Exploring the Tractability of the Capped Hose Model

Thomas Bosman[1] and Neil Olver[2]

1 **Dept. of Econometrics & Operations Research, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands**
`t.n.bosman@vu.nl`

2 **Dept. of Econometrics & Operations Research, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands; and**
CWI, Amsterdam, The Netherlands
`n.olver@vu.nl`

Abstract

Robust network design concerns the design of networks to support uncertain or varying traffic patterns. An especially important case is the *VPN problem*, where the total traffic emanating from any node is bounded, but there are no further constraints on the traffic pattern. Recently, Fréchette et al. [10] studied a generalization of the VPN problem where in addition to these so-called hose constraints, there are individual upper bounds on the demands between pairs of nodes. They motivate their model, give some theoretical results, and propose a heuristic algorithm that performs well on real-world instances.

Our theoretical understanding of this model is limited; it is APX-hard in general, but tractable when either the hose constraints or the individual demand bounds are redundant. In this work, we uncover further tractable cases of this model; our main result concerns the case where each terminal needs to communicate only with two others. Our algorithms all involve *optimally embedding* a certain auxiliary graph into the network, and have a connection to a heuristic suggested by Fréchette et al. for the capped hose model in general.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems, G.1.6 Optimization, G.2.2 Graph Theory

Keywords and phrases robust network design, VPN problem

Digital Object Identifier 10.4230/LIPIcs.ESA.2017.19

1 Introduction

Robust network design (RND) [2] is concerned with designing networks that can efficiently handle uncertain or varying utilization. The motivation comes primarily (though not exclusively) from communication networks. Let $G = (V, E)$, be a graph with edge costs that describes an existing, high-capacity network. A set of *terminals* $W \subseteq V$ is required to communicate over the network, and to enable this, we must reserve capacity on the edges of G for our exclusive use (this is in order to guarantee reliable performance). On each edge, we may buy multiple units of capacity (measured, say, in Mb/s); the cost of the edge represents the per-unit cost of capacity. In the RND framework, demand uncertainty is described by a *demand universe* $\mathcal{U}$. It is simply a set containing all of the demands that need to be routed; the choice of this set will be determined by operational needs or historical data. More precisely, each $D \in \mathcal{U}$ is a matrix where entry D_{ij} describes the demand (measured again, say, in Mb/s) from terminal i to terminal j. It turns out that the universe can always be taken to be a convex set, and will frequently be a polytope.

25th Annual European Symposium on Algorithms (ESA 2017).
Editors: Kirk Pruhs and Christian Sohler; Article No. 19; pp. 19:1–19:12
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

We will consider only single-path, oblivious routing (other routing schemes are possible, but less relevant to practice). This means that a solution to the RND problem must specify, for each pair of terminals $i, j \in W$, a path P_{ij} that will be used to route the demand between this pair. This path must be fixed ahead of time, and cannot be adjusted as a function of the current demand. Once all these paths have been fixed, a capacity reservation must be made on the network. For any edge $e \in E$, the capacity $u(e)$ must be chosen so that no matter which demand matrix $D \in \mathcal{U}$ is instantiated, the total amount of traffic traversing e does not exceed $u(e)$.

The most studied case of universe is the *hose model* [5, 8]. Here, each terminal $i \in W$ has an associated marginal b_i, and the universe $\mathcal{H}(\boldsymbol{b})$ consists of all demand matrices D for which $\sum_{j \in W} D_{ij} \leq b_i$ for all i, and $D_{ij} = D_{ji}$[1]. The optimization problem for the hose model is called the VPN problem, and it was shown by Goyal et al. [11] to be polynomially solvable.

While the hose model is particularly appealing, especially given its exact solvability, it is very natural to consider generalizations with more expressive modelling power. A number of such generalizations have been considered in the literature [7, 17, 10, 9]. One such generalization, the *capped hose model* was introduced by Fréchette, Shepherd, Thottan and Winzer [10]. It is very natural: in addition to the hose constraints b_i for each $i \in W$, there is an upper bound d_{ij} on the demand between a given pair $i, j \in W$. This leads to the capped hose polytope $\mathcal{H}^{\text{cap}}(\boldsymbol{b}, \boldsymbol{d})$. If $d_{ij} = \infty$ for all pairs $i, j \in W$, then this recovers the hose model; and if $b_i = \infty$ for all $i \in W$, then this recovers the *pipe model*, the somewhat trivial case where the problem is to route a single fixed demand matrix. We refer to Fréchette et al. [10] for further discussion and motivation.

As Fréchette et al. [10] observed, the problem of finding the cheapest solution in the capped hose model generalizes Steiner tree, and hence is APX-hard. Simply consider, for an arbitrarily chosen root $r \in W$, the choice $b_i = 1$ for all $i \in W$, and $d_{ir} = d_{ri} = 1$ for all $i \in W$, $d_{ij} = 0$ otherwise. Beyond this, the complexity and approximability of this problem is poorly understood. In particular, it is open as to whether there is a constant factor approximation algorithm. (The general robust network design problem is hard to approximate within polylog factors [17], but this construction does not apply to this restricted setting.) Moreover, the RND problem under $\mathcal{H}^{\text{cap}}(\boldsymbol{b}, \boldsymbol{d})$ is polynomially solvable for some choices of $\boldsymbol{b}$ and $\boldsymbol{d}$, for example when $\boldsymbol{d}$ is sufficiently large (recovering the hose model), or $\boldsymbol{b}$ is sufficiently large (recovering the pipe model). Our goal in this work is to expand the class of exactly solvable cases.

