Approximation Algorithms for Hop Constrained and Buy-At-Bulk Network Design via Hop Constrained Oblivious Routing

Chandra Chekuri ✉
University of Illinois, Urbana-Champaign, IL, USA

Rhea Jain ✉
University of Illinois, Urbana-Champaign, IL, USA

Abstract

We consider two-cost network design models in which edges of the input graph have an associated *cost* and *length*. We build upon recent advances in hop-constrained oblivious routing to obtain two sets of results.

We address multicommodity buy-at-bulk network design in the nonuniform setting. Existing poly-logarithmic approximations are based on the junction tree approach [20, 58]. We obtain a new polylogarithmic approximation via a natural LP relaxation. This establishes an upper bound on its integrality gap and affirmatively answers an open question raised in [20]. The rounding is based on recent results in hop-constrained oblivious routing [38], and this technique yields a polylogarithmic approximation in more general settings such as set connectivity. Our algorithm for buy-at-bulk network design is based on an LP-based reduction to h-hop constrained network design for which we obtain LP-based bicriteria approximation algorithms.

We also consider a fault-tolerant version of h-hop constrained network design where one wants to design a low-cost network to guarantee short paths between a given set of source-sink pairs even when $k-1$ edges can fail. This model has been considered in network design [42, 41, 7] but no approximation algorithms were known. We obtain polylogarithmic bicriteria approximation algorithms for the single-source setting for any fixed k. We build upon the single-source algorithm and the junction-tree approach to obtain an approximation algorithm for the multicommodity setting when at most one edge can fail.

2012 ACM Subject Classification Theory of computation → Discrete optimization; Theory of computation → Routing and network design problems

Keywords and phrases Buy-at-bulk, Hop-constrained network design, LP integrality gap, Fault-tolerant network design

Digital Object Identifier 10.4230/LIPIcs.ESA.2024.41

Related Version *Full Version*: https://arxiv.org/abs/2404.16725 [23]

Funding Chandra Chekuri and Rhea Jain are partially supported by NSF grants CCF-1910149 and CCF-1907937.

Acknowledgements We thank Mik Zlatin for helpful discussions on hop-constrained metric embeddings.

1 Introduction

Network design is a fundamental area of research in algorithms touching upon several related fields, including combinatorial optimization, graph theory, and operations research. A canonical problem is Steiner Forest: given a graph $G=(V,E)$ with non-negative edge costs $c : E \to \mathbb{R}_+$ and terminal pairs $s_i, t_i \in V$ for $i \in [r]$, the goal is to find $F \subseteq E$ that connects all s_i-t_i pairs while minimizing the total cost of F. Steiner Tree is the special case when $s_i = s$ for all i. Steiner Tree is referred to as *single-source* problem, while Steiner

Forest is a *multicommodity* problem. Both are NP-Hard and APX-hard to approximate, and extensively studied. Steiner Forest has a 2-approximation [54] and Steiner Tree has a $\ln 4 + \varepsilon$-approximation [17]. In this paper we address a class of *two-cost* network design problems. The input to these problems is a graph $G = (V, E)$ where each edge has a non-negative cost $c(e)$, and a non-negative length $\ell(e)$; $c(e)$ represents a fixed cost and $\ell(e)$ represents a hop constraint or routing cost. Throughout the paper, we let n and m denote $|V|$ and $|E|$ respectively. The goal is to choose a low-cost subgraph of G to satisfy connectivity and/or routing requirements for some given set of source-sink pairs. The two-cost model is important due to its ability to model a number of fundamental problems. Despite many advances over the years, there are still several problems that are important for both theory and applications, but are not well-understood. We consider two sets of problems, formally described below. We obtain several new approximation algorithms and resolve an open problem from [20].

Buy-at-Bulk. In buy-at-bulk network design, the goal is to design a low-cost network to support routing demands between given source-sink pairs. The cost of buying capacity on an edge to support the flow routed on it is typically a subadditive function; this arises naturally in telecommunication networks and other settings in which costs exhibit economies of scale. Buy-at-bulk is an important problem in practice and has been influential in the approximation algorithms literature since its formal introduction in [9]; see Section 1.3 for details. We study the multicommodity version, denoted MC-BaB. The input consists of a graph $G = (V, E)$, a set of r demand pairs $\{s_i, t_i\}_{i\in[r]}$ with demand $\delta(i)$ each, and a monotone sub-additive cost function $f_e : \mathbb{R}^+ \to \mathbb{R}^+$ for each edge $e \in E$. We need to route $\delta(i)$ units of flow from s_i to t_i for each $i \in [r]$. Given a routing, its cost is $\sum_{e\in E} f_e(x_e)$ where x_e is the total amount of flow sent on edge x_e. The goal is to find a routing of minimum cost. We focus on the *nonuniform* setting, where each edge has its own cost function f_e, and refer to the case where f_e is the same for all edges ($f_e = c_e \cdot f$ for some function f) as the *uniform* problem. We call an instance *single-source* if all pairs have the same source s, i.e. $\exists s \in V$ such that $s_i = s$ for all $i \in [r]$. One can simplify the problem by losing an approximation factor of 2 [5], wherein we can assume that each f_e has a simple piecewise linear form: $f_e(x) = c(e) + \ell(e) \cdot x$; we call $c(e)$ the *cost* and $\ell(e)$ the *length.* Even though buy-at-bulk is naturally defined via routing flows, the two-cost model allows one to recast it in a different light by considering fixed costs and hop lengths in the aggregate. In particular, the problem now is to choose a set of edges $F \subseteq E$ where the objective function value of F is defined as $\sum_{e\in F} c(e) + \sum_{i\in[r]} \delta(i)\ell_F(s_i, t_i)$, where $\ell_F(s_i, t_i)$ is the shortest path length between s_i and t_i in the graph induced by F. Notice that the choice of routing has been made implicit in the objective. The uniform versions of the problem can be handled via metric embedding techniques [9], however, the nonuniform problem has been particularly challenging. MC-BaB admits a poly-logarithmic approximation in the nonuniform setting [20, 58]. However, the integrality gap of a natural LP relaxation for MC-BaB has been unresolved for over fifteen years [20]. We also note that MC-BaB is hard to approximate within a $\Omega(\log^c n)$-factor even in uniform settings (see Section 1.3).

Hop-constrained network design. The goal is to design low-cost networks in which source-sink pairs are connected by paths with few edges. We say that a path has hop-length h if it has h edges – this corresponds to path length with $\ell(e) = 1$ for all e. Such a hop-length constraint is natural in many telecommunication networks and is extensively studied in theory and practice, see Section 1.3 for details. Here, we study h-Hop Constrained Steiner Forest

(h-HCSF). The input is a graph $G = (V, E)$ with non-negative edge costs $c : E \to \mathbb{R}_+$ and r terminal pairs $s_i, t_i \in V$. The goal is to find $F \subseteq E$ minimizing $c(F) := \sum_{e \in F} c(e)$ such that for all $i \in [r]$, there exists an s_i-t_i path in F of hop-length at most h. Although hop-constrained problems had admitted bicriteria approximations in single-source and spanning settings, multicommodity versions such as h-HCSF were harder to tackle, and until recently, no non-trivial approximation algorithms (even bicriteria) were known. In the past few years, this barrier has been overcome through the use of probabilistic tree embeddings with hop constraints [49, 38, 36]. These results allow us to project the graph onto a tree, solve the problem on the tree, and project back to the original graph with distances preserved up to small factors. This has led to the development of several *bicriteria* approximation algorithms, in which the returned subgraph F connects each terminal pair with a path of length at most $\text{polylog}(n) \cdot h$ and the total cost of F is at most $\text{polylog}(n)$ times the optimal [49, 36]. We note that hop-constrained probabilistic tree embeddings pose several new challenges beyond those of traditional metric embeddings [13, 35] since the trees are partial (do not contain all vertices). Most recently, Ghaffari, Haeupler and Zuzic gave a congestion-based hop-constrained tree embedding [38], providing a hop-constrained analog to Räcke's seminal tree-based oblivious routing result [68].

MC-BaB and h-HCSF are related two-cost problems. While h-HCSF imposes a strict hop-constraint on the paths, MC-BaB penalizes the hop-length for the pairs in the objective function. The relation allows some results for one to be translated to the other with care.

Fault-tolerance. Fault-tolerant network design has been studied in a variety of settings with numerous practical applications. Via Menger's theorem, two nodes are k-edge-connected if there exist k edge-disjoint paths between them, or, equivalently, if they remain connected despite the failure of any $k-1$ edges. Survivable Network Design Problems (SNDP) is a central problem in this context. The input is similar to Steiner Forest; in addition each $s_i t_i$ pair specifies a requirement $k_i \in \mathbb{Z}$ and the goal is to find a min-cost subgraph in which each $s_i t_i$ is k_i-edge-connected. Jain's seminal work [54] obtained a 2-approximation for this problem. In the hop-constrained setting, the corresponding variant of Menger's theorem does not hold (see [42] and Figure 1). Thus, there are two natural higher-connectivity generalizations of hop-constrained network design.

The first version is the focus of this paper and is the fault-based generalization of hop-constrained network design, introduced in [42]. We say $u, v \in V$ are (h, k)*-hop-connected* in G if there exists a path using at most h edges in $E \setminus Q$ for all $Q \subseteq E$, $|Q| < k$. For instance, s and t are $(h, 2)$-hop-connected in Figure 1. We define the (h, k)*-Fault-Tolerant Hop Constrained Network Design* problem, denoted (h, k)-Fault-HCND: given r terminal pairs $s_i, t_i \in V$ with associated connectivity requirements k_i, the goal is to find $F \subseteq E$ minimizing $c(F)$ such that for all $i \in [r]$, s_i, t_i are (h, k_i)-hop-connected. We let $k = \max_{i \in [r]} k_i$.

