Designing networks under uncertain demands (draft)

Neil Olver
Vrije Universiteit Amsterdam & CWI

Hausdorff Summer School on Combinatorial Optimization 2018

Version updated August 21, 2018

1 Notation

Characteristic vectors. The following notation is not standard, but will work very nicely for us. It's closely related to the Iverson bracket notation promoted by Knuth [40].

Given a groundset X and a subset $S \subseteq X$, we define $[S] : X \to \{0,1\}$ to be the characteristic function of S:

$$[S](i) = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{if } i \notin S \end{cases}.$$

We can also extend this to multisets: if S is a multiset, $[S](e)$ will denote the number of times element e is present in S.

Unordered pairs. We use $\binom{W}{2}$ to denote the set of unordered pairs of distinct elements of a subset W. For concreteness, if W is ordered, we can consider $\binom{W}{2} = \{(i,j) \in W^2 : i < j\}$. So $\sum_{i<j}(\cdots)$ and $\sum_{\{i,j\}\in\binom{W}{2}}(\cdots)$ are the same.

Miscellaneous. We use $[n]$ as shorthand for $\{1,2,\ldots,n\}$.

If $u : X \to \mathbb{R}$ and $S \subseteq X$, then $u(S)$ is shorthand for $\sum_{x\in S} u(x)$.

Given a graph $G = (V,E)$ and a subset $S \subseteq V$, $\delta_G(S)$ denotes the set of edges crossing S, i.e., $\delta_G(S) = \{e \in E : |e \cap S| = 1\}$. We write just $\delta(S)$ when there can be no confusion over the choice of G.

Given $G = (V,E)$ and $i, j \in V$, an **i-j-path** P is a path with endpoints i and j.

2 Uncertain demands: the robust network design model

2.1 Motivation and model

At least historically, the main motivation for the model discussed in these lectures comes from communications networks. While it's not hard to come up with applications in other settings (such as logistics), we'll stick to just this motivation (and we won't go into it much).

We are given an undirected graph $G = (V, E)$, with edge costs $c(e)$. A set of "terminals" $W \subseteq V$ are also given (which we can assume are labelled $1, 2, \ldots$). Imagine we are tasked with building a network that connects the terminals, by buying capacity on G. in order to ensure a certain level of service; by reserving the bandwidth, we ensure that the private network is not adversely affected by traffic from the rest of the network. Or G might represent something physical, with $c(e)$ representing something like the distance between the endpoints of e. Suppose, at any rate, that $c(e)$ represents a *cost per unit bandwidth* on that edge; if we buy $u(e)$ units of capacity, we will pay $c(e)u(e)$. So we're assuming a linear cost structure here. We will also assume that there is no upper bound to the amount of capacity we can buy on an edge.

Now if we just wanted to connect the terminals, this would give us the Steiner tree problem. But our notion of connectivity takes into account bandwidth: how fast must the connection be between various terminal pairs? Suppose that we require D_{ij} units of bandwidth (measured, say, in Mb/s) between terminal i and terminal j. (We consider this demand to be undirected; so D_{ij} and D_{ji} are the same thing). Given all of this, what is the cheapest way of buying capacity so that we can *simultaneously* route all the demands D_{ij}, for every pair $\{i, j\}$?

This is extremely simple, essentially because there is no sharing of capacity between different routing pairs. Compute a shortest path between every pair of terminals; say P_{ij} is a shortest path between i and j. We then reserve, cumulatively, an amount D_{ij} on path P_{ij}, for every $i \neq j \in W$. It is clear that this is optimal, and (using $\mathrm{SP}_c(i, j)$ to denote the length of a shortest path between i and j) has cost

$$\mathrm{cost}(D) = \sum_{i<j} D_{ij} \mathrm{SP}_c(i, j).$$

Not a very interesting problem so far...

Uncertain demands. However, the traffic pattern of a real-world network is typically not fixed; rather, it varies over time. Moreover, it is often difficult to measure or estimate traffic patterns reliably in large networks, even if these traffic patterns are roughly static. Robust network design deals with this uncertainty in traffic patterns via the methodology of robust optimization. We assume that the demand, while not fixed, comes from some prescribed *universe* of possible demands. The solution we give must be able to route any demand matrix in this universe.

Informally, our problem is thus: given a network $G = (V, E)$, costs $c(e)$, terminal set $W \subseteq V$ and universe $\mathcal{U} \subseteq \mathbb{R}_+^{\binom{W}{2}}$, find a cheapest capacity allocation $u : E \to \mathbb{R}_+$ so that we can "handle" every demand $D \in \mathcal{U}$.

But what do we mean by being able to "handle" a demand? There are various possibilities here. Probably the most immediately obvious definition would be the following:

Definition 2.1. (Dynamic routing.) Under dynamic routing, we say that u is feasible if for every $D \in \mathcal{U}$, the fractional multicommodity flow problem described by D is routable in u. In other words, for every $D \in \mathcal{U}$, we can find an i-j-flow g_{ij} for $\{i,j\} \in \binom{W}{2}$ of value D_{ij}, with $\sum_{i<j} g_{ij}(e) \leq u(e)$ for all $e \in E$.

However, there are some severe issues with this. Suppose we have found a feasible u, and allocate our network accordingly.

- To determine the routing of a particular pair i, j, we need to know precisely what the overall demand pattern D is.
- Even if we know D, determining the routing for all the pairs requires solving a multicommodity flow problem. There may not be enough time to do this.
- And even ignoring this: a small change in our demand pattern could cause a very large change in the required flows.

We'll return to dynamic routing in the last lecture (there are still reasons to study it). But for the next two lectures, we will focus on a much more "stable" way of routing demands in our network. Our solution will not be just a capacity allocation u: it will also be a prescription of how to route demand between any given pair. More precisely, we have the following.

Definition 2.2. Single-path routing (SPR). Under single-path routing, we specify, along with u, an i-j-path P_{ij} for each $\{i,j\} \in \binom{W}{2}$. (We call $\boldsymbol{P} = (P_{ij})_{i<j}$ the **path template**.) Given a demand matrix $D \in \mathcal{U}$, we will route all demand between i and j along P_{ij}; for feasibility, $u(e)$ must exceed the amount routed on edge e, for every edge e and every demand $D \in \mathcal{U}$.

There is also a fractional version of this, where we don't require that demand is routed along a single path, but we insist still on a fixed rule for how to route each demand that is independent of D. Again, we'll return to this in the final lecture. In contrast to dynamic routing, these routing schemes which cannot depend on the given demand are called *oblivious* routing schemes.

We are now ready to formally describe our robust network design problem, using single-path routing.

Input. Network $G = (V, E)$ with costs $c : E \to \mathbb{R}_+$; terminal set $W \subseteq V$; universe $\mathcal{U}$.

Output. Path template $\boldsymbol{P}$ and capacity allocation $u : E \to \mathbb{R}_+$ that is feasible for $\boldsymbol{P}$ and $\mathcal{U}$.

Goal. Minimize the cost $\sum_{e \in E} c(e)u(e)$.

One thing to notice is that it's really the path template $\boldsymbol{P}$ that is most crucial. Once this has been decided, there is a uniquely best choice of u that goes along with it, which is defined by

$$u_{\boldsymbol{P}}(e) := \max_{D \in \mathcal{U}} \sum_{i<j} D_{ij}[P_{ij}](e).$$

So we can think of the problem as being to choose $\boldsymbol{P}$, with a rather complicated objective function $\text{cost}(\boldsymbol{P}) = \sum_{e \in E} c(e)u_{\boldsymbol{P}}(e)$.

It's also worth noting that even to check if a given capacity allocation u is feasible for $\boldsymbol{P}$, we need to be able to optimize linear functions over $\mathcal{U}$. We will always assume this to be the case when talking about general universes, and certainly all the specific universes we will consider have this property.

The choice of the universe $\mathcal{U}$ makes a huge impact on the modelling and algorithmic aspects of the problem. We'll spend some time now investigating various different natural choices for $\mathcal{U}$, and connections to other network design problems. Along the way, we will learn something about the tractability (or not) of the RND problem.

Explicit universes. One fairly obvious way to choose a universe is to explicitly list its elements, by providing a list $D^{(1)}, D^{(2)}, \ldots, D^{(\ell)}$ of demand patterns we anticipate seeing.

We could set $\mathcal{U} = \{D^{(1)}, \ldots, D^{(\ell)}\}$, but it is convenient to assume that $\mathcal{U}$ is always convex, and the following shows that this is possible.

Lemma 2.3. *Let $\mathcal{V} \subseteq \mathbb{R}_+^{\binom{W}{2}}$, and let $\mathcal{U} = \operatorname{conv.hull}(\mathcal{V})$. Then the set of feasible solutions to* $\mathrm{RND_{SPR}}$ *is identical for both universes.*

Proof. This follows from the definition of $u_{\boldsymbol{P}}$ and the fact that maximizing a linear function over a set and its convex hull have the same value. □

So we can take $\mathcal{U} = \operatorname{conv.hull}(\{D^{(1)}, \ldots, D^{(\ell)}\})$; it is an arbitrary polytope, explicitly described by its vertices.

Consider the following choice of a vertex-defined universe, which we call $\mathcal{U}_{\mathrm{ST}}$. Pick an arbitrary terminal $r \in W$ as the "root". For each $i \in W$, let $D^{(i)}$ be the demand pattern requesting a single unit between i and r, with no demand at all between other pairs. Then $\mathcal{U}_{\mathrm{ST}} = \operatorname{conv.hull}(\{D^{(i)} : i \in W\})$.

What is a feasible solution to $\mathrm{RND_{SPR}}$ with this universe? We must choose a path P_{ir} for each $i \in W$ (all other paths in the template have no effect), and we set

$$u_{\boldsymbol{P}}(e) = \max_{D \in \mathcal{U}_{\mathrm{ST}}} \sum_{i \in W} D_{ij}[P_{ir}](e) = \begin{cases} 1 & \text{if } e \in P_{ir} \text{ for some } i \in W \\ 0 & \text{otherwise} \end{cases}.$$

(Here we assume that each P_{ir} is a simple path.)

Corollary 2.4. $\mathrm{RND_{SPR}}$ *is APX-hard.*

Rent-or-buy. Vertex-described universes are very general—any polytope can be so described. But has the issue that the number of potential demand patterns can often be extremely large. It is thus very natural to consider different ways of describing the universe. In particular, we can describe a polytope by its facets instead.

