# Approximating node-connectivity augmentation problems*

Zeev Nutov

The Open University of Israel

nutov@openu.ac.il

**Abstract**

We consider the (undirected) Node Connectivity Augmentation (NCA) problem: given a graph $J = (V, E_J)$ and connectivity requirements $\{r(u,v) : u,v \in V\}$, find a minimum size set $I$ of new edges (any edge is allowed) such that the graph $J \cup I$ contains $r(u,v)$ internally-disjoint $uv$-paths, for all $u,v \in V$. In Rooted NCA there is $s \in V$ such that $r(u,v) > 0$ implies $u = s$ or $v = s$. For large values of $k = \max_{u,v \in V} r(u,v)$, NCA is at least as hard to approximate as Label-Cover and thus it is unlikely to admit an approximation ratio polylogarithmic in $k$. Rooted NCA is at least as hard to approximate as Hitting-Set. The previously best approximation ratios for the problem were $O(k \ln n)$ for NCA and $O(\ln n)$ for Rooted NCA. In this paper we give an approximation algorithm with ratios $O(k \ln^2 k)$ for NCA and $O(\ln^2 k)$ for Rooted NCA. This is the first approximation algorithm with ratio independent of $n$, and thus is a constant for any fixed $k$. Our algorithm is based on the following new structural result which is of independent interest. If $\mathcal{D}$ is a set of node pairs in a graph $J$, then the maximum degree in the hypergraph formed by the inclusion minimal tight sets separating at least one pair in $\mathcal{D}$ is $O(\ell^2)$, where $\ell$ is the maximum connectivity in $J$ of a pair in $\mathcal{D}$.

*A preliminary version of this paper is [29].

# 1 Introduction

## 1.1 Problem definition

Let $\kappa_G(u,v)$ denote the maximum number of internally-disjoint $uv$-paths in a graph $G$. We consider the following fundamental problem in network design:

Node-Connectivity Augmentation (NCA):
*Instance:* A graph $J = (V, E_J)$, connectivity requirements $\{r(u,v) : u,v \in V\}$.
*Objective:* Find a minimum size set $I$ of new edges such that the graph $G = J \cup I$ satisfies

$$\kappa_G(u,v) \geq r(u,v) \quad \text{for all } u,v \in V \ . \tag{1}$$

In general, all graphs are assumed to be undirected, and may have parallel edges. In the Simplicity Preserving NCA (SPNCA) variant of NCA, the graph $J$ is simple, and the graph $G = J \cup I$ is required to be simple. Note that if SPNCA has a feasible solution, then $n \geq k+1$ must hold. For a problem instance at hand, let opt denote the optimal solution value, let $k = \max_{u,v \in V} r(u,v)$ denote the maximum connectivity requirement, and let $n = |V|$.

If all the connectivity requirements are "rooted", namely from a specific node $s$, then we have the following important particular case of NCA:

Rooted NCA:
*Instance:* A graph $J = (V, E_J)$, a root $s \in V$, and requirements $\{r(v) : v \in V\}$.
*Objective:* Find a minimum size set $I$ of new edges such that the graph $G = J \cup I$ satisfies $\kappa_G(s,v) \geq r(v)$ for all $v \in V$.

NCA is an extensively studied particular case of the following problem, that recently received a renewed attention:

Survivable Network Design (SND):
*Instance:* A complete graph on node set $V$ with edge-costs, and connectivity requirements $\{r(u,v) : u,v \in V\}$.
*Objective:* Find a minimum cost subgraph $G$ on $V$ that satisfies (1).

NCA is equivalent to SND with $0,1$-costs, when $E_J$ is the set of edges of cost 0, and any other edge is allowed and has cost 1. The case of $1,\infty$-costs of SND gives the min-size subgraph problems, when we seek a solution using the edges of cost 1 only.

## 1.2 Our Results

NCA admits an $O(k \ln n)$-approximation algorithm [21, 22], and is unlikely to admit a polylogarithmic approximation in $k$ even for $\{0, k\}$-requirements [28]. For rooted requirements, an $O(\ln n)$-approximation is known [21], and for $k = \Omega(n)$ this is tight [26, 22]. As for small values of $k$ the problem can be solved in polynomial time or admits a constant ratio approximation algorithm, the author posed in [28] the following question: Does NCA admit a $\rho(k)$-approximation algorithm? Here $\rho(k)$ is a functions that depends on $k$ only. We resolve this question for both NCA and SPNCA, thus obtaining a constant ratio for any constant $k$; furthermore, when $\ln^2 k = o(\ln n)$ our ratios are better than the ones in [21].

**Theorem 1.1** *Both* NCA *and* SPNCA *admit the following approximation ratios:*

- $O(k \ln^2 k)$ *for arbitrary requirements (improving* $O(k \ln n)$*);*
- $O(\ln^2 k)$ *for rooted requirements (improving* $O(\ln n)$*).*

*Here* $k = \max_{u,v \in V} r(u, v)$ *is the maximum requirement.*

As an intermediate problem, we consider NCA instances with $r(u, v) \leq \kappa_J(u, v) + 1$ for all $u, v \in V$. That is, given a set $\mathcal{D}$ of node pairs, we seek to increase the connectivity by 1 between pairs in $\mathcal{D}$, meaning $r(u, v) = \kappa_J(u, v) + 1$ for all $\{u, v\} \in \mathcal{D}$ and $r(u, v) = 0$ otherwise. Formally, intermediate problem we consider is as follows.

