Integrated Logistics: Approximation Algorithms Combining Facility Location and Network Design *

R. Ravi[†] Amitabh Sinha[†]

November 2001

Abstract

We initiate a study of the approximability of integrated logistics problems that combine elements of facility location and the associated transport network design.

In the simplest version, we are given a graph $G = (V, E)$ with metric edge costs c, a set of potential facilities $\mathcal{F} \subseteq V$ with facilty opening costs ϕ, a set of clients $D \subseteq V$ (each with unit demand), and a positive integer u (cable capacity). We wish to open facilities and construct a network of cables, such that every client is served by some open facility and all cable capacities are obeyed. The objective is to minimize the sum of facility opening and cable installation costs. With only one zero-cost facility and infinite u, this is the *Steiner tree* problem, while with unit capacity cables this is the *Uncapacitated Facility Location* problem. We give a $(\rho_{ST} + \rho_{UFL})$-approximation algorithm for this problem, where ρ_P denotes any approximation ratio for problem P.

For an extension when the facilities don't have costs but no more than p facilities may be opened, we provide a bicriteria approximation algorithm that has total cost at most $\rho_{p-MEDIAN} + 1$ times the minimum but opens up to $2p$ facilities.

Finally, for the general version with k different types of cables, we extend the techniques of [Guha, Meyerson, Munagala, STOC 2001] to provide an $O(k)$ approximation.

*Supported in part by a research grant from the Carnegie Bosch Institute, CMU, and by an NSF grant CCR-0105548.

[†]GSIA, Carnegie Mellon University, Pittsburgh, PA 15213. {ravi, asinha}@andrew.cmu.edu

1 Introduction

A ubiquitous problem faced by every corporation which manufactures and sells products to a geographically spread-out market is the following: Where should the factories be built, and how should the finished goods be transported to the markets, so as to minimize costs? Earlier work on facility location problems and network design problems have sought to address these two questions independently. In this paper, we initiate an integrated study of the overall problem; We define and study some simple versions of problems that combine the two objectives, and provide polynomial time approximation algorithms for them.

Consider the following scenario: A multinational corporation wishes to enter a promising new geographic market, characterized by demand at each city. It has also identified potential locations of its manufacturing facilities, and the associated costs. Suppose the shipping of the goods (from the facilities to the cities) is to be outsourced to a transport company. This transport company has only one type of truck, with a large capacity. For each truck, the transport company charges at a fixed rate per mile, and offers no discount in case the truck is not utilized to full capacity. The overall logistics problem facing the corporation is to decide on the location of its manufacturing facilities, and a shipping plan of the finished goods to each city, so that the total demand at each city is met and the total cost is minimized. Assume for the sake of simplicity that facilities have no capacity limitations.

If the facility location costs were not an issue (e.g., if the company had already decided where to open its facilities), the problem becomes a single sink edge installation problem [11]. (If several facilities are open, they can all be identified into a single sink node.) If the transport company charged in proportion to the amount shipped instead of the (discrete) number of trucks used, the problem becomes the uncapacitated facility location problem [18]. Both these problems have been well studied in the past. However, to the best of our knowledge, there has been no effort to study the problem in an integrated way that would allow one to exploit the savings that may result from making both decisions in a coordinated way to reduce the total cost of location and transportation. Our paper addresses this gap, and provides approximation algorithms for some simple versions of the integrated problem.

The first problem we consider is exactly as defined above. We call this the *capacitated cable facility location* problem (CCFL for short). A variant of this problem is the *median* version. Here there are no facility location costs, but we are not allowed to open more than p facilities. We call this the capacitated cable p-median problem (CCpM). Finally, we study an extension of CCFL where, for example, the transport company may provide a range of truck types, each with a different capacity and cost. We call this the *k-cable facility location* problem (KCFL for short).

All three problems generalize known NP-hard problems, and hence are NP-hard. We provide polynomial time approximation algorithms for these problems.

1.1 Previous work

While this is a first attempt to combine the facility location and transport network design objectives, a lot of work has been done on each of the individual problems. Shmoys, Tardos and Aardal [18] provided the first $O(1)$ approximation algorithm for the uncapacitated facility location problem. They used LP-rounding, thus also showing that the integrality gap of their IP formulation is 4. The bound on the integrality gap was improved to 3 by a primal-dual algorithm due to Jain and Vazirani [12]. Guha and Khuller [8] improved the approximation ratio using a greedy local-search algorithm. The current best known algorithm for uncapacitated facility location is a 1.728 approximation due to Charikar and Guha [5], using a combination of local search, greedy and LP-rounding techniques.

Charikar, Guha, Shmoys and Tardos [6] gave the first constant factor approximation algorithm for the p-median problem with metric costs. A local search technique by Korupulu, Plaxton and Rajaraman [14] provided an improved approximation, and this was further improved by Arya, Garg, Khandekar, Meyerson, Munagala and Pandit [2] to a factor of $3+\epsilon$, which is the best known at present.

Cable installation problems have also received a lot of attention in the recent past ([1], [3], [7], [9], [11], [15], [17]). Hassin, Ravi and Salman [11] provide a constant factor approximation for the single sink single cable version of the cable installation problem; we use their method as a subroutine. The currently best-known approximation for the multiple cable single sink edge installation problem is one that provides a constant factor guarantee by Guha, Meyerson and Munagala [9]; we use this method also in our solution to the general problem.