Consider the following choice of universe $\mathcal{U}_{\mathrm{ROB}}$, which generalizes $\mathcal{U}_{\mathrm{ST}}$. As with Steiner tree, fix a root terminal $r \in W$; this time the choice does matter, and we call r the "sink". We introduce an additional integer parameter M. We first give the explicit vertex description of the polytope: for any subset $S \subseteq W$ with $|S| \leq M$, we require that we can route a unit of demand for each $i \in W$ to r *simultaneously*. In other words, the demand $D^{(S)}$ defined by

$$D_{ij}^{(S)} = \begin{cases} 1 & \text{if } i \in S \text{ and } j = r \\ 0 & \text{otherwise} \end{cases},$$

and $\mathcal{U}_{\text{ROB}} = \text{conv. hull}(\{D^{(S)} : S \subseteq W, |S| \leq M\})$.

Now $\mathcal{U}_{\text{ROB}}$ has around $\binom{|W|}{M}$ vertices, which is very large if M is large. But we can describe $\mathcal{U}_{\text{ROB}}$ very efficiently instead as

$$\mathcal{U}_{\text{ROB}} = \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : D_{ij} = 0 \text{ for } j \neq r, \sum_{i \in W} D_{ir} \leq M, D_{ir} \leq 1 \; \forall i \in W \Big\}.$$

RND_{SPR} with this universe is precisely what is known as the *single-sink rent-or-buy problem*, which is described as follows (without reference to the RND framework). Each terminal $i \in W$ has a unit of demand to send to the sink, along a path P_{ir} that must be chosen. Each edge we can choose either to "rent", in which case we pay $c(e)$ for every unit of demand routed along e; or we may "buy" the edge, at a cost of $c(e) \cdot M$. Of course, we should clearly buy precisely if the number $n_{\boldsymbol{P}}(e)$ of paths P_{ir} containing e exceeds e. So given $\boldsymbol{P}$, the cost of our solution is $\sum_{e \in E} c(e) \min\{n_{\boldsymbol{P}}(e), M\}$.

Why exactly does RND_{SPR} with universe $\mathcal{U}_{\text{ROB}}$ model this problem? This is immediate from our facet description of $\mathcal{U}_{\text{ROB}}$: for any $\boldsymbol{P}$, $u_{\boldsymbol{P}}(e)$ is the solution to

$$\begin{aligned} \max \quad & \sum_{i \in W} D_{ir} \cdot [P_{ir}](e) \\ \text{s.t.} \quad & \sum_{i \in W} D_{ir} \leq M \\ & D_{ir} \leq 1 \qquad \forall i \in W \\ & D_{ir} \geq 0 \qquad \forall i \in W, \end{aligned}$$

which clearly has optimal value $\min\{n_{\boldsymbol{P}}(e), M\}$.

Describing polytopes via its facets opens up a much larger class of interesting "well-structured" polytopes. Here is one fairly natural such choice—in fact, the first universe considered (before the general RND model was defined).

$$\mathcal{H}(b) = \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : \sum_j D_{ij} \leq b_i \; \forall i \in W \Big\}. \tag{1}$$

Note that if $b_i = 1$ for all $i \in W$, $\mathcal{H}$ is just the fractional matching polytope on the complete graph on W. So the choice is very natural from a combinatorial optimization point of view as well.

We have seen that RND_{SPR} is certainly APX-hard, but this does not say anything about a specific choice of universe, such as $\mathcal{H}$. Can we solve RND_{SPR} with universe $\mathcal{H}$? This universe and problem has particular importance, and we will return to it in the next lecture. There are also many variants and generalizations of this choice, which we'll discuss a bit in the third lecture.

2.2 Upper and lower bounds to approximability

Let's return to $\mathcal{U}_{\text{ROB}}$, which encodes the single-sink rent-or-buy problem. We can generalize this problem in two ways that yield again well-known network design problems:

- Recall that in rent-or-buy, the cost we pay for an edge e is $c(e) \cdot \min\{n(e), M\}$, where $n_{\boldsymbol{P}}(e)$ is the number of terminals whose path to r uses e. The function $f(z) = \min\{z, M\}$ is of course a concave function. What if we allow *any* concave function? The result is called single-sink buy-at-bulk. Constant factor approximation algorithms are known [29, 24].
- There are also multicommodity versions of rent-or-buy and buy-at-bulk. We have pairs $(s_i, t_i)_{i=1}^k$, and for each $i \in [k]$, must choose an s_i-t_i-path P_i. Other than that, everything is the same: the cost on an edge is proprtional to $g(n_{\boldsymbol{P}}(e))$, where $n_{\boldsymbol{P}}(e)$ is the number of paths using edge e.

  Multicommodity rent-or-buy can still be approximated within constant factors [33]. But multicommodity buy-at-bulk *cannot*. Andrews [2] proved that under appropriate complexity assumptions, no $O(\log^{1/4-\epsilon} k)$-approximation algorithm is possible, for any $\epsilon > 0$.

So if we can find a universe for which RND_{SPR} corresponds to multicommodity buy-at-bulk, we will have shown a polylogarithmic hardness for RND_{SPR} in general.

Let's start with single-sink buy-at-bulk.

Lemma 2.5. *Let $g : \mathbb{R}_+ \to \mathbb{R}_+$ be an increasing concave function. Define*

$$\mathcal{U}_{\text{SSBAB}}(g) := \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : D_{ij} = 0 \text{ if } r \notin \{i, j\}, \sum_{i \in S} D_{ir} \leq g(|S|) \ \forall S \subseteq W \Big\}.$$

Then for any solution template $\boldsymbol{P}$, $u_{\boldsymbol{P}}(e) = g(n_{\boldsymbol{P}}(e))$.

Proof. Consider any $e \in E$. Let $X = \{i \in W : e \in P_{ij}\}$. Clearly

$$u_{\boldsymbol{P}}(e) = \max_{D \in \mathcal{U}_{\text{SSBAB}}(g)} \sum i \in X D_{ir} \leq g(|X|) = g(n_{\boldsymbol{P}}(e)).$$

Conversely, take $D_{ir} = \frac{g(|X|)}{|X|}$ for all $i \in X$, and $D_{ij} = 0$ for all other pairs. Then for any $S \subseteq W$,

$$\sum_{i \in S} D_{ir} = |S \cap X| \frac{g(|X|)}{|X|} \leq g(|S \cap X|)$$

by concavity. So $D \in \mathcal{U}_{\text{SSBAB}}(g)$, and hence $u_{\boldsymbol{P}}(e) \geq g(n_{\boldsymbol{P}}(e))$. □

Extending this to multiple commodities is not too difficult.

Lemma 2.6. *Let $g : \mathbb{R}_+ \to \mathbb{R}_+$ be an increasing concave function. Suppose $|W| = 2k$ is even; we think of pair $\{i, i+k\}$ as the source and sink of commodity $i \in [k]$. Define*

$$\mathcal{U}_{\text{BAB}}(g) := \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : D_{ij} = 0 \text{ if } |j - i| \neq k, \sum_{i \in S} D_{i,i+k} \leq g(|S|) \ \forall S \subseteq [k] \Big\}.$$

Then for any solution template $\boldsymbol{P} = (P_{i,i+k})_{i \in [k]}$, $u_{\boldsymbol{P}}(e) = g(n_{\boldsymbol{P}}(e))$.

Corollary 2.7 ([43])**.** $\mathrm{RND_{SPR}}$ *with universe* $\mathcal{U}_{\mathrm{BAB}}(g)$ *is (for an appropriate choice of* g*) hard to approximate within* $O(\log^{1/4-\epsilon}|W|)$ *for any* $\epsilon > 0$*.*

On the positive side, we have the following.

Proposition 2.8. *Let* $\mathcal{U}$ *be any universe over which we can efficiently optimize linear functions. Then we can compute, in polynomial time, an* $O(\log n)$*-approximation to* $\mathrm{OPT_{SPR}}$*.*

Proof. The proof is an easy application of a "big hammer": the FRT tree embedding theorem. We refer to [48] for more—we will explain it only briefly here.

The first observation is that while we have described RND in terms of graphs, it can equally well be formulated in terms of metrics. Given an instance G with costs c, replace (G, c) with (K_V, SP_c), i.e., the complete graph on V, with the cost of edge ij being given by the shortest-path length in G with respect to c. It is not hard to see that the resulting instance is equivalent to the new one: an edge $ij \in K_V$ represents some shortest i-j-path in G. A solution template for (G, c) can be mapped to a solution template for (K_V, SP_c) without increasing the cost, and vice versa. Moreover, the cost of a given solution $\boldsymbol{P}$ for (K_v, SP_c) depends linearly on the distances in the metric SP_c.

Suppose $T = (V, F)$ is a spanning tree of K_V, with edge costs $c_T : F \to \mathbb{R}_+$. This naturally induces a metric d_T, with $d_T(i, j)$ equal to the cost of the i-j-path in T.

Theorem 2.9 ([19])**.** *Let* $d : V \times V \to \mathbb{R}_+$ *be any finite metric space, with* $|V| = n$*. Then there exists a random spanning tree* T *of* K_V *such that for all* $i, j \in V$*,*

(i) $d_T(i, j) \geq d(i, j)$ *surely; and*

(ii) $\mathbb{E} d_T(i, j) \leq O(\log n) d(i, j)$*.*

Moreover, the support of the distribution of the random variable T *has size polynomial in* $|V|$ *and can be efficiently computed.*

Given this theorem, we can reduce our problem to solving $\mathrm{RND_{SPR}}$ on trees, losing $O(\log n)$. Simply take the random tree according to the distribution guaranteed by the theorem; solve the problem on this tree T, resulting in a path template $\boldsymbol{P}^{(\boldsymbol{T})}$; and then map this back to a path template $\boldsymbol{P}$ on the original instance, replacing each edge with a corresponding shortest path in G. Property (ii) ensures that the cost of the optimal solution on T is in expectation at most a factor $O(\log n)$ more expensive than OPT; and property (i) ensures that when we take a solution for T and move it back to G, the cost can only decrease.

But $\mathrm{RND_{SPR}}$ on a tree is trivial! There is a unique simple path between each pair of terminals, so the optimal choice for $\boldsymbol{P}$ is immediate, and all we need to do is be able to compute $u_{\boldsymbol{P}}(e)$. Since we are assuming we can optimize linear functions over $\mathcal{U}$, we are done. □

Open problem 1. There is a gap between the upper bound of $O(\log n)$ and the lower bound of $\Omega(\log^{1/4-\epsilon}(n))$ for the approximability of RND. The inapproximability result for buy-at-bulk is rather complicated, but perhaps a different choice of $\mathcal{U}$ provides lower bounds more easily.

2.3 Single-sink RND

A number of the universes we described have a "single sink" feature: which we can formalize as follows.

Definition 2.10. A universe $\mathcal{U}$ is called **single-sink** if there is a terminal $r \in W$, called the sink, such that for any $D \in \mathcal{U}$, $D_{ij} = 0$ if $r \notin \{i, j\}$.

