwar, *A tight bound on approximating arbitrary metrics by tree metrics*, Journal of Computer and System Sciences **69** (2004), no. 4, 485–497.

[13] J. Fingerhut, S. Suri, and J. Turner, *Designing least-cost nonblocking broadband networks*, Journal of Algorithms **24** (1997), no. 2, 287–309.

[14] S. Fiorini, G. Oriolo, L. Sanità, and D. Theis, *The VPN problem with concave costs*, SIAM Journal on Discrete Mathematics **24** (2010), no. 3, 1080–1090.

[15] L. Fleischer, J. Könemann, S. Leonardi, and G. Schäfer, *Simple cost sharing schemes for multicommodity rent-or-buy and stochastic steiner tree*, Symposium on Theory of Computing (STOC), ACM, 2006, pp. 663–670.

[16] N. Garg, A. Gupta, S. Leonardi, and P. Sankowski, *Stochastic analyses for online combinatorial optimization problems*, Symposium on Discrete Algorithms (SODA), SIAM, 2008, pp. 942–951.

[17] N. Garg, R. Khandekar, G. Konjevod, R. Ravi, F. Salman, and A. Sinha, *On the integrality gap of a natural formulation of the single-sink buy-at-bulk network design problem*, International Conference on Integer Programming and Combinatorial Optimization (IPCO), Springer, 2001, pp. 170–184.

[18] N. Goyal, N. Olver, and F. B. Shepherd, *The VPN conjecture is true*, Symposium on Theory of Computing (STOC), ACM, 2008, pp. 443–450.

[19] F. Grandoni and G. F. Italiano, *Improved approximation for single-sink buy-at-bulk*, International Symposium on Algorithms and Computation (ISAAC), Springer, 2006, pp. 111–120.

[20] F. Grandoni, V. Kaibel, G. Oriolo, and M. Skutella, *A short proof of the VPN tree routing conjecture on ring networks*, Operations Research Letters **36** (2008), no. 3, 361–365.

[21] F. Grandoni and T. Rothvoß, *Network design via core detouring for problems without a core*, International Colloquium on Automata, Languages and Programming (ICALP), Springer, 2010, pp. 490–502.

[22] S. Guha, A. Meyerson, and K. Munagala, *A constant factor approximation for the single sink edge installation problem*, SIAM Journal on Computing **38** (2009), no. 6, 2426–2442.

[23] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, and B. Yener, *Provisioning a virtual private network: a network design problem for multicommodity flow*, Symposium on Theory of Computing (STOC), ACM, 2001, pp. 389–398.

[24] A. Gupta, A. Kumar, M. Pal, and T. Roughgarden, *Approximation via cost-sharing: simpler and better approximation algorithms for network design*, Journal of the ACM **53** (2007), no. 3, 11.

[25] A. Gupta, A. Kumar, and T. Roughgarden, *Simpler and better approximation algorithms for network design*, Symposium on Theory of Computing (STOC), ACM, 2003, pp. 365–372.

[26] A. Gupta and M. Pál, *Stochastic Steiner trees without a root*, International Colloquium on Automata, Languages and Programming (ICALP), Springer, 2005, pp. 1051–1063.

[27] 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** (2007), no. 2, 482–503.

[28] G. F. Italiano, S. Leonardi, and G. Oriolo, *Design of trees in the hose model: The balanced case*, Operations Research Letters **34** (2006), no. 6, 601–606.

[29] R. Jothi and B. Raghavachari, *Improved approximation algorithms for the single-sink buy-at-bulk network design problems*, Scandinavian Workshop on Algorithm Theory (SWAT), Springer, 2004, pp. 336–348.

[30] D. R. Karger and M. Minkoff, *Building Steiner trees with incomplete global knowledge*, Symposium on Foundations of Computer Science (FOCS), IEEE Computer Society, 2000, pp. 613–623.

[31] A. Meyerson, K. Munagala, and S. Plotkin, *Cost-distance: two metric network design*, Symposium on Foundations of Computer Science (FOCS), IEEE Computer Society, 2000, pp. 624–630.

