Approximating the Single-Sink Link Installation Problem in Network Design*

F. S. Salman † *J. Cheriyan* ‡ *R. Ravi* § *S. Subramanian* ¶

May 1997, revised October 1998

Abstract

We initiate the algorithmic study of an important but NP-hard problem that arises commonly in network design. The input consists of

(1) An undirected graph with one sink node and multiple source nodes, a specified length for each edge, and a specified demand, dem_v, for each source node v.

(2) A small set of cable types, where each cable type is specified by its capacity and its cost per unit length. The cost per unit capacity per unit length of a high-capacity cable may be significantly less than that of a low-capacity cable, reflecting an *economy of scale*, i.e., the payoff for buying at bulk may be very high.

The goal is to design a minimum-cost network that can (simultaneously) route all the demands at the sources to the sink, by installing zero or more copies of each cable type on each edge of the graph. An additional restriction is that the demand of each source must follow a single path. The problem is to find a route from each source node to the sink and to assign capacity to each edge of the network such that the total costs of cables installed is minimized. We call this problem the single-sink link-installation problem.

For the general problem, we introduce a new "moat type" lower bound on the optimal value and we prove a useful structural property of near-optimal solutions: For every instance of our problem, there is a near-optimal solution whose graph is acyclic (with cost no more than twice the optimal cost). We present efficient approximation algorithms for key special cases of the problem that arise in practice. For points in the Euclidean plane, we give an approximation algorithm with performance guarantee $O(\log(D/u_1))$, where D is the total demand and u_1 is the smallest cable capacity. When the metric is arbitrary, we consider the case where the network to be designed is restricted to be *two-level*, i.e. every source-sink path has at most two edges. For this problem, we present an algorithm with performance guarantee $O(\log n)$, where n is the number of nodes in the input graph, and also show that this performance guarantee is nearly best possible.

Keywords: Minimum-cost Capacitated Network Design, Approximation Algorithms
AMS Subject Classification: 68Q25, 68R10, 90B10, 90B12

*A preliminary version of this paper appeared in the *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA '97)*, 619-628 (1997).

†GSIA, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213. Supported in part by an IBM Corporate Fellowship. Email: `fs2c+@andrew.cmu.edu`

‡Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. Supported in part by NSERC grant no. OGP0138432 (NSERC code OGPIN 007). Email: `jcheriyan@dragon.uwaterloo.ca`

§GSIA, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213. Supported in part by an NSF CAREER grant CCR-9625297. Email: `ravi+@andrew.cmu.edu`

¶Manager, Wireless Technology Group, Northern Telecom, Dallas TX 70007. Email: `sai@nortel.com`

1 Introduction

1.1 The problem

An oil company wishes to construct a network of pipelines to carry oil from several remote wells to a major refinery. For each edge of the network, the company can install either zero or more copies of a cheap but thin pipe (say, the diameter is 10 inches and the cost is $1,000 per mile) or zero or more copies of a more expensive but thicker pipe (say, the diameter is 100 inches and the cost is $2,000 per mile). The demand (actually, oil supply) at each of the oil wells is given. The goal is to build a minimum-cost network that has sufficient capacity at every edge to transport the oil to the refinery.

Notice one feature of the problem: The cost per unit length versus capacity (available by combination of different pipe types) is a staircase function, reflecting an *economy of scale*. Also, note that several copies of several pipe types may be used in parallel to accommodate the flow on one edge of the network.

The above network design problem is NP-hard. There are reductions from both the Steiner tree problem and the knapsack problem (see Subsection 2.2). These reductions suggest two inherent sources of hardness of our problem. One is the connectivity requirement – our problem is NP-hard even when only one cable type is available. The second is the choice of cables – the problem is NP-hard even on a graph consisting of a single edge. While there are several known approximation algorithms that attack NP-hard minimum-cost connectivity problems [AKR 95, GW 95, WGM+95], to the best of our knowledge, there is no (previous) approximation algorithm that considers costs based on the choice of different cable types. Our work gives the first results on this topic.

Our problem of designing a single-sink multi-source network at minimum cost is a fundamental and economically significant problem that arises in hierarchical design of telecommunication networks. In the lowest level of network design, switching centers (controllers) collect calls from customers (base cells) and in the next level traffic goes between pairs of controllers. Once a set of customers are assigned to a switching center, the single-sink, multi-source problem arises. An additional constraint on the telecommunication problem is that the flow of traffic for any demand must follow a single path to the sink in the network [BMW 95] – this arises from limitations on the capacity of routing tables at nodes, and in avoiding complex switching hardware to support bifurcating flow. We call such flow routes *indivisible*[1]. The availability of a small number of cables, strong economies of scale, and the large number of customers are characteristic to the problems in the telecommunication industry.

1.2 Our results

We start by showing NP-hardness of two very simple versions of our problem (Section 2.2). Next we formalize two known lower bounds and use them to derive a simple constant factor approximation algorithm for the case with a single cable type using existing ideas.

We continue by proving a structure theorem (Theorem 3.2): For every instance of our problem, there is a near-optimal solution whose graph is acyclic (the cost is no more than two times the optimal value).

The case that appears to be most relevant in the goegraphic instances of network design is when the graph is defined by points in the two-dimensional Euclidean plane. For this case, we present

[1] also called unsplittable [Kle 96] or non-bifurcated [Bar 96] in the literature.

an approximation algorithm with performance guarantee $O(\log(D/u_1))$, where D denotes the total demand and u_1 denotes the smallest cable capacity (Theorem 4.1). The analysis of the performance guarantee hinges on a new "moat type" lower bound on the optimal value that we introduce, which is valid also for the general metric case.

In the general case, when the metric is arbitrary, we focus on a restricted version of the problem: Instead of allowing the optimal solution to induce an arbitrary graph we restrict the graph to be *two-level*, i.e. every source-sink path has at most two edges. For this problem, we present an algorithm with nearly best-possible performance guarantee $O(\log n)$, where n is the number of nodes (Theorem 5.2).

1.3 Previous work

The problem we address arises commonly in practical network design, and has been widely studied in the Operations Research literature. One of the first papers on routing flows under a staircase cost function arises from the *Telepak* problem in network design [GR 71]. Specializations of this problem include the fixed charge problem [Bal 61, G71] while generalizations include the minimum concave cost flow problem[Z68, GSS80]. In a survey on network synthesis and design problem, Minoux [Min 89] discusses several variants of the problem and exact solution methods. This body of work does not enforce the indivisible flow constraint.

Balakrishnan, Magnanti and Wong [BMW 95] address the problem of expanding an existing telecommunication network, where they have the indivisible flow restriction. Magnanti, Mirchandani and Vachani [MMV 95] study the polyhedral and computational aspects of the design problem with two cables. One feature they highlight is the large gap between heuristic solutions and Lagrangian lower bounds. Due to this, even small instances of the problem cannot be solved to anywhere near optimality by state-of-the-art computational techniques. More recent study of our network design problem with multiple sinks is undertaken in [BG 96, BCGT 98, Bar 96]. These papers develop cutting plane methods by exploiting classes of valid inequalities for an appropriate formulation that only considers one or two cable types.