We focus on the setting where $b_i = 1$ for all $i \in W$ and $d_{ij} \in \{0, \infty\}$ (or equivalently, $d_{ij} \in \{0, 1\}$) for all $i, j \in W$, which we call the *masked hose model.* Instead of parametrizing an instance with $\boldsymbol{d}$ and $\boldsymbol{b}$, we can describe it via the *mask graph* H, which has vertex set W and an edge between each pair of terminals which may communicate. In other words, the universe is the set of all fractional matchings in H. The resulting *masked VPN problem* is a clean generalization of the standard VPN problem (the case where H is the complete graph), and is already very rich from a theoretical standpoint. Again, it is not known if a constant approximation factor is possible for arbitrary mask graphs, and the case where H is a star is APX-hard. It is harmless to restrict to connected mask graphs, since otherwise the problem can be solved separately on each connected component, and the resulting solutions overlaid in G.

[1] This is the symmetric hose model; an asymmetric variant which does not require $D_{ij} = D_{ji}$ is also possible, and different [14, 6, 13].

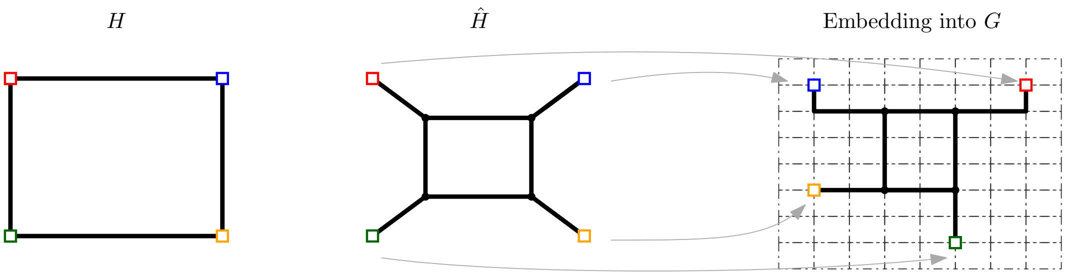


Figure 1 The embedding algorithm for H a cycle; in this example, G is a grid.

Our main result is the following

▶ **Theorem 1.** *The masked VPN problem is polynomially solvable if H is a cycle.*

The algorithm is based on embedding an appropriate auxiliary graph (see Figure 1). Let $\hat{H}$ denote the graph obtained by replacing each terminal i by a new node $\hat{\imath}$, and then adding back the terminal i along with the edge $\{i, \hat{\imath}\}$. We give each edge e of $\hat{H}$ a capacity of 1. An *embedding* of $\hat{H}$ into G is simply a mapping ϕ satisfying the following. Each node of $\hat{H}$ is mapped to a node of G, with $\phi(i) = i$ for all $i \in W$; and each edge $\{u, v\} \in E(\hat{H})$ maps to a path in G between $\phi(u)$ and $\phi(v)$. Any embedding implies a path in G between any adjacent pair of adjacent terminals $\{i, j\} \in E(H)$; simply the image under the embedding of the path $(i, \hat{\imath}, \hat{\jmath}, j)$. After assigning a capacity reservation $u(e) = |\{f \in E(\hat{H}) : e \in \phi(f)\}|$, it is easy to see that this yields a feasible solution to the masked VPN problem for H. Our algorithm simply finds the cheapest possible embedding of $\hat{H}$ into G; this can easily be done by dynamic programming. We show that this is optimal; an overview of the proof strategy can be found in Section 3.1

The cycle may seem like a very specific and restricted case. But understanding cycles has historically been an important stepping stone in the area towards more general results. The VPN Conjecture on the polynomial solvability of the hose model was first solved for the case where the network is a cycle [15, 12], and ideas from [12] were crucial for the resolution of the full conjecture.

We also prove the following.

▶ **Theorem 2.** *The masked VPN problem is exactly solvable if H is a tree with bounded degree.*

Technically, this result is much more straightforward, and we give the proof in Section 4. It exploits the well-known Dreyfus-Wagner algorithm for Steiner tree on a fixed number of terminals [4], which corresponds to the case where H is a bounded degree star. We make heavy use of the dual viewpoint, discussed in Section 2, in order to argue that the solution can be efficiently decomposed into Steiner tree problems. And while our focus is on exactly solvable cases, we remark that an $O(1)$-approximation without any degree bound can readily be obtained (see Theorem 15 in Section 4).

While this result does not require major technical novelty, it yields an interesting message. The algorithm can *also* be interpreted as an embedding algorithm. This time, however, there are multiple options regarding which graph to embed, and we have to choose the best. Begin by constructing $\hat{H}$ in the same fashion as above, splitting out each terminal. But now we go further; for each node $v \in V(\hat{H})$ with degree 4 or more, consider all possible ways of "blowing up" v into a tree with only degree 3 nodes (see Figure 2). Each possible way of blowing up

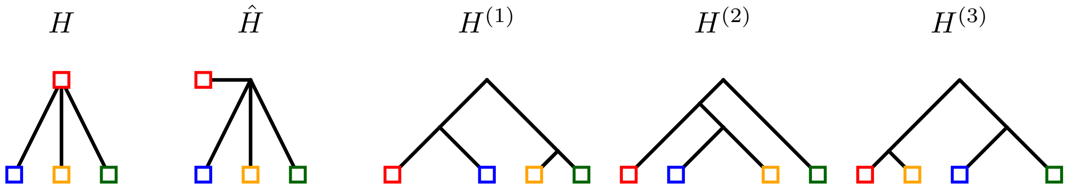


Figure 2 Potential graphs to embed in the case where H is a star.

each node v yields a graph whose embedding yields a solution. The algorithm computes the cheapest possible embedding from all of these possibilities; by using dynamic programming, combined with the assumption of bounded degree, this can be done in polynomial time. Again, this is precisely the idea of the Dreyfus-Wagner algorithm for Steiner tree [4], extended from H a star to H a tree. We also observe that if H was a path, then $\hat{H}$ has maximum degree 3, and so only $\hat{H}$ itself needs to be embedded.