[32] T. Rothvoß and L. Sanità, *On the complexity of the asymmetric VPN problem*, International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX), Springer, 2009, pp. 326–338.

[33] L. Sanità, *Robust network design*, Ph.D. thesis, Università Sapienza di Roma, 2009.

[34] A. Schrijver, *Combinatorial optimization. Polyhedra and efficiency*, Springer-Verlag, 2003.

[35] D. B. Shmoys and K. Talwar, *A constant approximation algorithm for the a priori traveling salesman problem*, International Conference on Integer Programming and Combinatorial Optimization (IPCO), Springer, 2008, pp. 331–343.

[36] C. Swamy and A. Kumar, *Primal–dual algorithms for connected facility location problems*, International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX), Springer, 2002, pp. 256–269.

[37] K. Talwar, *The single-sink buy-at-bulk LP has constant integrality gap*, International Conference on Integer Programming and Combinatorial Optimization (IPCO), Springer, 2002, pp. 475–486.

[38] A. van Zuylen, *Deterministic sampling algorithms for network design*, European Symposium on Algorithms (ESA), Springer, 2008, pp. 830–841.

Appendix A. Omitted Proofs.

Proof of Lemma 2.1. Let $x = \{x_e\}_{e \in E}$ be a given non-tree solution to Ssbb. The cost of this reservation is $g(x) := \sum_{e \in E} w(e)\phi(x_e)$. Observe that $g(\cdot)$ is concave. We will describe a polynomial-time procedure which computes a different (feasible) capacity reservation x' of not larger cost and such that the number of edges with zero capacity in x' is strictly larger that the same number in x. By applying this procedure a polynomial number of times one obtains the desired tree solution.

Let $P_1, \ldots, P_h$ be the flow paths corresponding to x (each one carrying a positive amount of flow). We can assume $h \leq |E| + |V|$ by standard max-flow techniques. Let $f(e)$ be the flow along edge e. Without loss of generality, $f(e) = x_e$: otherwise we could decrease x_e hence getting a cheaper feasible solution. We can also assume that each (undirected) edge $e = \{u, v\}$ carries a flow directed from u to v or vice versa. In fact, otherwise we could decrease $f(e)$ (and hence x_e) and obtain a cheaper feasible solution. Let us replace $\{u, v\}$ with the directed edge (u, v) in the first case, and with (v, u) otherwise. For a node v, by $f(v)$ we denote the sum of the flows of edges entering v minus the sum of the flows of edges leaving v. Observe that $f(v) = 0$ for all nodes but the sources v, where it is $-d(v)$, and the sink z, where it is $\sum_{v \in D} d(v)$.

Since x is not a tree solution, there must be two *conflicting* paths P_i and P_j that, after meeting, split at some node w. Let $k, h \in \{i, j\}$, $h \neq k$. We define P'_k as the subpath of P_k from w to the sink z, and $\tilde{P}_k := P'_k \setminus P'_h$. In particular, $\tilde{P}_i \cup \tilde{P}_j$ is the symmetric difference of P'_i and P'_j. Observe that $\tilde{P}_i$ and $\tilde{P}_j$ are non-empty sets of (directed) disjoint paths.

Let $y_k := \min_{e \in \tilde{P}_k}\{f(e)\} > 0$ be the minimum flow over any edge of $\tilde{P}_k$. Consider the flow f^k which is obtained by decreasing f by y_k on the edges of $\tilde{P}_k$, and increasing f by the same amount on the edges of $\tilde{P}_h$. Let E_v be the edges of $\tilde{P}_k \cup \tilde{P}_h$ incident to v. One of the following two cases must occur: (1) E_v contains an even number of edges of both $\tilde{P}_k$ and $\tilde{P}_h$; (2) E_v contains exactly one edge of $\tilde{P}_k$ and one of $\tilde{P}_h$, and these two edges either both leave v or both enter v. It is easy to check that in any case $f^k(v) = f(v)$, and hence f^k is a feasible flow.