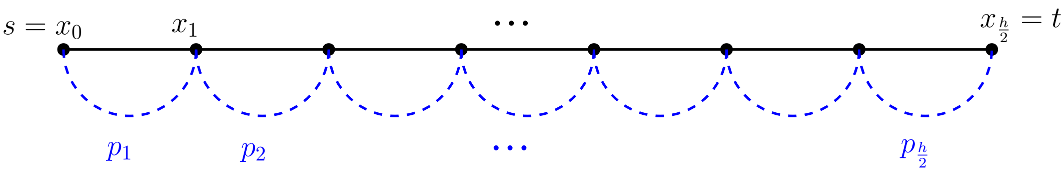

Figure 1 Each p_i has $\text{hop}(p_i) = h/2$. s and t are connected by a path of hop-length h despite the failure of any one edge. However, one needs hop-length $\Omega(h^2)$ to obtain two edge disjoint s-t paths.

The second version is the Hop-Constrained Survivable Network Design (HC-SNDP) problem: given a hop constraint h and terminal pairs s_i, t_i with connectivity requirements k_i, a feasible solution contains k_i edge-disjoint s_i-t_i paths, *each* of hop length at most h. The goal is to find a min-cost feasible subgraph. In Figure 1, s and t are $(h, 2)$-hop-connected but any two s-t disjoint paths must have at least one path of length $\Omega(h^2)$.

As noted in [42, 41], the two versions are both interesting and relevant from a telecommunication network design point of view. While there exist several algorithms via IP-solvers for both versions of hop-constrained problem (see Section 1.3), there are no known efficient approximation algorithms.

Now we discuss buy-at-bulk in the fault-tolerant setting, an important practical problem first studied in approximation by Antonakapolous et al. [6]. The goal is to find a min-cost subgraph that allows the demands to be routed even under failures which requires routing each demand along disjoint paths. It has proved to be a challenging problem. For instance, even in the single-source nonuniform setting, poly-logarithmic approximation algorithms are not known for protecting against a single edge failure! Progress has been made for some special cases (see Section 1.3). One can show that bicriteria algorithms for HC-SNDP would imply algorithms for fault-tolerant nonuniform Buy-at-Bulk and vice-versa. Our current techniques do not seem adequate to address these problems in their full generality even though there are natural LP relaxations. In this version we focus on the first version of fault-tolerant hop-constrained network design which is also challenging.

Set Connectivity. There are several problems where we seek connectivity between pairs of *sets*. In such problems, the input consists of set pairs $S_i, T_i \subseteq V$ instead of terminal pairs $s_i, t_i \in V$. The goal is to find paths connecting S_i to T_i for each $i \in [r]$. This problem was first introduced in the single source setting, known as Group Steiner Tree, and has been influential in network design (see Section 1.3). Recent progress has resulted in algorithms for the fault-tolerant version [27] and also for the hop-constrained version [49, 36]. One can naturally extend Buy-at-Bulk to this more general setting, which we consider in this work.

1.1 Overview of Results

We consider polynomial-time approximation algorithms. For Buy-at-Bulk we consider the standard approximation ratio. For Hop-Constrained Network Design we consider bi-criteria (α, β)-approximations where α is the approximation for the cost of the output solution and β is the violation in hop constraint. We let OPT_I and OPT_{LP} respectively denote the values of an optimum integral solution and of an optimum fractional solution to an underlying LP relaxation (Section 2).

Our first set of results address multicommodity settings. Theorem 1 resolves an open question posed by [20] by proving a polylogarithmic approximation for multicommodity Buy-at-Bulk with respect to the LP relaxation. The resulting algorithm provides a new approach to the problem.

► **Theorem 1.** *There is a randomized $O(\log D \log^3 n \log r)$-approximation for multicommodity Buy-at-Bulk with respect to OPT_{LP}, where $D = \max_{i\in[r]} \delta(i)$ is the maximum demand. The approach extends to Buy-at-Bulk Set Connectivity with an approximation ratio of $O(\log D \log^7(nr))$.*

There is an $O(\log^4 r)$-approximation for MC-BaB with respect to the optimal integral solution [20]; our result essentially matches this up to an $O(\log D)$ factor. Removing this factor is a direction for future research, see Remark 12. No previous approximation was known for Buy-at-Bulk Set Connectivity.

Theorem 2 describes a polylogarithmic bicriteria approximations for h-HCSF and the Set Connectivity version (referred to as h-HCSC) with respect to optimal fractional solutions.

▶ **Theorem 2.** *There exist polylogarithmic bicriteria approximation algorithms for hop-constrained network design problems with respect to* OPT_{LP}*. In h-Hop Constrained Steiner Forest, we obtain cost factor $\alpha = O(\log^2 n \log r)$ and hop factor $\beta = O(\log^3 n)$. In h-Hop Constrained Set Connectivity, we obtain cost factor $O(\log^5 n \log r)$ and hop factor $\beta = O(\log^3 n)$.*

There exist bicriteria approximations for both h-HCSC and h-HCSF with respect to optimal integral solutions via partial tree embeddings [49], which were later improved through the use of clan embeddings [36]. Our results match or improve on those in [49]; see Section 1.3. In addition to inherent interest and other applications, an important reason for proving Theorem 2 is to derive Theorem 1 via a reduction (see Section 3.1).

Our next set of results are on hop-constrained network design in the *fault-tolerant* setting. There were previously no known approximation algorithms for this problem. Theorems 3 obtains bicriteria approximation for the single-source setting for any fixed number of failures.

▶ **Theorem 3.** *There is a randomized $(O(k^3 \log^6 n), O(k^k \log^3 n))$-approximation for* single-source *(h, k)-Fault-HCND with respect to* OPT_{LP} *and runs in time $n^{O(k)}$.*

We consider the multicommodity setting and prove Theorem 4 for $k = 2$ via a reduction to single-source. We note that the result is with respect to the optimal integral solution.

▶ **Theorem 4.** *There is a randomized polynomial time $(O(\log^6 n \log^2 r), O(\log^3 n \log r))$-approximation for multicommodity $(h, 2)$-Fault-HCND with respect to* OPT_I*.*

There are several known advantages for considering LP based algorithms even when there exist combinatorial algorithms. Working with fractional solutions allows us to extend results to otherwise complex settings. For example, without much additional overhead, we are able to extend our results to set connectivity. We can also extend to the *prize collecting* versions using existing LP-based algorithms ([14] and follow-up work), in which each demand pair is associated with a penalty; the goal is to find an $F \subseteq E$ minimizing $c(F)$ plus the sum of penalties of demand pairs not adequately connected by F. It is sometimes possible to improve LP-based algorithms for prize-collecting versions as was done recently for Steiner Forest [1], however, it requires substantial problem-specific machinery.

Another important motivation is to address higher connectivity variants. Though the extension is nontrivial, we are able to use ideas from the LP-based multicommodity algorithms to obtain algorithms for the fault-tolerant variants of hop-constrained network design. We believe that this work provides a stepping stone to handle the challenges that remain in addressing fault-tolerant buy-at-bulk as well as extending Theorem 4 for $k > 2$. As we remarked earlier, all the fault-tolerant versions have natural LP relaxations, however, bounding their integrality gap is challenging.

1.2 Overview of Techniques

The proofs of Theorems 1 and 2 are inherently connected, as multicommodity buy-at-bulk and hop-constrained network design essentially reduce to each other if one is willing to lose polylogarithmic factors in the approximation ratios. Such a connection between the problems has been previously pointed out in the single-source setting with respect to integral solutions [50]; we extend this to an LP based relation with a bit of technical work. Our main contribution is to prove Theorem 2. We note that existing approximation algorithms for h-HCSF use distance-based embeddings [49]. Unlike single-cost network design problems, these hop-constrained tree embeddings cannot be directly used to argue about fractional solutions. This is because the embeddings are *partial* trees that do not include all vertices, so there is no natural projection of a fractional solution on G to a solution on T. Instead, we leverage a connection to the recent *congestion-based* hop-constrained partial tree embeddings [38]. We use this tool by solving the LP relaxation for h-HCSF and using the fractional x_e values to define a capacitated graph G' from which we sample trees. This allows us to solve the relevant problems on the tree and project the solution back to the input graph.

Our second set of results is for the fault-tolerant setting of hop-constrained problems. We approach the problems here via *augmentation*, where we start with an initial partial solution and repeatedly augment to increase the connectivity of terminal pairs whose requirement is not yet satisfied. A technical challenge in using congestion-based tree embeddings for higher connectivity problems is that polylogarithmically many paths in the input graph can map to the same path in the tree embedding. To overcome this, we rely on a recent powerful approach of Chen et al. for Survivable Set Connectivity [27] which has also found some extensions in [22]. The framework provides a clever way to assign capacities to edges, avoiding the aforementioned difficulty while ensuring that $O(\text{polylog}\, n)$ rounds of an oblivious dependent tree-rounding process are adequate to obtain a feasible solution. Unlike [27, 22] who applied this framework with Räcke's congestion based tree embeddings, we need to use hop-constrained version [38]. Unfortunately, the framework does not generalize in a nice way due to the additional complexity of hop-constraints. We were able to overcome this difficulty in the single-source setting by analyzing the diameter of components as the augmentation algorithm proceeds. The multicommodity setting poses non-trivial challenges; thus we restrict our attention to $k = 2$ which is often relevant in practice. We use the *junction-tree* technique developed in [20] for buy-at-bulk network design and later extended by [6] for fault-tolerant buy-at-bulk when $k = 2$. This approach reduces the multicommodity problem to the single-source setting. However, the argument relies on the integral optimum solution and hence Theorem 4 is only with respect to OPT_I. Our key contribution for the multicommodity setting with $k = 2$ is showing the existence of a good junction structure in the fault-tolerant hop-constrained setting.