Further, embedding algorithms of this form have been used before in RND, though only embeddings of *trees*. For the standard VPN problem, the optimal solution is simply the optimal embedding of a star with a leaf for each terminal [11]. This is very natural when one considers that the demand universe for the hose model – fractional matchings on the complete graph – is nothing more than the set of demands that are routable on such a star. A generalization of the hose model (different to the one discussed here) defines the universe to be the set of demands routable on a given capacitated tree (with leaf set equal to W) [17]. It has been conjectured that the optimal algorithm is given by the optimal embedding of this tree [18]; it is only known that this yields a constant factor approximation [17].

Partially motivated by this, Fréchette et al. proposed a tree embedding algorithm as a heuristic for the capped hose model. The tree they embed is chosen carefully, albeit heuristically, and they show that this performs well in practice. Our work can be seen as providing an initial theoretical basis for their approach, while suggesting that extending beyond trees may be beneficial.

2 Problem definition and preliminaries

An instance of the masked VPN (mVPN) problem consists of a graph $G = (V, E)$, with edge costs $c : E \to \mathbb{R}_+$ (where $\mathbb{R}_+$ denotes the nonnegative reals), a set of terminals $W \subseteq V$, and a second graph H which has W as its vertex set.

We use $\binom{W}{2}$ to denote the collection of unordered pairs of distinct terminals. The demand universe is defined as

$$\mathcal{H}^{\text{mask}}(H) := \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : \sum_j D_{ij} \leq 1 \ \forall i \text{ and } D_{ij} = 0 \text{ unless } \{i,j\} \in E(H) \Big\}.$$

Our goal is to specify a routing template $\mathcal{P} = \{P_{ij} \ : \{i,j\} \in E(H)\}$, where P_{ij} is a fixed (possibly non-simple) i-j-path used for traffic between terminal i and j. (P_{ij} and P_{ji} refer to the same path.) Given a set of routing paths, we are required to make a capacity reservation $u : E \to \mathbb{R}$, sufficient to route any traffic vector in $\mathcal{H}^{\text{mask}}(H)$. The minimum capacity requirement on edge e is therefore

$$u(e) = \max_{D \in \mathcal{H}^{\text{mask}}(H)} \sum_{\{i,j\} \in \binom{W}{2} : e \in P_{ij}} D_{ij}. \tag{1}$$

The resulting solution has cost $\sum_e c(e)u(e)$, and this we wish to minimize.

We will often take a dual viewpoint of (1). This viewpoint has been exploited before, see [1, 15, 11]. Since (1) is a fractional matching problem, its dual is a fractional vertex cover problem:

$$\begin{aligned} u(e) = \quad & \min \quad \sum_{i \in W} y_i(e) \\ & \text{s.t.} \quad y_i(e) + y_j(e) \geq 1 \qquad \forall \{i,j\} \in E(H),\ e \in P_{ij} \\ & \qquad\quad y_i(e) \geq 0\,. \end{aligned} \tag{2}$$

So, we may rephrase the problem as follows. Each terminal i buys a capacity vector y_i, with the property that $\{e \in E : y_i(e)+y_j(e) \geq 1\}$ contains an i-j-path, for each $\{i,j\} \in E(H)$. The goal is to minimize the total cost $\sum_i c(y_i)$, where $c(y_i) = \sum_{e \in E} c(e) y_i(e)$.

Let $\boldsymbol{y}$ denote the vector whose i'th component y_i is the capacity vector purchased by $i \in W$. At this point we note that, since (2) is a fractional vertex cover problem, $\boldsymbol{y}$ can always assumed to be half-integral. In fact, we will mainly be able to restrict ourselves to integral capacity vectors. In such a case it is convenient to express the solution as a collection of edge sets $\boldsymbol{Y} = (Y_i)_{i \in W}$ where $Y_i = \{e : y_i(e) = 1\}$.

▶ Remark. Through this dual viewpoint, a connection can be made with the work of Iglesias et al. [16]. With a completely different motivation, they consider essentially this problem, explicitly requiring integrality but also *connectivity* of the set of edges purchase by each terminal. Since we show that the optimal solutions satisfy these properties, our results apply in their setting as well.

3 The cycle case

We consider the case where H is a cycle. Let k denote the number of terminals, and assume for convenience that $W = \{1, 2, \ldots, k\}$, with the ordering corresponding to the cycle structure of H. We will interpret all references to terminals modulo k; so terminal 0 refers to terminal k, and terminal $k+1$ to terminal 1.

Our main technical theorem shows that there is always an optimal solution to the mVPN problem that satisfies a simple structure.

▶ **Theorem 3** (Hubbed solution)**.** *There exists an optimal solution to the* mVPN *problem on cycles such that:*
- *for each terminal i there exists a* hub *vertex h_i; and*
- *the routing path $P_{i,i+1}$ is given by concatenating shortest paths from i to h_i, from h_i to h_{i+1}, and from h_{i+1} to $i+1$.*

The optimal location of the hub vertices minimizes the cost

$$\sum_{i \in W} \mathrm{sp}_c(i, h_i) + \mathrm{sp}_c(h_i, h_{i+1}),$$

where sp_c denotes the shortest path distance with respect to the edge costs c. Since H is a cycle the optimal location of hubs $h_{i+1}, \ldots, h_{j-1}$ are independent of hubs $h_{j+1}, \ldots, h_{i-1}$ given the location of h_i and h_j, so we can find these hubs in polynomial time with dynamic programming, yielding Theorem 1.

3.1 Overview

The proof of Theorem 3 involves first showing that there is always an optimal solution of a certain nice form, albeit not yet of the hubbed form we are looking for. We first argue that we may restrict our focus to solutions $\boldsymbol{y}$ that are integral. Next, we show that there is an integral solution satisfying a certain structure theorem. Roughly speaking, this structure is similar to the hubbed structure we are looking for, but instead of a single hub h_i, there is an odd-cardinality set T_i; instead of a path between i and h_i, we have a $(\{i\} \triangle T_i)$-join; and instead of a path between h_i and h_{i+1}, we have a $(T_i \triangle T_{i+1})$-join. The final step of the argument is then to show that we can take $|T_i| = 1$ for each i, which is then precisely a hubbed solution. This step uses a rather non-obvious "rotation" of the solution to reduce the cardinality of the T_i's.

