Capacitated Network Design on Undirected Graphs

Deeparnab Chakrabarty* Ravishankar Krishnaswamy† Shi Li‡ Srivatsan Narayanan§

Abstract

In this paper, we study the approximability of *the capacitated network design problem* (Cap-NDP) on *undirected* graphs: Given $G = (V, E)$ with non-negative costs c and capacities u on its edges, source-sink pairs (s_i, t_i) with demand r_i, the goal is to find the minimum cost subgraph where the minimum (s_i, t_i) cut with u-capacities is at least r_i. When $u \equiv 1$, we get usual SNDP for which Jain gave a 2-approximation algorithm [9]. Prior to our work, the approximability of undirected Cap-NDP was not well understood even in the single source-sink pair case. In this paper, we show that the single-source pair Cap-NDP is label-cover hard in undirected graphs.

An important special case of single source-sink pair undirected Cap-NDP is the following *source location problem*. Given an undirected graph, a collection of sources S and a sink t, find the minimum cardinality subset $S' \subseteq S$ such that $\mathsf{flow}(S', t)$, the maximum flow from S' to t, equals $\mathsf{flow}(S, t)$. In general, the problem is known to be set-cover hard. We give a $O(\rho)$-approximation when $\mathsf{flow}(s, t) \approx_\rho \mathsf{flow}(s', t)$ for $s, s' \in S$, that is, all sources have max-flow values to the sink within a multiplicative ρ factor of each other.

The main technical ingredient of our algorithmic result is the following theorem which may have other application. Given a capacitated, undirected graph G with a dedicated sink t, call a subset $X \subseteq V$ *irreducible* if the maximum flow $f(X)$ from X to t is strictly greater than that from any strict subset $X' \subset X$, to t. We prove that for any irreducible set, X, the flow $f(X) \geq \frac{1}{2}\sum_{i \in X} f_i$, where f_i is the max-flow from i to t. That is, undirected flows are quasi-additive on irreducible sets.

1 Introduction

In the capacitated network design problem (Cap-NDP), we are given a graph $G = (V, E)$. Each edge e has a cost $c(e)$ and a capacity $u(e)$ which we assume to be non-negative integers. We are also given a collection of pairs of *terminals* $(s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k)$, where each s_i and t_i lies in V. Each pair is associated with an integer r_i. The objective is to find a *minimum cost* subgraph of G in which every s_i can send a flow of at least r_i units to t_i. We are *not* requiring these flows to be satisfied concurrently; the concurrent requirement leads to a different problem.

The problem generalizes many problems, the simplest of which is probably the minimum knapsack problem induced when the graph has two nodes and parallel edges between the same. When all the capacities are unit, and the graph is undirected, then the problem is what is called the *survivable* network design problem (SNDP) for which a 2-approximation is known [9]. SNDP is label-cover hard [4] in case of directed graphs; this already shows the hardness of the above problem for directed networks. In fact, Cap-NDP is label-cover hard for directed graphs even when there is a single pair of terminals [5, 3].

*Microsoft Research India, `dechakr@microsoft.com`
†Princeton University, `ravishan@cs.cmu.edu`
‡Princeton University, `shili@cs.princeton.edu`
§Carnegie Mellon University, `srivatsa@cs.cmu.edu`

For undirected graphs, algorithmically nothing better is known, and the hardness results were weaker. Hajiaghayi et al. [8] showed that the single pair Cap-NDP is as hard as the group Steiner tree problem. This implies a $\Omega(\log^2 n)$ hardness. At this point we should stress that we are *disallowing* picking multiple copies of an edge; if this were allowed, then a $O(\log k)$-approximation algorithm is known for undirected graphs [3]. In this paper, we prove that the single source-sink pair Cap-NDP is label-cover hard even in undirected graphs. More precisely, unless $NP \subseteq DTIME(n^{\text{polylog} n})$, for any $\delta > 0$, there is no $2^{\log^{1-\delta} n}$-approximation for the problem.