Recall that single-sink buy-at-bulk (as well as the special cases single-sink rent-or-buy and Steiner tree) have constant-factor approximation algorithms. The example showing polylog hardness for RND in general was the universe corresponding to multicommodity buy-at-bulk, which is certainly not single-sink. So it seems worth exploring single-sink universes in more detail.

Let's start with the explicit model. Here is one special case that may help to see why this is interesting. We are given sets $R_1, R_2, \ldots, R_\ell \subseteq W$. Let $\mathcal{U} = \{D^{(1)}, \ldots, D^{(\ell)}\}$ where $D^{(i)}_{jr} = [j \in R_i]$. So this means that for each $i \in [\ell]$, we must be able to send 1 unit from each $j \in R_i$ simultaneously. Compare this with group Steiner tree: we just need to be able to send 1 unit to any single node in each group.

A different way to think of single-sink RND is as follows [35]. Assume $\mathcal{U}$ is a polytope, and let $v^{(1)}, \ldots, v^{(\ell)}$ denote its vertices. (Let's not assume that the polytope is described explicitly, so this set may be very large.) Now define the set function $h : 2^W \to \mathbb{R}_+$ by

$$h(S) = \max_{j \in [\ell]} v^{(j)}(S) \qquad \forall S \subseteq W.$$

Then it's clear that $u_{\boldsymbol{P}}(e) = h(\{i \in W : e \in P_{ir}\})$. Such a function h is called **fractionally subadditive** or **XOS**. By adjusting $\mathcal{U}$, we can obtain any increasing fractionally subadditive function.

This clearly generalizes single-sink buy-at-bulk for a concave function g: take $h(S) := g(|S|)$. Also, submodular functions are fractionally subadditive: if we restrict h to being submodular, we obtain what one would naturally call "single-sink submodular network design".

This problem is very naturally, and variations of this have been suggested without any connection to network design [36, 4]. For example, consider the following problem [4]. We have various different pieces of data, indexed by $[\ell]$. Each piece of data is the same size. Each terminal is interested in a given subset $Z_i \subseteq [\ell]$ of the pieces of data. All the data is at the root, and we should send each terminal its desired pieces of data, at minimum cost. Crucially, we only need to send a given piece of data once over a given edge, no matter how many terminals eventually receive it.

This corresponds to single-sink submodular network design, with h given by a coverage function (and hence submodular):

$$h(S) = \Big|\bigcup_{i \in S} Z_i\Big|.$$

In [4], they show that some particular special cases are $O(1)$-approximable (under a somewhat different routing scheme than SPR).

Open problem 2. What is the approximability of single-sink RND? This is open, and also if we restrict to the explicit model, or to universes described by submodular functions, or by coverage functions. Nothing better than the $O(\log n)$ approximability coming from metric embedding is known.

Exercise 1. Suppose $\mathcal{U} = \{D^{(1)}, \ldots, D^{(k)}\}$, where each $D^{(k)}_{ir}$ is nonzero only for $i = k$. Provide a constant-factor approximation algorithm in this case. (Hint: it's not hard to see that one can restrict to the case where $D^{(k)}_{kr}$ is a power of two, losing only a constant factor.)

2.4 Modelling random demands

Many of the specific universes we have discussed (though not the hose universe) have been motivated by theoretical reasons, rather than being of specific interest in applications. So let's end with one that has a more applied feel.

Given that we are motivated by uncertain and/or varying demand patterns, it is natural to ask: what if we have an explicit stochastic model?

Imagine that the demand matrix D is drawn from a distribution μ. We can then ask for a solution (template $\boldsymbol{P}$ and capacity allocation u) with the property that

$$\mathbb{P}_{D\sim\mu}(u \text{ is feasible for routing } D \text{ according to } \boldsymbol{P}) \geq 1 - \epsilon, \tag{2}$$

where ϵ is some acceptable failure probability.

This falls within the purview of stochastic optimization—more precisely, this is a chance-constrained optimization problem. It does not fall into the RND framework. However, there is a way of using RND to obtain an approximation. This idea is more widely applicable, and has been heavily discussed in the general setting of robust optimization. The idea is to pick as the universe some subset $\mathcal{V}$ such that $\mathbb{P}_{D\sim\mu}(D \in \mathcal{V}) \geq 1-\epsilon$. Clearly, a solution to the robust problem with this universe will satisfy (2). It certainly need not be optimal (note that there are many possible choices for $\mathcal{V}$). Such a robust problem is called a *safe tractable approximation* [7, 6] to the chance-constrained problem.

Suppose that the demand D_{ij} between a given pair i, j is distributed uniformly on $[a_{ij}, a_{ij} + s_{ij}]$, and is independent for different pairs. Let $\hat{D}_{ij} = (D_{ij} - a_{ij})/s_{ij}$ for each i, j. Let $m := \left|\binom{W}{2}\right|$. Then an application of Hoeffding's inequality tells us that

$$\mathbb{P}\Big(\sum_{i<j} \big(\hat{D}_{ij} - \tfrac{1}{2}\big) > \tfrac{1}{2}\sqrt{m \log(1/\epsilon)}\Big) \leq \epsilon.$$

Thus we can choose

$$\mathcal{V} = \Big\{D \in \mathbb{R}_+^{\binom{W}{2}} : a_{ij} \leq D_{ij} \leq a_{ij} + s_{ij}\ \forall \{i,j\} \in \tbinom{W}{2}, \sum_{i<j} (D_{ij} - a_{ij})/s_{ij} \leq \tfrac{m}{2} + \tfrac{\sqrt{m}}{2}\sqrt{\log(1/\epsilon)}\Big\}.$$

Note that this is a polyhedron. Solving RND using the universe $\mathcal{V}$ will yield a solution that is feasible for the chance-constrained problem—we won't violate our capacity with probability more than ϵ.

But we can do a lot better. Hoeffding also tells us that

$$\mathbb{P}\Big(\sum_{i<j}(\hat{D}_{ij}^2 - \tfrac{1}{3}) > \tfrac{1}{2}\sqrt{m\log(1/\epsilon)}\Big) \le \epsilon,$$

leading to the alternative choice

$$\mathcal{V}' = \Big\{D \in \mathbb{R}_+^{\binom{W}{2}} : a_{ij} \le D_{ij} \le a_{ij} + s_{ij}\ \forall\{i,j\} \in \tbinom{W}{2}, \sum_{i<j}(D_{ij} - a_{ij})^2/s_{ij}^2 \le \tfrac{m}{3} + \tfrac{1}{2}\sqrt{m\log(1/\epsilon)}\Big\}.$$

This is not a polyhedron, rather it is the intersection of an ellipsoid with a box. Nonetheless, optimizing over this set is (relatively) tractable, being a convex quadratic optimization problem. It is not hard to see that $\mathcal{V}' \subset \mathcal{V}$, and in fact $\mathcal{V}'$ is *much* smaller if $|W|$ is large. (Look in a diagonal direction: $\hat{D}_{ij} = \alpha$ for a fixed α; for $\mathcal{V}$, α can be as large as a constant, whereas for $\mathcal{V}'$, $\alpha = O(1/\sqrt{m})$.) So solving RND using the universe $\mathcal{V}'$ will likely lead to better results than $\mathcal{V}$.

$\mathcal{V}'$ is an example of an *ellipsoidal uncertainty set*; these have been studied a lot as tools for tackling chance-constrained problems [7, 9].

2.5 Some further remarks

Asymmetric demands. We choose to consider demands to be undirected, so that D_{ij} and D_{ji} refer to the same demand. Allowing asymmetric demands—thinking of demand from i to j as being different from demand from j to i—turns out to not be a substantial generalization, though.

Let's see why. Suppose we have a universe $\mathcal{U} \subseteq \mathbb{R}_+^{W\times W}$, where demand matrices are not necessarily symmetric. Let's allow also P_{ij} and P_{ji} to differ. Now define $\bar{W} = W \cup W'$, where $W' = \{1', 2', \ldots, |W|'\}$. We add W' to the instance by adding i' at the same location (i.e., connected by a 0-cost edge) to i. We now define a new universe

$$\bar{\mathcal{U}} = \Big\{\bar{D} \in \mathbb{R}_+^{\binom{\bar{W}}{2}} : \bar{D}_{ij'} = D_{ij} \text{ and } \bar{D}_{i'j} = D_{ji} \text{ for all } i < j \in W, \text{ for some } D \in \mathcal{U}\Big\}.$$

We also define an associated template $\bar{\boldsymbol{P}}$ by $\bar{P}_{ij'} = P_{ij}$ and $\bar{P}_{i'j} = P_{ji}$ for all $i < j$. It is then easy to see that the new symmetric universe is completely equivalent to the asymmetric universe $\mathcal{U}$.

If we consider asymmetric demands and also restrict $P_{ij} = P_{ji}$, things are no longer quite identical. But it doesn't seem like an interesting variant—and quite likely, in most situations there will always be an optimal solution with $P_{ij} = P_{ji}$.

Directed graphs. We will restrict ourselves only to undirected graphs. One could certainly define a model for directed graphs (and asymmetric demands). Not much is known here. Since metric embedding techniques no longer apply, it's unlikely that much positive can be said about general universes. It's at least as hard as directed Steiner tree, but probably much harder.

One may still hope for positive results for particularly nice universes. What about $\mathcal{H}$? It's certainly APX-hard, but I'm not sure if an interesting approximation ratio is possible.

Notes

The RND model with general universes was introduced by Ben-Ameur and Kerivin [5]. Quite a bit earlier, the VPN problem (i.e., RND with the hose universe) was introduced by Fingerhut [20], and later and independently, Duffield et al. [14].

Chekuri [12] gives a nice survey of algorithmic aspects of RND, though it is fairly out of date. A lot of the material in these lectures is also discussed in my thesis [42].

Robust optimization in general is a huge topic; see the book by Ben-Tal, El Ghaoui and Nemirovski [6].

There is a large literature on the various network design problems mentioned: Steiner tree, rent-or-buy and buy-at-bulk, e.g., [11, 33, 32, 22].

Proposition 2.8 was noticed by a few people, but I first heard it from Anupam Gupta.

Oblivious routing is very important in the context of minimizing congestion as well. A breakthrough result of Räcke [45] says that one can find a (fractional) oblivious routing whose congestion is only a factor $O(\log n)$ worse than the minimum congestion routing, for any demand.

The case where the cost on an edge is not linear in the required capacity but concave in the capacity—corresponding to "economies of scale" has also been considered [21, 46].

Some papers [44, 35] consider a version of single-sink network design where the routing scheme is dynamic, but capacity allocation must be integral. One reason to consider this is that the ILP formulation for the problem is a very natural cut formulation: for every cut S with $r \notin S$, the capacity of the cut should be at least $f(W \cap S)$. This is NP-hard even if the universe is explicit and described by only two traffic matrices [35], and is approximable within a constant factor on a ring network [44].