3.2 Integrality

It will be convenient to work with integral solutions. If k is even, so that H is bipartite, then each fractional matching problem in (2) has an integral optimum. We have to work a bit harder in the case where k is odd.

▶ **Lemma 4.** *There exists an integral optimal solution to the cycle* mVPN *problem.*

Proof. Since any strict subgraph of a cycle is bipartite, the only case where (2) does not have an integral optimal solution, is if it corresponds to a vertex cover problem on the complete cycle. This only happens if the routing path between *every* pair of neighbours uses the edge. So suppose $e = \{u, v\}$ is used on every routing path in a solution $\boldsymbol{y}$. We claim the integral solution $\boldsymbol{Z}$, given by taking Z_i to be the edges of a shortest i-v-path for all terminals i, costs no more than $\boldsymbol{y}$.

Let D be a traffic vector with $D_{i,i+1} = \frac{1}{2}$ for all i. Now if we route D according to the solution $\boldsymbol{y}$, the flow between i and $i+1$ can be split into half a unit of i-v flow and half a unit of v-$(i+1)$-flow, since every routing path passes through e, and thus v.

So D induces a unit i-v flow for each $i \in W$. So $\boldsymbol{y}$ has sufficient capacity to route 1 unit of flow from each $i \in W$ simultaneously. But this costs at least as much as the sum of the shortest paths from each terminal to v, as required. ◀

For the remainder of the proof we will therefore assume that each terminal buys a set of edges. It will be useful to partition these edges into a different collection of sets based on the routing paths they support.

▶ **Definition 5.** A *feasible solution* $\bar{\boldsymbol{X}}$ consists of edge sets $\bar{X}_i$ and $\bar{X}_{i,i+1}$ for each i, such that all edge sets are disjoint and for each i there exists a path $P_{i,i+1}$ connecting terminal i to $i+1$ with

$$P_{i,i+1} \subseteq \bar{X}_i \cup \bar{X}_{i,i+1} \cup \bar{X}_{i+1}.$$

The cost of the solution is $\sum_i c(\bar{X}_i) + \sum_i c(\bar{X}_{i,i+1})$.

One should think of $\bar{X}_i$ as the edges on both $P_{i,i+1}$ and $P_{i-1,i}$ and $\bar{X}_{i,i+1}$ as the remaining edges on $P_{i,i+1}$. Such a solution can be transformed into a feasible solution in original form by setting $X_i = \bar{X}_i \cup \bar{X}_{i,i+1}$ for all i. Indeed, the definition does not confer any advantage to choosing any of the sets $\bar{X}_{i,i+1}$ to be nonempty. But as we will see in the next section, this formulation provides a natural way to express the structure of a feasible solution.

3.3 Structure theorem

For the remainder we will assume that G is a complete graph satisfying the triangle inequality. We may simply replace G with its metric completion to ensure this, and it allows us to bypass some substantial technical awkwardness.

▶ **Definition 6.** Let $\boldsymbol{T}$ be a collection consisting of an odd cardinality sets $T_i \subseteq V \setminus W$ for each terminal i. Then a *$\boldsymbol{T}$-solution* $\bar{\boldsymbol{X}}$ consists of a collection of edge sets $\bar{X}_i$ and $\bar{X}_{i,i+1}$ for each terminal i satisfying:
1. $\bar{X}_i$ is a perfect matching on $T_i \triangle \{i\}$,
2. $\bar{X}_{i,i+1}$ is a perfect matching on $T_i \triangle T_{i+1}$.

The cost of the solution is $\sum_i c(\bar{X}_i) + \sum_i c(\bar{X}_{i,i+1})$.

It is good to observe that Property 1 and 2 of this definition imply that $\bar{\boldsymbol{X}}$ is a feasible solution. The odd degree vertices of $\bar{X}_i \,\dot\cup\, \bar{X}_{i,i+1} \,\dot\cup\, \bar{X}_{i+1}$ are precisely

$$T_i \triangle \{i\} \triangle T_i \triangle T_{i+1} \triangle T_{i+1} \triangle \{i+1\} = \{i, i+1\},$$

implying that i and $i+1$ are in the same component.

The restriction in the above definition that $T_i \cap W = \emptyset$ is without loss of generality, since we may always modify any given instance by replacing each terminal i in the instance and the solution with a new dummy node $\bar{i}$, and then replacing i in the instance at the same location, attaching the terminal to this dummy node by an edge of zero cost, and including $\{i, \bar{i}\}$ in $\bar{X}_i$.

The following lemma shows that we do not lose anything if we restrict ourselves to $\boldsymbol{T}$-solutions.

▶ **Lemma 7** (Weak Structure Lemma). *Any feasible solution $\bar{\boldsymbol{X}}$ may be transformed into a $\boldsymbol{T}$-solution $\bar{\boldsymbol{Y}}$ of no higher cost for some $\boldsymbol{T}$ satisfying $T_i \subseteq V(\bar{X}_i) \setminus \{i\}$ for all $i \in W$.*

Proof. Define:
- $\bar{Y}'_i = \bar{X}_i \cap E(P_{i-1,i}) \cap E(P_{i,i+1})$, and
- $\bar{Y}'_{i,i+1} = E(P_{i,i+1}) \setminus (\bar{X}_i \cup \bar{X}_{i+1})$.

We then obtain $\bar{\boldsymbol{Y}}$ from $\bar{\boldsymbol{Y}}'$ by shortcutting paths, so that $\bar{Y}_i$ is a collection of vertex disjoint edges for each $i \in W$.

Now for each terminal i choose the vertex set T_i such that $T_i \triangle \{i\}$ is the set of vertices incident to an edge in $\bar{Y}_i$. We claim that $\bar{\boldsymbol{Y}}$ is a $\boldsymbol{T}$-solution.