1.1 A Source Location Problem

Consider a capacitated network with a dedicated sink t and a subset of terminals $S \subseteq V$. Given a subset $X \subseteq S$, let $f(X) \in \mathbb{R}$ denote the maximum flow that can be sent from X to t. We use f_i as a shorthand for $f(\{i\})$. It is a standard fact that f is monotone and submodular: indeed, f is monotone because adding a new source can only increase the total flow, and f is submodular since the marginal flow due to a terminal decreases as X becomes larger. Consider the following problem: find the minimum cardinality subset $X \subseteq S$ such that $f(X) = f(S)$. This is a special case of Cap-NDP where we create a super-source s_1 and connect it to each source $s \in S$ by an edge of cost 1 and capacity $f(\{s\})$. Each edge in the original graph has cost 0 and capacity equal to its original capacity. The sink t_1 is the vertex t, and the requirement is r_1 is set to $f(S)$. Then the minimum cost graph H which can support a flow of r_1 between s_1 and t_1 exactly corresponds to the minimum cardinality set of sources which can send a flow $f(S)$ to the sink t.

The above problem is a special case of a widely studied class of problems known as *source location problems*. In its generality, each vertex $v \in G$ has a cost $c(v)$ and demand $d(v)$, and the goal is to pick a minimum cost subset X such that for all v, $f(X, v) \geq d(v)$. We obtain the above problem by setting $c(v) = 1$ for $v \in S$, and infinity otherwise, and demands $d(t) = 1$, and 0 otherwise. Source location problems have been studied extensively (see, for instance, [11]), and many special cases are known to be polynomial time solvable ([13, 14, 2]). In general, the problem is as hard as set cover ([12, 1]), and in fact, the reductions therein show that the above special case is also set-cover hard. Algorithmically, there is a logarithmic approximation [12], and for the above problem such a result follows by noting that problem is a special case of submodular set cover [15].

1.1.1 Irreducible Sets

In order to understand the above problem better, in this paper we study the *behavior* of the function f on undirected graphs. In particular, given a set $X \subseteq S$, can we lower bound $f(X)$ in terms of the f_i's for $i \in X$? Our main result answers this question affirmatively for a natural class of terminal sets X which we call *irreducible sets*. A subset $X \subseteq V$ is *irreducible* if $f(X) > f(X')$ for all *strict subsets* $X' \subsetneq X$. That is, removing any vertex from X strictly decreases the maximum flow that can be sent to the sink.

We now state our main positive result which shows that on such sets, the submodular function f behaves 'almost' additively.

Theorem 1. *Given an* undirected *graph G and any irreducible set X, $f(X) \geq \frac{1}{2}\sum_{i \in X} f_i$. Furthermore, there is an (X, t) flow of value $f(X)$ such that the total out-flow from terminal i is at least $f_i/2$, for all $i \in X$.*

We note that an analogous result is not true for directed graphs; in fact, the ratio $\sum_{i \in X} f_i / f(X)$ may be as large as $|X|$. To see this consider a collection of terminals with an arc to a vertex v and an arc from vertex v to t. All these arcs have capacity 1. Furthermore, each terminal has a direct arc to t of capacity ε. Note that X is irreducible; each terminal sends nonzero flow through its 'private' arc. However, $\sum_{i \in X} f_i / f(X)$ tends to $|X|$ as ε tends to 0. It's instructive to note that if the arcs were undirected, then the set X becomes *reducible*; the direct arcs aren't private anymore and other terminals can send flow through them. Thus the above example also shows irreducibility is necessary for the above theorem to hold.

We believe the condition of irreducibility is a natural extremal condition. For instance, in the source location problem above, it is easy to see that any reasonable solution will be irreducible. Therefore, we believe the theorem above can have many applications, we will illustrate the source location application in the following section. To take another example, in a telecommunication setting, the above theorem states that in an undirected capacitated network, any irreducible set of transmitters can transmit *concurrently* at, at least, half their maximum rates. The condition of irreducibility may be imposed by the central designer interested in the total throughput, to reduce operative costs.

1.1.2 Application to the above Source Location Problem.

Our main corollary of Theorem 1 is the following. Call an instance of the problem ρ-regular, if all the f_i's are within a ρ-multiplicative factor of each other.