3 The VPN Theorem

In this section, we focus solely on the hose universe—recall (1). Throughout, we assume an instance $(G = (V, E), W, c, \mathcal{H}(b))$ of the RND problem with universe $\mathcal{H}(b)$ and with SPR routing is given; we call this the *VPN problem.*

We will always assume that $b_i \leq \frac{1}{2} \sum_{j \in W} b_j$ for each $i \in W$. If not, the hose constraint for i can never be tight, and modifying b_i to be equal to $\sum_{j \neq i} b_j$ instead has no effect on the universe.

Hubbed solutions. There is a special kind of routing template that will be of particular interest. Let $r \in V$ be given, as well as an i-r-path R_i for each $i \in W$. We call $\boldsymbol{R} = (R_i : i \in W)$ an **r-hubbing** (or just a hubbing if we wish to leave r implicit). The routing template associated with $\boldsymbol{R}$, which we denote routing($\boldsymbol{R}$), is defined by routing i to r via R_i, and then from r to j via R_j. We call the resulting solution an **r-hubbed solution**. Note that the resulting path may not be simple. We will refer to the minimal capacity vector associated with routing($\boldsymbol{R}$) more directly as $u_{\boldsymbol{R}}$: it has a very simple description.

Lemma 3.1.

$$u_{\boldsymbol{R}}(e) = \sum_{i \in W} b_i \cdot [R_i](e).$$

Note that for a fixed r, the cheapest r-hubbed solution is straightforward to compute. We have, for any r-hubbing $\boldsymbol{R}$,

$$\text{cost}(\boldsymbol{R}) := \text{cost}(\text{routing}(\boldsymbol{R})) = \sum_{e\in E} c(e)u_{\boldsymbol{R}}(e) = \sum_{i\in W} \text{cost}(R_i).$$

Thus if we choose R_i' to be a shortest (with respect to c) i-r-path, we have $\text{cost}(\boldsymbol{R'}) \leq \text{cost}(\boldsymbol{R})$.

Theorem 3.2 ([20, 30])**.** *There exists an $r \in W$ such that the cheapest r-hubbed solution costs at most $2(1-\frac{b_{\min}}{B})\text{OPT}$, where $b_{\min} = \min_{i\in W} b_i$ and $B = \sum_{i\in W} b_i$.*

Proof. Let $\alpha = (1-\frac{b_{\min}}{B})^{-1}$. Consider the single demand matrix $\bar{D}$ defined by

$$\bar{D}_{ij} = \begin{cases} \alpha\frac{b_ib_j}{B} & \text{if } i \neq j \\ 0 & \text{otherwise} \end{cases}.$$

We have that for any $i \in W$,

$$\sum_{j\in W} \bar{D}_{ij} = \alpha \cdot \frac{b_i}{B}\sum_{j\neq i} b_j = \alpha b_i \frac{B-b_i}{B} \leq b_i. \tag{3}$$

Hence $\bar{D} \in \mathcal{U}_{\text{TREE}}(b)$, and so $\text{OPT} \geq \text{cost}(\bar{D})$, the cost to optimally route just the single demand matrix $\bar{D}$. So we have

$$\text{OPT} \geq \sum_{i<j} \text{SP}_c(i,j)\bar{D}_{ij} = \tfrac{\alpha}{2}\sum_{i\in W}\sum_{j\in W} \text{SP}_c(i,j)\frac{b_ib_j}{B}. \tag{4}$$

For $j \in W$, let $\boldsymbol{R}^{(j)}$ be a cheapest j-hubbing. Then $\text{cost}(\boldsymbol{R}^{(j)}) \leq \sum_{i\in W} b_i\text{SP}_c(i,j)$, by using shortest paths. Now consider the following weighted average:

$$\begin{aligned} \frac{1}{B}\sum_{j\in W} b_j \,\text{cost}(\boldsymbol{R}^{(j)}) &\leq \frac{1}{B}\sum_{j\in W} b_j \sum_{i\in W} b_i\text{SP}_c(i,j) \\ &\leq 2\alpha \cdot \text{OPT} \qquad\qquad \text{by (4)}. \end{aligned}$$

Since the cheapest of the $\boldsymbol{R}^{(j)}$'s costs at most this average, the result follows. ☐

Exercise 2. Show that this result is tight in general (when restricting to a hub in W).

But could it be that the cheapest hubbed solution is always *optimal*? This became known as the *VPN Conjecture.*

VPN Theorem ([27])**.** There exists an optimal solution to the VPN problem that is a hubbed solution.

Note that this immediately implies that the VPN problem is polynomially solvable: simply consider each possible choice of $r \in W$, determine the cost $\sum_{i\in W} \text{SP}_c(i,r)$ of an optimal r-hubbed solution, and choose the cheapest.

We will actually prove a slightly different, though equivalent, statement.

VPN Theorem, packing form ([27])**.** Let $\boldsymbol{P}$ be any solution template. Then there exists a *random* hubbing $\boldsymbol{R}$ such that $\mathbb{E}[u_{\boldsymbol{R}}] \le u_{\boldsymbol{P}}$.

This immediately implies that the expected cost of the random hubbed solution is not larger than the cost of the provided solution, and hence the cost version of the VPN theorem.

Remark. The first part of the proof described here is a little different to the published one. The original proof proceeds by proving the "Pyramidal Routing Conjecture" introduced by Grandoni et al. [28], who also show that this Pyramidal Routing Conjecture is equivalent to the VPN Conjecture.

The dual perspective. The following dual perspective is very useful (and we'll see it again in the next section). Given a routing template $\boldsymbol{P}$, we have (unpacking the definition of $\mathcal{H}(b)$)

$$\begin{aligned} u_{\boldsymbol{P}}(e) = \max &\sum_{i<j} D_{ij} \cdot [P_{ij}](e) \\ \text{s.t.} \qquad &\sum_{j \in W} D_{ij} \le b_i \qquad \forall i \in W \\ &D_{ij} \ge 0 \qquad \forall \{i,j\} \in \tbinom{W}{2}. \end{aligned}$$

By strong LP duality, we obtain

$$\begin{aligned} u_{\boldsymbol{P}}(e) = \min &\sum_{i \in W} b_i y_i(e) \\ \text{s.t.} \qquad & y_i(e) + y_j(e) \ge [P_{ij}](e) \qquad \forall \{i,j\} \in \tbinom{W}{2} \\ & y_i \ge 0 \qquad \forall i \in W. \end{aligned} \tag{5}$$

This provides us with a very nice way of describing solutions to the VPN problem. Instead of describing it via a routing template $\boldsymbol{P}$ and associated capacity vector $u_{\boldsymbol{P}}$, we describe it via the dual solution $\boldsymbol{y} := (y_i : i \in W)$. The capacity vector associated with $\boldsymbol{y}$ is then simply

$$u_{\boldsymbol{y}} := \sum_{i \in W} b_i y_i,$$

leading to an associated cost

$$\text{cost}(\boldsymbol{y}) := \sum_{i \in W} b_i \,\text{cost}(y_i) = \sum_{e \in E} c(e) \sum_{i \in W} b_i y_i(e).$$

So given a dual solution $\boldsymbol{y}$, we can see its cost directly—no need to solve a fractional matching problem for each edge. We can think of y_i as being capacity "purchased" by i. The only thing we have to concern ourselves with is *feasibility* of our dual solution: $y_i(e) + y_j(e) \ge [P_{ij}](e)$ for all $\{i,j\} \in \binom{W}{2}$ and $e \in E$. But we don't need to fix $\boldsymbol{P}$, we can state feasibility "for some $\boldsymbol{P}$" as simply:

for every pair $\{i,j\} \in \binom{W}{2}$, $\{e \in E : y_i + y_j \ge 1\}$ contains an i-j-path.

Given a feasible dual solution $\boldsymbol{y}$, we can define (non-uniquely) routing($\boldsymbol{y}$) to be any path template satisfying the above.

Now $\boldsymbol{y}$ can certainly be fractional; we can assume it is half-integral though, since it is well-known that there is always a half-integral optimum solution to (5). It is quite helpful to consider integral solutions however, at least for intuition. In that case, we can write the dual solution instead as a collection of sets $\boldsymbol{N} := (N_i : i \in W)$, where $N_i = \{e \in E : y_i = 1\}$. Feasibility becomes even simpler: $N_i \cup N_j$ must contain an i-j-path for each $\{i, j\} \in \binom{W}{2}$. Somewhat whimsically, we will call integral feasible dual solution *weeds*. (An attempt will be made later to justify this name, but it should not be taken very seriously.) So in what follows, we will often start by just assuming that we are given a weed solution $\boldsymbol{N}$; afterwards, we will check that all arguments do extend to arbitrary dual solutions.

Reducing to $b = 1$. Mostly for notational simplicity, it is convenient to observe that it suffices to prove the theorem for the case $b_i = 1$ for all $i \in W$. To see this, first observe that we can assume $b_i \in \mathbb{N}$, by scaling b up by a large enough factor. Next, we can replace a terminal with $b_i \in \{2, 3, \dots, \}$ with b_i distinct terminals, each connected to the site of the original terminal by edges of cost 0. So let $W' = \{(i, r) : i \in W, r \in [b_i]\}$, and set $b_{(i,r)} = 1$ for all $(i, r) \in W'$.

Given any dual solution $\boldsymbol{y}$ to the old instance, we can get a solution $\boldsymbol{y'}$ to the new instance of equal cost by defining $y'_{(i,r)} = y_i$ for all $i \in W, r \in [b_i]$. And given a solution $\boldsymbol{y'}$ to the new instance, we can get a solution $\boldsymbol{y}$ to the old instance of no larger cost by define $y_i = y'_{(i,r^*)}$, where $r^* = \arg\min_r \operatorname{cost}(y'_{i,r})$. So the two instances are equivalent.

Vine solutions. We have introduced one kind of special solution, namely hubbed solutions. Now we will introduce another. Let $\tau \subseteq V$ be a subset of the nodes of odd cardinality. We define a **τ-vine** to be a vector $\boldsymbol{M} = (M_i : i \in W)$, where M_i is a $(\tau \triangle \{i\})$-join for each $i \in W$. (Recall that for any set $T \subseteq V$ of even cardinality, a T-join is a subset $F \subseteq E$ such that the odd-degree nodes of (V, F) are precisely T.)