1.2 Our results

The CCFL problem with unit demands at the clients generalizes both the Steiner tree (ST) and uncapacitated facility location (UFL) problems. In the next section (Section 2), we present a $\rho_{ST} + \rho_{UFL}$ approximation algorithm for CCFL, where ρ_P is any approximation factor achievable for the problem P. We do this by carefully combining solutions to appropriately set up Steiner tree and UFL problems that capture two natural lower bounds for our problem. With the current best factors available, this is a 3.277-approximation algorithm. We also present an integer programming formulation of the problem, and show that its integrality gap is no more than 5. For the case where clients have arbitrary demands and the entire demand for a client must be served by the same facility, we provide a 5.005 approximation (Section 2.7).

For CCpM, in Section 3, we provide a bicriteria approximation that delivers a solution of cost at most $(\rho_{p-MEDIAN} + 2)$ times the optimum while opening up to $2p$ medians. Again, our method combines approximate solutions to a corresponding p-median problem and a 2-approximation for a newly defined p-Steiner forest problem. With the current best approximation factor for the p-median problem, this is a $(5 + \epsilon, 2)$ bicriteria factor for the (total cost, number of medians) problem.

Finally, in Section 4, we study the KCFL problem where k different cable types (or truck sizes) are available to us, each with a different cost and capacity. We provide an $O(k)$ approximation for this problem, by extending and adapting the algorithm of Guha et al [9] to incorporate the choices for facility location.

2 The capacitated-cable facility location problem

2.1 Problem definition

The *capacitated-cable facility location* problem (CCFL) is defined as follows. We are given an undirected graph $G = (V, E)$. There is a weight function on the edges, $c : E \to I\!R^+$, which satisfies the triangle inequality. The *clients* (markets) consist of a subset of nodes, $D \subset V$. The set of *potential facilities*, $\mathcal{F} \subset V$, is also part of the input. Each potential facility $j \in \mathcal{F}$ has a *facility opening cost* of ϕ_j. We are also given an integer $u > 0$, which is the *capacity* of the cable type available to us.

Each client has a demand of one unit, which needs to be serviced by routing one unit of flow to it from some *open* facility. On any edge, we are only allowed to install integral amounts of the cable. If we install z_e copies of the cable on edge e, we can route uz_e units of flow through it, and it costs us $c_e z_e$. Hence our total cost is the cost of all cables installed plus the cost of all the facilities we have opened. The objective of CCFL is to open facilities and install cables connecting clients to open facilities such that no capacity constraint is violated, all clients are served, and the total cost is minimized.

2.2 Hardness and relation to other problems

If there is only a single potential facility ($|\mathcal{F}| = 1$) and u is infinity, then the problem reduces to the *Steiner tree* problem. If there is a single facility and $1 < u < \infty$, CCFL is the *single-sink, single-cable edge installation* problem. If $u = 1$ but $|\mathcal{F}| > 1$, CCFL is the *uncapacitated facility location* problem. All these problems have been studied in the past, and all three are known to be `MAX-SNP`-hard. Hence CCFL is also `MAX-SNP`-hard.

2.3 Lower bounds

We begin with two lemmas which provide lower bounds to an optimal solution of CCFL.

Lemma 1 *Consider a UFL (uncapacitated facility location) instance defined as follows. The set of clients and potential facilities remain the same as in the CCFL instance, but for all edges e, we set the edge cost to be c_e/u. Then the cost of an optimal solution to this UFL instance is a lower bound on the optimal solution to CCFL.*

Proof: Consider the optimal solution to CCFL. In the UFL instance, open all facilities which were opened by CCFL. Every client in CCFL is able to send one unit of flow to an open facility. Construct these flow paths. Now for each client, assign it to the facility it is assigned to in CCFL. The cost of this assignment is at most $\frac{1}{u}$ of that of the flow path used by this client in the CCFL solution, by triangle inequality. This constitutes a feasible solution to UFL, of cost no more than that of the CCFL solution. Hence an optimal UFL solution has cost at most that of the optimal CCFL solution. $\square$

Lemma 2 *Consider the graph $G' = (V \cup \{r\}, E \cup E')$ where $E' = \{(j,r) : j \in \mathcal{F}\}$ and $c_{(j,r)} = \phi_j$. Define the set of terminals to be $R = D \cup \{r\}$. Then the cost of a minimum Steiner tree in G' is a lower bound on the optimal solution of CCFL.*

Proof: Consider the optimal solution to CCFL. The set of edges in the CCFL solution, along with the edges (j, r) such that facility j is opened in the CCFL solution, constitutes a Steiner tree in G' of the same cost as the CCFL solution. Dropping all but one copy of edges which have multiplicity more than 1 in the CCFL solution only reduces the cost. Hence an optimal Steiner tree must cost no more than the optimal CCFL solution. □

We use the two lower bounds in Lemma 1 and Lemma 2 (and approximation algorithms for these two problems) to build our solution. We use a flow rerouting algorithm introduced by Hassin, Ravi and Salman [11] to efficiently construct our solution.