Corollary 1. *For ρ-regular instances, there is a 2ρ-approximation to the source location problem.*

Proof. The algorithm is extremely simple: starting with S, keep on deleting vertices in any order as long as the deletion doesn't decrease the total flow to t, ending with a subset $X \subseteq S$. Now, by the nature of this procedure, $f(X) = f(S)$, and also X is irreducible as deleting any more vertices decreases the flow. We can now appeal to Theorem 1 to get that $f(S) = f(X) \geq \frac{1}{2}\sum_{i \in X} f_i \geq \frac{1}{2}|X| f_{min}$, where $f_{min} = \min_i f_i$. This gives $|X| \leq 2f(S)/f_{min}$. However, any solution X^* with $f(X^*) = f(T)$ satisfies $|X^*| f_{max} \geq \sum_{i \in X^*} f_i \geq f(S)$, implying $|X^*| \geq f(S)/f_{max} \geq |X| \cdot \frac{f_{min}}{2f_{max}}$. The proof follows from the ρ-regularity assumption. $\square$

We remark that *even with this regularity assumption.*, the directed version is inapproximable to within a factor of $o(\log n)$.[1] Furthermore, our approximation factor is optimal (assuming the unique games conjecture [10]): source location on undirected regular instances captures the *vertex cover problem* in regular graphs, which is inapproximable to within a factor of $2 - \varepsilon$, assuming the unique games conjecture [10, 6].

1.1.3 Proof Technique

How would one prove a theorem as Theorem 1? Arguably, one needs a handle on the structure of the cuts separating a subset of terminals from a given vertex. In undirected graphs the structure of cuts has been extensively investigated. For instance, there exist cactus representations for all the minimum cuts of the graph, and the Gomory-Hu tree captures the local edge connectivities of the graph. However, we are unaware of structural results capturing cuts separating a *set* of terminals from a sink.

Another syntactic way would be to deduce inequalities involving the flow function for various subsets of the irreducible set and combine them to obtain the stated result. For instance, suppose $|X| = 3$. Then, essentially, there are a constant number of types of vertices and edges, depending on which cuts they appear in. Subsequently, one can write inequalities capturing the cut conditions, and can obtain Theorem 1 for this special case. In fact, it may be a illuminating exercise for the reader to try this out. However, for larger sets, although this process is possible, it may not be feasible to do in a 'brute force' manner.

[1] Set cover is hard even restricted to instances with the regularity property, i.e., instances with uniform set sizes, via a simple approximation-preserving reduction from general instances.

Our approach can be thought of as a clever way of performing the above 'inequality setting'. We first show a mapping from *every* graph on to the k-dimensional hypercube, where k is the number of terminals. This mapping is 'cut-expanding': for every subset of terminals, the min-cut separating their images from the image of the sink is larger than that in the original graph. Furthermore, and this is the non-trivial part, certain min-cuts remain unchanged. This includes, in particular, the cut separating all the terminals from the sink. With this mapping, we show that the theorem need only be proved for the 'hypercube networks' that we construct. Our mapping is very similar to those used to generate what are called *mimicking networks*, and was first described by [7]. Once we go to the hypercube network, we show that the setting up of inequalities can be performed easily.

2 Proof of Theorem 1

Let G be an undirected network with an irreducible set T of terminals. We let k denote the number of terminals, that is, $|T|$. Also, for a subset $S \subseteq V(G)$, let $\delta_G(S)$ denote the capacity of the cut $(S, V(G) \setminus S)$ in G.

We let Hyp_k be the graph associated with the k-dimensional Boolean hypercube $\{0,1\}^k$. The vertices of Hyp_k are the k-dimensional Boolean vectors, and there is an edge between two vectors iff they differ in exactly one coordinate. We let H_k denote the graph Hyp_k along with an extra vertex t^* connected to the all 1s vector, 1^k.

2.1 From Graphs to Hypercubes

We now describe a mapping $\Phi : V(G) \to V(H_k)$ along with a capacity assignment to the edges of H_k. Let $S_i \subseteq V(G)$ be the inclusion-wise minimal min-cut separating the terminal set $T_i := T \setminus i$ from t. Obviously, all $j \in T \setminus i$ lies in S_i. Note that $\delta_G(S_i) = f(T \setminus i) < f(T)$ by the irreducibility of T. Therefore, $i \notin S_i$, since otherwise the minimum cut separating T from t would be strictly less than $f(T)$, violating the max-flow-min-cut theorem. This is the place where irreducibility is crucially used.