We note that any vine $\boldsymbol{M}$ (no matter what τ is) is a weed: for any $\{i, j\} \in \binom{W}{2}$, $M_i \cup M_j \supseteq M_i \triangle M_j$; and $M_i \triangle M_j$ is an $\{i, j\}$-join. The motivation for this definition will not be immediately clear, but we can observe that vines are a special class of weeds that are somehow a little better behaved. Having fixed τ, we can check for each $i \in W$ separately that it is a $(\tau \triangle \{i\})$-join, and after that, we know that we have a valid weed solution—we don't need to check a property for every $\{i, j\} \in \binom{W}{2}$.

A second thing we can observe about vines is that not only are they a special class of weeds, but hubbing solutions are a special class of vines. An r-hubbing $\boldsymbol{R}$ is exactly an $\{r\}$-vine; hubbing soluions are exactly τ-vines for τ restricted to be a singleton.

The remainder of the proof of Theorem 3 will split cleanly into two parts.

Part 1. Given any dual solution $\boldsymbol{y}$, we show that there is a *random* vine solution $\boldsymbol{M}$ such that $\mathbb{E}[u_{\boldsymbol{M}}] \leq u_{\boldsymbol{y}}$.

Part 2. Given a τ-vine $\boldsymbol{M}$ (for any choice of τ), there exists an $r \in V$ and an $\{r\}$-vine $\boldsymbol{R}$ such that $u_{\boldsymbol{R}} \leq u_{\boldsymbol{M}}$. (In this part, no randomization is necessary.)

Proof of Part 1.

Let's begin by assuming our starting dual solution is integral, and described by the weed $\boldsymbol{N}$.

Our weed $\boldsymbol{N}$ has unfortunately little structure; it could be rather overgrown. We'll need to hack it up a bit to uncover our more nicely structured vine... In particular, there is no τ in sight; $M_i \cup M_j$ could look like anything (as long as it connects i and j), and i and j might not be connected in $M_i \triangle M_j$.

Theorem 3.3. *Let $\boldsymbol{N}$ be a weed. Then there exists a random vine $\boldsymbol{M}$ so that $\mathbb{E}\ \sum_i [M_i] \leq \sum_{i \in W}[N_i]$.*

Proof. Let $\tau^{(i)} = \text{odd-set}(N_i) \triangle \{i\}$, where odd-set$(N_i)$ denotes the set of nodes of odd degree in (V, N_i). We define $M_j^{(i)}$ by

$$M_j^{(i)} = N_i \triangle P_{ij},$$

where $P_{ij} \subseteq N_i \cup N_j$ for all $\{i,j\} \in \binom{W}{2}$. Then $\boldsymbol{M^{(i)}}$ is a $\tau^{(i)}$-vine:

$$\text{odd-set}(M_j^{(i)}) = \text{odd-set}(N_i) \triangle \text{odd-set}(P_{ij}) = \tau^{(i)} \triangle \{j\}.$$

Moreover, for any pair $\{i,j\} \in \binom{W}{2}$,

$$[M_j^{(i)}] + [M_i^{(j)}] = [N_i \triangle P_{ij}] + [N_j \triangle P_{ij}] \leq [N_i] + [N_j],$$

using the fact that $P_{ij} \subseteq N_i \cup N_j$. Thus

$$\sum_{i \in W} \sum_{j \in W} [M_j^{(i)}] \leq \sum_{i \in W} \sum_{j \in W} [N_i] = |W| \cdot \sum_{i \in W} [N_i],$$

and the result follows. □

Non-integral solutions.

Theorem 3.4. *Let $\boldsymbol{y}$ be any dual feasible solution. Then there exists a random vine $\boldsymbol{M}$ with $\mathbb{E} \sum_{i \in W} [M_i] \leq \sum_{i \in W} y_i$.*

Proof. For any $\gamma \in [0,1]$ and $i, j \in W$, define

$$M_j^{(i,\gamma)} = \{e : y_i(e) \geq \gamma\} \triangle P_{ij}.$$

Define the random vine $\boldsymbol{M}$ by: $M_j = M_j^{(i,\gamma)}$ for all $j \in W$, where γ is chosen uniformly from $[0,1]$ and i uniformly from W. It is staightforward to confirm that this is indeed a vine, using a similar argument to the integral case. We claim this satisfies the requirements.

Note that $M_j^{(i,\gamma)} \cup M_i^{(j,1-\gamma)}$ contains P_{ij}, by feasibility of y. Thus

$$[M_j^{(i,\gamma)}] + [M_i^{(j,1-\gamma)}] \leq [\{e : y_i \geq \gamma\}] + [\{e : y_j \geq 1 - \gamma\}].$$

Thus

$$\tfrac{1}{|W|}\sum_{i\in W}\sum_{j\in W}\int_0^1 [M_j^{(i,\gamma)}]d\gamma \le \tfrac{1}{|W|}\sum_{i\in W}\sum_{j\in W}[\{e : y_i \ge \gamma\}] = \sum_{i\in W} y_i,$$

and the result follows. □

Remark. As mentionted, this part of the proof differs from the one described in [27]. It replaces precisely the notion of Pyramidal Routing of Grandoni et al. [28]. It is not a truly different proof however; there is a 1-1 mapping between vine solutions and solutions to the Pyramidal Routing problem, and this argument was entirely inspired by the original one. The interested reader will have little difficulty making the correspondence after reading the appropriate notions in [27]. Nonetheless, the description here is (in hindsight) arguably cleaner and more natural; at least to me, this approach reduces some of the mystery present in the old proof.

Proof of Part 2.

The following also differs somewhat from the proof in the paper. This version came out of some discussions with Michel Goemans.

Proposition 3.5. *Let $\boldsymbol{V}$ be a τ-vine, for some odd-cardinality $\tau \subseteq V$. Then there exists an $r \in \tau$ and an r-hubbing $\boldsymbol{R}$ with $u_{\boldsymbol{R}} \le u_{\boldsymbol{V}}$.*

Proof. Let $F = \dot{\bigcup}_{i\in W} M_i$ be the disjoint union of the M_i's. We wish to find a node $r \in \tau$ so that there are edge-disjoint paths from the terminals to r in F.

A certificate that we cannot use a particular node $v \in \tau$ as the hub is that there is some set S so that $|\delta_F(S)| < |W \setminus S|$. Call such a set a *bad* set. So if there is no good choice of r, then this means that every node in τ is contained in some bad set. Or in other words: our goal is to show that $\tau \setminus X \neq \emptyset$, where X is the set of nodes contained in a bad set.

We now note two important properties of bad sets.

Lemma 3.6. *If S is a bad set, then $|S \cap \tau|$ is even.*

Proof. If $|S \cap \tau|$ is odd, then for every $i \notin S$, $|S \cap \tau \triangle \{i\}|$ is also odd, and hence M_i must cross S. Thus $|\delta_F(S)| \ge |W \setminus S|$, and so S is not bad. □

Lemma 3.7. *If S_1 and S_2 are bad sets, then at least one of $S_1 \setminus S_2$ and $S_2 \setminus S_1$ are also bad.*

Proof. Define the slack $\mathrm{sl}(S)$ of any set to be just $|\delta_F(S) - |W \setminus S|$; so bad sets are precisely those with negative slack.

We will need the standard fact that the cut function is "posimodular": for any $X_1, X_2 \subseteq V$,

$$|\delta_F(X_1)| + |\delta_F(X_2)| \ge |\delta_F(X_1 \setminus X_2)| + |\delta_F(X_2 \setminus X_1)|.$$

This can be confirmed by enumerating the various ways an edge can contribute to both sides of this inequality.

Define the slack $\mathrm{sl}(S)$ of any set to be just $|\delta_F(S) - |W \setminus S|$; so bad sets are precisely those with negative slack. Then since $S \to |W \setminus S|$ is modular, $\mathrm{sl}(\cdot)$ is also posimodular. Combined with the fact that S_1 and S_2 are both bad,

$$\mathrm{sl}(S_1 \setminus S_2) + \mathrm{sl}(S_2 \setminus S_1) \leq \mathrm{sl}(S_1) + \mathrm{sl}(S_2) < 0;$$

the lemma follows. □

It follows that we can find a partition $\mathcal{P}$ of X into bad sets. But $|S \cap \tau|$ is even for each $S \in \mathcal{P}$, by Lemma 3.6, and so $|X \cap \tau|$ is even too. But $|\tau|$ is odd, and so $\tau \setminus X$ is indeed nonempty. □

Notes

Several researchers [1, 18, 31, 39] noticed at around the same time that the cheapest shortest path tree solution seemed to always be optimal, and not just within a factor of 2. This prompted them to independently formulate the VPN Conjecture. Important progress was made by Hurkens, Keijsper and Stougie [38], who proved that the result is true if G is a cycle (and some other cases too). Grandoni et al. [28] later gave a much simplified proof on ring networks; their approach was crucial to the final resolution of the VPN Conjecture [27].

Part 2 of the original proof [25] was simplified by András Sebő using Gomory-Hu trees, and this version can be found in the journal version [27]. The proof of part 2 given here is not much simpler, but is slightly more elementary, avoiding the machinery of Gomory-Hu trees.

4 Comparing routing schemes

So far, we have focused exclusively on single-path routing. It is the most combinatorially interesting, has the closest connection to other fundamental network design problems, and is well-motivated by applications in communications networks. But other routing schemes are interesting, and it is of interest to compare them and understand how crucial the routing scheme might be. Here we discuss some results and comparisons, as well as open problems.

4.1 Multipath routing

Multipath routing keeps the "fixed routing" aspect of SPR, but has the fractional aspect of dynamic routing. Instead of specifying a path P_{ij} for each $\{i, j\} \in \binom{W}{2}$, we specify a unit i-j-flow f_{ij}. Note that this is a unit flow—and unlike dynamic routing, it *cannot* depend on the choice of D. Given a $D \in \mathcal{U}$, $f_{ij}(e)$ denotes the *fraction* of the overall demand D_{ij} between i and j that uses edge e. For a capacity allocation u to be feasible, it must satisfy

$$u(e) \geq \max_{D \in \mathcal{U}} \sum_{i<j} D_{ij} f_{ij}(e) \qquad \forall e \in E.$$

We denote by $\mathrm{RND}_{\mathrm{MPR}}$ the problem of finding a cheapest solution to a given RND problem under multipath routing.

Does the fractional nature of the problem make it easier? In fact yes: here we see that we can efficiently solve $\mathrm{RND}_{\mathrm{MPR}}$ efficiently, under rather mild assumptions on $\mathcal{U}$. The theorem can be seen as a special case of more general results in robust optimization [8].