1.3 Related Work

1.3.1 Group Steiner Tree and Set Connectivity

Group Steiner tree (GST) was first introduced by Reich and Widmayer [70]. Garg, Konjevod and Ravi studied approximation algorithms for the problem and provided a randomized rounding scheme, giving an $O(\log n \log r)$-approximation when the input graph G is a tree [37]. This was extended to an $O(\log^2 n \log r)$-approximation for general graphs via tree embeddings [13, 35]. These results are essentially tight; there is no efficient $\log^{2-\varepsilon}(r)$ approximation for GST [52], and there is an $\tilde{\Omega}(\log n \log r)$ lower bound on the integrality gap of GST when the input graph is a tree [51]. Approximation algorithms when allowing for quasi polynomial running time have been studied, see [26, 44, 39]. Multicommodity Set Connectivity was first introduced by Alon et al. [2] in the online setting. The first polylogarithmic approximation ratio for offline Set Connectivity was given by Chekuri, Even, Gupta, and Segev [21].

In the higher connectivity setting, we are given an additional parameter k and the goal is to connect each S_i-T_i pair with k edge-disjoint paths. Gupta et al. showed that Survivable Group Steiner Tree has a polylogarithmic approximation when $k = 2$ [46]. Chalermsook, Grandoni, and Laehkanukit [18] studied the higher connectivity version of Set Connectivity. They gave a bicriteria approximation algorithm, where the solution only contains k/β edge-disjoint paths between each pair and costs at most α times the optimal, for $\alpha, \beta \in \text{poly}\log(n)$. This approximation is via Räcke congestion-based tree embeddings and Group Steiner Tree rounding ideas from [37]. In recent work by Chen et al. [27], similar ideas are used to give the first true polylogarithmic approximation for Survivable Set Connectivity.

1.3.2 Buy-at-Bulk Network Design

Buy-at-bulk network design was introduced to the algorithms literature by Salman et al. [73] in 1997 in the single-source setting; the multicommodity setting was first considered by Awerbuch and Azar [9]. The model was first introduced to capture multiple cable types with different capacities, in which the cost per unit of bandwidth decreases as the capacity increases, giving us the general subadditive cost function we consider in this work. This problem had been previously considered in practical settings and operations research; see [72, 15, 15] for a few such examples. Since then, it has been extensively studied in a variety of different settings; we limit our discussion to relevant results in undirected graphs for the offline edge-cost Buy-at-Bulk problem.

We begin with a discussion of the uniform setting: when each edge has the same cost function f_e. In the single-source setting, the first constant factor approximation was given by Guha, Meyerson, and Munagala [45], later improved by Talwar [74], Gupta, Kumar, and Roughgarden [47], and finally Grandoni and Italiano [43]. In the multicommodity setting, Awerbuch and Azar gave an $O(\log n)$ via probabilistic tree embeddings [9, 13, 35]; this bound improves to $O(\log r)$ via a refinement of the distortion in tree embeddings [48]. Andrews showed an $\Omega(\log^{1/4-\varepsilon}(n))$ hardness of approximation factor for MC-BaB [4].

In the nonuniform setting, Meyerson, Munagala, and Plotkin gave a randomized $O(\log r)$-approximation [65] for single-source BaB. Chekuri, Khanna, and Naor [24] obtained a deterministic algorithm by derandomizing the algorithm in [65] via an LP relaxation which also established that the LP's integrality gap is $O(\log r)$. The first nontrivial approximation for the general multicommodity case was a simple randomized greedy algorithm given by Charikar and Karagiozova [19] with an approximation factor of $\exp(O(\sqrt{\ln n \ln\ln n}))$. Chekuri et al. gave an $O(\log^4 r)$-approximation for nonuniform multicommodity buy-at-bulk [20]. They also give a simple greedy combinatorial $O(\log^3 r \log D)$-approximation. In the case that $D = \text{poly}(n)$, Kortsarz and Nutov improved the approximation to $O(\log^3 n)$ [58]. On hardness of approximation, multicommodity has an $\Omega(\log^{1/2-\varepsilon}(n))$ lower bound [4] and single source has an $\Omega(\log\log n)$ lower bound [29].

We briefly discuss Buy-at-Bulk with protection. Recall that the goal is to route along k edge-disjoint/ node-disjoint routes (note that node-connectivity generalizes edge-connectivity). This protected setting was first introduced by Antonakapolous et al., who give an $O(1)$-approximation when $k = 2$ in the single cable, node-connectivity setting [6]. They also show when $k = 2$ that an α-approximation with respect to the LP for single-source implies an $O(\alpha \log^3 n)$-approximation for multicommodity via junction arguments, both in the uniform and nonuniform settings. Chekuri and Korula subsequently gave two results in the single-source node-connectivity setting: $(\log n^{O(b)})$-approximation for $k = 2$ with b cables and $(2^{O(\sqrt{n})})$-approximation for any fixed k in the nonuniform case [25]. In the edge-connectivity setting, Gupta, Krishnaswamy, and Ravi gave an $O(\log^2 n)$-approximation for the multicommodity uniform setting with $k = 2$ [46]. Obtaining a polylogarithmic approximation algorithm for $k \geq 3$ in the uniform setting and $k \geq 2$ in general remains an important open problem.

1.3.3 Hop-Constrained Network Design

Hop constrained network design was first introduced by Balakrishnan and Altinkemer [10]. It has since been shown that hop constraints have many applications, including easier traffic management [8, 66], faster communication [31, 76], improved tolerance to failure [75, 71], and more [60, 40, 30]. Unfortunately, hop constrained network design problems have been shown to be hard; even MST with hop constraints has no $o(\log n)$-approximation in polynomial time [12]. The majority of the approximation algorithms and IP formulations for hop constrained network design problems have been limited to simple connectivity problems, such as MST [30, 76, 67, 3, 57, 69, 64], Steiner Tree [59, 55], and k-Steiner Tree [50, 56]. One reason for this limitation is the hardness of approximating more complex problems. For example, hop-constrained Steiner Forest has no polynomial time $o(2^{\log^{1-\varepsilon}(n)})$-approximation [34]. Note that the hardness results are when the hop constraints are strict. Much less is known on the hardness when hop constraints can be relaxed by constant or poly-logarithmic factors. Natural generalizations of Steiner Tree, including Steiner Forest and Set Connectivity, were not studied in the hop-constrained setting until recently.

Recent progress on hop-constrained tree embeddings has led to significant progress in many hop-constrained network design problems. This work was initiated by Haeupler, Hershkowitz, and Zuzic [49]. They extended the line of work on distance-based probabilistic tree embeddings (see [13, 35]) to the hop-constrained setting. In particular, they show that although hop-constrained distances are inapproximable by metrics, one can approximate hop-constrained distances with partial trees that contain only a fraction of nodes of the original graph. This led to the development of bicriteria approximations for several offline problems (k-Steiner Tree, Relaxed k-Steiner Tree, Group Steiner Tree, Group Steiner Forest), as well as some online and oblivious problems. Note that bicriteria approximations are necessary due to the hardness result mentioned earlier. For Hop-Constrained Steiner Forest, they obtain a cost factor $\alpha = O(\log^3 n)$ and hop factor $\beta = O(\log^3 n)$ with respect to the optimal integral solution. For Hop-Constrained Set Connectivity, they obtain a cost factor $\alpha = O(\log^6 n \log r)$ and hop factor $O(\log^3 n)$. Many of these results were then improved by Filtser [36] via *clan*-embeddings, in which embeddings contain multiple copies of each vertex in the original graph. In particular for HCSF they improve the cost factor to $\tilde{O}(\log^2 n)$, hop factor to $\tilde{O}(\log^2 n)$ and for HCSC they improve the cost factor to $\alpha = \tilde{O}(\log^4 n \log k)$, hop factor $\beta = \tilde{O}(\log^2 n)$. Most recently, Ghaffari, Haeupler, and Zuzic [38] extended their hop-constrained tree embedding results to give an oblivious routing scheme with low congestion as well as distance dilation. This is analogous to Räcke's work with congestion-based tree embeddings [68] without hop constraints.

Hop-constrained network design in the fault-tolerant model has not been considered in approximation before this work; however, there is a fair amount of literature on algorithms via IP-solvers. These models have been shown to be of importance in several practical settings, including telecommunication networks [33] and transportation shipments [11, 61]. HC-SNDP was first considered in the single-pair setting [53]; this is essentially min-cost flow if one allows for $O(k)$ hop distortion. There is extensive literature on IP formulations in the single-pair setting which we omit here. In the past decade, the single source setting [63] and multicommodity setting [32, 62, 33, 16] have been studied as well. Fault-HCND was recently introduced by [42] under the name "Network Design Problem with Vulnerability Constraints" (NDPVC), where they provide several IP formulations for the $k = 2$ case (protecting against a single edge failure). This work was improved in [41] and extended to $k \geq 3$ by [7].

Organization: We describe the LP relaxations and relevant background on tools used in the paper in Section 2. We discuss the first set of results (Theorems 1 and 2) in Section 3. We discuss the second set of results (Theorems 3 and 4) on fault-tolerance in Section 4. Due to space constraints we are unable to provide many of the involved technical details in the main body, and encourage the reader to see the full version [23].

2 Preliminaries

2.1 Linear Programming Relaxations

Following [20], we define path-based LP relaxations for both Buy-at-Bulk (see **LP-BB**) and hop-constrained network design (see **LP-HC**). For $i \in [r]$, let $\mathcal{P}_i$ denote the set of all simple s_i-t_i paths. For the hop-constrained setting, let $\text{hop}(p)$ denote the hop-length of any path p, and let $\mathcal{P}_i^h$ denote the set of all simple s_i-t_i paths with hop-length at most h. In both relaxations, we define variables $x_e \in [0,1]$ for $e \in E$ and $f_p \in [0,1]$ for $p \in \bigcup_{i\in[r]} \mathcal{P}_i$ in **LP-BB** and $p \in \bigcup_{i\in[r]} \mathcal{P}_i^h$ in **LP-HC**; these are indicators for whether or not an edge e or path p is used. The relaxations are described below.