2.4 Algorithm

We first run approximation algorithms for an uncapacitated facility location instance and a Steiner tree instance by transforming our problem as described in Lemmas 1 and 2. We then merge the two solutions to obtain a feasible solution of cost no more than the sum of these two approximate solutions (in a **Merge phase**).

To carry out the Merge phase, we adapt a re-routing algorithm described in [11]. We first open all facilities identified by the earlier two phases. Consider the subtrees associated with the facilities opened in the Steiner tree phase. If such a subtree has at most u clients, this subtree along with the facility it is attached to is a feasible solution, without adding any additional copies of the cable.

On the other hand, a subtree that has more than u clients is not feasible right away, since more cables have to be installed along the tree to route all the demand in this overloaded subtree. This is where we use the UFL solution - we clump the demands in these overloaded subtrees into subtrees which are disjoint with respect to edge capacities such that each new subtree has exactly u clients each (with one remaining subtree with at most u clients attached to the facility opened in the original overloaded subtree). The fact that such a clumping is possible was proved in [11]; we describe it in detail in Algorithm 1 and prove it in Lemma 3. For each such clump, we use the UFL solution to select the client which is closest to an open facility in the UFL solution, and install one cable from this client to its nearest open facility. The idea is that since each client can pay a $1/u$ fraction of the cable cost to the facility assigned to it by the UFL phase, u such clients in a clump can together pay for one full cable from a client to an open facility if this distance is the cheapest distance among these u clients. In order to achieve this, we need to re-route flow from the $u - 1$ other clients to our selected client in a clump. However, this rerouting takes place along the original Steiner tree solution at no extra cost since the subtrees obtained in the clumping are disjoint with respect to edge capacities. We finally prune the solution by getting rid of unused facilities and cables.

The algorithm is formally described in Figure 1.

2.5 Analysis

We have argued that both the underlying UFL instance and the associated Steiner tree problem are lower bounds for our CCFL instance. Hence the facilities opened by these two phases can be paid for by these two lower bounds.

The cables purchased by the Steiner tree phase can be paid for by the Steiner tree lower bound. We also install fresh cables in the Merge phase. Each cable has exactly u demand flowing through it. Each of the terminals which use this cable were assigned a facility whose distance is at least the length of the cable in the UFL phase. Hence we can charge the cost of this cable to the cost in the UFL solution.

Lemma 3 (due to Hassin, Ravi and Salman [11]) *The solution produced by our algorithm is feasible for CCFL.*

Proof: By standard flow cancellation arguments, no cable is used in both an upward and a downward direction. The flow cancellation only reduces flow in the upward direction. If any cable has an upward flow, this flow has value at most $u-1$, and this may potentially be cancelled by downward flow when a client in the subtree below it is part of a picked pair.

Downward flow is assigned to any cable at most once, and the quantity of flow assigned is at most $u-1$. After such an assignment, all the clients in this subtree are deleted from further consideration. □

Recall that ρ_{ST} and ρ_{UFL} denote the currently best known approximation ratios for the Steiner tree and UFL problems respectively. We have the following theorem.

Theorem 1 **Algorithm CCFL** *is a $\rho_{ST} + \rho_{UFL}$ approximation algorithm for CCFL.*

Proof: This follows from Lemmas 1, 2 and 3. □

The current best approximation algorithm for the Steiner tree problem is the one by Robins and Zelikovsky [16], which achieves an approximation factor of 1.549. Charikar and Guha [5] have the current best approximation for UFL, with a performance ratio of 1.728. With these values for ρ_{ST} and ρ_{UFL}, Theorem 1 gives a 3.277 approximation.

Algorithm CCFL

UFL phase:
- Convert into UFL instance by changing edge costs to c_e/u.
- Solve UFL (approximately).
  - Let $\mathcal{F}_1$ denote the facilities opened.
  - For a client i, let $\phi(i)$ be its assigned facility.

Steiner tree phase:
- Create a new root node r.
- For every $j \in \mathcal{F}$, add an edge (j, r) with cost ϕ_j.
- Define the terminal set $R := D \cup \{r\}$.
- Solve (approximately) the Steiner tree problem.
  - Let T denote this tree.
  - Orient all edges to point towards the root along T.
  - Let $\mathcal{F}_2$ be the set of facilities through which there are edges in T.
  - Let T_j be the subtree of T rooted at j, for all $j \in T$.

Merge phase:
- Open all facilities in $\mathcal{F}_1 \cup \mathcal{F}_2$.
- For all $j \in \mathcal{F}_2$, do:
  - Let D_j be the set of clients in T_j.
  - Install cable on all edges in T_j.
  - While $|D_j| > u$ do:
    - Let V' be the set of nodes at which the incoming demand on each edge is less than u, but the total demand is at least u.
    - For all $v \in V'$ do:
      - For every child w of v, let T_w be the subtree rooted at w.
      - Let (i_w, j_w) be the nearest client-facility pair in T_w.
      - Pick the cheapest $\lfloor D_v/u \rfloor$ such pairs.
      - Install one cable on each such picked pair (i_w, j_w).
      - Route all demand in T_w to j_w via i_w.
      - Route remaining demand (in other subtrees T_w of children of v) to either some picked pair or to w, in such a way that all newly installed cables are saturated. This means that the total remaining demand to v is less than u.
      - Remove all satisfied demands from D_j.