Theorem 4.1. *As long as we can optimize linear functions over $\mathcal{U}$ efficiently,* $\mathrm{RND}_{\mathrm{MPR}}$ *is efficiently solvable as well.*

Proof. Consider the $\mathrm{RND}_{\mathrm{MPR}}$ problem, in the following form:

$$\begin{aligned} \min \quad & \sum_{e \in E} c(e) u(e) \\ \text{s.t.} \quad & u(e) \geq \max_{D \in \mathcal{U}} \sum_{i<j} D_{ij} f_{ij}(e) \qquad \forall e \in E \\ & f_{ij} \text{ is a unit } i\text{-}j\text{-flow} \qquad \forall \{i,j\} \in \tbinom{W}{2}. \end{aligned} \tag{6}$$

We note that this is a convex program, since the right hand side of (6), being a maximum over linear functions, is convex.

By the equivalence of optimization and separation, it suffices to determine either that a given solution $(\boldsymbol{f}, u)$ is feasible for this convex program, or otherwise find a violated constraint. The flow constraints are easy to check. For a given $e \in E$, checking (6) requires optimizing a linear function over $\mathcal{U}$, which we assumed can be done. □

The above requires the ellipsoid method, so it is not fast. If $\mathcal{U}$ is a polytope with a small number of constraints (for example, the hose universe), we can use duality in a similar way to what we have seen in Section 3 to obtain a completely explicit LP for the problem.

Theorem 4.2. *Suppose that $\mathcal{U}$ is a polyhedron defined by k constraints*[1]*. Then* $\mathrm{RND}_{\mathrm{MPR}}$ *can be written as a linear program with $O(|W|^2|E|)$ constraints and $O(k|E|)$ variables.*

Proof. Let $\sum_{i<j} a_{ij}^{(\ell)} D_{ij} \leq b^{(\ell)}$, for $\ell \in [k]$, be the constraints defining $\mathcal{U}$. By strong LP duality, for any fixed $\boldsymbol{f}$ and $e \in E$,

$$\max_{D \in \mathcal{U}} \sum_{i<j} D_{ij} f_{ij}(e) \quad = \quad \begin{aligned} \min \quad & \sum_{\ell=1}^{k} b^{(\ell)} y_\ell(e) \\ \text{s.t.} \quad & \sum_{\ell=1}^{k} a_{ij}^{(\ell)} y_\ell(e) \geq f_{ij}(e) \qquad \forall \{i,j\} \in \tbinom{W}{2}. \end{aligned}$$

(Here, $(y_\ell(e))_{\ell=1\dots,k}$ are the dual variables.

[1] It also suffices if $\mathcal{U}$ can be written as a projection of a polyhedron with k constraints, i.e., that $\mathcal{U}$ has extension complexity k.

We return to the convex program in the previous proof, and use this dual formulation in (6). This yields

$$\begin{aligned}
\min \quad & \sum_{e\in E} c(e)u(e) \\
\text{s.t.} \quad & u(e) \geq \sum_{\ell=1}^{k} b^{(\ell)} y_\ell(e) \qquad \forall e \in E \\
& \sum_{\ell=1}^{k} a_{ij}^{(\ell)} y_\ell(e) \geq f_{ij}(e) \qquad \forall e \in E, \{i,j\} \in \tbinom{W}{2} \\
& f_{ij} \text{ is a unit } i\text{-}j\text{-flow} \qquad \forall \{i,j\} \in \tbinom{W}{2}.
\end{aligned}$$

□

Open problem 3. Since RND_{MPR} is polynomially solvable, and RND_{SPR} is polylog-hard in general, this implies that there can be a polylog gap between SPR and MPR. What about an $\Omega(\log n)$ gap? This should be the case, but I don't know a construction.

Exercise 3. Consider RND_{MPR} for the hose universe. By the previous theorem, we can solve this as a linear program, but one can still ask the question: is there always a hubbed solution that is optimal? In other words, is $\text{OPT}_{\text{SPR}} = \text{OPT}_{\text{MPR}}$?

Show that this is *not* the case, by providing an explicit counterexample. (Only three terminals are needed…)

4.2 Dynamic routing

Although impractical for reasons already discussed, dynamic routing is still of interest. One reason in particular is that it provides a clear lower bound that applies to any other routing scheme one might come up with; something less restrictive than SPR or MPR perhaps, but still plausible; see, e.g., [49, 47].

We will use, e.g., RND_{FR} and OPT_{FR} when referring to the dynamic routing model; this stands for "fractional routing".

Chekuri et al. [13] proved that RND_{FR} is NP-hard. On the positive side, we can get an $O(\log n)$ approximation, using exactly the same tree embedding approach as used for Proposition 2.8. In fact, that proof shows something more.

Proposition 4.3. *For any universe* $\mathcal{U}$, $\text{OPT}_{\text{SPR}} \leq O(\log n)\text{OPT}_{\text{FR}}$.

Proof. This is easily seen from the proof of Proposition 2.8, and the fact that SPR and FR are the same on a tree. We have, using the notation in the proof of Proposition 2.8,

$$\text{OPT}_{\text{SPR}}(G) \leq O(\log n)\mathbb{E}[\text{OPT}_{\text{SPR}}(T)] = O(\log n)\mathbb{E}[\text{OPT}_{\text{FR}}(T)] \leq O(\log n)\mathbb{E}[\text{OPT}_{\text{FR}}(G)]. \qquad \square$$

We can do better for specific choices of the universe, for example, the hose universe:

Proposition 4.4. *In the hose model $\mathcal{H}$,* $\text{OPT}_{\text{SPR}} \le 2\text{OPT}_{\text{FR}}$.

Proof. This follows from the proof of Theorem 3.2. That proof really showed that $\text{OPT}_{\text{SPR}} \le 2\,\text{cost}(\bar{D})$, where $\bar{D}$ was a single demand matrix in $\mathcal{H}$. Clearly $\text{OPT}_{\text{FR}} \ge \text{cost}(\bar{D})$. □

Open problem 4. $\text{OPT}_{\text{SPR}} \le 2\text{OPT}_{\text{FR}}$, but is a tighter bound possible? This is open. With Bruce Shepherd, I have some partial results that give a better bound for ℓ_1 metrics. If G is an expander graph, I do not know the answer (nor a strong belief).

In, fact, for the hose model, there is an even more basic open problem.

Open problem 5. What is the complexity status of OPT_{FR} for the hose model? Is it NP-complete, or polynomially solvable?

By the equivalence of separation and optimization, this is the same as asking: given a capacity allocation u, can one efficiently check if all demands in $\mathcal{H}$ can be routed, and if not, find a violating demand?

Can the ratio between OPT_{SPR} and OPT_{FR} really be as large as $\Omega(\log n)$? Considering the metric embedding argument for the upper bound, it is clear that a bad example must be one that cannot be approximated by a distribution over tree metrics to a ratio better than $O(\log n)$. One standard class of examples of such a poorly embeddable networks are *expander graphs*. That provides part of the motivation for the following construction.

Definition 4.5. A graph $G = (V, E)$ is a *c-expander* for some constant $c > 0$ if for every $S \subset V$ with $|S| \le |V|/2$, we have $|\delta(S)| \ge c|S|$.

The existence of expanders is trivial (consider just a complete graph), but much more surprising is the existence of *constant degree* expanders. In fact, a uniformly random d-regular graph is an expander (for some positive c) with high probability, and explicit constructions also exist; see, e.g., [37] for a survey, and also [41]. By choosing d large enough, we can make c as large as we like (even just by making parallel copies of edges).

So let $G = (V, E)$ be a 1-expander on n nodes with constant degree d. Now add a special sink node r to V to obtain our instance $\bar{G} = (\bar{V}, \bar{E}) = (V \cup \{r\}, E \cup \{vr : v \in V\})$, with terminal set $W = \bar{V}$. The edges in E have cost 1, and the edges adjacent to r have cost $\log n$.

The demand universe we use is $\mathcal{U}_{\text{ROB}}$, with $M \approx n/\log n$. Recall this means that each terminal may send at most 1 unit of demand to r, but the total demand never exceeds $n/\log n$.

Lemma 4.6. *There is an optimal solution $\boldsymbol{P}$ to* RND_{SPR} *whose support $\bigcup_{i \in V} P_{ir}$ is a tree.*

Proof. Let $\boldsymbol{P}$ be an optimal solution chosen so that $\sum_{e \in E} n_{\boldsymbol{P}}(e)^2$ is maximal, where $n_{\boldsymbol{P}}(e) = |\{i \in V : e \in P_{ir}\}|$.

Suppose that the support of $\boldsymbol{P}$ is not a tree. Then there must be two terminals $i, j \in V$ and a node $v \in V$ so that $v \in P_{ir}$ and $v \in P_{jr}$, but where the subpath Q_i of P_{ir} from v and the subpath Q_j of P_{jr} from v are distinct.

Consider the template $\boldsymbol{P^{(1)}}$ obtained from $\boldsymbol{P}$ by modifying only P_{ir} to use Q_j from v, and the template $\boldsymbol{P^{(2)}}$ obtained by modifying P_{jr} to use Q_i from v. Then $(n_{\boldsymbol{P^{(1)}}}(e) + n_{\boldsymbol{P^{(2)}}}(e))/2 = n_{\boldsymbol{P}}(e)$ for every $e \in E$.

Since the function $z \to \min(z, M)$ is concave, and $u_{\boldsymbol{Q}}(e) = \min(n_{\boldsymbol{Q}}(e), M)$ for any $\boldsymbol{Q}$, it follows immediately that

$$\Big(\sum_{e \in E} c(e) u_{\boldsymbol{P^{(1)}}}(e) + \sum_{e \in E} c(e) u_{\boldsymbol{P^{(2)}}}(e)\Big)\Big/2 \leq \sum_{e \in E} c(e) u_{\boldsymbol{P}}(e).$$

Furthermore the strict convexity of z^2 implies that

$$\Big(\sum_{e \in E} n_{\boldsymbol{P^{(1)}}}(e)^2 + \sum_{e \in E} n_{\boldsymbol{P^{(2)}}}(e)^2\Big)\Big/2 > \sum_{e \in E} n_{\boldsymbol{P}}(e)^2.$$

Together, these contradict our choice of $\boldsymbol{P}$. ☐

Theorem 4.7 ([26]). *For the instance described,* $\text{OPT}_{\text{SPR}} = \Omega(\log n)\text{OPT}_{\text{FR}}$.

Proof. We first show that $\text{OPT}_{\text{FR}} = O(n)$, by providing an explicit solution. We set $u(e) = 1$ for all $e \in E$, and $u(e) = 2/\log n$ for $e \in \delta(r)$; this has cost $2n + dn/2 = O(n)$. We need to show that any given $D \in \mathcal{U}_{\text{ROB}}$ is feasible. Since all demand involves the sink r, to check feasibility it suffices to check the cut condition: for any set $S \subseteq V$ (so $r \notin S$), the total demand $\sum_{i \in S} D_{ir}$ inside of S should not exceed the capacity $\sum_{e \in \delta_{\bar{G}}(S)} u(e)$ of the cut.