Mansour and Peleg [MP 94] have results on a variant of our problem. In their model, there are multiple sinks and multiple sources, there is *only one* type of cable, and installing an edge has a fixed cost (similar to our model) as well as a variable cost per unit flow. By applying light-weight distance-preserving spanners [ADD+92], they obtain an $O(\log n)$-approximation algorithm for their network design problem with n nodes. It is easy to apply the method of Mansour and Peleg to the case with only one sink, and *only one* cable type, and to improve the logarithmic approximation to a constant-factor approximation (Subsection 2.3).

2 Preliminaries

2.1 Formalizing the problem

We are given an underlying undirected graph $G = (V, E), |V| = n$. A subset S of nodes is specified as sources of traffic and a single sink t is specified. Each source node $s_i \in S$ has an integer-valued demand dem_i. All the traffic of the source set is to be routed to t. The edges of G have lengths $\ell : E \to \mathbb{R}^+$. Without loss of generality, we assume that for every pair of nodes v, w, we can use the shortest-path distance $dist(v, w)$ as the length of the edge between v and w, i.e., we take the metric completion of the given graph. The edges of the network must be installed by purchasing one or more copies from among a small set of cables, where each cable type $i \in \{1, \ldots, q\}$ has a specified capacity u_i and a specified cost c_i per unit length. The indexing of the cables is such that $u_1 \leq u_2 \leq \ldots \leq u_q$, $c_1 \leq c_2 \leq \ldots \leq c_q$ and $c_1/u_1 \geq c_2/u_2 \geq \ldots \geq c_q/u_q$. Let $\sigma_{ij} := \dfrac{c_i/u_i}{c_j/u_j}$, $i \in \{1, \ldots, q-1\}$, $j > i$. Then, a type i cable is σ_{ij} times as expensive as a type j cable, per unit of capacity per unit of length. We refer to σ_{ij} as the "economy of scale" factor between the ith and the jth cable types.

A solution to our network design problem can be characterized by specifying for each source s_i, a path to t and a combination of cables to be used on each arc of the network induced by the paths. We will call the traffic of source s_i commodity i, $i = 1, \ldots, k$. Let P_i be the path for commodity i and let $N = (V_N, A)$ be the graph induced by the union of P_i, $i = 1, \ldots, k$.

Let f_e^i denote the amount of flow of commodity i on edge e of A and f_e denote the total flow on edge e. That is,

$$f_e^i = \begin{cases} dem_i, & if\ e \in P_i \\ 0, & \text{otherwise.} \end{cases}, \text{ and } \qquad f_e = \sum_{i=1}^{k} f_e^i.$$

Let $\nu_1^e, \ldots, \nu_q^e$ be the number of copies of cable types 1 to q to be installed on edge e, where q is the number of available cable types. ($\nu_i^e = 0$ implies that no cable of type i is installed on edge e).

Let $P = (P_1, \ldots, P_k)$ denote the routing and $\nu^e = (\nu_1^e, \ldots, \nu_q^e)$ denote the choice of cables on edge e that accommodates f_e (induced by P). Then, P and $\nu = (\nu^e, e \in A)$ characterize a feasible solution to the problem formulated below.

$$\begin{array}{ll} \min\limits_{P,\nu} & \sum\limits_{e \in A} \{dist(e) \sum\limits_{j=1}^{q} (c_j \cdot \nu_j^e)\} \\ s.t. & \\ & \sum\limits_{i: e \in P_i} dem_i \; - \; \sum\limits_{j=1}^{q} (u_j \cdot \nu_j^e) \; \leq \; 0, \quad e \in A \quad (1) \\ & \nu_j^e \in \{0, 1, 2, \ldots\}, \quad j \in \{1, \ldots, q\}, \quad e \in A \\ & P_i \text{ is a } s_i - t \text{ path}, \quad i = 1, \ldots, k \end{array}$$

Equivalently, the constraints (1) can be rewritten as:

$$f_e \; - \; \sum_{j=1}^{q} (u_j \cdot \nu_j^e) \; \leq \; 0, \quad e \in A \quad (2)$$

Note that the optimal choice of the routing depends on the choice of cables, as they determine the cost of edges. Yet, the optimal choice of cables on each edge depends on the amount of flow on each edge, which is determined by the routing decision. Hence, an optimal solution requires the decision of the routing and the cable choices simultaneously.

2.2 Hardness of the Problem

It is easy to see that the single-sink edge-installation problem is NP-hard. There are reductions from both the Steiner tree problem and the knapsack problem.

A special case of our problem when there is only one cable type with capacity large enough to hold all of the demand is equivalent to a Steiner tree problem with the sources and the sink as the terminal nodes. Hence, our problem is NP-hard [GJ 79] even when only one cable type is available with unlimited capacity. In this case, the problem is that of finding a minimum cost routing under fixed costs on the edges.

Another simple special case of our problem with one source node and a single edge is also

NP-hard. In this case, the problem reduces to finding the minimum cost choice of cables on the edge such that the total capacity of the cables covers the demand of the source. This problem is an integer min-knapsack problem with the additional economies of scale restrictions on data. The integer min-knapsack problem was shown to be NP-hard by Lueker [Lue 75] by a transformation from the subset sum problem. This transformation is still valid under the economies of scale restrictions.

2.3 Single Cable Type Case

In Subsection 1.3 we mentioned that a constant factor approximation can be obtained by applying the method of Mansour and Peleg [MP 94] to the case of single sink, and single cable type. In this subsection we give a proof of this claim. The idea is to route through a Light Approximate Shortest-Path Tree (LAST) [KRY 93].