Prune phase:
- Remove all cables on which flow is zero.
- Close all facilities which have no demand.

Figure 1: Algorithm for CCFL

2.6 IP formulation and its gap

There is a natural integer programming formulation of CCFL, which we describe next. We show that the techniques used in our approximation algorithm described above extend to providing a constant factor rounding algorithm for the linear relaxation of the IP formulation.

IP_{CCFL} is an integer program formulation of CCFL. Variable y_j is an indicator variable which is 1 iff facility j is opened. The number of copies of the cable on edge e is counted by z_e. Finally, f_e^i is the flow of the demand from client i along edge e. For a vertex set S, define $\delta^+(S) = \{(u,v) \in E : u \in S, v \notin S\}$. Define $\delta^-(S) = \delta^+(V \setminus S)$, and for a vertex v, define $\delta^+(v) = \delta^+(\{v\})$. The first four constraints are flow constraints, while the last is a connectivity constraint which strengthens the linear relaxation of IP_{CCFL}. These constraints enforce that for any set S containing a client, either it must contain a facility or must have at least one cable leaving the set (to connect a client within it to a facility in the solution).

$$
\begin{array}{llll}
\min & \sum_j \phi_j y_j + \sum_e z_e c_e & & (IP_{CCFL}) \\
& \sum_{e \in \delta^+(i)} f_e^i - \sum_{e \in \delta^-(i)} f_e^i & \geq 1 & \forall i \in D \\
& \sum_i f_e^i - z_e u & \leq 0 & \forall e \in E \\
& \sum_{e \in \delta^-(j)} f_e^i - \sum_{e \in \delta^+(j)} f_e^i & \leq y_j & \forall i \in D, \forall j \in \mathcal{F} \\
& \sum_{e \in \delta^+(v)} f_e^i - \sum_{e \in \delta^-(v)} f_e^i & = 0 & \forall i \in D, \forall v \in V \setminus (\mathcal{F} \cup \{i\}) \\
& \sum_{e \in \delta^+(S)} z_e + \sum_{j \in S} y_j & \geq 1 & \forall S \subseteq V : S \cap D \neq \emptyset \\
& z_e, y_j, f_e^i & & \text{non-negative integers}
\end{array}
$$

Let gap_{ST} and gap_{UFL} denote the currently known upper bounds on the integrality gap of the undirected cut formulation of Steiner tree problem and the standard IP formulation of the uncapacitated facility location problem respectively, that are obtainable by polynomial-time rounding algorithms.

Theorem 2 *The integrality gap of IP_{CCFL} is no more than $gap_{ST}+gap_{UFL}$.*

Proof: Consider an optimal solution to the linear relaxation (denoted LP_{CCFL}) of IP_{CCFL}. The linear relaxation of our UFL instance described in Lemma 1 is exactly LP_{CCFL} with the last constraint ignored. Hence an optimal solution to LP_{CCFL} costs no less than the solution to our UFL instance. If we use the rounding algorithm with gap gap_{UFL} in the UFL phase of our algorithm, this solution costs no more than gap_{UFL} times the value of an optimal solution to LP_{CCFL}.

Similarly, the linear relaxation of our Steiner tree instance described in Lemma 1 is identical to LP_{CCFL} if we ignore the first four (flow) constraints. Hence an optimal solution of the Steiner tree relaxation is a lower bound on the cost of the optimal solution of LP_{CCFL}. Therefore the cost of the Steiner tree phase of our algorithm (using the rounding algorithm with ratio gap_{ST}) is no more than gap_{ST} times the value of an optimal solution to LP_{CCFL}. □

The current best bounds on gap_{UFL} and gap_{ST} are 3 [12] and $(2 - \frac{2}{|D|+1})$ [4] respectively. Hence the integrality gap of this formulation of CCFL is less than 5.

2.7 Non-uniform demands

The above algorithm clearly generalizes to the case of non-uniform demands at the clients, provided we are allowed to split the demand at each client to different facilities. If the demands are unsplittable, the problem becomes more interesting. However, Hassin et al. [11] showed how their (single sink) problem can be solved in the unsplittable demand case with a slight increase in the approximation ratio. Clients which have more than u demand can be sent directly to their nearest facilities, incurring an additional factor of at most 2. To assign the remaining clients, we proceed as before. We now aggregate demands to lie between u and $2u$, and again use the UFL bound at most twice. Hence the approximation ratio for this problem is $2\rho_{UFL} + \rho_{ST} = 5.005$.

3 The capacitated cable p-median problem

3.1 Problem definition

The *capacitated cable p-median* problem (CCpM) is a minor variant of CCFL. Facilities can be opened for free in this version, but we are not allowed to open more than p facilities (called *medians* in this context). Everything else is as in CCFL. An (α, β) approximation for CCpM consists of a solution which uses βp medians and costs no more than α times the best possible solution which uses no more than p medians.

We first consider a simplified version where every node in the graph is a client node and also an eligible median. Let $\rho_{p-MEDIAN}$ denote the best known approximation factor for the p-median problem. We provide a $(\rho_{p-MEDIAN} + 1, 2)$-approximation for this restricted version of CCpM.