Now $\sum_{e \in \delta_{\bar{G}}(S)} u(e) = |S| \cdot \frac{2}{\log n} + |\delta_G(S)|$. If $|S| \geq n/2$, then this exceeds M just from the first term. If $|S| < n/2$, then $|\delta_G(S)| \geq |S| \geq \sum_{i \in S} D_{ir}$. In either case, the cut condition is satisfied, and so u is feasible.

Now we show that $\text{OPT}_{\text{SPR}} = \Omega(n \log n)$. Let $\boldsymbol{P}$ be an optimal solution whose support is a tree (call it T), the existence of which is guaranteed by Lemma 4.6. Call an edge of T *heavy* if it is used by at least M paths (otherwise call it light).

First, let $Z = \{i \in V : P_{ir} \text{ has no heavy edges}\}$. Since $c(e) = \log n$ for $e \in \delta(r)$, the cost of the solution is at least $|Z| \log n$. So we can assume $|Z| \leq n/2$, otherwise we are already done.

We will ignore terminals in Z from now on. Let T' be the tree obtained by deleting Z from T. Let F be the forest on $V \setminus Z$ obtained by removing all heavy edges from T', as well as r (note that all edges adjacent to r in T' were heavy).

Let $C_1, \ldots, C_k$ be the connected components of F; so $\sum_{i=1}^{k} |C_i| = |V \setminus F| \geq n/2$. Call a component C_j *small* if $|C_j| \leq \sqrt{n}$.

There are two cases.

Case 1. The total number of terminals in small components is at least $n/4$.

Then there are at least $\sqrt{n}/4$ small components. Hence T has at least $\sqrt{n}/4$ heavy edges, yielding a cost of at least $n^{3/2}/(4 \log n)$ just from the heavy edges—much more than $\Omega(n \log n)$.

Case 2. The total number of terminals in large components is at least $n/4$.

We will ignore the cost of heavy edges this time and focus only on light edges. Note that we can think of each terminal $i \in V$ as paying (once) for all the light edges on its path P_{ir}, since that precisely divides up the total cost of the light edge.

Consider some large component C_j. There is some node $v \in C_j$ such that all paths P_{ir} from a terminal $i \in C_j$ passes through v. (v will be one endpoint of a heavy edge.) But since G is d-regular, the number of nodes within a distance $\frac{1}{4}\log_d(n)$ of v is at most $\sum_{i=0}^{\frac{1}{4}\log_d(n)} d^i = O(n^{1/4})$. Thus most nodes in C_j will pay more than $\frac{1}{4}\log_d(n)$ along light edges to reach v, leading to a total cost of $\Omega(|C_j|\log_d(n))$ attributable to terminals in C_j. Summing over all large components, we find the desired $\Omega(n \log n)$ cost.

□

Notes

Theorem 4.1 and Theorem 4.2 are special cases of more general results about robust linear programs [8]. They are very useful, and have been rediscovered independently a number of times in various settings such as oblivious congestion minimization and robust network design [18, 3, 1, 38].

5 Some generalization and variants of the VPN Problem

5.1 A generalization of the VPN Problem

Recall the hose universe $\mathcal{H}(b)$. This consisted of all demand matrices which described a fractional b-matching on the complete graph on W. Here's a slightly different way of describing $\mathcal{H}(b)$. Let T be a star graph with leaf set W, and a single non-leaf node v. We give edge vi capacity b_i, for each $i \in W$. We then define

$$\mathcal{H}(b) = \{D \in \mathbb{R}_+^{\binom{W}{2}} : D \text{ is routable on } T\}.$$

This should be clear: there is a unique way of routing a given demand D, and this will be feasible as long as we don't exceed the capacity of any edge, which precisely means $\sum_{j \neq i} D_{ij} \leq b_i$ for all $i \in W$.

But why a star? It's extremely natural to consider any edge-capacitated tree T, again with the leaf set of T being precisely W. We then define

$$\mathcal{U}_{\text{TREE}}(T) = \{D \in \mathbb{R}_+^{\binom{W}{2}} : D \text{ is routable on } T\}.$$

Here is a different perspective. Any demand matrix D can be alternatively specified by a weighted complete graph on the terminals, with edge uv having weight D_{uv}. The hose universe can be interpreted as imposing singleton cut constraints on this graph: we must be able to route all demands such that for any $u \in W$, the weight of the cut $\delta(\{u\})$ in the demand graph does not exceed its marginal b_u. It is natural to study universes defined by more general cut families; each cut in a given family has a maximum capacity, and a demand is valid as long as it does not violate any of these "cut constraints". In other words, we are given a family $\mathcal{S}$ of nontrivial subsets of W; for each $S \in \mathcal{S}$, an upper bound $b_S \in \mathbb{N}$ is prescribed, and any symmetric feasible demand D must satisfy

$$\sum_{i \in S, j \notin S} D_{ij} \leq b_S \qquad \forall S \in \mathcal{S}.$$

$\mathcal{U}_{\text{TREE}}(T)$ corresponds exactly to the case where the sets in $\mathcal{S}$ form a nested family described by T.

5.1.1 A natural algorithm

Definition 5.1. A **hub location function** is a map $\phi : V(T) \to V$ satisfying $\phi(i) = i$ for all $i \in W$.

Given a hub location function ϕ, a **ϕ-hubbing** $R = (R_e)_{e \in E(T)}$ is a vector where for each $e = vw \in E(T)$, R_e is a $\phi(v)$-$\phi(w)$-path. A **hubbing** is a ϕ-hubbing for some choice of ϕ. A **hubbed solution** refers to a path template $\boldsymbol{P}$ obtained from some hubbing.

Notice that if T is a star with centre x, a ϕ-hubbing precisely matches the definition of a $\phi(x)$-hubbing introduced in the context of the hose model only. This is a natural generalization.

Lemma 5.2 ([43])**.** *The cheapest hubbed solution can be found in polynomial time.*

Proof. This is a straightforward dynamic program. Let T_x denote the subtree of T below x, for any internal node x of T. For $x \in V(T)$ and $v \in V$, let $C(x, v)$ denote the minimum cost of $\sum_{f=yz \in T_x} \text{SP}_c(\phi(y), \phi(z))$ under the restriction that $\phi(x) = v$. This can easily be computed bottom-up. □

5.1.2 The Generalized VPN Conjecture

How good is this algorithm? Notice that if T is a star, we know that it is optimal! This is precisely the VPN Theorem. Could this be true for arbitrary trees? I conjecture this to be the case.

Conjecture 5.3 (Generalized VPN Conjecture)**.** *For any capacitated tree T, the cheapest hubbed solution is always an optimal solution to* RND_{SPR} *for universe* $\mathcal{U}_{\text{TREE}}(T)$.

Open problem 6. This conjecture remains open, even for the smallest interesting case beyond a star: take T to be the union of two stars, with an edge between them, with all capacities unit.

The notion of a vine can be extended in a natural way to this case. Part 2 of the proof *can* be generalized (unpublished joint work with Michel Goemans, Yusuke Kobayashi and Rico Zenklusen). But part 1 seems to be problematic.

5.1.3 A constant factor approximation algorithm

Here we show that the proposed algorithm is at least a constant factor approximation. This holds even if we restrict $\phi(v) \in W$ for each $v \in V(T)$.

A random hub location function. As an upperbound to the cost of the algorithm, we will (carefully) pick a random hub location function, and take the expected cost of the solution it produces. Each hub location function in the support of the distribution will hub only at terminals (again, this matches what we saw before for the hose model).

The random hub location function ϕ is constructed as follows. We construct ϕ in bottom-up fashion. At the leaves, $\phi(i) = i$ of course. Consider a node $v \in V(T)$ for which ϕ has already been fixed for all children of v. We simply set $\phi(v) = \phi(C_v)$ where C_v is a child of v chosen uniformly at random.

What is the expected cost of the resulting solution? We simply need to determine, for each $e = vw \in E(T)$, the expected value of $\mathrm{SP}_c(\phi(v), \phi(w))$. The overall expected cost is thus

$$\sum_{i<j} \sum_{e=vw \in E(T)} \mathrm{SP}_c(i,j) \cdot \mathbb{P}(\{\phi(v), \phi(w)\} = \{i,j\})$$
$$= \sum_{i<j} \mathrm{SP}_c(i,j) \cdot \mathbb{P}(\exists vw \in E(T) : \phi(v) = i \text{ and } \phi(w) = j).$$

Let $\bar{D}_{ij} = \mathbb{P}(\exists vw \in E(T) : \phi(v) = i \text{ and } \phi(w) = j)$. So the expected cost of our solution is simply $\mathrm{cost}(\bar{D})$.

Lower bound. Just as in the factor 2 bound for the VPN problem, the lower bound we will use is very simple. We will show that $\bar{D}/2 \in \mathcal{U}_{\text{TREE}}(T)$. Since the cost to route a single demand in the universe is clearly a lower bound (for any routing scheme), and $\mathrm{cost}(\bar{D}/2) = \frac{1}{2}\mathrm{cost}(\bar{D})$, the theorem follows.

So fix any edge $f = xy \in E(T)$, labelled so that y is a child of x. Let L denote the set of leaves below y in T. We wish to show that $\sum_{i \in L} \sum_{j \in W \setminus L} \bar{D}_{ij} \le 2$, or in other words, that $\mathbb{E}|S| \le 2$, where

$$S = \{vw \in E(T) : \phi(v) \in L \text{ and } \phi(w) \notin L\}.$$

Let $x_0 = y$, and let x_t be the parent of x_{t-1}, with $x_\ell = r$. (So $x_1 = x$.) Let F_t denote the set of arcs between x_t and all of its children other than x_{t-1}, for $t \in [\ell]$. Let $Z = \max\{t \in [\ell] : \phi(x_t) \in L\}$. The edge $\{x_Z, x_{Z+1}\}$ is always in S, unless $Z = \ell$; let's just count 1 for this. And if $Z \ge t$, then

$F_t \in S$. This captures all edges in S. So

$$\begin{aligned}
\mathbb{E}|S| &\leq 1 + \sum_{t=1}^{\ell} \mathbb{P}(Z \geq t)|F_t| \\
&= 1 + \sum_{t=1}^{\ell} |F_t| \prod_{r=1}^{t} \frac{1}{|F_r| + 1} \\
&= 1 + \sum_{t=1}^{\ell} \prod_{r=1}^{t-1} \frac{1}{|F_r| + 1} - \sum_{t=1}^{\ell} \prod_{r=1}^{t} \frac{1}{|F_r| + 1} \\
&= 2 - \prod_{r=1}^{\ell} \frac{1}{|F_r| + 1} \\
&< 2.
\end{aligned}$$

This bound is not just a bound on the approximation ratio of the hierarchical hubbing algorithm, but it also bounds the gap between SPR and dynamic routing by 2. (Which is essentially tight, even in the case of a star).