By construction, $\bar{Y}_i$ is a perfect matching on $T_i \triangle \{i\}$. To see that $\bar{Y}_{i,i+1}$ is a perfect matching on $T_i \triangle T_{i+1}$, note that since $\bar{Y}'_i$ and $\bar{Y}'_{i+1}$ are both contained in the path $P_{i,i+1}$, we may write

$$\bar{Y}'_{i,i+1} = E(P_{i,i+1}) \triangle \bar{Y}_i \triangle \bar{Y}_{i+1}.$$

Thus the odd degree nodes of $\bar{Y}'_{i,i+1}$ are precisely

$$\{i, i+1\} \triangle T_i \triangle \{i\} \triangle T_{i+1} \triangle \{i+1\} = T_i \triangle T_{i+1}.$$

As the odd degree nodes in $\bar{Y}'_{i,i+1}$ and $\bar{Y}_{i,i+1}$ are equal, the result follows. ◀

▶ **Definition 8.** A *strong $\boldsymbol{T}$-solution* is a $\boldsymbol{T}$-solution with the additional properties:
- **(i)** $\bar{X}_i \,\dot\cup\, \bar{X}_{i,i+1} \,\dot\cup\, \bar{X}_{i+1}$ consists of a single i-$(i+1)$-path, and
- **(ii)** each edge in $\bar{X}_{i,i+1}$ is incident to one vertex in T_i and one in T_{i+1}.

Notice that in a strong $\boldsymbol{T}$-solution $\bar{\boldsymbol{X}}$, $|T_i| = |T_j|$ for all $i, j \in W$.

▶ **Lemma 9** (Strong Structure Lemma). *Any $\boldsymbol{T}$-solution $\bar{\boldsymbol{X}}$ can be transformed into a strong $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$ of no higher cost, with $R_i \subseteq T_i$ for all $i \in W$.*

For the proof of this lemma we will need two auxilliary lemmas.

▶ **Lemma 10.** *Let $\bar{\boldsymbol{X}}$ be a $\boldsymbol{T}$-solution such that for some $i \in W$, $\bar{X}_i \dot\cup \bar{X}_{i,i+1} \dot\cup \bar{X}_{i+1}$ does not satisfy property (i) of Definition 8. Then there exists an $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$ of no higher cost, with $R_j \subseteq T_j$ for all $j \in W$, and $R_i \subsetneq T_i$.*

Proof. Since i and $i+1$ are the only vertices that do not have degree 2 in $\bar{X}_i \dot\cup \bar{X}_{i,i+1} \dot\cup \bar{X}_{i+1}$, the connected component containing i and $i+1$ contains an i-$(i+1)$-path; call it P.

Define $R_i = T_i \cap V(P)$, $R_{i+1} = T_{i+1} \cap V(P)$, and $R_j = T_j$ for all other $j \in W$. Note that $R_i \subsetneq T_i$. We will now construct a $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$ as follows. Define $\bar{Y}_j = \bar{X}_j$ for all $j \notin \{i, i+1\}$, and $\bar{Y}_{j,j+1} = \bar{X}_{j,j+1}$ for all $j \notin \{i-1, i, i+1\}$. Now define $\bar{Y}_{i,i+1} = \bar{X}_{i,i+1} \cap P$, so it is a perfect matching on $R_i \triangle R_{i+1}$ with $c(\bar{Y}_{i,i+1}) \leq c(\bar{X}_{i,i+1})$. Also let $\bar{Y}_i = \bar{X}_i \cap P$, which is a perfect matching on $R_i \triangle \{i\}$. We have $c(\bar{Y}_i) = c(\bar{X}_i) - c(Q)$, where $Q = \bar{X}_i \setminus P$ is a perfect matching on $T_i \setminus R_i$. To define $\bar{Y}_{i-1,i}$, first let

$$\bar{Y}'_{i-1,i} = \bar{X}_{i-1,i} \triangle (\bar{X}_i \setminus P).$$

Notice that the odd degree nodes of $\bar{Y}'_{i-1,i}$ are precisely

$$(T_{i-1} \triangle T_i) \triangle (T_i \setminus R_i) = T_{i-1} \triangle R_i = R_{i-1} \triangle R_i.$$

Now, by discarding any cycles and shortcutting paths, we can choose $\bar{Y}_{i-1,i}$ to be a perfect matching on $R_{i-1} \triangle R_i$ that costs no more than $\bar{Y}'_{i-1,i}$. So we have $c(\bar{Y}_{i-1,i}) \leq c(\bar{X}_{i-1,i}) + c(Q)$.

We make precisely the symmetric construction to define $\bar{Y}_{i+1}$ and $\bar{Y}_{i+1,i+2}$. We have obtained the required $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$. ◀

▶ **Lemma 11.** *Let $\bar{\boldsymbol{X}}$ be a $\boldsymbol{T}$-solution that satisfies Property (i) of Definition 8 but where Property (ii) fails for some terminal i. Then there exists an $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$ of no higher cost, with $R_j \subseteq T_j$ for all $j \in W$, and $R_i \subsetneq T_i$.*

Proof. Suppose w.l.o.g. $e = \{u, v\}$ is an edge in $\bar{X}_{i,i+1}$ with both $u, v \in T_i$. Because of Property (i) we know there exists a u-v-path in $\bar{X}_{i-1} \dot\cup \bar{X}_{i-1,i} \dot\cup \bar{X}_i$, say Q.