In Section 3.4 we extend it to the case where every node may be a client, a potential median, both, or neither, and provide a $(\rho_{p-MEDIAN} + 2, 2)$ approximation.

3.2 Overview of our approach

Our approach is essentially the same as before, with appropriate modifications. The proof of the following lemma is identical to the proof of Lemma 1.

Lemma 4 *Consider a p-median instance as follows. The set of clients and potential facilities remain the same as in the CCpM instance, but for all edges e, we set the edge cost to be c_e/u. Then an optimal solution to this p-median instance is a lower bound on the optimal solution to CCpM.*

Definition 1 *Given a graph $G = (V, E)$ with edge costs, the p-Steiner forest problem is to find a minimum cost forest with at most p trees.*

Lemma 5 *The p-Steiner forest problem can be solved optimally in polynomial time.*

Proof: A minimum spanning tree with the $p - 1$ heaviest edges deleted is an optimal solution to the p-Steiner forest problem. □

Lemma 6 *A minimum cost p-Steiner forest in G is a lower bound on the optimal solution of CCpM.*

Proof: Consider the optimal solution to CCpM, and delete all but one copy of edges which have multiplicity more than one. This constitutes a p-Steiner forest in G of no greater cost than the CCpM solution. □

Our algorithm is now straightforward. We solve the above two problems on our input instance. For each tree in the p-Steiner forest solution, designate any node as its median. (The p-median instance can only be solved approximately, to a factor $\rho_{p-MEDIAN} = 3 + \epsilon$ [2].) We then reroute exactly as described in the **Merge** phase of **Algorithm CCFL**.

3.3 Analysis

Lemma 3 continues to hold and ensures that we do not violate any capacities in the solution we construct. Lemmas 4 and 6 bound the cost of the two stages of our solution. However, since each of our phases chooses p medians, we may end up with a solution which has as many as $2p$ medians.

Theorem 3 *There is a $(\rho_{p-MEDIAN} + 1, 2)$ approximation algorithm for the restricted version of CCpM.*

3.4 Unrestricted version of CCpM

We now relax the simplification that every node is a client as well as a possible median. In the unrestricted case, a node may be a client, a possible median, both, or neither (Steiner node). As before, let $\mathcal{F} \subseteq V$ denote the set of possible medians. We can continue to use the $(3 + \epsilon)$ approximation for p-median for the p-median phase. However, our new p-Steiner forest problem is as follows. We wish to compute a minimum cost forest which has at most p trees, such that every client is in some tree, and each tree has at least one possible median.

The following is an integer program formulation of p-Steiner forest. Let y_r be an indicator variable indicating whether or not we designate node r to be a median. Let z_e denote whether or not we pick edge e, and for any $S \subseteq V$, let $\delta(S) = \{(u, v) \in E : u \in S, v \notin S\}$.

$$\begin{array}{rll}
\min & \sum_{e \in E} c_e z_e & (IP_{CCpM}) \\
z(\delta(S)) + y(S) & \geq 1 & \forall S \subseteq V, S \cap D \neq \emptyset \\
\sum_{r \in \mathcal{F}} y_r & \leq p & \\
z, y & \text{non-negative integers} &
\end{array}$$

The dual of the linear relaxation of IP_{CCpM} is given below:

$$\begin{array}{rll}
\max & \sum_{S \subseteq V, S \cap D \neq \emptyset} u_S - \lambda p & (DP_{CCpM}) \\
\sum_{S : e \in \delta(S)} u_S & \leq c_e & \forall e \in E \\
\sum_{S : r \in S} u_S & \leq \lambda & \forall r \in \mathcal{F} \\
u, \lambda & \geq 0 &
\end{array}$$

The following is a corollary of a lemma about the integrality ratio of the Steiner tree IP formulation, proved in [4].

Corollary 1 *The primal-dual algorithm of [4] applied to IP_{CCpM} provides a polynomial time 2 approximation for the p-Steiner forest problem.*

Proof: In the absence of the second constraint, IP_{CCpM} is exactly the undirected Steiner tree IP formulation. The dual of its linear relaxation is exactly DP_{CCpM} with the second constraint ignored.

Consider the primal-dual algorithm for Steiner tree of [4]. We can run the same algorithm here to search for a *locally optimal* dual solution. In the algorithm, we raise the value of dual variables corresponding to *minimally violated sets* simultaneously. A *violated* set is a subset of the vertex set which is either not connected by edges selected in the primal solution, or does not contain a median. If necessary, we simultaneously raise λ, as long as the number of connected components in the primal is more than p. We also construct a primal solution alongside, driven by the complementary slackness conditions. Since all variables are being raised simultaneously, we can define a notion of "time", such that the dual variables are being raised at the rate of one unit per unit time. Whenever two minimally violated sets have duals large enough to satisfy a constraint to equality, we add the tightened edge to our primal solution, and replace the two sets by their union (which is a new minimally violated set).

Clearly, while there are more than p components the total dual value is increasing. A locally optimal solution is obtained when there are at most p components, each containing a potential median. By imposing an ordering to break ties, we can find a locally maximal dual solution and a corresponding primal solution which has at most p connected components, each with at least one median.