Let S denote the inclusion-wise min-cut separating T from t. We claim that $S_i \subseteq S$ for all $i \in T$. Suppose $S_i \nsubseteq S$. The sets $S_i \cup S$ and $S_i \cap S$ respectively separate T and T_i from the sink, so $\delta_G(S_i \cup S) \geq \delta_G(S)$ and $\delta_G(S \cap S_i) > \delta_G(S_i)$. Note that the second inequality is strict, thanks to the minimality of S_i. Thus $\delta_G(S_i \cup S) + \delta_G(S_i \cap S) > \delta_G(S) + \delta_G(S_i)$, contradicting the submodularity of the cut function.

Given S and S_i's for all $i \in T$, we define the mapping Φ as follows. If $v \notin S$, then $\Phi(v) = t^*$. For $v \in S$, $\Phi(v)$ is the element of Hyp_k such that $\Phi(v)_i = 0$ if $v \in S_i$; $\Phi(v) = 1$ otherwise. Observe the following: (a) $\Phi(t) = t^*$; (b) for $i \in T$, $\Phi(i)$ is the unit vector $\mathbf{e}_i$ which has 0's in all but the ith coordinate. This follows from our previous discussion that $i \notin S_i$ but $i \in S_j$ for all $j \neq i$.

We now describe the capacities on the edges of H_k. Initially all edges have capacity 0. For each edge $(u, v) \in E(G)$ of capacity c_{uv}, we will add capacities on the edges of $E(H_k)$. If both u and v are outside S, we do nothing. If both u and v are in S, and thus $\Phi(u)$ and $\Phi(v)$ lie in $V(\mathsf{Hyp}_k)$, then we add capacity c_{uv} on all the edges of the *canonical path* between $\Phi(u)$ to $\Phi(v)$. The canonical path from x to y in Hyp_k is $x =: x_0, x_1, \ldots, x_k := y$ where x_i agrees with y on the first i coordinates, and with x in the last $(k-i)$ coordinates. Note that x_i could be the same as x_{i+1} if x and y have the same ith coordinate. If $u \in S$ and $v \notin S$, then we add a capacity c_{uv} on all edges on the canonical path from $\Phi(u)$ to 1^k, and also to the edge $(1^k, t^*)$.

To differentiate between G and H_k, given a subset X of terminals, we henceforth let $f_G(X)$ denote $f(X)$, that is, the maximum flow from X to t in G. We let $f_H(X)$ denote the maximum flow from $\Phi(X)$ to

t^* in H_k with edge capacities as described above. Here, we use $\Phi(X)$ as a shorthand for $\{\Phi(x) : x \in X\}$.

Theorem 2. *Given a graph $G = (V, E)$ and an irreducible set of terminals $T \subseteq V$ of size k, the mapping $\Phi : V(G) \to V(H_k)$ as described above along with the capacity assignment on $E(H_k)$, has the following properties.*

(i) $f_G(X) \le f_H(X)$ for all subsets $X \subseteq T$. In particular, for singletons $X = \{i\}$.

(ii) $f_G(T) = f_H(T)$.

(iii) $f_G(T') = f_H(T')$ for all subsets $T' \subseteq T$ of size $k-1$.

Proof. (i) Consider any flow in the graph G from X to t. For any edge $(u, v) \in E(G)$ carrying positive flow, if u and v are both outside S, then $\Phi(u) = \Phi(v) = t^*$, so we do nothing. If both are inside S, then send the same amount of flow from $\Phi(u)$ to $\Phi(v)$ along the canonical path in the hypercube. By the capacity assignment, this is a feasible flow. If exactly one of them, say u, is in S, then we use the canonical path from u to 1^k, followed by the edge $(1^k, t^*)$. This shows a feasible flow of value $f_G(X)$ from $\Phi(X)$ to t^* in H_k.

(ii) From (i), it suffices to show that the capacity of the $(1^k, t^*)$ edge in H_k equals the (T, t) min-cut $\delta_G(S)$. By our construction, the $(1^k, t^*)$ gets capacity c_{uv} only for edges (u, v) with exactly one end point in S. This is precisely $\delta_G(S)$.