Definition 2.1 ([KRY 93]) *Let G be a graph with non-negative edge lengths. A tree T rooted at vertex t is called an (α, β)-LAST if the following conditions are satisfied ($\alpha, \beta \geq 1$):*

1. *The distance of every vertex v from t in T is at most α times the distance between v and t in G.*
2. *The length of T is at most β times the length of an MST of G.*

Lemma 2.1 *Let C^* be the optimal cost for the network design problem with a single cable type. Let G' be a complete graph on node set $S \cup \{t\}$ where length of an edge is the shortest path length in G between the end points of the edge. Let T be an (α, β)-LAST of G' rooted at the sink node t and C_T be the cost of routing dem_i through the $s_i - t$ path in T for all i and using as many copies of the cable as necessary. Then, $C_T \leq (\alpha + 2\beta)C^*$.*

Proof: Let $dist_T(s_i, t)$ denote the length of the $s_i - t$ path P_i in T.

$$C_T = \sum_{e \in T} c_1 \lceil \tfrac{f_e}{u_1} \rceil \cdot dist_T(e) \leq \sum_{e \in T} c_1 \left(\frac{\sum_{i: e \in P_i} dem_i}{u_1} + 1 \right) \cdot dist_T(e)$$

$$= c_1 \sum_{s_i \in S} \sum_{e \in P_i} dist_T(e) \tfrac{dem_i}{u_1} + c_1 \sum_{e \in T} dist_T(e)$$

$$\leq \tfrac{c_1}{u_1} \sum_{s_i \in S} dist_T(s_i, t) dem_i + c_1 \sum_{e \in T} dist_T(e)$$

Since $dist_T(s_i, t) \leq \alpha \cdot dist(s_i, t)$ for all $s_i \in S$ and $\sum_{e \in T} dist_T(e) \leq \beta \ w(MST(G'))$, where $w(MST(G'))$ is the weight (sum of edge lengths) of an MST of G', we get

$$C_T \leq \alpha \frac{c_1}{u_1} \sum_{s_i \in S} dem_i \cdot dist(s_i, t) \ + \ \beta c_1 w(MST(G')).$$

The term $\frac{c_1}{u_1} \sum_{s_i \in S} \{dem_i \cdot dist(s_i, t)\}$ is a lower bound on C^* since dem_i must be routed a distance of at least $dist(s_i, t)$ and be charged at least at the rate c_1/u_1 per unit length. This lower bound is called the *routing lower bound*. In the general case when there are q cable types, c_1/u_1 is replaced by the cheapest rate c_q/u_q in the bound. In addition, $\frac{1}{2} c_1 \ w(MST(G'))$ is another lower bound on C^*, which is called the *MST lower bound*. The reasoning is as follows. We must connect the nodes in S to t and install at least one copy of the (cheapest) cable on each connecting edge.

Then, the cost of a Steiner tree with terminal set $S \cup \{t\}$ and cost $dist(e) \cdot c_1$ on each edge e is a lower bound and the length of an MST on $S \cup \{t\}$ times c_1 is within a factor 2 of this lower bound. These two lower bounds are essentially due to Mansour & Peleg [MP 94], though they study a different model. Thus, using the two lower bounds we get $C_T \leq (\alpha + 2\beta)C^*$. Note that to obtain a solution in the original graph G, edges of T are replaced with the corresponding shortest paths in G. Clearly, the new solution also has cost C_T. □

Constructing an $(\alpha, 1 + \frac{2}{\alpha-1})$-LAST as in [KRY 93] for $\alpha = 3$ gives the following corollary.

Corollary 2.2 *There is a 7-approximation algorithm for the single-sink edge installation problem with a single cable type.*

Note that when all the nodes in G except the sink node are source nodes, routing through an (α, β)-LAST of G gives an $(\alpha + \beta)$-approximation. Then, a $(2\sqrt{2} + 2)$-approximation is obtained using an $(\alpha, 1 + \frac{2}{\alpha-1})$-LAST of [KRY 93] for $\alpha = \sqrt{2} + 1$.

For the general case of multiple cable types, routing through an (α, β)-LAST and buying as many copies of the cheapest (i.e. the thinnest) cable type as necessary provides an approximate solution with a worst case bound of $(\alpha\sigma_{1q} + 2\beta)$ times the optimal cost (recall that σ_{1q} is the economies of scale factor between the thinnest and the thickest cables). However, in practice there are strong economies of scale between cable types. Hence, we focus on the case when σ_{1q} is large, possibly larger than poly-logarithmic in the number of nodes, i.e., $\sigma_{1q} > (\log n)^{\Omega(1)}$.

2.4 Multiple cable type case – An example where naive heuristics produce poor solutions

Here is an example to show that heuristics based on routing through an MST, a shortest paths tree or a LAST produce poor solutions. Suppose we have n source nodes $s_1, \ldots, s_n$ each with unit demand, at unit distance to each other and at distance $\sqrt{n}$ to the sink node t (see Figure 1a). There are only two types of cables T1 and T2, where a T1 cable has capacity $u_1 = 1$ and costs $c_1 = 1$ per unit length, whereas a T2 cable has capacity $u_2 = n$ and costs $c_2 = \sqrt{n}$ per unit length. (Note that $\sigma_{12} = \sqrt{n}$). An optimal solution with cost $2n - 1$ is obtained by installing a T2 cable for the edge (t, s_1), and using T1 cables to build a "star" centered at s_1 that has nodes $s_2, \ldots, s_n$ as leafs, i.e., by installing T1 cables on the edges $(s_1, s_2), \ldots, (s_1, s_n)$ (Figure 1b). A shortest paths tree (with root t) is a poor solution, since it has n edges of length $\sqrt{n}$, implying a cost of $n\sqrt{n}$, which is roughly $\sqrt{n}/2$ times the optimal cost (Figure 1c). An (arbitrary) minimum spanning tree is a poor solution: for example, the path $t, s_1, s_2, \ldots, s_n$ is a minimum spanning tree, and it requires at least $n - \sqrt{n}$ unit-length edges of capacity $\geq \sqrt{n}$ for a total cost $\geq n\sqrt{n}$, which is roughly $\sqrt{n}/2$ times the optimal cost (Figure 1d). (Though our optimal solution routes on a minimum spanning tree, this can be avoided by perturbing the distances.) Another heuristic is to use a spanner, or rather a light approximate shortest-paths tree (LAST). However, a LAST based on the previous minimum spanning tree, i.e. the path $t, s_1, s_2, \ldots, s_n$, turns out to be even costlier than the minimum spanning tree. One such LAST has edges $(t, s_{i\sqrt{n}})$ and paths $s_{i\sqrt{n}}, s_{i\sqrt{n}+1}, \ldots, s_{(i+1)\sqrt{n}-1}$ for $i \in \{1, 2, \ldots, \sqrt{n}\}$.

3 Structure of Solutions

It is possible to have a unique optimal solution such that the graph induced by edges with positive flow contains cycles (see Figure 2). However, we show in this section that there is a solution which

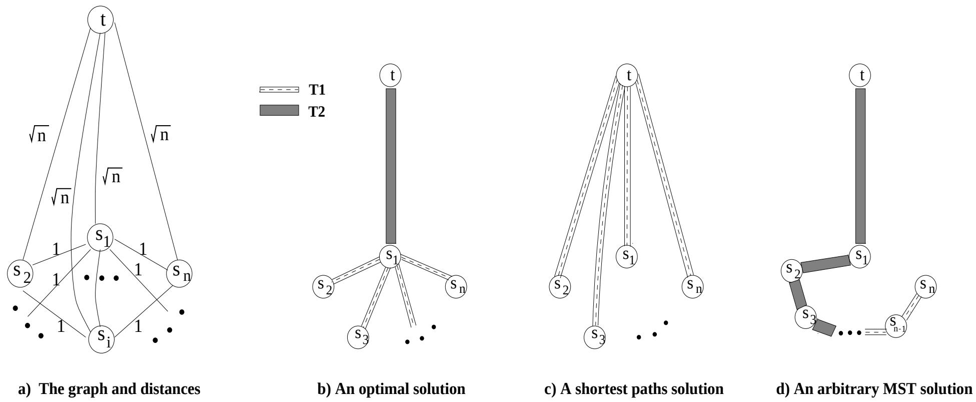


Figure 1: An example where naive heuristics produce poor solutions

induces a tree with cost at most twice the cost of an optimal solution. The key idea of the proof is to associate with every edge e chosen by an optimal solution an "adversary price" $\mathcal{C}_e$, where $\mathcal{C}_e$ is the length of e times the cheapest cost per unit capacity (per unit length) among all cable types installed on e by the optimal solution, and to route the traffic through the cheapest $s_i - t$ paths with respect to the adversary prices.