Lemma 5.3 in [4] proves that at time t, the cost of any tree constructed in the primal is no more than twice the total dual collected minus twice the time t. Clearly λ is never more than the final value of the time t, since λ is also raised at the same rate of one per unit time. Therefore, the cost of each component is at most twice the value of the duals collected by moats within it minus twice λ. Summing this inequality over the p components, we get that the primal solution has cost at most twice the value of this locally maximal dual solution to DP_{CCpM}. $\square$

Theorem 4 *There is a polynomial time $(\rho_{p-MEDIAN}+2, 2)$ bicriteria approximation algorithm for the general version of the capacitated-cable p-median problem.*

Proof: This follows from Lemmas 4, 3 and Corollary 1. □

The current best value of $\rho_{p-MEDIAN}$ is $3+\epsilon$ [2]. Hence our algorithm is a $(5+\epsilon, 2)$ bicriteria approximation to the general cable-capacitated p-median problem.

4 The multiple-cable facility location problem

We now consider an extension of CCFL. Instead of just one cable type (or truck type), we have a suite of k cable types. Cable type i has fixed cost σ_i and variable (per unit) cost δ_i. That is, for using one copy of cable type i on edge e and transporting f_e flow through it, our cost is $(\sigma_i + f_e\delta_i)c_e$. This is equivalent to cables having fixed costs and capacities (within an approximation factor of two – see [7], [9]), which is why we call it a generalization of CCFL. We call this problem the *k-cable facility location* problem, or KCFL.

In KCFL, if $|\mathcal{F}| = 1$, then the problem reduces to the *single sink edge installation* problem. Guha, Meyerson and Munagala provided a constant factor approximation algorithm for this problem [9]. In this section, we show how their algorithm can be adapted to incorporate many facilities. We will closely follow their paper, and many lemmas will not be proved here because they require only minor or notational changes from their paper. We use GMM to denote their paper and algorithm.

4.1 Algorithm

We begin with a brief overview of the GMM algorithm for the single sink edge installation problem. It can be shown that there is a near-optimal solution which is a tree, and where the flow path from any node to the sink uses cables in non-decreasing order of fixed costs. GMM build this tree in a bottom-up fashion. They start with the set of all clients, and build the first layer using a Steiner tree. They are able to charge this cost to a connectivity component of the cost of an optimal solution. They then find points of aggregation of sufficient demand along this tree, and solve a *lower bounded facility location* problem[1] to send some of that demand to the root. The rest of the demand is *aggregated* at certain points. The amount of aggregation is such that it justifies the use of the next type of cable, which

[1]The lower bounded facility location can also be solved to within a constant factor, see [10], [13].

has a higher fixed cost but a lower incremental cost. This process is repeated for all the cable types. The details may be read in their paper [9].

For our purposes, we begin by pre-processing the cables as in GMM. We order the cables in order of increasing fixed cost, and then retain cables such that for any i, we have $\sigma_i \leq \alpha\sigma_{i+1}$ and $\delta_{i+1} \leq \alpha\delta_i$, for some predefined constant $\alpha \in (0, \frac{1}{2})$. We also define b_i to be such that $\sigma_{i+1} + \delta_{i+1}b_i = 2\alpha(\sigma_i + \delta_i b_i)$. Intuitively, b_i is the demand quantity at which the cost of using cable types i and $i+1$ are (almost) the same. We also define $u_i = \sigma_i/\delta_i$. Again, u_i is the point in building the part of the solution using cable type i when the problem shifts its focus from being a Steiner-tree-like problem to being a Shortest-path-aggregation-like problem, since after this threshold, the routing cost of any cable can pay for the fixed cost of installing it on any edge with this much flow.

We are now ready to state our algorithm. In the following, the changes we introduce are italicized, while the rest is the original GMM algorithm.

Iterate over cable types in order of increasing fixed cost:

1. **Steiner tree**: *Augment the graph by adding a "sink" and connecting it to every facility with edge cost equal to the facility cost.* All other edges have cost σ_i. Construct an approximately optimal Steiner tree with the terminals being the sources and the newly added sink. Walking along this tree, identify edges which have u_i demand and "cut" the tree at these edges.

2. **Consolidate**: For every tree in the forest created in the preceding step *not attached to a facility paid for by the Steiner tree*, transfer the total demand in the tree back to one of its sources with probability proportional to the demand at that source.

3. **Shortest path tree**: Set up a lower bounded facility location instance as follows. On every node, the lower bound is b_i and cost is 0. *On facility nodes, the lower bound is* 0 *and cost is the facility cost.* Edge costs are now δ_i per unit length. Solve this LBFL problem approximately.

4. **Aggregate**: *Consider only nodes which are not assigned to facilities opened by the previous step.* For each such node, as in Step 2, we transfer the total demand in that component of the solution back to one of the sources with probability proportional to the demand at that source.

In each stage, we also make available all facilities opened at all previous stages, at cost zero. In the last stage, the Shortest path tree step has no b_i

since there are no more cable types. Hence all the demand aims to reach only facilities. There is no Aggregate step for the last cable type, since all the demand has reached open facilities.