Let us define a new solution $\bar{\boldsymbol{Y}}$ equal to $\bar{\boldsymbol{X}}$ except for:

$$\begin{aligned} \bar{Y}_i &= (\bar{X}_i \setminus E(Q)) \cup \{e\} \\ \bar{Y}_{i,i+1} &= \bar{X}_{i,i+1} \cup (\bar{X}_i \cap E(Q)) \setminus \{e\}. \end{aligned}$$

The i-$(i+1)$-path in $\bar{\boldsymbol{X}}$ is still feasible in $\bar{\boldsymbol{Y}}$, and we can get an $(i-1)$-i-path $\bar{\boldsymbol{Y}}$ from the respective path in $\bar{\boldsymbol{X}}$, by replacing the subpath Q with the edge e. Thus, $\bar{\boldsymbol{Y}}$ is a feasible solution.

Let P be the maximal path in $\bar{Y}_i$ that contains e. Since every edge on P is used both on some $(i-1)$-i-path and i-$(i+1)$-path, we can replace P by an edge connecting the endpoints in $\bar{Y}_i$ and retain a feasible solution, with the property that

$$V(\bar{Y}_i) \setminus \{i\} \subsetneq V(\bar{X}_i) \setminus \{i\} = T_i,$$

and $V(\bar{Y}_j) = V(\bar{X}_j)$ for $j \neq i$. By Lemma 7 it now follows that we can find an $\boldsymbol{R}$-solution $\bar{\boldsymbol{Z}}$ with $R_i \subseteq V(Y_i) \setminus \{i\} \subsetneq T_i$ and $R_j \subseteq T_j$ for $j \in W$, as required. ◀

Proof. Lemma 9 We arrive at our Lemma from the fact that we can alternatingly apply Lemmas 10 and 11 to a $\boldsymbol{T}$-solution $\bar{\boldsymbol{X}}$ until we have a strong $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$. Since every time we apply Lemma 10, $\sum_{i \in W} |T_i|$ strictly decreases, this procedure must terminate in a finite number of steps. ◀

3.4 From a $\boldsymbol{T}$-solution to an optimal embedding

▶ **Observation 12.** *Suppose $\bar{\boldsymbol{X}}$ is a strong $\boldsymbol{T}$-solution with $|T_i| = 1$ for all $i \in W$. Then $\bar{\boldsymbol{X}}$ is a hubbed solution.*

As we will see, as long as we have a strong $\boldsymbol{T}$-solution that is not a hubbed solution (implying that $|T_i| = \alpha > 1$ for some α and all i), we can find an $\boldsymbol{R}$-solution such that $\boldsymbol{R}$ is strictly smaller than $\boldsymbol{T}$.

▶ **Lemma 13.** *Given a strong $\boldsymbol{T}$-solution $\bar{\boldsymbol{X}}$ with $|T_i| > 1$ for all i, there exists a strong $\boldsymbol{R}$-solution $\bar{\boldsymbol{Y}}$ of no higher cost with $R_i \subsetneq T_{i+1}$ for all i.*

Proof. We claim that we can find a new solution $\bar{\boldsymbol{Y}}$ with $V(\bar{Y}_i) = T_{i+1} \setminus \{u_{i+1}\} \cup \{i\}$ for some node $u_{i+1} \in T_{i+1}$. It then follows by Lemma 7 and Lemma 9 that we can find a strong $\boldsymbol{R}$-solution $\bar{\boldsymbol{Z}}$ with

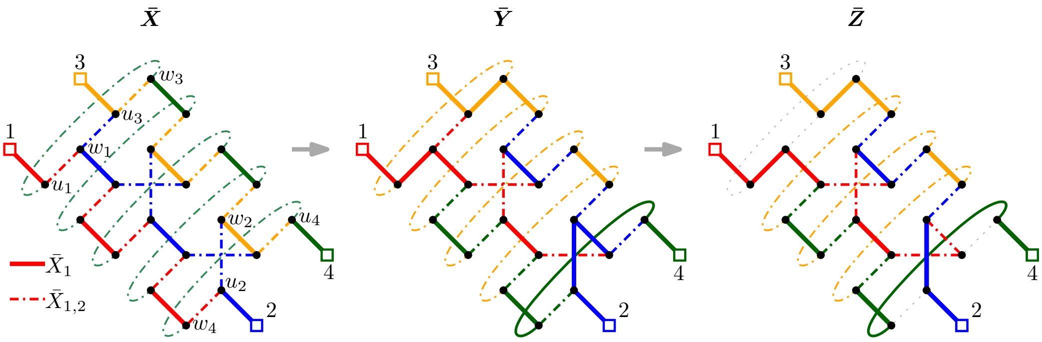
$$R_i \subseteq V(\bar{Y}_i) \setminus \{i\} = T_{i+1} \setminus \{u_{i+1}\},$$

which implies the required result.

For each terminal i we define $u_i \in T_i$ as the node matched to i in $\bar{X}_i$, and $w_i \in T_{i+1}$ as u_i if $u_i \in T_{i+1}$, or the vertex matched to u_i in $\bar{X}_{i,i+1}$ otherwise. Finally let L_i denote the i-u_i-w_i path in $\bar{X}_i \cup \bar{X}_{i,i+1}$.

Now take a solution $\bar{\boldsymbol{Y}}$ equal to $\bar{\boldsymbol{X}}$ except for

$$\bar{Y}_i = \{\{i, w_i\}\} \cup \bar{X}_{i+1} \setminus L_{i+1}$$
$$\text{and} \qquad \bar{Y}_{i,i+1} = \bar{X}_{i+1,i+2} \setminus L_{i+1} \setminus L_{i+2}.$$



We will show that $|T_i| > 1$ implies that $\bar{Y}_i \cup \bar{Y}_{i,i+1} \cup \bar{Y}_{i+1}$ contains an i-$(i+1)$-path. Note that:

$$\bar{Y}_i \cup \bar{Y}_{i,i+1} \cup \bar{Y}_{i+1} = \{\{i, w_i\}, \{i+1, w_{i+1}\}\} \cup E(P_{i+1,i+2} \setminus L_{i+1} \setminus L_{i+2}).$$

As $w_j \in T_{j+1}$ for all j, clearly $P_{i+1,i+2}$ contains a w_i-w_{i+1}-subpath. Since $|T_j| > 1$ for all $j \in W$, we must have that L_{i+1} and L_{i+2} are vertex disjoint, and therefore $E(P_{i+1,i+2} \setminus L_{i+1} \setminus L_{i+2})$ induces a single non-empty connected component.