(iii) Let $T' = T \setminus i$. From (i), it suffices to exhibit a cut in H_k separating $\Phi(T \setminus i)$ and t^* of value $f_G(T \setminus i) = \delta_G(S_i)$. We claim that the ith *dictator cut* suffices. That is, the cut separating vertices $D_i := \{x \in \mathsf{Hyp}_k : x_i = 0\}$ from the rest of the vertices in H_k. Firstly note that D_i contains $\Phi(T \setminus i)$ and t^* lies outside D_i. So this is a valid $(\Phi(T \setminus i), t^*)$ cut. Furthermore, the only edges crossing this cut belong to Hyp_k.

Consider an edge (x, y) in Hyp_k crossing D_i with say $x_i = 0$. The capacity on this edge is contributed by edges (u, v) which have (x, y) in the canonical path from $\Phi(u)$ to $\Phi(v)$. In particular, $\Phi(u)_i = 0$ and $\Phi(v)_i = 1$; that is, $u \in S_i$ and $v \notin S_i$ and $(u, v) \in \delta_G(S_i)$. Furthermore, since this is a dictator cut, no canonical path crosses this cut more than once. In particular, the capacity of this cut is exactly the total capacity of these edges (u, v), and thus is precisely $\delta_G(S_i)$. $\square$

2.2 Bounding the Flow on the Hypercube Graph H_k

Lemma 1. $f_H(T) \ge \frac{1}{2} \sum_{i \in T} f_H(i)$.

Proof. For $1 \le i < k$, let L_i denote the set of edges $(x, y) \in E(\mathsf{Hyp}_k)$ such that x has precisely i ones and y has $(i + 1)$ ones. Moreover, let L_k consist of the single edge $(1^k, t^*)$. We abuse notation and let L_i also denote the total capacity of the edges in L_i. Recall, $\Phi(i) = \mathbf{e}_i$. Thus the 'singleton cut' separating $\mathbf{e}_i$ from the remaining vertices is an upper bound on $f_H(i)$. Furthermore, all these singleton cuts are disjoint, and their union is $L_0 \cup L_1$. This gives

$$L_0 + L_1 \ge \sum_{i \in T} f_H(i). \tag{1}$$

Observe that for any $1 \le i \le k$, the edge set L_i separates t^* from $\Phi(T)$. Therefore, we get

$$L_i \ge f_H(T). \tag{2}$$

Finally, recall from the proof of (iii) of Theorem 2, that each dictator cut D_i has value $f_G(T \setminus i) < f_G(T) = f_H(T)$. Since each edge of the hypercube appears in exactly one dictator cut D_i, by adding this over all $1 \leq i \leq k$, we get

$$\sum_{0 \leq i \leq k-1} L_i \leq k \cdot f_H(T). \tag{3}$$

Using (2) for $2 \leq i \leq k-1$, the above inequality becomes

$$L_0 + L_1 \leq 2 \cdot f_H(T). \tag{4}$$

Comparing (1) and (4) gives the lemma. $\square$

Theorem 2 and Lemma 1 imply the first part of Theorem 1.

To prove the second part, we introduce a dummy source s to G and connect it to every vertex in T with capacity of (s, i) edge being $f_i/2$. We claim that the minimum cut in this network is of value precisely $\sum_{i \in T} f_i/2$. If so, then the resulting maxflow will imply the second part of Theorem 1.

Suppose not, and let the mincut be $(Z, V \cup s \setminus Z)$ with $s \in Z$. Let $X := T \cap Z$. Let F be the edges in $\delta(Z)$ which have endpoints in V. Let C be the total capacity of the edges in F. Since the mincut is $< \sum_{i \in T} f_i/2$, we get that $C < \sum_{i \in X} f_i/2$. However, F separates X from t, and thus the maxflow from X to t is $\leq C$. But $X \subseteq T$ is irreducible as well, and thus this violates part one of Theorem 1. To see this irreducibility of X note that if $f(X \setminus i) = f(X)$, by submodularity of f, this would imply $f(T \setminus i) = f(T)$ as well.

3 Hardness of single source-sink pair Cap-NDP

We show that the single source undirected Cap-NDP is label cover hard. The reduction is actually from the directed instances which showed label-cover hardness for directed Cap-NDP [5, 3].