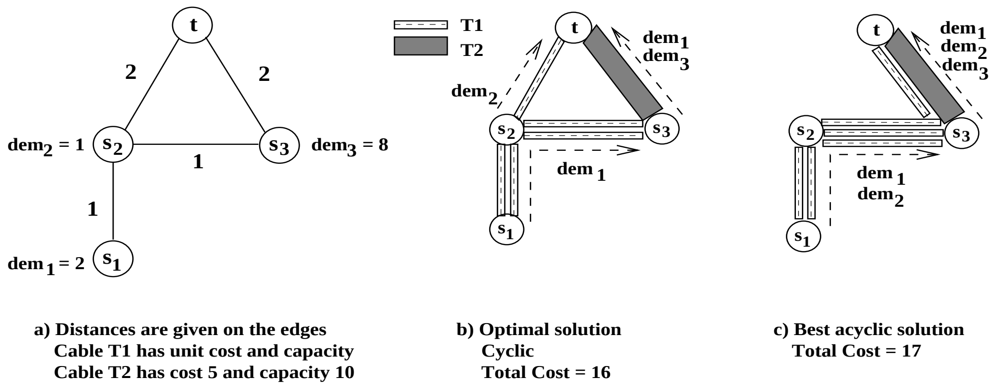


Figure 2: An example where the unique optimal solution is cyclic

Let (P^*, ν^*) be an optimal routing and choice of cables. Let $N^* = (V^*, A^*)$ be the graph induced by P^* and C^* be the cost of an optimal solution.

Lemma 3.1 *Let κ_e be the index of the thickest cable installed on edge e in ν^*. That is, $\kappa_e = \max\{j : \nu_j^{e*} > 0\}$. Associate a cost $\mathcal{C}_e$ for each edge e in A^*, where $\mathcal{C}_e := dist(e) \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}}$. Then, LB_1 defined below is a lower bound on the optimal cost, i.e.*

$$C^* \geq LB_1 := \sum_{i=1}^{k} \{dem_i \sum_{e \in P_i^*} \mathcal{C}_e\}$$

Proof: The lemma follows since LB_1 corresponds to the cost of a solution where every commodity flows along an optimal path using "cheapest per capacity" cables of an optimal solution fractionally.

More rigorously,

$$LB_1 = \sum_{i=1}^{k} \{dem_i \sum_{e \in P_i^*} (dist(e) \frac{c_{\kappa_e}}{u_{\kappa_e}})\} = \sum_{e \in A^*} \{dist(e) \cdot (\sum_{i=1}^{k} f_e^i) \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}}\} = \sum_{e \in A^*} \{dist(e) \cdot f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}}\}.$$

Let $k(1)$ to $k(l)$ be the indices of cables used in edge e in ν^*, where $k(l) = \kappa_e$. Without loss of generality we can allocate the flow induced by P^* on e, f_e, such that all but the thickest cable are saturated. Then, $f_e = u_{k(1)}\nu_{k(1)} + u_{k(2)}\nu_{k(2)} + \ldots + u_{k(l)}(\nu_{k(l)} - 1) + rem_e$, where we use $\nu_{k(i)}$ as a shorthand notation for $\nu_{k(i)}^{e*}$ and rem_e for the remaining flow in the unsaturated cable. Then,

$$f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} = \frac{u_{k(1)}}{u_{k(l)}} \cdot c_{k(l)}\nu_{k(1)} + \frac{u_{k(2)}}{u_{k(l)}} \cdot c_{k(l)}\nu_{k(2)} + \ldots + \frac{u_{k(l)}}{u_{k(l)}} \cdot c_{k(l)}(\nu_{k(l)} - 1) + \frac{rem_e}{u_{k(l)}} \cdot c_{k(l)}.$$

By economies of scale, $\frac{u_{k(i)}}{u_{k(l)}} \leq \frac{c_{k(i)}}{c_{k(l)}}$, $i = 1, \ldots, l-1$. Hence, replacing $\frac{u_{k(i)}}{u_{k(l)}}$ by $\frac{c_{k(i)}}{c_{k(l)}}$ for $i = 1, \ldots, l-1$ in the above equation gives

$$f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \leq c_{k(1)}\nu_{k(1)} + c_{k(2)}\nu_{k(2)} + \ldots + c_{k(l)}(\nu_{k(l)} - 1) + \frac{rem_e}{u_{k(l)}} \cdot c_{k(l)}.$$

Since $\frac{rem_e}{u_{k(l)}} \leq 1$, we have $f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \leq \sum_{i=1}^{l} c_{k(i)}\nu_{k(i)} = \sum_{j=1}^{q} c_j \nu_j^e$. Thus,

$$LB_1 = \sum_{e \in A^*} dist(e) \cdot f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \leq \sum_{e \in A^*} \{dist(e) \sum_{j=1}^{q} (c_j \cdot \nu_j^e)\} = C^*.$$

□

Now we are ready to prove the following theorem.