4.2 Properties of a near-optimum solution

If the set of facilities to be opened has already been decided for us, then the problem reduces to the single-sink edge installation problem. This is achieved by identifying all opened facilities into a new node called "sink", and updating the metric appropriately. This transformation allows us to show the existence of a near-optimum solution to KCFL which satisfies the properties mentioned in Theorem 5 below. Our algorithm also constructs a solution which satisfies these properties, and we compare ourselves with this near-optimal solution.

Theorem 5 (Theorem 3.2 in GMM) *There exists a solution to KCFL which uses cable type $i+1$ on a link only if at least b_i demand is being routed across that link, and which routes all demand which entered a node using cable i, out of that node using cables i and $i+1$. This solution pays at most $\frac{2}{\alpha}+1$ times the optimum.*

Proof: Consider an optimum solution of KCFL. Take its set of open facilities, and identify them to a sink. The resulting solution is now a (possibly sub-optimal) single-sink solution. GMM show that there is a near-optimal solution to this single-sink instance which obeys the properties enumerated in the theorem. Hence we can transform our KCFL-optimal solution to a solution which satisfies these properties. We next reverse our "identification-of-facilities" operation, that is, we "separate" the facilities. This yields a solution to KCFL which is not too far from the optimum, and which satisfies the properties mentioned in the theorem. □

The next two lemmas can be proved using nothing more than the definition of u_i and b_i.

Lemma 7 (Lemmas 3.3 and 3.4 in GMM) *For all i, we have $u_i \leq b_i \leq u_{i+1}$.*

Define $f_i(D) = \sigma_i + \delta_i(D)$ to be the per-unit-distance cost of routing D units of flow on cable type i.

Lemma 8 (Lemma 3.5 in GMM) *For any i and $D \geq b_i$, we have $f_{i+1}(D) \leq 2\alpha f_i(D)$.*

4.3 Analysis

Our analysis will follow and make some modifications to the analysis in the paper by Guha, Meyerson and Munagala. Let d_v be the original demand of node v. Let D_v^i be the demand at node v at stage i in the algorithm.

Let T_i, P_i and N_i be the Steiner tree, Shortest path and consolidation-aggregation step costs respectively, at iteration i. Also, let T_i^I, T_i^F and T_i^ϕ denote the incremental (variable), fixed and facility opening cost components respectively of the Steiner tree step at iteration i. P_i^I, P_i^F and P_i^ϕ are similarly defined. The following Lemmas are proved for the single sink version in GMM, and can be adapted to hold in our setting with facility costs too.

Lemma 9 (Lemma 4.1 in GMM)[2] *At the end of every consolidation and aggregation step, $E[D_v^i]$ at each node which has not yet been connected to an open facility is d_v.*

Proof: The main idea is that at the end of every consolidation and aggregation step, we re-route any demand which has not yet been connected to client nodes with probability proportional to the original demand at each node. Some nodes are connected to open facilities, and no demand is ever re-routed to any such node. Hence for any client node which still hasn't been connected to an open facility, its expected demand at any stage $E[D_v^i]$ is exactly d_v. □

Since the consolidation and aggregation steps occur only after all excess demand has been removed, the total demand at each such step is no more than the cable capacity. Hence, we have the following.

Lemma 10 (Lemma 4.2 in GMM) *At every i, we have $E[N_i] \leq T_i + P_i$.*

Proof: The consolidation step occurs only along the Steiner tree which has just been built. It uses no extra edges, and hence costs no more than T_i. Similarly, the aggregation step costs no more than P_i. Hence the total consolidation-aggregation cost at step i, $E[N_i]$, is no more than $T_i + P_i$. □

Lemma 11 (Lemmas 4.4 and 4.8 in GMM) *At every i, we have $P_i^F \leq P_i^I$ and $T_i^I \leq T_i^F$.*

Proof: At the start of every Shortest path tree stage, we know that the demand at any active node is at least u_i, since this is guaranteed by the

[2] Some of the results from GMM have been reworded for contextual clarity, while others have been extended by us to incorporate facility costs.

preceding Steiner tree step. By the definition of u_i, the fixed cost at the shortest-path step is no more than the incremental cost, hence $P_i^F \leq P_i^I$.

Similarly, at the start of every Steiner tree step we have at least b_i demand at every active node. Again, using the definition of b_i, we can prove that $T_i^I \leq T_i^F$. □

Now let C_i^* be the total cost of cables of type i in the near-optimum solution. Define $C^* = \sum_i C_i^*$ to be the total cable cost of the near-optimum solution. Let ϕ^* be the cost of the facilities opened by the near-optimum solution. Our main result is a consequence of the following two theorems.

Theorem 6 (Lemma 4.3 in GMM) *For every i, we have $E[P_i^I + P_i^\phi] = O(C^* + \phi^*)$.*

Proof Sketch: Consider the near-optimum solution, and replace all cables of type less than i by cable type i. For each $j < i$, the incremental cost of the new solution is a small fraction (α^{i-j}) of the incremental cost of the optimal solution's cable-j portion. The set of facilities opened in this new solution, combined with the new cables, constitutes a feasible solution for our problem of cost no more than $\phi^* + \frac{C^*}{(1-\alpha)}$. Since the lower bounded facility location problem can be solved within a constant factor ([10], [13]), the theorem follows. □

Theorem 7 (Lemmas 4.6 and 4.7 in GMM) *For every i, we have $E[T_i^F + T_i^\phi] = O(C^* + \phi^*)$.*

Proof Sketch: Consider the near-optimum solution, and consider only those nodes which are candidate terminals in our stage i. Since it already has sufficient demand (b_i), thanks to stage $i-1$, the expected cost incurred when we modify the near optimum solution to install a type i cable on the path from this node to an open facility is small, that is, within a constant of the original cost of the near-optimum solution.