(**LP-BB**)

$$\begin{aligned}
\min \sum_{e\in E} c(e)x_e &+ \sum_{i\in[r]} \delta(i) \sum_{p\in\mathcal{P}_i} \ell(p) f_p \\
\text{s.t.} \sum_{p\in\mathcal{P}_i} f_p &= 1 && \forall i \in [r] \\
\sum_{p\in\mathcal{P}_i, e\in p} f_p &\le x_e && \forall e \in E, i \in [r] \\
x_e, f_p &\ge 0
\end{aligned}$$

(**LP-HC**)

$$\begin{aligned}
\min \sum_{e\in E} c(e)x_e & \\
\text{s.t.} \sum_{p\in\mathcal{P}_i^h} f_p &= 1 && \forall i \in [r] \\
\sum_{p\in\mathcal{P}_i^h, e\in p} f_p &\le x_e && \forall e \in E, i \in [r] \\
x_e, f_p &\ge 0
\end{aligned}$$

LP-HC can be extended to the fault-tolerant setting, as described in Section 4.

▶ Remark 5. **LP-HC** and **LP-BB** can each be solved in polynomial time via separation oracle on their respective duals.

▶ **Lemma 6.** *We assume that given any solution (x, f) to **LP-BB** or **LP-HC** that $x_e \ge \frac{1}{m^2}$ by increasing the approximation ratio by at most $1 + 1/(m-1)$.*

2.2 Hop-Constrained Congestion-Based Embeddings

The results in this paper use the congestion-based hop-constrained embeddings developed in [38]. Given a graph $G = (V, E)$, a *partial tree embedding* (T, M) is a rooted tree T such that $V(T) \subseteq V$, along with a mapping $M : E(T) \to 2^E$ that maps each edge $uv \in E(T)$ to a path M_{uv} from u to v in G. We let $P_T(u, v) \subseteq E(T)$ denote the unique u-v path in T. We extend the definition of M_{uv} for $uv \notin E(T)$ to be the concatenation of $M_{e_1}, \ldots, M_{e_t}$, where $e_1, \ldots, e_t = P_T(u, v)$. For $e \in E$, let $\text{flow}(M_{uv}, e)$ denote the number of times M_{uv} traverses e. We use the following lemma.

▶ **Lemma 7** ([38])**.** *For every complete capacitated graph $G = (V, E, x)$ and every $\varepsilon \in (0, 1/3)$, there exists a distribution $\mathcal{T}$ over partial tree embeddings such that:*

1. (Exclusion probability) *For each $v \in V$, $\mathbf{Pr}_{(T,M)\sim\mathcal{T}}[v \in V(T)] \ge 1 - \varepsilon$,*
2. (Hop dilation) *For each $(T, M) \sim \mathcal{T}$, $\forall u, v \in V(T)$, $\text{hop}(M_{uv}) \le O\left(\frac{\log^3 n}{\varepsilon}\right)$,*

3. (Congestion) *For* $e \in E$, $\mathbf{E}\left[\sum_{u,v \in V(T)} \text{flow}(M_{uv}, e) \cdot x_{uv}\right] \leq O\left(\log n \log \frac{\log n}{\varepsilon}\right) x_e$.

Furthermore, if the ratio $\frac{\max_{e \in E} x(e)}{\min_{e \in E} x(e)}$ *is at most* $\text{poly}(n)$*, the height of all trees in the support of* $\mathcal{T}$ *is bounded by* $O(\log n)$*. Following [38], we refer to these distributions as* $\mathcal{D}^{(1)}$*-routers.*

For a partial tree embedding (T, M) and an edge $e' \in E(T)$, we let $A_{e'}, B_{e'} \subseteq V(T)$ denote the two components of $T \setminus e'$. Let $E(A_{e'}, B_{e'})$ denote the set of edges of G with one endpoint in $A_{e'}$ and the other in $B_{e'}$; these are the edges on the cut $\delta_{G[V(T)]}(A_{e'})$. We let $y(e') = \sum_{e \in E(A_{e'}, B_{e'})} x_e$; we call this the *capacity* of a tree edge e'. For ease of notation, we refer to the *inverse* of the mapping function M as $M^{-1}(e) = \{e' \in E(T) : e \in M_{e'}\}$.

▶ **Lemma 8.** *Let* $e \in E$ *and let* $\mathcal{T}$ *be a* $\mathcal{D}^{(1)}$*-router.*

$$\mathop{\mathbf{E}}_{(T,M)\sim\mathcal{T}}\left[\sum_{e' \in M^{-1}(e)} y(e')\right] \leq O(\log n \log \tfrac{\log n}{\varepsilon}) x_e.$$

▶ Remark 9. We use $\mathcal{D}^{(1)}$-routers with x_e values from **LP-HC** as capacities. We assume the height of the trees in the support of $\mathcal{T}$ are at most $O(\log n)$, since $x_e \in [\frac{1}{m^2}, 1]$ for all $e \in E$ by Lemma 6. We also use Lemma 7 on graphs that are not complete. This is without loss of generality; we can set $x_e = \frac{1}{m^2}$ for all $e \notin E$, so the total amount of flow carried on edges not in E is at most $\frac{1}{m}$. It is easy to verify that this does not affect our results.

2.3 Group Connectivity and Oblivious Rounding

Motivated by higher connectivity variants of Set Connectivity (see Section 1.3), Chalermsook, Grandoni, and Laekhanukit [18] describe a rounding algorithm based on Räcke's seminal congestion-based tree embeddings [68] that is oblivious to the set pairs. We use the following lemma summarizing their result, and refer to the algorithm as TreeRounding(T, x, f).

▶ **Lemma 10** **([18, 27]).** *Consider an instance of Set Connectivity on an* n*-node tree* $T = (V, E)$ *with height* h *and let* $x : E \to [0, 1]$*. Suppose* $A, B \subseteq V$ *are disjoint sets and suppose* $K \subseteq E$ *such that* x *restricted to* K *supports a flow of* $f \leq 1$ *between* A *and* B*. There is a randomized algorithm that is oblivious to* A, B, K *(hence depends only on* x *and value* f*) that outputs a subset* $E' \subseteq E$ *such that (i) The probability that* $E' \cap K$ *connects* A *to* B *is at least a fixed constant* ϕ *and (ii) For any edge* $e \in E$*, the probability that* $e \in E'$ *is* $\min\{1, O(\frac{1}{f} h \log^2 n) x(e)\}$*.*

3 Multicommodity Two-Cost Network Design

3.1 LP-Based Reduction from Buy-at-Bulk to Hop-Constraints

We consider Buy-at-Bulk network design and describe an LP-based reduction to corresponding hop-constrained network design problems via the following lemma:

▶ **Lemma 11.** *Suppose we are given an* (α, β)*-approximation for HCSF on a graph with* $r \cdot \text{poly}(n)$ *vertices with respect to its optimal fractional solution. Then there exists an* $O(\log D(\alpha \log n + \beta))$*-approximation for multicommodity Buy-at-Bulk with respect to* OPT_{LP}*, where* $D = \max_{i \in [r]} \delta(i)$*. This reduction can be extended to the Set Connectivity setting.*

Note that Theorem 1 follows directly as a corollary of Lemma 11 and Theorem 2. We will discuss the reduction in the set connectivity setting, as this generalizes multicommodity. Let $G = (V, E)$ be a graph with edge costs $c : E \to \mathbb{R}_{\geq 0}$, edge lengths $\ell : E \to \mathbb{R}_{\geq 0}$, and demand pairs $\{S_i, T_i\}_{i \in [r]}$ with demands $\delta(i)$. Recall that our goal is to find $F \subseteq E$ minimizing $c(F) + \sum_{i \in [r]} \delta(i) \ell(p_i)$, where p_i is the shortest S_i-T_i path in F.

The reduction consists of two components. First, we show that we can assume $\delta(i) = 1$ for all $i \in [r]$, $\ell(e) = 1$ for all $e \in E$ with an $O(\log D)$ loss in the approximation ratio. The idea is to partition the set of terminal pairs based on their demands and solve each subinstance separately, and also subdivide each edge into paths of length $\ell(e)$, where each new edge has length 1. There are some technical difficulties in handling settings where the maximum edge length is large, as this reduction could introduce exponentially many variables. This can be handled via scaling tricks on the fractional solution. Second, we use an LP solution to "guess" the optimal hop-requirement for each demand pair; we then partition demand pairs based on this hop-constraint and solve each subinstance separately (see Algorithm 1). The proofs of cost and correctness are simple and deferred to the full version [23].

Algorithm 1 Reducing BaB to Hop-Constrained Network Design.

$F \leftarrow \emptyset$
$(x, f) \leftarrow$ feasible solution to **LP-BB**
For $i \in [r]$, $h_i \leftarrow \sum_{p \in \mathcal{P}_i} \ell(p) f_p$
for $j = 1, \ldots, \log n + 1$ **do**
  $T_j \leftarrow \{i : h_i \in [2^{j-1}, 2^j)\}$
  $F_j \leftarrow (\alpha, \beta)$-approx for HCSC with terminal pairs T_j, hop constraint 2^{j+1}
  $F \leftarrow F \cup F_j$
return F

▶ Remark 12. We believe that the $\log D$ factor is not needed in the upper bound on the integrality gap and can remove this factor in some special cases. In general, this has been difficult to avoid in several previous approaches for nonuniform buy-at-bulk. One reason is we rely on hop-constrained tree embeddings which assume that $\ell(e) = 1$ for each edge.

3.2 Hop-Constrained Network Design Problems via Oblivious Routing

In this section we discuss Theorem 2. We give an overview of the algorithm and prove key lemmas; detailed proofs of correctness follow from $\mathcal{D}^{(1)}$-router guarantees and are deferred to the full version [23]. Let $G = (V, E)$ with edge costs $c(e)$ and hop-constraint h denote the given input. Recall that our goal is to find a subgraph $F \subseteq E$ minimizing $c(F)$ in which each demand pair is connected with a path of hop-length at most h.