We constructed two feasible flows f^i and f^j, which induce two capacity reservations x^i and x^j, respectively. Note that both x^i and x^j contain at least one more zero entry with respect to x. Observe also that $x = \alpha x^j + (1 - \alpha)x^i$, where $\alpha = \frac{y_i}{y_i + y_j}$. Being $g(\cdot)$ concave, $g(x) \geq \alpha\, g(x^j) + (1 - \alpha)g(x^i)$, and hence $g(x) \geq \min\{g(x^i), g(x^j)\}$. In words, one of the two capacity reservations x^i and x^j is not more expensive than x. The claim follows. ☐

Proof of Lemma 2.2. By Lemma 2.1, there is always an optimal tree solution, and consequently a capacity reservation with integral values in $\{0, 1, \ldots, |D|\}$.

Hence, it is sufficient to construct a cable of capacity $(1 + \varepsilon)^i$ and cost $\phi((1 + \varepsilon)^i)$ for each $i = 0, 1, \ldots, \lceil \log_{1+\varepsilon} |D| \rceil$. This induces a (polynomial-size) CabSsbb instance of cost at most $(1 + \varepsilon)$ times the optimal cost for the input Ssbb instance. In fact, any time the optimum solution installs a capacity x_e on edge e, we can rather install a cable of capacity x'_e, $x_e \leq x'_e \leq (1 + \varepsilon)x_e$ on the same edge. The cost of this cable is $\phi(x'_e) \leq \phi((1 + \varepsilon)x_e) \leq (1 + \varepsilon)\phi(x_e)$. The claim follows. ☐

Proof of Lemma 2.3. Let $\mathcal{I}$ be the input UnsSsbb instance. Of course, $OPT_{\text{CabSsbb}}(\mathcal{I}) \leq OPT_{\text{UnsSsbb}}(\mathcal{I})$. Hence, it is sufficient to show that any CabSsbb solution S to $\mathcal{I}$, and in particular a ρ-approximate solution, can be turned into a tree solution U of cost at most twice the cost of S: this solution U induces a feasible solution to the original problem of cost at most $2\rho \cdot OPT_{\text{CabSsbb}}(\mathcal{I}) \leq 2\rho \cdot OPT_{\text{UnsSsbb}}(\mathcal{I})$.

Let $\phi^*(x_e)$ be the minimum-cost of a multi-set of cable types of capacity at least $x_e \geq 0$. Define

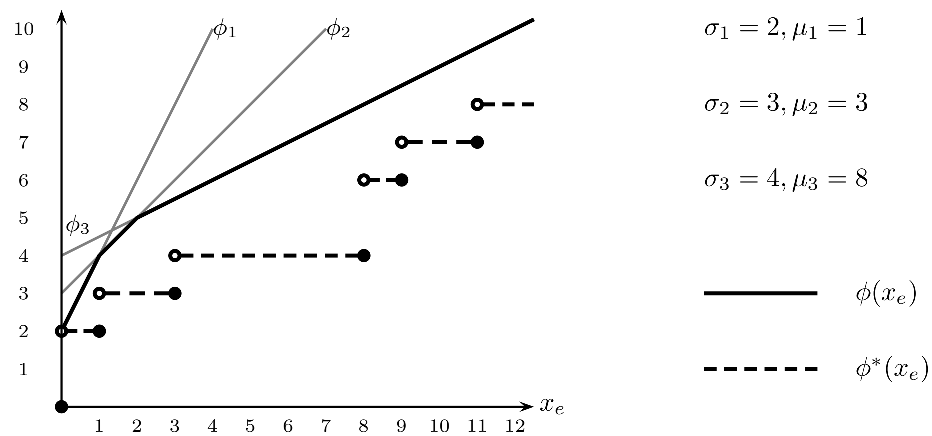
$$\phi_i(x_e) = \begin{cases} 0 & \text{if } x_e = 0; \\ \sigma_i + \delta_i \cdot x_e & \text{if } x_e > 0, \end{cases}$$

and let $\phi(x_e) := \min_{i=1,\ldots,k}\{\phi_i(x_e)\}$ (see also [37] and Figure 3). Observe that $\phi(\cdot)$ is concave and piecewise linear (for $x_e > 0$) with at most k slopes. It can be easily computed in polynomial time.