This solution is a candidate solution for the Steiner tree stage, and hence a lower bound. Since we can find a Steiner tree within a constant factor of optimum [4], we are done. □

Theorem 8 *There is an $O(k)$ approximation algorithm for KCFL.*

Proof: This follows from Lemma 10, Lemma 11, Theorem 6 and Theorem 7. □

The precise details of the single sink edge installation problem can be read in the paper by Guha, Meyerson and Munagala [9]. They are able to prove much stronger versions of Theorems 6 and 7, which allows them to obtain an $O(1)$ approximation for their problem. However, they do not have facility costs. We have shown that we can incorporate facility costs, but we are only able to prove a weaker bound on the cost at each stage. We note that in our solution, the cable costs continue to meet the bounds proved in GMM, but the facility costs do not. In other words, the cable cost of our solution to KCFL is in fact within a constant factor of the best possible. It is an intriguing open question to bound the facility costs to within a constant too.

5 Open questions

Our approximation algorithm for CCpM provides a solution which uses twice as many medians as we are allowed to. An algorithm which is *uni-criteria*, that is, obeys the median restriction exactly, would be very desirable.

Our approach does not work for the median version of KCFL, as it would open $O(kp)$ medians. We also note that for KCFL, the cable cost of our solution is still within a constant factor of the best possible. The approximation ratio is $O(k)$ only due to the facility location costs. If we can tailor our algorithm to keep the facility costs bounded, our approach may yield a constant factor approximation for KCFL.

References

[1] M. Andrews and L. Zhang, "The access network design problem", Proc. of the 39th Ann. IEEE Symp. on Foundations of Computer Science, 1998.

[2] V. Arya, N. Garg, R. Khandekar, V. Pandit, A. Meyerson and K. Munagala. "Local search heuristic for k-median and facility location problems", Proc. 33rd ACM Symposium on the Theory of Computing, 2001.

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

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

[5] M. Charikar and S. Guha, "Improved combinatorial algorithms for the facility location and k-median problems", Proc. 40th Ann. IEEE Symposium on Foundations of Computer Science, 1999.

[6] M. Charikar, S. Guha, D. Shmoys and E. Tardos, "A constant-factor approximation algorithm for the k-median problem", Proc. 31st ACM Symposium on Theory of Computing, 1999.

[7] N. Garg, R. Khandekar, G. Konjevod, R. Ravi, F.S. Salman and A. Sinha. "On the integrality gap of a natural formulation of the single-sink buy-at-bulk network design problem", Proc. 8th Conference on Integer Programming and Combinatorial Optimization, 2001. Preliminary version appeared as "A mathematical formulation of a transportation problem with economies of scale", Carnegie Bosch Institute Working paper 01-1, 2001.

[8] S. Guha and S. Khuller. "Greedy strikes back: Improved facility location algorithms", Proc. 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[9] S. Guha, A. Meyerson and K. Munagala. "Improved combinatorial algorithms for single sink edge installation problems", Proc. 33rd ACM Symposium on the Theory of Computing, 2001.

[10] S. Guha, A. Meyerson and K. Munagala. "Hierarchical placement and network design problems", Proc. 41st Ann. IEEE Symposium on Foundations of Computer Science, 2000.

[11] R. Hassin, R. Ravi and F.S. Salman. "Approximation algorithms for a capacitated network design problem", Approximation Algorithms for Combinatorial Optimization, 2000.

[12] K. Jain and V.V. Vazirani. "Primal-dual approximation algorithms for metric facility location and k-median problems", Proc. 40th Ann. IEEE Symposium on Foundations of Computer Science, 1999.

[13] D. Karger and M. Minkoff. "Building Steiner trees with incomplete global knowledge", Proc. 41st Ann. IEEE Symposium on Foundations of Computer Science, 2000.

[14] M. Korupulu, C. Plaxton and R. Rajaraman. "Analysis of a local search heuristic for facility location problems", Proc. 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[15] 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, 1994; Also presented at the DIMACS workshop on Robust Communication Networks, 1998.

[16] G. Robins and A. Zelikovsky. "Improved Steiner tree approximation in graphs", Proc. 10th ACM-SIAM Symposium on Discrete Algorithms, 1999.

[17] F.S. Salman, J. Cheriyan, R. Ravi and S. Subramanian, "Buy-at-bulk network design: Approximating the single-sink edge installation problem", Proc. 8th ACM-SIAM Symposium on Discrete Algorithms, 1997.

[18] D.B. Shmoys, E. Tardos and K. Aardal. "Approximation algorithms for the facility location problem", Proc. 29th ACM Symposium on the Theory of Computing, 1997.