Algorithm 2 (α, β)-approximation for Hop Constrained Network Design.

$F \leftarrow \emptyset$
$\mathcal{T} \leftarrow \mathcal{D}^{(1)}$-router on G with capacities x_e, exclusion prob $\varepsilon = \frac{1}{4(h+1)}$
for $j = 1, \ldots, \tau$ **do**
  Sample $(T, M) \sim \mathcal{T}$, with induced capacities y as defined in Section 2
  $F'_j \subseteq E(T) \leftarrow$ Some collection of tree paths
  $F_j \leftarrow \cup_{e' \in F'_j} M_{e'}$, this is the mapping of F'_j to the input graph G
  $F \leftarrow F \cup F_j$
return F

We employ a tree based rounding scheme described in Algorithm 2. The algorithm solves **LP-HC**, obtains a fractional solution (x, f), and considers a $\mathcal{D}^{(1)}$-router with capacities x_e and exclusion probability $\varepsilon = O(1/h)$. It repeatedly samples partial tree embeddings and buys tree paths through some randomized process to be described separately for each problem.

▶ **Lemma 13.** *Suppose that in every iteration j, the probability that $e' \in E(T)$ is added to F_j' is at most $\gamma \cdot y(e')$. Then the total cost of Algorithm 2 is at most $O(\gamma\tau \log^2 n)\sum_{e\in E} c(e)x_e$.*

We show that the tree T preserves S_i-T_i flow with good probability. Since set demand pairs generalize vertex demand pairs, the following lemma holds for all settings.

▶ **Lemma 14.** *Fix a demand pair $S_i, T_i \subseteq V$ and an iteration $j \in \tau$. With probability at least $1/2$, y supports a flow of at least $1/2$ between S_i and T_i on T.*

Proof. By construction, y supports at least as much flow between S_i and T_i on T as x does on $G[V(T)]$, where $G[V(T)]$ denotes the induced subgraph of G by $V(T)$. Let $z_e = \sum_{p\in\mathcal{P}_i^h, V(p)\subseteq V(T), e\in p} f_p$. Since x, f is a feasible solution to **LP-HC**, $x_e \geq z_e$ for all $e \in E$, so we restrict our attention to flow supported by z. Let Z denote the total S_i-T_i flow supported by z, i.e. $Z = \sum_{p\in\mathcal{P}_i^h, V(p)\subseteq V(T)} f_p$. It suffices to show that $Z \geq \frac{1}{2}$ with probability at least $\frac{1}{2}$.

First, note that for all $p \in \mathcal{P}_i^h$, $\text{hop}(p) \leq h$, so $|V(p)| \leq h+1$. Thus by union bound, $\mathbf{Pr}[V(p) \not\subseteq V(T)] \leq \sum_{v\in V(p)} \mathbf{Pr}[v \notin V(T)] \leq \varepsilon(h+1)$. Therefore, $\mathbf{E}[Z] = \sum_{p\in\mathcal{P}_i} f_p \mathbf{Pr}[V(p) \subseteq V(T)] \geq \sum_{p\in\mathcal{P}_i} f_p(1-\varepsilon(h+1)) = (1-\varepsilon(h+1)) = \frac{3}{4}$. Since $\sum_{p\in\mathcal{P}_i} f_p = 1$ (constraint (2b) on **LP-HC**), $Z \in [0,1]$. Applying Markov's inequality on $1-Z$ gives $\mathbf{Pr}\left[Z \leq \frac{1}{2}\right] = \mathbf{Pr}[1-Z \geq \frac{1}{2}] \leq 2\,\mathbf{E}[1-Z] \leq \frac{1}{2}$. Thus $Z \geq \frac{1}{2}$ with probability at least $\frac{1}{2}$. ◀

Hop-Constrained Steiner Forest. We let $s_i, t_i \subseteq V$, $i \in [r]$ denote the terminal pairs. We follow Algorithm 2 with $\tau = \log_2(2r)$. For each iteration $j \in [\tau]$, for each $i \in [r]$, if y supports a flow of at least $\frac{1}{2}$ between s_i and t_i, we set $F_j' \leftarrow F_j' \cup P_T(s_i, t_i)$. It is a simple consequence of Lemma 14 that for each $i \in [r]$, $j \in [\tau]$, F_j contains an s_i-t_i path of length at most $O(\log^3 n)h$ with probability at least $\frac{1}{2}$; thus repeating and applying a union bound over all terminal pairs gives us the desired hop factor. The cost factor follows immediately from Lemma 13 with $\gamma = 2$.

Hop-Constrained Set Connectivity. Let $S_i, T_i \subseteq V$, $i \in [r]$ denote the demand pairs, In Set Connectivity, the fractional flow between S_i and T_i may be spread out amongst many vertex pairs $(s_i, t_i) \in S_i \times T_i$. Therefore, we cannot follow the same simple algorithm for Hop-Constrained Steiner Forest. Instead, we use the Oblivious Tree Rounding framework described in Section 2. In each iteration $j \in [\tau]$ of Algorithm 2, we let $F_j' = \text{TreeRounding}(T, y, \frac{1}{2})$. We set $\tau = \frac{2}{\phi}\log 2r$, where ϕ is the probability of success given by Lemma 10.

It is not difficult to see that for any $i \in [r]$, $j \in [\tau]$, F_j connects S_i and T_i with a path of hop-length at most $O(\log^3 n)$ with probability at least $\frac{\phi}{2}$; this follows from Lemma 14 and Oblivious Tree Rounding guarantees given by Lemma 10. We obtain the hop factor by repeating and applying a union bound over all terminal pairs. The cost factor follows from Lemma 13, since by Lemma 10, the probability any tree edge e' is included in F_j' is at most $O(\log^3 n)y(e')$.

4 Fault-Tolerant Hop-Constrained Network Design

In this section, we provide polylogarithmic bicriteria approximations for (h,k)-Fault Tolerant Hop-Constrained Network Design. We are given $G = (V, E)$ with edge costs $c(e)$ and terminal pairs s_i, t_i with connectivity requirements k_i. A solution $F \subseteq E$ is feasible if for every $i \in [r]$, for every subset $Q \subseteq E$ of less than k_i edges, $F \setminus Q$ contains need an s_i-t_i path of hop-length at most h. We modify the LP relaxation **LP-HC** for the fault-tolerant setting.

LP Relaxation. We define variables $f_p^Q\ \forall p \in \mathcal{P}_i^h, \forall Q \subseteq E, |Q| < k$, where f_p^Q is an indicator for whether the path p carries s_i-t_i flow that avoids edges in Q. We replace the first constraint in **LP-HC** with

$$\sum_{p \in \mathcal{P}_i^h, p \cap Q = \emptyset} f_p^Q = 1 \quad \forall i \in [r], \forall Q \subseteq E \text{ s.t. } |Q| < k_i.$$

We replace the second constraint in **LP-HC** with

$$\sum_{p \in \mathcal{P}_i^h, e \in p} f_p^Q \leq x_e \quad \forall i \in [r], \forall e \in E, \forall Q \subseteq E \text{ s.t. } |Q| < k_i.$$

To solve this LP, for each $Q \subseteq E$, consider the LP restricted to variables f^Q and constraints corresponding to Q. This is exactly **LP-HC**. Thus the polytope for Fault-HCND is the intersection of $\binom{m}{k} = n^{O(k)}$ polytopes which we can separate over by Remark 5.

In Section 4.1, we focus on the single-source setting and prove Theorem 3. In Section 4.2, we prove Theorem 4 via a reduction from multicommodity to single-source when $k = 2$. The results in this section rely on several non-trivial technical tools and ideas [27, 38, 20]. In this extended abstract we omit several details and refer the reader to the full version [23].

4.1 Single-Source Fault Tolerant HCND

We are given a source terminal s and sink terminals $t_1, \ldots, t_r$ with requirements k_i each. We will assume that $k_i = k$ for all i; this can be done with an additional factor of k in the cost approx factor by partitioning the set of terminals based on k_i and considering each set independently.

We employ the *augmentation framework*. For $\ell \in [k]$, we say a partial solution H_ℓ satisfies (h_ℓ, ℓ)-hop-connectivity if $\forall i \in [r]$, s and t_i are (h_ℓ, ℓ)-hop-connected. Our goal is to *augment* H to satisfy $(h_{\ell+1}, \ell + 1)$-hop connectivity for some $h_{\ell+1}$ proportional to h_ℓ. We modify the tree-rounding scheme given in Section 3.2 to solve the augmentation problem. We follow an approach introduced by [27] to scale down x_e values of existing edges and "large" edges, and then follow Algorithm 2 with these modified x_e values. Let $\sigma = O\left(\log n \cdot \log \frac{\log n}{\varepsilon}\right)$ denote the congestion of $\mathcal{D}^{(1)}$-routers, i.e. $\mathbf{E}\left[\sum_{u,v \in V(T)} \text{flow}(M_{uv}, e) x_{uv}\right] \leq \sigma x_e$. The algorithm to solve the augmentation problem is given in Algorithm 3.

We assume H_ℓ is minimal; i.e. for all $e \in H_\ell$, $H_\ell \setminus \{e\}$ does not satisfy (h_ℓ, ℓ)-hop-connectivity. This can be accomplished by removing each $e \in H_\ell$ and checking if $H_\ell \setminus \{e\}$ satisfies (h_ℓ, ℓ)-hop-connectivity. We also assume that if either endpoint of e has hop-distance greater than h to s, then $x_e = 0$. This assumption is true for all optimal fractional solutions, since x_e only needs to be nonzero if $e \in p$ for some $p \in \mathcal{P}_i^h$. We write H to denote $H_\ell \cup \text{LARGE}$. We let $\text{hop}_G(u, v) = \text{dist}_G(u, v)$ denote the minimum hop-length between two nodes in a graph G (we will drop G if it is clear from context). The diameter of $S \subseteq V$ is $\text{diam}(S) = \max_{u,v \in S} \text{hop}(u, v)$. When we discuss the diameter of the graph, we are referring to the set of all non-isolated vertices. For $S \subseteq V$, $F \subseteq E$, we let $\delta_F(S)$ denote the set of all edges in F with exactly one endpoint in S. It is not difficult to see that given the above assumptions, H is connected and $\text{diam}(H) \leq 2h_\ell$.