This result can be extended, at the cost of increasing the constant, to arbitrary capacities; see Olver and Shepherd [43]. The obvious generalization of the above distribution for ϕ doesn't work, but a fairly minor tweak of it does.

Open problem 7. Olver and Shepherd [43] prove a factor of 8 for arbitrary capacities (and again, also bounding the gap between SPR and dynamic routing), but the argument is probably not tight. Can this bound be improved?

5.2 The capped hose model

Another natural generalization of the hose model is the *capped hose model*, introduced by Fréchette et al. [23]. We basically intersect the hose model with individual upper bounds on individual pairwise demands. So for $b : W \to \mathbb{R}_+$ and $p : \binom{W}{2} \to \mathbb{R}_+$, we define

$$\mathcal{H}_{\text{CAP}}(b, p) = \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : \sum_{j \neq i} D_{ij} \leq b_i \ \ \forall i \in W, D_{ij} \leq p_{ij} \ \ \forall i < j \Big\}.$$

A special case of this that still contains the VPN problem (with unit marginals) is the *mased hose model*. It is the above, with $b_i = 1$ for all $i \in W$, and $p_{ij} \in \{0, 1\}$ for all $i < j$. Instead of describing this via p, we can describe it by a graph H on the vertex set W representing the support of p.

$$\mathcal{H}_{\text{MASK}}(H) = \Big\{ D \in \mathbb{R}_+^{\binom{W}{2}} : \sum_{j \neq i} D_{ij} \leq b_i \ \ \forall i \in W, D_{ij} = 0 \ \ \forall \{i, j\} \notin H \Big\}.$$

We first observe that the masked hose model (and hence also the capped hose model) are APX-hard. Consider H a star, with $r \in W$ being the root. Then it's not hard to see that RND_{SPR} with universe $\mathcal{H}_{\text{MASK}}(H)$ precisely describes the Steiner tree problem.

Next, suppose H is a complete bipartite graph, with bipartition $W = W_1 \cup W_2$. This means we are dividing the terminals into two groups, call them senders and receivers; and a nonzero demand is always between a sender and a receiver. The resulting RND problem is called the *asymmetric VPN problem*, which has been studied quite a bit [34, 32, 15, 16, 46]. It is APX-hard but constant-factor approximable. The same is true for a complete multipartite graph [17].

Aside from the case where H is complete, there are at least some other polynomially solvable cases.

Theorem 5.4 ([10])**.** *Suppose H is a tree with bounded degree tree, or a cycle. Then* $\mathrm{RND_{SPR}}$ *for universe $\mathcal{H}_{\mathrm{CAP}}(H)$ can be solved exactly in polynomial time.*

Consider the case of a cycle, which is the most interesting. The algorithm, interestingly, is very similar to the one discussed for the generalized VPN problem. Define the graph $\hat{H}$ by replacing each $i \in W$ with a replacement $\hat{i}$, and then adding back the node i but attached only to $\hat{i}$. We compute an optimal embedding of $\hat{H}$ into G. In other words, we determine a map $\phi : V(\hat{H}) \to V$ such that $\phi(i) = i$ for all $i \in W$, and which minimizes

$$\sum_{vw \in E(\hat{H})} \mathrm{SP}_c(\phi(v), \phi(w)).$$

This can be done, again with dynamic programming. (It's worth noting though that for a general H, optimally embedding $\hat{H}$ is NP-hard. It's closely related to the *zero-extension problem.*)

Beyond this, not much is known. In particular, a constant factor approximation for general H is not known.

Open problem 8. What is the approximability of $\mathrm{RND_{SPR}}$ under the masked or capped hose models? Is there a constant factor approximation algorithm?

References

[1] A. Altın, E. Amaldi, P. Belotti, and M. C. Pınar. Provisioning virtual private networks under traffic uncertainty. *Networks*, 49(1):100–115, January 2007.

[2] M. Andrews. Hardness of buy-at-bulk network design. In *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 115–124, 2004.

[3] D. Applegate and E. Cohen. Making intra-domain routing robust to changing and uncertain traffic demands: understanding fundamental tradeoffs. In *Proceedings of ACM SIGCOMM*, pages 313–324, New York, NY, USA, 2003. ACM.

[4] S. Barman, S. Chawla, , and S. Umboh. Network design with coverage costs. In *Proceedings of APPROX*, pages 48–63, 2014.

[5] W. Ben-Ameur and H. Kerivin. New economical virtual private networks. *Communications of the ACM*, 46(6):69–73, June 2003.

[6] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization.* Princeton Series in Applied Mathematics. Princeton University Press, August 2009.

[7] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, November 1998.

[8] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1):1–13, August 1999.

[9] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88(3):411–424, September 2000.

[10] T. Bosman and N. Olver. Exploring the tractability of the capped hose model. In *Proceedings of the 25th Annual European Symposium on Algorithms (ESA)*, pages 19:1–19:12, 2017.

[11] J. Byrka, F. Grandoni, T. Rothvoß, and L. Sanità. Steiner tree approximation via iterative randomized rounding. *Journal of the ACM*, 60(1):6:1–6:33, 2013.

[12] C. Chekuri. Routing and network design with robustness to changing or uncertain traffic demands. *SIGACT News*, 38(3):106–128, September 2007.

[13] C. Chekuri, G. Oriolo, M. G. Scutellà, and F. B. Shepherd. Hardness of robust network design. *Networks*, 50(1):50–54, April 2007.

[14] 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. In *Proceedings of SIGCOMM*, pages 95–108, New York, NY, USA, 1999. ACM.

[15] F. Eisenbrand and F. Grandoni. An improved approximation algorithm for virtual private network design. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 928–932, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[16] F. Eisenbrand, F. Grandoni, G. Oriolo, and M. Skutella. New approaches for virtual private network design. *SIAM Journal on Computing*, 37(3):706–721, 2007.

[17] F. Eisenbrand and E. Happ. Provisioning a virtual private network under the presence of non-communicating groups. In *Proceedings of the 6th International Conference on Algorithms and Complexity*, volume 3998 of *Lecture Notes in Computer Science*, pages 105–114. Springer, 2006.

[18] T. Erlebach and M. Rüegg. Optimal bandwidth reservation in hose-model VPNs with multi-path routing. In *Proceedings of INFOCOM*, pages 2275–2282, 2004.

[19] J. Fakcharoenphol, S. Rao, and K. Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *Journal of Computer and System Sciences*, 69(3):485–497, November 2004.

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

[21] S. Fiorini, G. Oriolo, L. Sanità, and D. O. Theis. The VPN problem with concave costs. Accepted to SIAM Journal on Discrete Mathematics, 2010.

[22] L. Fleischer, J. Könemann, S. Leonardi, and G. Schäfer. Simple cost sharing schemes for multicommodity rent-or-buy and stochastic Steiner tree. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 663–670, New York, NY, USA, 2006. ACM.

[23] A. Fréchette, F. B. Shepherd, M. K. Thottan, and P. J. Winzer. Shortest path versus multi-hub routing in networks with uncertain demand. In *Proceedings of IEEE INFOCOM*, pages 710–718, 2013.

[24] A. Goel and I. Post. One tree suffices: A simultaneous O(1)-approximation for single-sink buy-at-bulk, April 2010.

[25] N. Goyal, N. Olver, and B. Shepherd. The VPN Conjecture is true. In *Proceedings of the 40th ACM Symposium on Theory of Computing (STOC)*, 2008.

[26] N. Goyal, N. Olver, and F. B. Shepherd. Dynamic vs. oblivious routing in network design. In *Proceedings of the 17th Annual European Symposium on Algorithms (ESA)*, volume 5757 of *Lecture Notes in Computer Science*, pages 277–288, 2009.

[27] N. Goyal, N. Olver, and F. B. Shepherd. The VPN conjecture is true. *J. ACM*, 60(3):17:1–17:17, June 2013.

[28] 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(3):361–365, 2008.

[29] S. Guha, A. Meyerson, and K. Munagala. A constant factor approximation for the single sink edge installation problems. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 383–388, New York, NY, USA, 2001. ACM.

[30] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, and B. Yener. Provisioning a virtual private network: a network design problem for multicommodity flow. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 389–398, New York, NY, USA, 2001. ACM Press.

[31] A. Gupta and A. Kumar. Private communication, 2001.

[32] A. Gupta, A. Kumar, M. Pál, and T. Roughgarden. Approximation via cost sharing: Simpler and better approximation algorithms for network design. *Journal of the ACM*, 54(3):11, 2007.

[33] A. Gupta, A. Kumar, and T. Roughgarden. A constant-factor approximation algorithm for the multi-commodity rent-or-buy problem. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 333–344, Los Alamitos, CA, USA, 2002. IEEE Computer Society.

[34] A. Gupta, A. Kumar, and T. Roughgarden. Simpler and better approximation algorithms for network design. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 365–372, New York, NY, USA, 2003. ACM.

[35] R. R. Guru Guruganesh, Jennifer Iglesias and L. Sanità. Single-sink fractionally subadditive network design. In *Proceedings of the 25th European Symposium on Algorithms (ESA)*, pages 46:1–46:13, 2017.

[36] A. Hayrapetyan, C. Swamy, and E. Tardos. Network design for information networks. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 933–942, 2005.

[37] S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(04):439–562, August 2006.

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

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

[40] D. Knuth. Two notes on notation. arXiv:9205211, 1992.

[41] A. Marcus, D. A. Spielman, and N. Srivastava. Interlacing families I: Bipartite Ramanujan graphs of all degrees. arXiv:1304.4132, 2013.

[42] N. Olver. *Robust network design*. PhD thesis, McGill University, 2010.

[43] N. Olver and F. B. Shepherd. Approximability of robust network design. In *Proceedings of the 21st ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1097–1105, 2010.

[44] G. Oriolo, L. Sanità, and R. Zenklusen. Network design with a discrete set of traffic matrices. *Operations Research Letters*, 41(4):390–396, 2013.

[45] H. Räcke. Optimal hierarchical decompositions for congestion minimization in networks. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing (STOC)*, pages 255–264, New York, NY, USA, 2008. ACM.

[46] T. Rothvoß and L. Sanità. On the complexity of the asymmetric VPN problem. In *Proceedings of the 12th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 326–338, 2009.

[47] M. G. Scutellà. On improving optimal oblivious routing. *Operations Research Letters*, 37(3):197–200, May 2009.

[48] D. P. Williamson and D. B. Shmoys. *The Design of Approximation Algorithms*. Cambridge University Press, 2011.

[49] M. Żotkiewicz and W. Ben-Ameur. More adaptive robust stable routing. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–6. IEEE, November 2009.