We next show that

$$\phi^*(x_e) \leq \phi(x_e) \leq 2\phi^*(x_e).$$

Figure 3 Example of functions $\phi^*(x_e)$ and $\phi(x_e)$, $x_e \geq 0$.



For $x_e = 0$ this trivially holds. Now fix a value of $x_e > 0$ and let i be the largest cable, used for obtaining capacity x_e at cost $\phi^*(x_e)$. Then $\phi^*(x_e) \geq \sigma_i$ and $\phi^*(x_e) \geq \delta_i \cdot x_e$, hence $\phi_i(x_e) = \sigma_i + \delta_i \cdot x_e \leq 2\phi^*(x_e)$. On the other hand choose i such that $\phi(x_e) = \phi_i(x_e)$. We can then install x_e units of capacity with $\lceil \frac{x_e}{\mu_i} \rceil$ copies of cable i. Thus $\phi^*(x_e) \leq \lceil \frac{x_e}{\mu_i} \rceil \sigma_i \leq \sigma_i + x_e \cdot \delta_i = \phi_i(x_e) = \phi(x_e)$.

Now, let $x = \{x_e\}_{e \in E}$ be the capacity reserved by S, and $g^*(x) := \sum_{e \in E} w(e)\phi^*(x_e)$ be its cost. Consider the Ssbb instance associated to the capacity cost function $\phi(\cdot)$, of total cost

$$g(x) := \sum_{e \in E} w(e)\phi(x_e) \leq 2 \sum_{e \in E} w(e)\phi^*(x_e) = 2g^*(x).$$

By Lemma 2.1, we can construct a new tree solution x' of cost $g(x') \leq g(x)$. We eventually install on each edge e, $\lceil x'_e/\mu_i \rceil$ copies of cable type i, where i minimizes the cost $\sigma_i \lceil x'_e/\mu_i \rceil$: this defines the solution U.

Let j be the cable minimizing $\phi(x'_e)$, i.e. $\phi_j(x'_e) = \phi(x'_e)$. Observe that $\sigma_i \lceil x'_e/\mu_i \rceil \leq \sigma_j \lceil x'_e/\mu_j \rceil \leq \sigma_j + x'_e \cdot \delta_j = \phi_j(x'_e) = \phi(x'_e)$. We can conclude that the cost of U is at most $\sum_{e \in E} w(e)\phi(x'_e) = g(x') \leq 2g^*(x)$. The claim follows. □

Proof of Lemma 6.2.

(1) We prove the claim by induction. The claim for $t = 1$ is a straightforward consequence of the properties of the aggregation algorithm. Assume that the claim is true for some $t \geq 1$. Consider any $d \in D_t$. Let w be the sampled node which collects the flow from d during the t-th collection step, and let $d'_t(w) = b\,\mu_{i(t)}$, for some integer $b \geq 1$. Given b,

$$\Pr[d \in D_{t+1} | d \in D_t] = \frac{1}{b}\frac{E[d''_t(w)]}{\mu_{i(t+1)}} = \frac{1}{b}\frac{d'_t(w)}{\mu_{i(t+1)}} = \frac{1}{b}\frac{b\,\mu_{i(t)}}{\mu_{i(t+1)}} = \frac{\mu_{i(t)}}{\mu_{i(t+1)}}.$$

We can conclude that

$$\Pr[d \in D_{t+1} | d \in D_0] = \Pr[d \in D_{t+1} | d \in D_t] \cdot \Pr[d \in D_t | d \in D_0] = \frac{\mu_{i(t)}}{\mu_{i(t+1)}}\frac{1}{\mu_{i(t)}} = \frac{1}{\mu_{i(t+1)}}.$$

(2) The subgraph G' induced by OPT satisfies $c(G') \leq \sum_s \frac{1}{\sigma_s} OPT(s)$. Since G' spans $D_0 \cup \{r\}$, the latter quantity is also an upper bound on the cost of an optimum Steiner tree T^* over $D_0 \cup \{r\}$. The claim follows by observing that $A_0 \leq \sigma_1 \rho_{\mathrm{ST}}\, c(T^*)$.