Algorithm 3 Augmentation Algorithm for Fault Tolerant Hop Constrained Network Design.

$(x, f) \leftarrow$ fractional solution to LP relaxation.
$\text{LARGE} \leftarrow \{e : x_e > \frac{1}{4\ell\sigma}\}$
$\text{SMALL} \leftarrow \{e : x_e \leq \frac{1}{4\ell\sigma}\}$
$\tilde{x}_e \leftarrow \begin{cases} x_e & e \in \text{SMALL} \\ \frac{1}{4\ell\sigma} & e \in H_\ell \cup \text{LARGE} \end{cases}$
$\mathcal{T} \leftarrow \mathcal{D}^{(1)}$-router on G with capacities $\tilde{x}_e$, exclusion prob $\varepsilon = \frac{1}{4(h+1)}$
for $j = 1, \ldots, \tau = O(\ell \log n)$ **do**
  Sample $(T, M) \sim \mathcal{T}$, with capacities y as defined in Section 2
  $F'_j \subseteq E(T) \leftarrow \text{TreeRounding}(T, y, \frac{1}{4})$
  $F \leftarrow F \cup \left(\bigcup_{e \in F'_j} M(e')\right)$
return $F \cup \text{LARGE}$

Analysis Overview. We say $Q \subseteq E$ is a *violating edge set* if $|Q| \leq \ell$ and there exists $i \in [r]$ such that $H \setminus Q$ does not contain a path of length h_ℓ from s to t_i. Since H satisfies (h_ℓ, ℓ)-hop connectivity, $|Q| = \ell$ for any violating edge set. To prove the correctness of Algorithm 3, we need to show that $H \cup F \setminus Q$ contains a short s-t_i path for every violating edge set Q and every terminal t_i. For the remainder of the analysis, we fix a violating edge set Q.

▷ Claim 15. $H \setminus Q$ has at most two connected components; both must contain a terminal.

Proof. We first show all components of $H \setminus Q$ contain a terminal. Suppose there exists a component C of $H \setminus Q$ that does not contain a terminal. Since H is connected, there exists some $e \in \delta_Q(C)$. Let $Q' = Q \setminus \{e\}$. Since H satisfies (h_ℓ, ℓ)-hop-connectivity, every terminal contains a path of hop length at most h_ℓ to s in $H \setminus Q'$. Since C contains no terminals and e is the only edge in $\delta_{Q'}(C)$, any s-t_i path for any $i \in [r]$ must traverse e an even number of times; thus any simple s-t_i path does not traverse e. Therefore for $i \in [r]$, any s-t_i path in $H \setminus Q'$ exists in $H \setminus Q$ as well, contradicting the fact that Q is violating.

Next, we show $H \setminus Q$ has at most 2 components. Suppose $H \setminus Q$ has at least three components C_1, C_2, C_3. Let $s \in C_1$. Since H is connected, there exists some $e \in \delta_Q(C_3)$. In particular, C_1 and C_2 remain disconnected in $H \setminus (Q \setminus \{e\})$. Since C_2 must contain a terminal, this contradicts the assumption that H satisfies (h_ℓ, ℓ)-hop-connectivity. ◁

Henceforth, we let C_s denote the connected component of $H \setminus Q$ containing s, and C_t the other component if it exists. The analysis of Algorithm 3 consists of two key parts, (1) bounding the diameter of C_s and C_t, and (2) showing that with good probability, there exists a round of TreeRounding that includes a path directly from C_s to C_t that avoids Q. We sketch the proofs of both parts.

(1) Diameter of Connected Components. In [28], Chung and Garey show that for any graph G with diameter d, if we remove ℓ edges and the resulting graph is connected, then its diameter is at most $d(\ell + 1) + O(\ell)$. We simplify their proof and generalize it to bound the diameter of components even when the resulting graph is no longer connected, with slightly looser bounds.

▶ **Lemma 16.** *All connected components of $H \setminus Q$ have diameter at most $O(\ell) \cdot h_\ell$.*

Proof. We prove the claim for C_s, as the argument for C_t is analogous. Let $d = \text{diam}(C_s)$, and let $u, v \in C_s$ be such that $\text{hop}(u, v) = d$. Let $V_i = \{w \in C_s : \text{hop}(u, w) = i\}$ for $i \in [d]$. Let C'_s be obtained from C_s by contracting each V_i into a single node v_i to form a path of length d and adding an isolated node t. Let H' be obtained from H by contracting each V_i into a node v_i and contracting C_t to a node t.

We will show that H' has diameter at least $\frac{d}{4\ell+2}$. Since C_s was constructed from H by deleting Q, H' can be constructed from C'_s by adding back some subset $Q' \subseteq Q$ of at most ℓ edges. Let $x_1, x_2, \ldots, x_j$ be the subset of vertices of C'_s that are incident to edges in Q', ordered in increasing distance from u in C'_s. Since $|Q'| \leq \ell$, $j \leq 2\ell$. We partition the path $C'_s \setminus \{t\}$ into $j+1$ segments: u to x_1, x_1 to x_2, …, x_{j-1} to x_j, x_j to v. The total hop-length of the path from u to v in C'_s is d, so there must be at least one segment of length $\geq \frac{d}{j+1}$. Since this segment does not have any edges in Q' (aside from the endpoints), the diameter of this segment in H' is at least $\frac{d}{2(j+1)}$. Thus $\text{diam}(H') \geq \frac{d}{2(j+1)} \geq \frac{d}{4\ell+2}$. Since H' was constructed by contracting edges in H, $\text{diam}(H) \geq \text{diam}(H')$. Thus $\frac{d}{4\ell+2} \leq \text{diam}(H') \leq \text{diam}(H) \leq 2h_\ell$, so $d = \text{diam}(C_s) \leq (8\ell + 4)h_\ell$. ◀

(2) Good flow on trees. We use a similar argument to that of Lemma 14 to show that in each iteration, y supports a flow of at least $\frac{1}{4}$ from C_s to C_t without using edges in $M^{-1}(Q)$ with constant probability. Crucially, we use the fact that Algorithm 3 scales down capacities x_e to $O(1/\ell\sigma)$ for all $e \in Q$; thus y cannot carry too much flow on $M^{-1}(Q)$. We can restrict attention to flow given by f_p^Q variables to obtain the desired bound.

▶ Remark 17. The main difficulty in extending this argument to the multicommodity setting lies in the fact that the components of H may have large diameter.

4.2 Multicommodity Fault-HCND when $k = 2$

We consider Fault-HCND in the multicommodity setting with $k = 2$. Recall that given terminal pairs s_i, t_i for $i \in [r]$, our goal is to obtain a subgraph $F \subseteq E$ such that for all $e \in F$, $i \in [r]$, $F \setminus \{e\}$ contains an s_i-t_i path with hop length at most h. We solve this via reduction to single-source and prove the following lemma:

▶ **Lemma 18.** *Suppose we are given an (α, β)-approximation for single-source $(h, 2)$-Fault-HCND with respect to OPT_{LP}. Then, there exists an $(O(\alpha \log^2 r), O(\beta \log r))$-approximation for $(h, 2)$-Fault-HCND in the general multicommodity setting with respect to OPT_I.*

Let $\mathcal{T} = \cup_{i\in[r]}\{s_i, t_i\}$ denote the set of all terminals (so $|\mathcal{T}| = 2r$). We employ the concept of a *junction structure* with root u and hop length h, which we define as a subgraph H in which for all terminals $t \in H \cap \mathcal{T}$, u and t are $(h, 2)$-hop-connected. The *density* of H is the ratio of its cost to the number of terminal pairs covered, i.e. $c(H)/|\{i : s_i, t_i \in H\}|$.

The goal of this section is to prove the following two lemmas: the first showing that a good junction structure exists, and the second providing an efficient algorithm to find it. Recall that OPT_I denotes the optimal *integral* solution to the given Fault-HCND instance.

▶ **Lemma 19.** *There exists a junction structure with hop length $4h(\log r + 1)$ and density $O(\frac{1}{r})\text{OPT}_I$.*

▶ **Lemma 20.** *Given an (α, β)-approximation for single-source $(h, 2)$-Fault-HCND with respect to OPT_{LP}, there exists an efficient algorithm to obtain a junction structure with hop length βh and density $O(\alpha \log r)$ times the optimal density of a junction structure with hop length h.*

We will outline the proof of Lemma 19 in Section 4.2.1. The proof of Lemma 20 uses standard techniques [20, 6]; we defer this to the full version [23]. By Lemmas 19 and 20 with hop length $h' = 4h\log(r+1)$, there exists an algorithm to obtain a junction structure of density at most $O\left(\frac{\alpha \log r}{r}\right) \mathrm{OPT}_I$. Lemma 18 follows by iteratively buying low-density junction structures until all terminals are covered.

▶ Remark 21. The single-source reduction to obtain a good junction structure can be extended to $k > 2$. The bottleneck for $k > 2$ is proving existence of a good junction structure.

4.2.1 Existence of good junction structure

Let E' denote an optimal solution to the instance of $(h,2)$-Fault-HCND, so $\mathrm{OPT}_I = c(E')$. We will partition E' into several junction structures, that, in total, cover a large fraction of the terminal pairs. For each $i \in [r]$, let $C_i \subseteq E'$ denote a minimal set of edges such that s_i and t_i are $(h,2)$-hop-connected. It is easy to verify that for all $i \in [r]$, C_i is 2-edge-connected and $\mathrm{diam}(C_i) \leq h$.