Now

$$V(P_{i+1,i+2} \setminus L_{i+1} \setminus L_{i+2}) = \begin{cases} V(P_{i+1,i+2}) \setminus \{i+1, u_{i+1}, i+2\} & \text{if } u_{i+1} \neq w_{i+1} \\ V(P_{i+1,i+2}) \setminus \{i+1, i+2\} & \text{otherwise} \end{cases}.$$

But as L_i and L_{i+1} are vertex disjoint, $w_i \neq u_{i+1}$. Therefore $P_{i+1,i+2} \setminus L_{i+1} \setminus L_{i+2}$ contains a w_i-w_{i+1}-path, implying that $\bar{Y}_i \cup \bar{Y}_{i,i+1} \cup \bar{Y}_{i+1}$ contains an i-$(i+1)$-path.

We conclude that $\bar{\boldsymbol{Y}}$ is a feasible solution. Finally note that:

$$\begin{aligned} V(\bar{Y}_i) &= V(\bar{X}_{i+1}) \setminus \{i+1, u_{i+1}\} \cup \{i, w_i\} \\ &= T_{i+1} \triangle \{i+1\} \setminus \{i+1, u_{i+1}\} \cup \{i, w_i\} \\ &= T_{i+1} \setminus \{u_{i+1}\} \cup \{i\}, \end{aligned}$$

where in the last equality we have used that $w_i \in T_{i+1}$. This proves our claim and hence the lemma. ◀

Recall that $|T_i| = |T_j|$ for all $i, j \in W$ in a strong $\boldsymbol{T}$-solution. By repeatedly applying Lemma 13, we obtain an optimal $\boldsymbol{T}$-solution with $|T_i| = 1$ for all $i \in W$, which by Observation 12 is a hubbed solution. This completes the proof of Theorem 3.

4 The tree case

We consider the case where H is a bounded-degree tree. Since H is bipartite, we may restrict ourselves to integral solutions to (2). We first show that there is an optimal solution of a particular form, which we refer to as a *hubbed solution*.

▶ **Lemma 14.** *There exists an optimal solution $\boldsymbol{Y}$ to the tree* MVPN *problem, such that, for some choice $h_{ij} \in V$ for each $\{i,j\} \in E(H)$ (which we call* hub vertices*), Y_i is the edge set of a Steiner tree with terminals $\{i\} \cup \{h_{ij} : \{i,j\} \in E(H)\}$.*

Proof. We prove that we can transform an arbitrary solution $\boldsymbol{Y}$ into a feasible solution $\boldsymbol{Z}$ of the required form.

Choose an arbitrary terminal and consider H to be rooted at this node. Let $C(i)$ denote the set of children of terminal i in H. We construct $\boldsymbol{Z}$ as follows.

We initialize $Z_i = \emptyset$ for all leaf terminals i. Now suppose we have defined Z_j for all the children of a node i. Then define

$$Z_i' = \bigcup_{j \in C(i)} \{e \in Y_i \cup Y_j : e \in P_{ij}\} \setminus Z_j.$$

Now let Z_i be the connected component of (V, Z_i') that contains i. By working up from the leaves of H, this clearly defines $\boldsymbol{Z}$.

Since $Z_i \subseteq \bigcup_{j \in \{i\} \cup C(i)} Y_j$ for all i, and $Z_i \cap Z_j = \emptyset$ for all $j \in C(i)$, $\boldsymbol{Z}$ costs no more in $\boldsymbol{Y}$.

To see that $\boldsymbol{Z}$ is indeed feasible and of the required form, note that for any terminal i and child $j \in C(i)$, by definition $Z_i' \cup Z_j$ must contain an i-j-path. If Z_j is empty, clearly Z_i must contain an i-j path. We set $h_{ij} = j$ and we are done. If not, then there exists a vertex h_{ij} in the single nonempty connected component of Z_j such that Z_i' contains a path from i to h_{ij}. But that path must be contained in the connected component of Z_i' that contains i, which is exactly Z_i, as required. ◀

With this structural lemma in place, Theorem 2 follows easily.

Proof. Theorem 2 We can solve the Steiner tree problem for a fixed number of terminals in polynomial time [4]. Therefore, for H a tree of bounded degree, finding an optimal solution reduces to finding the location of the hub vertices. We will show that we can do this efficiently with dynamic programming.

Suppose we root H at some terminal r. For each terminal i we let $\zeta(i,h)$ denote the minimum cost of the edge sets bought by all terminals in the subtree rooted at i, over all hubbed solutions such that the hub between i and its parent is located at h.

Now, define $\mathrm{MSt}(X)$ for $X \subseteq V$ as the minimum cost of a Steiner tree on terminal set X. We can calculate $\zeta(\cdot,\cdot)$ recursively as follows:

$$\zeta(i,h) = \min_{(h_j)_{j\in C(i)} \in V^{C(i)}} \mathrm{MSt}(\{i,h\} \cup \{h_j : j \in C(i)\}) + \sum_{j\in C(i)} \zeta(j,h_j).$$

In other words, we simply try all possible combinations of choices of $h_{j_1}, h_{j_2}, \dots$ for $j_1, j_2, \dots \in C(i)$. As the degree of H is bounded, the number of combinations is polynomially bounded. Since we can solve the Steiner tree instance in polynomial time as well, the result follows. ◀

As remarked in the introduction, a constant factor approximation can easily be obtained when H is an arbitrary tree, again from the structural lemma.