Theorem 3.2 *There exists a routing P which induces a tree T with cable choices ν such that the cost of this solution denoted by C_T satisfies $C_T \leq 2\,C^*$.*

Proof: Let T be a shortest path tree of N^* rooted at t with respect to costs $\mathcal{C}_e$ on $e \in A^*$. Route each commodity i through the unique path P_i from s_i to t in T. For each edge $e \in P$ use as many copies of cable κ_e as necessary, i.e. $\nu_{\kappa_e} = \lceil \frac{f_e}{u_{\kappa_e}} \rceil$ copies of cable κ_e. Then, the cost of this feasible solution is

$$C_T = \sum_{e \in T} dist(e) \cdot c_{\kappa_e} \cdot \nu_{\kappa_e}.$$

Since $\nu_{\kappa_e} \leq \frac{f_e}{u_{\kappa_e}} + 1$,

$$\begin{aligned} C_T &\leq \sum_{e \in T} dist(e) \cdot c_{\kappa_e} \cdot \frac{f_e}{u_{\kappa_e}} + \sum_{e \in T} dist(e) \cdot c_{\kappa_e} \\ &= \sum_{i=1}^{k} dem_i \sum_{e \in P_i} dist(e) \frac{c_{\kappa_e}}{u_{\kappa_e}} + \sum_{e \in T} dist(e) \cdot c_{\kappa_e}. \end{aligned}$$

As P_i's are shortest paths in N^* with costs $\mathcal{C}_e = dist(e)\frac{c_{\kappa_e}}{u_{\kappa_e}}$, $e \in A^*$, it follows that

$$C_T \leq \sum_{i=1}^{k} dem_i \sum_{e \in P_i^*} \mathcal{C}_e + \sum_{e \in T} dist(e) \cdot c_{\kappa_e}.$$

Therefore, the first summand above is at most LB_1 of Lemma 3.1. The second summand is also a lower bound on C^* since C^* includes the cost of at least one copy of the thickest cable on every edge in A^*, and $T \subset N^*$. Therefore, $C_T \leq 2C^*$. □

Theorem 3.2 motivates the problem of finding a minimum cost tree routing as an approximate solution to the general network design problem. However, note that the Steiner tree problem is a special case of the min cost tree routing problem. Thus, the minimum cost tree network design problem is also NP-hard.

4 Euclidean case - an approximation algorithm

In this section, we present an algorithm for our network design problem in the case when the nodes are represented as points in the plane and the length function is the (Euclidean) distance. We have the following theorem.

Theorem 4.1 *The Euclidean single-sink edge-installation problem can be approximated within a factor* $O(\min\{\log \frac{D}{u_1},\ \log \frac{\ell_{max}}{\ell_{min}}\})$, *where* D *is the total demand,* u_1 *is the capacity of the lowest-capacity cable,* ℓ_{max} *is the longest distance between a pair of nodes, and* ℓ_{min} *is the shortest distance between a pair of nodes.*

The approximation algorithm for the Euclidean case proceeds by by successively gridding the plane and constructing the network hierarchically. The performance ratio is proven by collecting several layers of "moat type" lower bounds and paying for the links laid in each layer of gridding using the lower bounds collected from that layer.

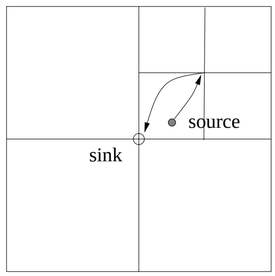
4.1 The algorithm

Let ℓ_{max} (ℓ_{min}) be the longest (shortest) distance between any pair of nodes. The topmost layer of gridding is a single square with minimum side length, centered at the sink node enclosing all the nodes, and hence has side length at most $2\ell_{max}$. We refine a square by partitioning it into four equal subsquares. We continue our refinement until every square in the lowest level of gridding either has side length at most $\ell_{max}/(D/u_1)$ (recall that D is the total demand) or contains at most one source. Thus, the number of layers of gridding used overall is $O(\min\{\log(D/u_1),\ \log(\ell_{max}/\ell_{min})\})$.

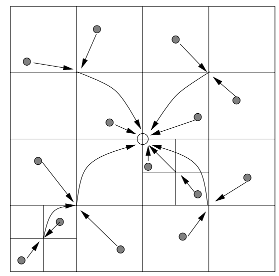
Based on this gridding, the construction of the network is done recursively by routing all the flow through the centers of the squares at any layer. That is, flow within a square is aggregated at its center. In a generic recursive step, suppose that we have a square of side length 2ℓ that contains points with total demand Dem. This demand can be partitioned into Dem_j for $j = 1, 2, 3, 4$ in each of the four subsquares of side length ℓ into which this square is divided. Assume that each of these demands has been already routed to the center of the subsquare where it arises. We now sketch how to route these demands to the center of the bigger square one level up in the gridding.

If $Dem_j = 0$, then we do not build any edges between the center of square j and the center of the bigger square. Suppose $u_i \leq Dem_j < u_{i+1}$, for some $i \in \{0, \cdots, q\}$. There are two cases. In the first case, $u_i \leq Dem_j < \rho_i u_i$. We then install $\lceil \frac{Dem_j}{u_i} \rceil$ copies of cable type i from the center of square j to the center of the big square. These cables have length $\frac{\ell}{\sqrt{2}}$. In the second case, $\rho_i u_i \leq Dem_j < u_{i+1}$ and we simply use a single copy of cable type $(i+1)$ to route the demand.

We have to be more careful in performing the recursive routing for demands that are near the sink. In particular, consider a demand that is very close to the sink in the northeastern quadrant of the first level of gridding. If this were the only demand, it is too expensive to route it to the center of the northeast square and then reroute it back to the sink (See Figure 3 (a)). We route the demand of any square with a corner at the sink directly to the sink. (See Figure 3 (b)). Thus, under this scheme, if a node v is at a distance d from the sink, its demand can be routed to the center of a square of side length at most $O(d)$.



(a)



(b)

Figure 3: Part (a) illustrates that care is required in routing from the squares closest to the sink. Part (b) illustrates the recursive routing strategy. If a square at level i does not contain the sink at one of its corners, then we route its demand to the center of the square at level $i-1$ enclosing it. The demand of a square that contains the sink at one of its corners is routed directly to the sink.

It is clear that the algorithm runs in polynomial time.

4.2 A moat lower bound

First we define a moat lower bound in its full generality and then apply it in the analysis of the above algorithm.

Consider a subset X of the node set V that excludes the sink node t. Let Dem be the total demand of the source nodes in X and let w be the minimum distance of any node in X to t, i.e. $w = \min_{x \in X} dist(x,t)$. A total of at least Dem flow has to travel a distance of at least w to reach the sink in any network. For any subset of nodes $X \subseteq V - \{t\}$, the ball around X of radius w defines a "moat" of width w separating demand Dem from the sink. The moat lower bound captures the cost of sending all the flow of value Dem together a distance of w even after utilizing the economies of buying at bulk.

Suppose that $u_i \leq Dem < u_{i+1}$, for some $i \in \{0, \ldots, q\}$, where we define $u_0 = 0$ and $u_{q+1} = \infty$ for convenience. Define the threshold multiplicity between the i^{th} and the $(i+1)^{st}$ cable as $\rho_i = \frac{c_{i+1}}{c_i}$, for $i = 1, \ldots, q$-1. In addition, let $\rho_0 = 1$ and $\rho_q = \infty$. If $u_i \leq Dem < \rho_i u_i$, for $i \in \{1, \ldots, q\}$, then a lower bound on the unitlength cable cost crossign this moat is $LB_K = \frac{Dem}{u_i} c_i$. The reason is that to pay at the cheaper rate $\frac{c_{i+1}}{u_{i+1}}$, we must buy an integral number of cables and buying a copy of cable type i+1 is costlier than paying $\frac{Dem}{u_i} c_i$. On the other hand, if $\rho_i u_i \leq Dem < u_{i+1}$, for $i \in \{0, \ldots, q-1\}$, then the lower bound is $LB_K = c_{i+1}$. As $\frac{Dem}{u_i} c_i \geq c_{i+1}$, buying any combination of cables 1 to i (by paying at least $\frac{Dem}{u_i} c_i$) will be more costly than a single cable of type i+1. We summarize the moat lower bound in the next proposition.