Let $B(u,\ell) = \{v \in G : \mathrm{hop}_G(u,v) \leq \ell\}$, and let $B'(u,\ell)$ denote the maximal 2-edge-connected component of $G[B(u,\ell)]$ containing u. It is not difficult to see that for any $u \in V$ and $p \in \mathbb{N}$, $B'(u,ph)$ is a valid junction structure with root u and hop length $4hp$. Thus the difficulty lines in showing that such a structure has good density.

We say $B'(u,\ell)$ *captures* $i \in [r]$ if $C_i \subseteq B'(u,\ell)$ and *intersects* $i \in [r]$ if i is not captured but there is some node $v \in C_i \cap B'(u,\ell)$. The main technical lemma is the following:

▶ Lemma 22. *If $i \in [r]$ is intersected by $B'(u,hp)$, then i is captured by $B'(u,h(p+1))$.*

(Proof Sketch). We first show that $C_i \subseteq B(u,h(p+1))$. This is because i is intersected by $B'(u,hp)$, so there exists some $v \in C_i \cap B'(u,hp)$. Since $\mathrm{diam}(C_i) \leq h$, all nodes in C_i are within hop-length h of v, and it is simple to see that v is within hop-length ph of s. Thus every node in C_i has hop-distance at most $h+ph$ to s. Showing that $C_i \subseteq B'(u,h(p+1))$ follows from transitivity of 2-edge-connectivity along with the fact that C_i and $B'(u,h(p+1))$ are each 2-edge-connected. ◀

As a corollary, there exists some $p \leq \log r + 1$ for which $B'(u,hp)$ captures at least as many pairs as it intersects. To construct the desired junction structures, let $H_1^* = B'(s,hp)$ be such a structure that captures at least as many pairs as it intersects. We remove all pairs that H_1^* captures or intersects, remove all edges in $B'(s,hp)$, and recurse on the remaining terminals and graph. Once all terminal pairs have been either captured or intersected, we obtain junction structures $H_1^*, H_2^*, \ldots$ with $\sum_j c(H_j^*) \leq c(E') = \mathrm{OPT}_I$. Since each iteration captures at least as many terminal pairs as it intersects, the total number of captured terminals is at least $\frac{r}{2}$. Thus there must be some H_τ^* that has density at most the average, which is $2\mathrm{OPT}_I/r$.

References

1 Ali Ahmadi, Iman Gholami, MohammadTaghi Hajiaghayi, Peyman Jabbarzade, and Mohammad Mahdavi. *2-Approximation for Prize-Collecting Steiner Forest*, pages 669–693. SIAM, 2024. doi:10.1137/1.9781611977912.25.

2 Noga Alon, Baruch Awerbuch, Yossi Azar, Niv Buchbinder, and Joseph Naor. A general approach to online network optimization problems. *ACM Transactions on Algorithms (TALG)*, 2(4):640–660, 2006.

3 Ernst Althaus, Stefan Funke, Sariel Har-Peled, Jochen Könemann, Edgar A. Ramos, and Martin Skutella. Approximating k-hop minimum-spanning trees. *Operations Research Letters*, 33(2):115–120, 2005. doi:10.1016/j.orl.2004.05.005.

4 M. Andrews. Hardness of buy-at-bulk network design. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 115–124, 2004. doi:10.1109/FOCS.2004.32.

5 Matthew Andrews and Lisa Zhang. The access network design problem. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, FOCS '98, page 40, USA, 1998. IEEE Computer Society.

6 Spyridon Antonakopoulos, Chandra Chekuri, Bruce Shepherd, and Lisa Zhang. Buy-at-bulk network design with protection. *Mathematics of Operations Research*, 36(1):71–87, 2011. doi:10.1287/moor.1110.0484.

7 Okan Arslan, Ola Jabali, and Gilbert Laporte. A flexible, natural formulation for the network design problem with vulnerability constraints. *INFORMS Journal on Computing*, 32(1):120–134, 2020. doi:10.1287/ijoc.2018.0869.

8 G. R. Ash, R. H. Cardwell, and R. P. Murray. Design and optimization of networks with dynamic routing. *The Bell System Technical Journal*, 60(8):1787–1820, 1981. doi:10.1002/j.1538-7305.1981.tb00297.x.

9 B. Awerbuch and Y. Azar. Buy-at-bulk network design. In *Proceedings 38th Annual Symposium on Foundations of Computer Science*, pages 542–547, 1997. doi:10.1109/SFCS.1997.646143.

10 Anantaram Balakrishnan and Kemal Altinkemer. Using a hop-constrained model to generate alternative communication network design. *ORSA Journal on Computing*, 4(2):192–205, 1992. doi:10.1287/ijoc.4.2.192.

11 Anantaram Balakrishnan and Christian Vad Karsten. Container shipping service selection and cargo routing with transshipment limits. *European Journal of Operational Research*, 263(2):652–663, 2017. doi:10.1016/j.ejor.2017.05.03.

12 Judit Bar-Ilan, Guy Kortsarz, and David Peleg. Generalized submodular cover problems and applications. *Theoretical Computer Science*, 250(1):179–200, 2001. doi:10.1016/S0304-3975(99)00130-9.

13 Yair Bartal. On approximating arbitrary metrices by tree metrics. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, pages 161–168, New York, NY, USA, 1998. Association for Computing Machinery. doi:10.1145/276698.276725.

14 Daniel Bienstock, Michel X Goemans, David Simchi-Levi, and David Williamson. A note on the prize collecting traveling salesman problem. *Mathematical programming*, 59(1-3):413–420, 1993.

15 Daniel Bienstock and Oktay Günlük. Capacitated network design—polyhedral structure and computation. *INFORMS Journal on Computing*, 8(3):243–259, 1996. doi:10.1287/ijoc.8.3.243.

16 Quentin Botton, Bernard Fortz, Luis Gouveia, and Michael Poss. Benders decomposition for the hop-constrained survivable network design problem. *INFORMS Journal on Computing*, 25(1):13–26, 2013. doi:10.1287/ijoc.1110.0472.

17 Jarosław Byrka, Fabrizio Grandoni, Thomas Rothvoß, and Laura Sanità. Steiner tree approximation via iterative randomized rounding. *Journal of the ACM (JACM)*, 60(1):1–33, 2013.

18 Parinya Chalermsook, Fabrizio Grandoni, and Bundit Laekhanukit. *On Survivable Set Connectivity*, pages 25–36. SIAM, 2015. doi:10.1137/1.9781611973730.3.

19 Moses Charikar and Adriana Karagiozova. On non-uniform multicommodity buy-at-bulk network design. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 176–182, New York, NY, USA, 2005. Association for Computing Machinery. doi:10.1145/1060590.1060617.

20 C. Chekuri, M. T. Hajiaghayi, G. Kortsarz, and M. R. Salavatipour. Approximation algorithms for nonuniform buy-at-bulk network design. *SIAM Journal on Computing*, 39(5):1772–1798, 2010. doi:10.1137/090750317.

21 Chandra Chekuri, Guy Even, Anupam Gupta, and Danny Segev. Set connectivity problems in undirected graphs and the directed steiner network problem. *ACM Trans. Algorithms*, 7(2), March 2011. doi:10.1145/1921659.1921664.

22 Chandra Chekuri and Rhea Jain. Approximation Algorithms for Network Design in Non-Uniform Fault Models. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming (ICALP 2023)*, volume 261 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 36:1–36:20, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ICALP.2023.36.

23 Chandra Chekuri and Rhea Jain. Approximation algorithms for hop constrained and buy-at-bulk network design via hop constrained oblivious routing, 2024. arXiv:2404.16725.

24 Chandra Chekuri, Sanjeev Khanna, and Joseph (Seffi) Naor. A deterministic algorithm for the cost-distance problem. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '01, pages 232–233, USA, 2001. Society for Industrial and Applied Mathematics.

25 Chandra Chekuri and Nitish Korula. Single-sink network design with vertex connectivity requirements. In Ramesh Hariharan, Madhavan Mukund, and V Vinay, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 2 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 131–142, Dagstuhl, Germany, 2008. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.FSTTCS.2008.1747.

26 Chandra Chekuri and Martin Pal. A recursive greedy algorithm for walks in directed graphs. In *46th annual IEEE symposium on foundations of computer science (FOCS'05)*, pages 245–253. IEEE, 2005.

27 Q. Chen, B. Laekhanukit, C. Liao, and Y. Zhang. Survivable network design revisited: Group-connectivity. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 278–289, Los Alamitos, CA, USA, November 2022. IEEE Computer Society. doi:10.1109/FOCS54457.2022.00033.

28 F. R. K. Chung and M. R. Garey. Diameter bounds for altered graphs. *Journal of Graph Theory*, 8(4):511–534, 1984. doi:10.1002/jgt.3190080408.

29 Julia Chuzhoy, Anupam Gupta, Joseph (Seffi) Naor, and Amitabh Sinha. On the approximability of some network design problems. *ACM Trans. Algorithms*, 4(2), May 2008. doi:10.1145/1361192.1361200.

30 Geir Dahl. The 2-hop spanning tree problem. *Operations Research Letters*, 23(1):21–26, 1998. doi:10.1016/S0167-6377(98)00029-7.

31 Jérôme De Boeck and Bernard Fortz. Extended formulation for hop constrained distribution network configuration problems. *European Journal of Operational Research*, 265(2):488–502, 2018. doi:10.1016/j.ejor.2017.08.01.

32 I. Diarrassouba, V. Gabrel, A. R. Mahjoub, L. Gouveia, and P. Pesneau. Integer programming formulations for the k-edge-connected 3-hop-constrained network design problem. *Networks*, 67(2):148–169, 2016. doi:10.1002/net.21667.