▶ **Theorem 15.** *The* mVPN *problem where H is a tree has a 2α-approximation, where α is the approximation ratio for Steiner tree.*

Proof. Root H at an arbitrary terminal. Again, let $C(i)$ denote the set of children of terminal i. Take any optimal solution $\boldsymbol{Y}$. Then define a new solution $Z_i := \bigcup_{j\in\{i\}\cup C(i)} Y_j$, which costs at most $2OPT$. Now for each terminal i, Z_i contains a Steiner tree on i and its children $C(i)$.

Let X_i be an α-approximate Steiner tree on i and its children $C(i)$. Then $\boldsymbol{X}$ is a feasible solution, and $c(\boldsymbol{X}) \le \alpha \cdot c(\boldsymbol{Z}) \le 2\alpha \cdot OPT$. ◀

At the time of writing the best known approximation for Steiner tree is $\ln 4 + \epsilon < 1.39$ [3], so the mVPN problem where H is a tree is approximable within 2.78.

5 Conclusion

Our results for H a tree can be extended easily to the capped hose model where the support (edges with $d_{ij} > 0$) forms a tree. If the support of $\boldsymbol{d}$ is a cycle, but $\boldsymbol{b}$ and $\boldsymbol{d}$ are otherwise arbitrary, the situation is unclear. There is a natural analog of the embedding algorithm. First, ensure that the components of $\boldsymbol{b}$ and $\boldsymbol{d}$ are all minimal, i.e., no component can be decreased without changing the uncertainty set. Then compute the cheapest embedding of the weighted version of the graph used in Section 3; edges $\{i, h_i\}$ get weight b_i, and edges $\{i, i+1\}$ get weight $d_{i,i+1}$. We leave it as an open question whether this algorithm is always optimal. More speculatively, we feel that our results suggest that embedding algorithms may play a deeper role in the subject than is currently apparent.

References

1 A. Altın, E. Amaldi, P. Belotti, and M. C. Pınar. Provisioning virtual private networks under traffic uncertainty. *Networks*, 49(1):100–115, 2007. `doi:10.1002/net.v49:1`.

2 Walid Ben-Ameur and Hervé Kerivin. New economical virtual private networks. *Commun. ACM*, 46(6):69–73, 2003. `doi:10.1145/777313.777314`.

3 Jarosław Byrka, Fabrizio Grandoni, Thomas Rothvoss, and Laura Sanità. Steiner tree approximation via iterative randomized rounding. *Journal of the ACM (JACM)*, 60(1):6, 2013. `doi:10.1145/1806689.1806769`.

4 Stuart E. Dreyfus and Robert A. Wagner. The Steiner problem in graphs. *Networks*, 1(3):195–207, 1971.

5 N. G. Duffield, Pawan Goyal, Albert Greenberg, Partho Mishra, K. K. Ramakrishnan, and Jacobus E. van der Merwe. A flexible model for resource management in virtual private networks. In *Proc. SIGCOMM*, pages 95–108, 1999. doi:10.1145/316188.316209.

6 Friedrich Eisenbrand, Fabrizio Grandoni, Gianpaolo Oriolo, and Martin Skutella. New approaches for virtual private network design. *SIAM J. Comput.*, 37(3):706–721, 2007.

7 Friedrich Eisenbrand and Edda Happ. Provisioning a virtual private network under the presence of non-communicating groups. In *Proc. CIAC*, pages 105–114, 2006. doi:10.1007/11758471_13.

8 J. A. Fingerhut, S. Suri, and J. S. Turner. Designing least-cost nonblocking broadband networks. *J. Algorithms*, 24(2):287–309, August 1997. doi:10.1006/jagm.1997.0866.

9 S. Fiorini, G. Oriolo, L. Sanità, and D. O. Theis. The VPN problem with concave costs. *SIAM J. Discrete Math.*, 24(3):1080–1090, 2010.

10 Alexandre Fréchette, F. Bruce Shepherd, Marina K. Thottan, and Peter J. Winzer. Shortest path versus multi-hub routing in networks with uncertain demand. In *Proc. INFOCOM*, pages 710–718, 2013. doi:10.1109/INFCOM.2013.6566857.

11 Navin Goyal, Neil Olver, and F. Bruce Shepherd. The VPN Conjecture is true. *J. ACM*, 60(3):17:1–17:17, 2013. doi:10.1145/2487241.2487243.

12 Fabrizio Grandoni, Volker Kaibel, Gianpaolo Oriolo, and Martin Skutella. A short proof of the VPN tree routing conjecture on ring networks. *Oper. Res. Lett.*, 36(3):361–365, 2008. doi:10.1016/j.orl.2007.10.008.

13 Fabrizio Grandoni, Thomas Rothvoß, and Laura Sanità. From uncertainty to nonlinearity: Solving virtual private network via single-sink buy-at-bulk. *Mathematics of Operations Research*, 36(2):185–204, 2011. doi:10.1287/moor.1110.0490.

14 Anupam Gupta, Amit Kumar, Martin Pál, and Tim Roughgarden. Approximation via cost sharing: Simpler and better approximation algorithms for network design. *J. ACM*, 54(3):11, 2007. doi:10.1145/1236457.1236458.

15 C. A. J. Hurkens, J. C. M. Keijsper, and L. Stougie. Virtual private network design: A proof of the tree routing conjecture on ring networks. *SIAM J. Discrete Math.*, 21(2):482–503, 2007.

16 Jennifer Iglesias, Rajmohan Rajaraman, R. Ravi, and Ravi Sundaram. Designing overlapping networks for publish-subscribe systems. In *Proc. APPROX*, pages 381–395, 2015. doi:10.4230/LIPIcs.APPROX-RANDOM.2015.381.

17 N. Olver and F. B. Shepherd. Approximability of robust network design. In *Proc. SODA*, pages 1097–1105, 2010. doi:10.1137/1.9781611973075.89.

18 Neil Olver. A note on hierarchical hubbing for a generalization of the VPN problem. *Oper. Res. Lett.*, 44(2):191–195, 2016. doi:10.1016/j.orl.2015.12.020.