Proposition 4.2 *For any node set X excluding t, of total demand Dem, at minimum distance w*

to t*,* $LB_K \cdot w$ *is a lower bound on the optimal cost, where*

$$LB_K = \begin{cases} \frac{Dem}{u_i} c_i, & \textit{if } u_i \leq Dem < \rho_i u_i, \textit{ for some } i \in \{1, \ldots, q\} \\ c_{i+1}, & \textit{if } \rho_i u_i \leq Dem < u_{i+1}, \textit{ for some } i \in \{0, \ldots, q-1\}. \end{cases}$$

We can also collect lower bounds from several disjoint moats.

Proposition 4.3 *For any set of disjoint moats, the sum of the lower bounds generated by such moats is also a lower bound on the optimal value.*

Let us examine closely the definition of a moat in the Euclidean case. Let X be a set of nodes. A moat around X is defined by a closed line in R^2 which contains X in the inside and leaves the sink node outside. If the moat has width w, then the region of the plane occupied by the moat is the annular region including all points that are outside this line within Euclidean distance w from the line (Figure 4a). A collection of moats is *disjoint*, if the regions occupied by any two moats in the collection do not intersect (Figure 4b).

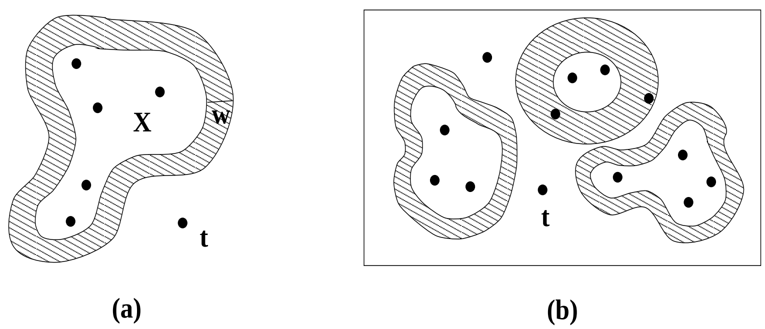


Figure 4: A moat around node set X of width w is shown in part (a). A disjoint moat collection is shown in part (b).

4.3 The performance ratio

We now bound the worst-case performance of our algorithm. For each layer of gridding, we use a disjoint collection of moats and bound the cost of cables installed at that layer using the lower bounds accumulated from these moats.

For a given layer i where each square has side length ℓ, we consider the moat of width ℓ around every square in this layer except the four squares closest to the sink. Of course, a square generates a nonzero lower bound only if it contains nonzero demand. We can split the moats of this layer into 9 classes of disjoint moat collections. One such class is shown in Figure 5. Other classes can be obtained by translating all the squares enclosed by the moats in this class by one square in one of the 8 neighboring directions (see Figure 5). Let LB_i be the maximum lower bound generated by any of these 9 classes of moats of layer i. By averaging, LB_i is at least 1/9 of the sum of the lower bounds generated by all squares in this layer.

The cost of the cables constructed in one layer of gridding is at most $\sqrt{2}$ times the lower bound generated by all squares in this layer. If a square has demand Dem_j and side length ℓ, the lower bound it generates (LB) is ℓ times LB_K. There are two corresponding cases in the

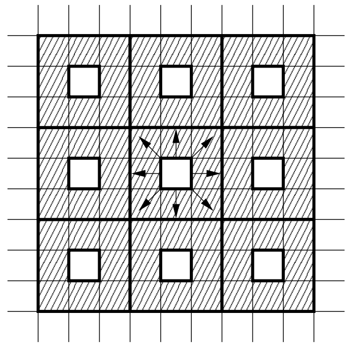


Figure 5: One class of disjoint moats in a given layer of gridding is shown. The arrows represent the 8 translation directions used to define the other 8 classes of this layer.

computation of LB_K and the choice of the cables in the algorithm. In the first case $Dem_j \geq u_i$. Thus, $\lceil \frac{Dem_j}{u_i} \rceil \leq 2(\frac{Dem_j}{u_i})$ so that unit length cost of cables used is within a factor 2 of the lower bound LB_K defined in Proposition 4.2. Since the cables have length $\ell/\sqrt{2}$, the cost is at most $\sqrt{2}\ell LB_K = \sqrt{2}LB$. In the second case, we incur unit length cost of c_{i+1}, which is equal to LB_K. Thus, the cost of cables used is $(\ell/\sqrt{2})LB_K$ which is less than LB. As a result, the cables installed at layer i have cost at most $9\sqrt{2}LB_i$.

Consider the last layer of gridding, with side-length at most $\ell_{max}u_1/D$. Note that the total cost incurred in routing all the demands to the centers of the squares in this layer is at most $O(\ell_{max}c_1)$. This is because the demand of each source s_i, dem_i, in each square of the last gridding has to be sent a distance of at most $O(1)$ times the side-length at unit length cost of at most $\lceil dem_i/u_1 \rceil c_1$. Since the MST lower bound is at least $1/2\ell_{max}c_1$, it is clear that the costs incurred in the last layer of gridding can be charged to the MST lower bound (to prevent further gridding).

The number of layers of gridding is $O(\min\{\log \frac{D}{u_1},\ \log \frac{\ell_{max}}{\ell_{min}}\})$, and the cost of the cables at each layer is at most $9\sqrt{2}$ times the optimal cost. Therefore, the cost of the solution constructed is within $O(\min\{\log \frac{D}{u_1},\ \log \frac{\ell_{max}}{\ell_{min}}\})$ times the optimal cost.

5 Two-level networks

In this section we consider a simpler version of our problem when the network in which the demand is to be routed is two-level, i.e. every source-sink path has at most two edges, and the distances arise from a general metric. This problem can be considered to limit the number of aggregation of flows. By requiring the network to be two-level, every source node is restricted to aggregate its flow at most once. This special case of the network design problem has also been addressed in [Min 89].

We first show that this problem is as hard as the set cover problem and then give a $O(\log n)$-approximation algorithm.