Consider a collection $\mathcal{G}$ of graphs obtained as follows. V consists of the following vertices. A set A of nodes partitioned into sets $A_1, \ldots, A_k$. A set B partitioned into $B_1, \ldots, B_k$. There are directed arcs of cost 0 and capacity 1 all of which are directed from some node in A_i to some node in B_j. There are nodes $a_1, \ldots, a_k$ and similarly $b_1, \ldots, b_k$. There is an arc (a_i, v) of capacity ∞ and cost C, for all $v \in A_i$. Similarly, there is an arc (v, b_j) of capacity ∞ and cost C, for all $v \in B_j$. Finally, there is an arc (s, a_i) of cost 0, capacity ∞ for all $i \in [k]$, and an arc (b_j, t) of cost 0, capacity ∞ for all $j \in [k]$. Let's call the capacity ∞ edges *big* edges. There is only one pair (s, t) with requirement R for some R (see Figure 1(a)). The reductions of [5, 3] show that single source Cap-NDP is label-cover hard even on these instances.

Theorem 3 ([5, 3]). *Unless $NP \subseteq DTIME(n^{\mathrm{polylog}(n)})$, there is no $2^{\log^{1-\delta}(n)}$-approximation algorithm for* Cap-NDP *for directed graphs coming from class $\mathcal{G}$.*

We now show how we obtain the hardness result for undirected graphs. If we simply make all edges undirected, the instance is not necessarily hard since the flows may travel along reverse directions. Given an undirected graph G obtained from the above instance by removing directions, we describe a simple trick that makes all capacity-1 edges (the edges between A and B) directed from left to right. This is enough for the hardness result.

Let M denote the number of capacity-1 edges. We add nodes s' and t' to V, edges (s, s'), (t', t) of cost 0 and capacity $M/2$. Furthermore, we add edges (s', v) for all $v \in B$ and (t', v) for all $v \in A$. The capacity of these edges are $d(v)/2$, where $d(v)$ is the number of capacity-1 edges incident to v. The costs of all these edges are 0. Finally, we change the capacities of the capacity-1 edges to $1/2$. The demand r_{st} is set to $R + M/2$ (see Figure 1(b)).

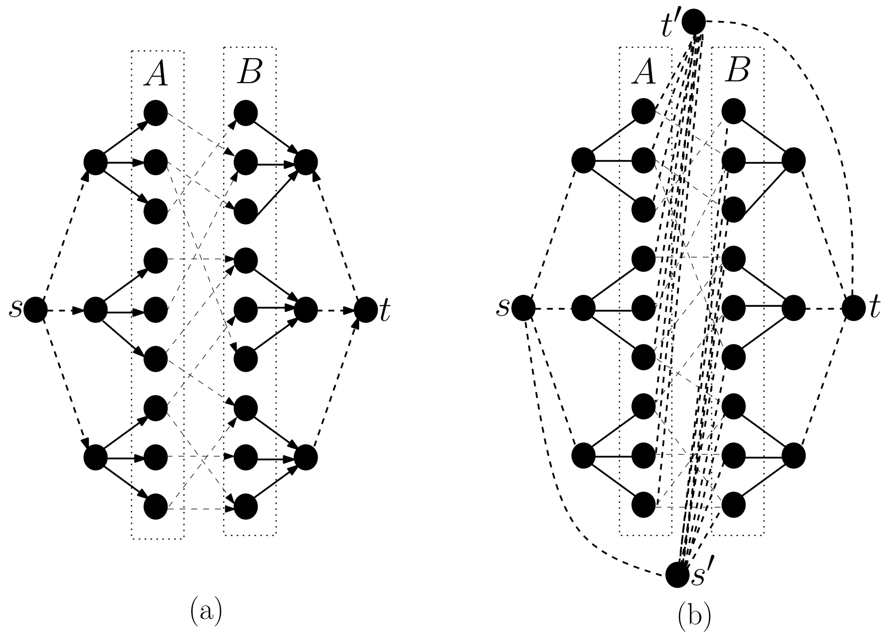

Figure 1: The graph on the left side is the hard instance of Cap-NDP for directed graphs, and the graph on the right side is the hard instance of Cap-NDP for undirected graphs. Solid lines represent edges of cost C and dashed lines represent edges of cost 0. All edges in the left graph have capacities ∞, except for the edges from A to B, which have capacities 1. In the right graph, (s, s') and (t', t) have capacities $M/2$, where M is the number of edges between A and B. An edge between s' and $v \in B$ has capacity $d(v)/2$, where $d(v)$ is the number of edges between v and A. An edge between $v \in A$ and t' has capacity $d(v)/2$, where $d(v)$ is the number of edges between v and B. The edges between A and B have capacities $1/2$. All other edges have capacity ∞.