(3) Suppose we could route the flow along the same paths of OPT, but at cost $\delta_{i(k')} = \delta_k$ per unit capacity. Let C' be the cost of this routing. Observe that $C' \leq \sum_s \frac{\delta_k}{\delta_s} OPT(s)$. Now replace each flow path with a shortest path to the sink, and let C'' be the corresponding cost: trivially, $C'' \leq C'$. By (1), in the final step each demand sends μ_k units of flow to the sink with probability $\frac{1}{\mu_k}$, and at a cost of δ_k per unit flow and unit length. Hence $E[A_{k'}] = C''$. The claim follows.

(4) We will construct, for any given j, a random graph G_t spanning $D'_t \cup \{r\}$, of expected cost

$$E[c(G_t)] \leq \sum_{s>j} \frac{1}{\sigma_s} OPT(s) + \sum_{s\leq j} \frac{\alpha\, \delta_{i(t)}}{\delta_s \sigma_{i(t+1)}} OPT(s).$$

The latter cost, multiplied by $\rho_{\mathrm{ST}}\sigma_{i(t+1)}$, gives the desired bound for a proper choice of j.

We initially add to G_t all the edges where OPT installs at least one cable of type $j+1$ or larger. The cost of these edges is at most $\sum_{s>j} \frac{1}{\sigma_s} OPT(s)$. Then we consider each $d \in D_0$. Let $P_1, \ldots, P_h$ be the flow paths carrying the (unit) demand of d in OPT, and let $f_i \in (0,1]$ be the flow carried by P_i. (Observe that $\sum_{i=1}^{h} f_i = 1$). If $d \in D'_t$, which happens with probability $\frac{1}{\mu_{i(t)}} \frac{\alpha\, \sigma_{i(t)}}{\sigma_{i(t+1)}} = \frac{\alpha\, \delta_{i(t)}}{\sigma_{i(t+1)}}$, we choose one of the flow paths P_i at random, according to the probability distribution induced by the f_i's, and add it to G_t. Observe that the final graph G_t spans $D'_t \cup \{r\}$ as required. Consider any edge of G_t introduced during the second phase only. For the sake of simplicity, assume a unique cable of type $s \leq j$ is installed on e by OPT. (The same type of analysis can be carried over on a cable-by-cable basis). Each path P_i using e, makes e belong to G_t with probability $\frac{\alpha\, \delta_{i(t)} f_i}{\sigma_{i(t+1)}}$. It follows from the union bound that e belongs to G_t with probability at most $\frac{\alpha\, \delta_{i(t)} x^*_e}{\sigma_{i(t+1)}}$, where $x^*_e \leq \mu_s$ is the flow carried by the cable installed on e. Therefore e contributes with at most $\frac{c(e)\alpha\, \delta_{i(t)} \mu_s}{\sigma_{i(t+1)}}$ to the cost of G_t in expectation. On the other side, OPT pays $c(e)\sigma_s$ for the cable installed on e. Hence the total cost of the edges added to G_t in the second phase only is at most $\sum_{s\leq j} \frac{c(e)\alpha\, \delta_{i(t)}\mu_s}{\sigma_{i(t+1)}} \frac{1}{c(e)\sigma_s} OPT(s) = \sum_{s\leq j} \frac{\alpha\, \delta_{i(t)}}{\sigma_{i(t+1)}\delta_s} OPT(s)$. The claim follows.

(5) Let us charge to each $d \in D_t$ the corresponding collection and redistribution cost during the t-th round. Source d pays $\sigma_{i(t)} w(d, D'_t \cup \{r\})$ to send $\mu_{i(t)}$ units of flow to the sampled nodes during the t-th collection step. By the proof of (1), with probability $\frac{\mu_{i(t)}}{\mu_{i(t+1)}}$, d receives $\mu_{i(t+1)}$ units of flow back during the t-th redistribution step, at a cost of $\sigma_{i(t+1)} w(d, D'_t \cup \{r\})$. Hence the redistribution cost of d is, in expectation, $\frac{\sigma_{i(t+1)}\mu_{i(t)}}{\mu_{i(t+1)}} \frac{1}{\sigma_{i(t)}} = \frac{\delta_{i(t+1)}}{\delta_{i(t)}}$ times the collection cost of d. The claim follows. □