Proposition 5.1 *The minimum cost two-level network design problem is NP-hard. Furthermore, there is no polynomial time approximation algorithm for the two-level problem whose performance ratio is better than $(1 - o(1))\ln n$ unless $P = NP$.*

Proof: The proof is by reduction from the set cover problem. Consider an instance of a set cover problem with a collection C of subsets of a finite set S, a weight $w(S_i) \in Z^+$ for each set S_i in C, and a positive integer K. The set cover problem is to determine if C contains a subset C' such that total weight of the sets in C' is at most K and every element of S belongs to at least one set in C'. We consider the following instance of the two-level network design problem. Construct the two-level input graph $G = (V, E)$ for the network design problem as follows. Let there be a node s_i corresponding to each element in S with unit demand and a node v_j for each set S_j in C with zero demand. In addition, V contains the sink node t. For each set S_j, E contains an edge of zero length between v_j and all the source nodes corresponding to the elements contained in S_j, as well as an edge of length $w(S_j)$ between v_j and t. Suppose one type of cable with unit cost per unit length and ∞ capacity is available. Then, there exists a minimum cost two-level network solution with cost at most K if and only if there is a set cover C' with weight at most K. Consider a two-level network solution with cost K or less. Since in this solution each demand must follow a single path of at most two edges, each source node is connected to one set containing it. Thus the collection of sets which send a positive amount of flow to the sink is a cover with total weight at most K. Now, let C' be a set cover with weight at most K. We can find a route for each source by connecting the source to one of the sets containing it, in C'. Clearly, the total cost of the cables installed will be at most K.

Note that the above reduction is also approximation-preserving, so the current hardness results [F96, RS 97, AS 97] shows that there is no polynomial time approximation algorithm for the two-level problem whose performance ratio is better than $(1 - o(1)) \ln n$ unless $P = NP$. $\Box$

We present the following result that is nearly best possible.

Theorem 5.2 *The two-level link-installation problem can be approximated within a factor $O(\log n)$ where n is the number of nodes in the input graph.*

The key idea of the proof of the above theorem is to define an appropriate (very large size) set cover problem. It is well known that the greedy algorithm yields logarithmically bounded approximate solutions for the set cover problem [Ch 79], but the crucial step in the algorithm is to find a greedy set. In our case, the problem of finding a single greedy choice is computationally hard but we devise a constant factor approximation for this problem thereby proving the above theorem; for more details on how a constant factor approximation for a greedy step yields a logarithmic approximation for the set cover problem, see e.g. [YC 95, Theorem 10].

5.1 The corresponding set cover problem

The two level problem can be modeled as a set covering problem as follows: The elements to be covered are the sources, each with a demand. The "sets" used to cover them are called *stars*. A star consists of a center (any node in the graph except the sink) and leaves that include the sink node and a subset of the sources. A star has cost equal to the total cost of the *cheapest* choice of cables to route all the demand of the sources it contains via its center to the sink. Note that a star represents a level of aggregation of flows since all the demand within a star that is aggregated at the center node is sent through more economical thick cables to the sink. A solution to the two-level problem naturally decomposes into a set of stars (one level routes define starts with only one leaf, namely the sink). Hence an optimal solution to the two level problem is the same as an optimal solution to the set covering problem defined above.

5.2 Finding a greedy star

To implement an iteration of the greedy algorithm, we need to find a greedy star – a star of minimum ratio cost, of the ratio of total cable cost of the star divided by the total demand routed by the star. As there are exponentially many stars, we proceed by approximating the ratio cost within a constant factor. At any given step k of the greedy algorithm, let the total remaining demand to be routed be D^k. We first guess the total demand routed by the minimum ratio star at this stage within a factor of two, and for every such range, we find a cheapest star of roughly this much demand. Formally, consider the demand ranges $[1,2), [2,4), \ldots, [2^{\log \frac{D^k}{2}}, 2^{\log D^k}]$, and for every range, suppose we can compute the minimum *total* cost star whose demand falls within this range. Now suppose that the minimum *ratio* cost star routes a total of D_r demand at total cost C_r, where $D_r \in [2^i, 2^{i+1})$. Let C_i be the minimum cost of any star that routes demand in the range $[2^i, 2^{i+1})$ to the sink. Then, the ratio cost of this star is near-optimal. In particular, if this star routes demand D_i, then $\frac{C_i}{D_i} \leq \frac{2C_r}{D_r}$ since $C_i \leq C_r$ and $D_r \leq 2D_i$.

One last problem remains – the problem of finding a star of minimum cost that routes demand in a given range, say $[2^i, 2^{i+1})$, to the sink. However, this requires solving integer min-knapsack problems by the following reduction – We first guess the center of the star; there are at most n guesses. Then, for each center node, we want to find a set of sources (to connect to the center) that have a total demand of at least 2^i, with total minimum cost of routing the demand of that set to the center. In this sense, we want to fill a knapsack of demand at least 2^i corresponding to this choice of center. The items used to fill the knapsack are the remaining sources s_i each with demand dem_i. The cost of an item s_i is the cost of routing dem_i from s_i to the center. If the edge connecting s_i to the center, say e, has length $dist(e)$, this cost is $dist(e)$ times the value of another integer min-knapsack problem corresponding to the choice of cables for this edge (recall that given flow on an edge, finding minimum cost cables to cover this flow is a min-knapsack problem).

The integer min-knapsack problem is NP-hard, but can be solved in $O(qDem)$ time by dynamic programming ([GG65]). Alternatively, we can transform this integer knapsack problem to a 0-1 knapsack problem with $\hat{n} = \sum_{j=1}^{q} \log\lceil\frac{dem_i}{u_j}\rceil = O(q \log D)$ binary variables in $O(\hat{n})$ time (see [MT 90]). Then, applying any of the fully polynomial time approximation schemes for the 0-1 min-knapsack problem we obtain approximate solutions to our problem that obey the worst-case bounds for such schemes. For instance, a $(1+\epsilon)$-approximate solution can be obtained in $O(\hat{n}^2/\epsilon)$ time by the PTAS of Gens and Levner [GL 79]. Alternatively, a 2-approximation solution can be obtained in $O(\hat{n} \log \hat{n})$ time by the greedy algorithm of Gens and Levner [GL 79].

To summarize, we estimate the costs of the different sources for a given choice of center using the knapsack approximation. Using these costs, we solve yet another knapsack problem that gives an approximately minimum cost of routing a total of at least 2^i demand to this center. The *total* cost of the star however, must include the cost of routing the demand aggregated at the center to the sink. This is also a problem of choice of cables from the center to the sink for a given value of total demand, and can be approximated using the knapsack framework.

We repeat this procedure for every demand range i and every choice of the center node. By choosing the star that has minimum ratio of total cost to demand among all iterations, we get a constant factor approximation to the minimum ratio cost star.

Note that the approximation factor for finding a minimum ratio cost star only multiplies the performance ratio of the greedy algorithm [YC 95]. Since we use a constant factor approximation for finding a minimum ratio star, we get the performance ratio claimed in Theorem 5.2.

6 Open problems

Theorem 3.2 motivates the problem of finding a minimum cost tree routing as an approximate solution to the general network design problem. Although the problem is still NP-hard, approximating it may be easier than the general case.