33 I. Diarrassouba, A. R. Mahjoub, and I. M. Almudahka. Optimization algorithms for the k edge-connected l-hop-constrained network design problem. *Soft Computing*, February 2024. doi:10.1007/s00500-023-09541-7.

34 Michael Dinitz, Guy Kortsarz, and Ran Raz. Label cover instances with large girth and the hardness of approximating basic k-spanner. *ACM Trans. Algorithms*, 12(2), December 2016. doi:10.1145/2818375.

35 Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *Journal of Computer and System Sciences*, 69(3):485–497, 2004. Special Issue on STOC 2003. doi:10.1016/j.jcss.2004.04.011.

36 A. Filtser. Hop-constrained metric embeddings and their applications. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 492–503, Los Alamitos, CA, USA, February 2022. IEEE Computer Society. doi:10.1109/FOCS52979.2021.00056.

37 Naveen Garg, Goran Konjevod, and Ramamoorthi Ravi. A polylogarithmic approximation algorithm for the group steiner tree problem. *Journal of Algorithms*, 37(1):66–84, 2000. Preliminary version in Proc. of ACM-SIAM SODA 1998.

38 Mohsen Ghaffari, Bernhard Haeupler, and Goran Zuzic. Hop-constrained oblivious routing. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2021, pages 1208–1220, New York, NY, USA, 2021. Association for Computing Machinery. doi:10.1145/3406325.3451098.

39 Rohan Ghuge and Viswanath Nagarajan. Quasi-polynomial algorithms for submodular tree orienteering and directed network design problems. *Mathematics of Operations Research*, 47(2):1612–1630, 2022.

40 L. Gouveia, P. Patricio, A.F. de Sousa, and R. Valadas. Mpls over wdm network design with packet level qos constraints based on ilp models. In *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428)*, volume 1, pages 576–586 vol.1, 2003. doi:10.1109/INFCOM.2003.1208708.

41 Luis Gouveia, Martim Joyce-Moniz, and Markus Leitner. Branch-and-cut methods for the network design problem with vulnerability constraints. *Computers & Operations Research*, 91:190–208, 2018.

42 Luis Gouveia and Markus Leitner. Design of survivable networks with vulnerability constraints. *European Journal of Operational Research*, 258(1):89–103, 2017. doi:10.1016/j.ejor.2016.09.003.

43 Fabrizio Grandoni and Giuseppe F. Italiano. Improved approximation for single-sink buy-at-bulk. In Tetsuo Asano, editor, *Algorithms and Computation*, pages 111–120, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

44 Fabrizio Grandoni, Bundit Laekhanukit, and Shi Li. O (log2 k/log log k)-approximation algorithm for directed steiner tree: a tight quasi-polynomial-time algorithm. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 253–264, 2019.

45 Sudipto Guha, Adam Meyerson, and Kamesh Munagala. A constant factor approximation for the single sink edge installation problems. In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, STOC '01, pages 383–388, New York, NY, USA, 2001. Association for Computing Machinery. doi:10.1145/380752.380827.

46 Anupam Gupta, Ravishankar Krishnaswamy, and R. Ravi. Tree embeddings for two-edge-connected network design. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '10, pages 1521–1538, USA, 2010. Society for Industrial and Applied Mathematics.

47 Anupam Gupta, Amit Kumar, and Tim Roughgarden. Simpler and better approximation algorithms for network design. In *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 365–372, New York, NY, USA, 2003. Association for Computing Machinery. doi:10.1145/780542.780597.

48 Anupam Gupta, Viswanath Nagarajan, and R Ravi. An improved approximation algorithm for requirement cut. *Operations Research Letters*, 38(4):322–325, 2010.

49 Bernhard Haeupler, D. Ellis Hershkowitz, and Goran Zuzic. Tree embeddings for hop-constrained network design. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2021, pages 356–369, New York, NY, USA, 2021. Association for Computing Machinery. doi:10.1145/3406325.3451053.

50 Mohammad Taghi Hajiaghayi, Guy Kortsarz, and Mohammad R Salavatipour. Approximating buy-at-bulk and shallow-light k-steiner trees. *Algorithmica*, 53:89–103, 2009.

51 Eran Halperin, Guy Kortsarz, Robert Krauthgamer, Aravind Srinivasan, and Nan Wang. Integrality ratio for group steiner trees and directed steiner trees. *SIAM Journal on Computing*, 36(5):1494–1511, 2007.

52 Eran Halperin and Robert Krauthgamer. Polylogarithmic inapproximability. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 585–594, 2003.

53 David Huygens, Ali Ridha Mahjoub, and Pierre Pesneau. Two edge-disjoint hop-constrained paths and polyhedra. *SIAM Journal on Discrete Mathematics*, 18(2):287–312, 2004. doi:10.1137/S0895480102419445.

54 K. Jain. A factor 2 approximation algorithm for the generalized Steiner network problem. *Combinatorica*, 21(1):39–60, 2001.

55 Erez Kantor and David Peleg. Approximate hierarchical facility location and applications to the shallow steiner tree and range assignment problems. In Tiziana Calamoneri, Irene Finocchi, and Giuseppe F. Italiano, editors, *Algorithms and Complexity*, pages 211–222, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

56 M. Reza Khani and Mohammad R. Salavatipour. Improved approximations for buy-at-bulk and shallow-light k-steiner trees and $(k, 2)$-subgraph. *J. Comb. Optim.*, 31(2):669–685, February 2016. doi:10.1007/s10878-014-9774-5.

57 Jochen Könemann, Asaf Levin, and Amitabh Sinha. Approximating the degree-bounded minimum diameter spanning tree problem. *Algorithmica*, 41:117–129, 2005.

58 Guy Kortsarz and Zeev Nutov. Approximating some network design problems with node costs. *Theoretical Computer Science*, 412(35):4482–4492, 2011. doi:10.1016/j.tcs.2011.04.013.

59 Guy Kortsarz and David Peleg. Approximating shallow-light trees. In *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '97, pages 103–110, USA, 1997. Society for Industrial and Applied Mathematics.

60 Larry J. LeBlanc, Jerome Chifflet, and Philippe Mahey. Packet routing in telecommunication networks with path and flow restrictions. *INFORMS Journal on Computing*, 11(2):188–197, 1999. doi:10.1287/ijoc.11.2.188.

61 Serge Lhomme. Vulnerability and resilience of ports and maritime networks to cascading failures and targeted attacks. In Routledge, editor, *Maritime Networks. Spatial Structures and Time Dynamic.* Routledge, October 2015. URL: https://hal.science/hal-01275157.

62 A. Ridha Mahjoub, Michael Poss, Luidi Simonetti, and Eduardo Uchoa. Distance transformation for network design problems. *SIAM Journal on Optimization*, 29(2):1687–1713, 2019. doi:10.1137/16M1108261.

63 A. Ridha Mahjoub, Luidi Simonetti, and Eduardo Uchoa. Hop-level flow formulation for the survivable network design with hop constraints problem. *Networks*, 61(2):171–179, 2013. doi:10.1002/net.21483.

64 Madhav V Marathe, R Ravi, Ravi Sundaram, S.S Ravi, Daniel J Rosenkrantz, and Harry B Hunt. Bicriteria network design problems. *Journal of Algorithms*, 28(1):142–171, 1998. doi:10.1006/jagm.1998.0930.

65 Adam Meyerson, Kamesh Munagala, and Serge Plotkin. Cost-distance: Two metric network design. *SIAM Journal on Computing*, 38(4):1648–1659, 2008. doi:10.1137/050629665.

66 C. Monma and Diane Sheng. Backbone network design and performance analysis: A methodology for packet switching networks. *IEEE Journal on Selected Areas in Communications*, 4(6):946–965, 1986. doi:10.1109/JSAC.1986.1146400.

67 Hasan Pirkul and Samit Soni. New formulations and solution procedures for the hop constrained network design problem. *European Journal of Operational Research*, 148(1):126–140, 2003. doi:10.1016/S0377-2217(02)00366-1.

68 Harald Räcke. Optimal hierarchical decompositions for congestion minimization in networks. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, STOC '08, pages 255–264, New York, NY, USA, 2008. Association for Computing Machinery. doi:10.1145/1374376.1374415.

69 R. Ravi. Rapid rumor ramification: approximating the minimum broadcast time. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 202–213, 1994. doi:10.1109/SFCS.1994.365693.

70 Gabriele Reich and Peter Widmayer. Beyond steiner's problem: A vlsi oriented generalization. In *International Workshop on Graph-theoretic Concepts in Computer Science*, pages 196–210. Springer, 1989.

71 André Rossi, Alexis Aubry, and Mireille Jacomino. Connectivity-and-hop-constrained design of electricity distribution networks. *European Journal of Operational Research*, 218(1):48–57, 2012. doi:10.1016/j.ejor.2011.10.006.

72 B. Rothfarb and M. Goldstein. The one-terminal telpak problem. *Operations Research*, 19(1):156–169, 1971. doi:10.1287/opre.19.1.156.

73 F Sibel Salman, Joseph Cheriyan, R Ravi, and Sairam Subramanian. Buy-at-bulk network design: Approximating the single-sink edge installation problem. In *Proceedings of the eighth annual ACM-SIAM symposium on Discrete algorithms*, pages 619–628, 1997.

74 Kunal Talwar. The single-sink buy-at-bulk lp has constant integrality gap. In William J. Cook and Andreas S. Schulz, editors, *Integer Programming and Combinatorial Optimization*, pages 475–486, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

75 Kathleen A. Woolston and Susan L. Albin. The design of centralized networks with reliability and availability constraints. *Computers and Operations Research*, 15(3):207–217, 1988. doi:10.1016/0305-0548(88)90033-0.

76 İbrahim Akgün and Barbaros Ç. Tansel. New formulations of the hop-constrained minimum spanning tree problem via miller-tucker-zemlin constraints. *European Journal of Operational Research*, 212(2):263–276, 2011. doi:10.1016/j.ejor.2011.01.051.