The intuition for the above construction is as follows. Since the capacity-$1/2$ edges (original capacity-1 edges) and the newly added edges have cost 0, we can assume they are included in the solution. With these edges, we can send $M/2$ units flow from s to t in the natural way: the flows go from s to s', then to vertices in B, then to vertices in A, to t' and finally to t. The flows use all the capacities of these edges. The remaining task is to select some other edges so that we can send R units flow in the residual graph. Notice that in the residual graph, all the capacity-$1/2$ edges are directed from left to right, with capacities 1. It is easy to see that the new added edges are useless in the residual graph. Thus, the remaining problem is equivalent to the original instance (Figure 1(a)) of Cap-NDP for directed graphs.

Now we give a more formal proof. Consider a solution to the undirected Cap-NDP. We may assume all the cost 0 edges are picked. Let F be the non-zero cost edges in the solution. Note that all of these are of the form (a_i, v) for some $v \in A_i$, or (v, b_j) for some $v \in B_j$. We abuse notation and let F also denote the corresponding arcs in the original digraph.

Claim 1. *F, along with the 0-cost arcs, is a valid solution for the directed* Cap-NDP *instance.*

Proof. Let $S \subseteq A$ be the set $\{v : (a_i, v) \in F, \text{ for some } i\}$. Similarly, let $T \subseteq B$ be the set of endpoints in B neighboring to some edge in F. We claim that the edges with one endpoint in S and the other in T, which we denote as $E(S : T)$, satisfies $|E(S : T)| \geq R$. Assuming this, we are done since the arcs are indeed directed from S to T, and since each vertex in S can receive R units of flow from s, and each vertex in T can send R units of flow to t (since these are neighboring to F), we get a feasible solution for the directed case.

Consider the following cut in the *undirected* graph with F and 0-cost edges. On the s side we have $s, \{a_1, \ldots, a_k\}, s'$ and $S \cup B \setminus T$. The t side contains the complement, that is, $t, \{b_1, \ldots, b_k\}, t'$ and $T \cup A \setminus S$. Observe there are no big edges in the cut. Big edges are either of the form (s, a_i),(t, b_j) or (a_i, v),(b_j, v). The first type are inside the s side or the t side; the second type has only F and the endpoints are made sure to be on the same side of the cut.

Therfore, the cut edges are precisely $E(S : T)$, $E(A \setminus S : B \setminus T)$ and the new edges $E(s' : T) \cup E(S : t')$. Let the capacites of these three sets be C_1, C_2, C_3. $C_1 = \frac{1}{2}|E(S : T)|$ is the quantity of interest. $C_3 = \frac{1}{2}d(S) + \frac{1}{2}d(T)$, where $d(X)$ is the shorthand for $\sum_{v \in X} d(v)$. $2C_2 = d(A \setminus S) - |E(A \setminus S : T)| = d(A) - d(S) - (d(T) - |E(S : T)|) = d(A) - d(S) - d(T) + |E(S : T)|$. Thus the total capacity of this cut is $C_1 + C_2 + C_3 = \frac{1}{2}(d(A) + 2|E(S : T)|) = M/2 + |E(S : T)|$, since $d(A)$ is nothing but the number of capacity 1 edges. The capacity of the cut is $\geq M/2 + R$ since F is a feasible solution, which implies $|E(S : T)| \geq R$. Therefore, F with the 0-cost arcs form a valid solution to the directed problem as well. □

The above claim, along with Theorem 3, gives the following theorem.

Theorem 4. *Unless $NP \subseteq DTIME(n^{\text{polylog}(n)})$, there is no $2^{\log^{1-\delta}(n)}$-approximation for undirected, single source-sink pair* Cap-NDP.