In the more general minimum cost capacitated network design problem arising in the telecommunications industry, special pieces of hardware called *concentrators* [BMW 95, Min 89] are required to aggregate the traffic from several thin cables in a single thick cable. We are given a list of concentrators of various types (inputs being a combination of cable types of total bandwidth equal to the output cable bandwidth), each with an associated fixed cost. Whenever traffic is aggregated, appropriate concentrators have to be used by paying the corresponding fixed cost. Moreover, traffic requirements may be specified between multiple sources and multiple sinks. As before, the flow must be indivisible and routed by purchasing integral copies of cables, whose cost versus capacity is a step function representing an economy of scale. The goal is to find the minimum total cost network.

Approximating this more general problem with concentrators remains open. Note however that the fixed costs of concentrators can be incorporated to the approximation algorithm for the 2-level network problem. For a given range of demand to be aggregated, the cost of the concentrator to be installed can be approximated. In the greedy step of the algorithm, after the bicriteria algorithm outputs a star of certain cost and demand, the approximate cost of the concentrator can be added to the total cost of the star.

Subsequent to the appearence of a preliminary version of this paper in [SCR+97], Awerbuch and Azar [AA97] gave a randomized $O(\log^2 n)$-approximation algorithm for the general edge installation problem with many cable types and many sources and sinks.

References

[AKR 95] A. Agrawal, P. Klein and R. Ravi, "When trees collide: An approximation algorithm for the generalized Steiner problem on networks," SIAM J. Computing **24** 1995.

[ADD+92] I. Althöfer, G. Das, D. Dobkin, D. Joseph and J. Soares, "On sparse spanners of weighted graphs," Discrete and Computational Geometry, (1992).

[AS 97] S. Aurora and M. Sudan, "Improved low degree testing and its applications," Proc. 29th ACM Annual Symp. on Theory of Computing, 485-495, 1997.

[AA97] B. Awerbuch and Y. Azar, "Buy at bulk network design," Proc. 38th Ann. IEEE Symposium on Foundations of Computer Science, 542-547, 1997.

[BMW 95] A. Balakrishnan, T. Magnanti and R. T. Wong, "A decomposition algorithm for local access telecommunications network expansion planning," Operations Research, **43**, 58-76, 1995.

[Bal 61] M. L. Balinski, "Fixed cost transportation problems," Nav. Res. Log. Quart. **8**, 41-54, 1961.

[Bar 96] F. Barahona, "Network design using cut inequalitites," SIAM J. on Optimization **6**, 823-837, 1996.

[BCGT 98] D. Bienstock, S. Chopra, O. Gunluk and C-Y. Tsai, "Minimum cost capacity installation for multicommodity network flows,", Mathematical Programming, **81**, 177-199, 1998.

[BG 96] D. Bienstock and O. Gunluk, "Capacitated network design - Polyhedral structure and computation," INFORMS J. on Computing **8**, 243-259, 1996.

[Ch 79] V. Chvátal, "A greedy heuristic for the set-covering problem," Mathematics of Operations Research, **4**, 233-235, 1979.

[F96] U. Feige, "A threshold of ln n for approximating set cover," Proc. 28th ACM Annual Symp. on Theory of Computing, 314-318, 1996.

[GSS80] G. Gallo, C. Sandi and C. Sodini, "An algorithm for the min concave cost flow problem," Eur. J. Opnl. Res. **4**, 248-255, 1980.

[GJ 79] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.

[GL 79] G. V. Gens and E. V. Levner, "Computational complexity of approximation algorithms for combinatorial problems," Lecture Notes in Computer Science, **74**, 292-300, 1979.

[GG65] P. C. Gilmore and R. E. Gomory, "Multi-stage cutting stock problems of two and more dimensions," Operations Research, **13**, 94-120, 1965.

[GW 95] M. X. Goemans and D. P. Williamson, "A general approximation technique for constrained forest problems," SIAM J. Computing, **24**, 1995.

[GR 71] M. Goldstein and B. Rothfarb, "The one terminal telepak problem," Operations Research, **19**, 156-169, 1971.

[G71] P. Gray, "Exact solution of the fixed charge transportation problem," Operations Research **19**, 1529-1538, 1971.

[KRY 93] S. Khuller, B. Raghavachari and N. E. Young, "Balancing minimum spanning and shortest path trees," Algorithmica **14** (1993), 305–322.

[Kle 96] J. Kleinberg, "Single-source unsplittable flow," Proc. 37th Ann. IEEE Symposium on Foundations of Computer Science, 68-77, 1996.

[Lue 75] G. S. Lueker, "Two NP-complete problems in nonnegative integer programming," Report No. 178, Computer Science Laboratory, Princeton University, Princeton, NJ, 1975.

[MMV 95] T. L. Magnanti, P. Mirchandani and R. Vachani, "Modeling and solving the two-facility capacitated network loading problem," Operations Research, **43**, 142–157, 1995.

[MP 94] Y. Mansour and D. Peleg, "An approximation algorithm for minimum-cost network design," The Weizman Institute of Science, Rehovot, 76100 Israel, Tech. Report CS94-22, 10 pages.

[MT 90] S. Martello and P. Toth, *Knapsack problems: algorithms and computer implementations*, John Wiley & Sons, England, 1990.

[Min 89] M. Minoux, "Network synthesis and optimum network design problems: Models, solution methods and applications," Networks, **19**, 313–360, 1989.

[RMR+93] R. Ravi, M. V. Marathe, S. S. Ravi, D. J. Rosenkrantz and H. B. Hunt, "Many birds with one stone: Multi-objective approximation algorithms," Proc. 25th ACM Symp. on Theory of Computing, 438-448, 1993.

[RS 97] R. Raz and S. Safra, "A sub-constant error-probability low-degree test, and a sub-constant error-probability pcp characterization of np," Proc. 29th Annual ACM Symp. on Theory of Computing, 314-318, 1997.

[SCR+97] F.S. Salman, J. Cherijan, R. Ravi and S. Subramanian, "Buy-at-bulk network design: Approximating the single-sink edge installation problem," In proc. of the 8th Ann. ACM-SIAM Symposium on Discrete Algorithms, 619-628, 1997.

[WGM+95] D. P. Williamson, M. X. Goemans, M. Mihail and V. V. Vazirani, "A primal-dual approximation algorithm for generalized Steiner network problems," Combinatorica, **15**, 1995.

[YC 95] B. Yu and J. Cheriyan, "Approximation algorithms for feasible cut and multicut problems", Proc. Algorithms — ESA'95, 3rd Annual European Symposium, P. Spirakis (Ed.), LNCS 979, Springer, New York, 394–408, 1995.

[Z68] W. I. Zangwill, "Minimum concave cost flows in certain networks," Mgmt. Sci., **14**, 429-450, 1968.