4 Conclusion

We conclude the note with a couple of observations. There is a special case of Cap-NDP, which has been called the k-bipartite flow problem by [8], where given a bipartite graph with node costs and unit capacity edges, the goal is to find subsets of nodes A, B from the left and right part of minimum total cost such that the edge connectivity between A and B is at least k. In directed graphs this generalizes the densest k-subgraph problem (the version where one needs to pick the minimum number of vertices which has at least k induced edges). A similar reduction as above shows that the undirected case, and thus undirected, single source Cap-NDP, is as hard as the densest k-subgraph problem. If the goal is to just pick a min-cost subset A from one part and the set B is fixed, then a logarithmic approximation exists, and a reduction as above shows that the problem is as hard as the set cover problem.

However, one should note that the inapproximability described above only rules out unicriteria results. For instance, we haven't ruled out a solution of cost $\text{polylog}(n)OPT$ which sends $\geq R/2$ flow (The reduction of [8] show a logarithmic hardness for any solution sending $\Omega(R)$ flow.) In fact, for the k-bipartite flow stated above, there is a solution [1] via the Räcke decomposition into trees, which obtains a solution of cost equaling OPT and sends $R/\text{polylog}(n)$ flow. We think this direction may be feasible; as a starting point we ask whether there is a $(O(\text{polylog}(n)), O(1))$-approximation for the k-bipartite flow problem.

References

[1] K. Andreev, C. Garrod, D. Golovin, B. Maggs, and A. Meyerson, *Simultaneous source location*, Trans. on Algorithms (TALG) **6** (2009), no. 1, 1–17. 2, 9

[2] K. Arata, S. Iwata, K. Makino, and S. Fujishige, *Locating sources to meet flow demands in undirected networks*, J. of Algorithms **42** (2002), 54 – 68. 2

[3] D. Chakrabarty, C. Chekuri, S. Khanna, and N. Korula, *Approximability of capacitated network design*, Proceedings, MPS Conference on Integer Programming and Combinatorial Optimization (IPCO), 2011. 1, 2, 6

[4] Y. Dodis and S. Khanna, *Design networks with bounded pairwise distance*, ACM Symp. on Theory of Computing (STOC), 1999. 1

[5] G. Evens, G. Kortsarz, and W. Slany, *On network design: fixed charge flows and the covering steiner problem.*, Trans. on Algorithms (TALG) **1** (2005), no. 1, 74–101. 1, 6

[6] U. Feige, *Vertex cover is hardest to approximate on regular graphs*, Tech. report, Weizmann Institute, 2004. 3

[7] T. Hagerup, J. Katajainen, N. Nishimura, and P. Ragde, *Characterizations of multiterminal flow networks and computing flows in networks of bounded treewidth*, Journal of Computer and System Sciences (JCSS) **57** (1998), 366–375. 4

[8] M. T. Hajiaghayi, R. Khandekar, G. Kortsarz, and Z. Nutov, *Capacitated network design problems: hardness, approximation algorithms, and connections to group steiner tree.*, 2012. 2, 8, 9

[9] K. Jain, *A factor 2 approximation algorithm for the generalized Steiner network problem.*, Combinatorica **21** (2001), no. 1, 39–60. 1

[10] S. Khot and O. Regev, *Vertex cover might be hard to approximate to within* $2 - \varepsilon$., Proceedings of FOCS (2003). 3

[11] M. Labbe, D. Peeters, and J.-F. Thisse, *Location on networks*, Handbook in OR and MS **8** (1995), 551–624. 2

[12] K. Makino M. Sakashita and S. Fujishige, *Minimum cost source location with flow requirements.*, Proceedings, Latin American Theoretical Informatics Symposium, 2006. 2

[13] H. Tamura, M. Sengoku, S. Shinoda, and T. Abe, *Location problems on undirected flow networks.*, IEICE Trans. **E** (1990), no. 73, 1989–1993. 2

[14] ______, *Some covering problems in location theory on flow networks*, IEICE Trans. **E** (1992), no. 75, 678–683. 2

[15] L. Wolsey, *An analysis of the greedy algorithm for the submodular set covering problem.*, Combinatorica **2** (1982), no. 4, 385–393. 2