$\mathcal{D}$-NCA:
*Instance:* A graph $J = (V, E_J)$ and a set $\mathcal{D}$ of unordered node pairs from $V$.
*Objective:* Find a minimum size edge-set $I$ such that the graph $G = J \cup I$ satisfies

$$\kappa_G(u, v) \geq \kappa_J(u, v) + 1 \quad \text{for all } \{u, v\} \in \mathcal{D} . \tag{2}$$

Given an edge-set or a graph $J$ and disjoint node-sets $X, Y$ let $\delta_J(X, Y)$ denote the set of edges in $J$ that have one endnode in $X$ and the other in $Y$; let $\delta_J(X) = \delta_J(X, V \setminus X)$. For $S \subseteq V$ let $\Gamma_J(S) = \Gamma(S) = \{v \in V \setminus S : \delta_J(u, v) \neq \emptyset \text{ for some } u \in S\}$ denote the set of *neighbors* of $S$ in the graph $J$.

**Definition 1.1** *Given an instance of* $\mathcal{D}$-NCA *or of* $\mathcal{D}$-SPNCA*, we say that* $S \subseteq V$ *is* $uv$-tight *if* $u \in S$, $v \notin S$, $\delta_J(S, v) = \delta_J(u, v)$, *and* $\kappa_J(u, v) = |\Gamma_{J \setminus \{v\}}(S)| + \delta_J(u, v)$. *We say that* $S$ *is* tight *if it is* $uv$*-tight for some* $\{u, v\} \in \mathcal{D}$. *Let* $\mathcal{C}_J(\mathcal{D})$ *denote the set of inclusion-minimal tight sets in* $J$ *w.r.t.* $\mathcal{D}$*, and in the case of rooted requirements let* $\mathcal{C}^s_J(\mathcal{D}) = \{C \in \mathcal{C}_J(\mathcal{D}) : s \notin C\}$.

The proof of Theorem 1.1 is based on the following theorem, which is of independent interest.

**Theorem 1.2** *Suppose that* $\max_{\{u,v\}\in\mathcal{D}} \kappa_J(u,v) = \ell$ *for an instance of* $\mathcal{D}$-NCA *or of* $\mathcal{D}$-SPNCA*. Then the maximum degree in the hypergraph* $(V, \mathcal{C}_J(\mathcal{D}))$ *is at most* $(4\ell+1)^2$*. For rooted requirements, the maximum degree in the hypergraph* $(V, \mathcal{C}^s_J(\mathcal{D}))$ *is at most* $2\ell+1$.

We believe that the result in the theorem reveals a fundamental property which will have further applications, and may to be useful to design approximation algorithms for various SND problems. Specifically, the approach in this paper was later used by the author in [25] to obtain an $O(k \ln k)$-approximation for Rooted SND with arbitrary costs, which is currently the best known ratio for the problem.

Theorems 1.1 and 1.2 are proved in Sections 2 and 3, respectively. Section 4 concludes with some open problems.

## 1.3 Previous and related work

Variants of SND, and especially of NCA, were vastly studied. See surveys in [20, 9]. While the edge-connectivity variant of SND – the so called Steiner Network problem – admits a 2-approximation algorithm due to Jain [15], no such algorithm is known for SND. For directed graphs, Dodis and Khanna [5] showed that $\{0,1\}$-SND – the so called Directed Steiner Forest problem – is at least as hard to approximate as Label-Cover, which implies that the problem is unlikely to admit a polylogarithmic approximation ratio. By extending the construction of [5], Kortsarz, Krauthgamer, and Lee [18] showed a similar hardness result for Undirected $\{0,k\}$-SND; the same hardness is valid even for $\{0,1\}$-costs, namely, for NCA, see [26]. However, the edge-connectivity variant of NCA – the so called Edge-Connectivity Augmentation problem admits a polynomial time algorithm due to Frank [8].

In general, for small requirements, undirected variants of SND are substantially easier to approximate than the directed ones. For example, Undirected Steiner Tree/Forest admits a constant ratio approximation algorithm, while the directed variants are not known to admit even a polylogarithmic approximation ratio. The currently best known approximation lower bound for Directed Steiner Tree is $\Omega(\ln^{2-\varepsilon} n)$ [12], while a long standing best known ratio is $O(n^\varepsilon/\varepsilon^3)$ in $O(|n|^{4/\varepsilon} n^{2/\varepsilon})$ time [2]; this gives an $n^\varepsilon/\varepsilon^3$-approximation scheme. In what follows, we survey results for general SND/NCA, Rooted SND/NCA, and the $k$-Connected Subgraph ($k$-CS) problem, for both general and $0,1$-costs; the latter is a famous particular case of SND when $r(u,v) = k$ for all $u,v \in V$. See a survey in [20]. We consider the cases of

| Costs | Req. | Approximability | |
|---|---|---|---|
| | | *Undirected* | *Directed* |
| general | general | $O(\min\{k^3 \ln n, n^2\}$ [4], $k^{\Omega(1)}$ [1] | $O(n^2)$, $\Omega(2^{\ln^{1-\varepsilon} n})$ [5] |
| general | rooted | $O(\min\{k \ln k, n\})$ [25], $\Omega(\ln^2 n)$ [22] | $O(n)$, $\Omega(\ln^2 n)$ [12] |
| general | $k$-CS | $O\left(\ln k \cdot \ln \frac{n}{n-k}\right)$ [27] | $O\left(\ln k \cdot \ln \frac{n}{n-k}\right)$ [27] |
| metric | general | $O(\ln k)$ [3] | $O(n^2)$, $\Omega(2^{\ln^{1-\varepsilon} n})$ [5] |
| metric | rooted | $O(\ln k)$ [3] | $O(n)$, $\Omega(\ln^2 n)$ [12] |
| metric | $k$-CS | $2 + \frac{k-1}{n}$ [19] | $2 + \frac{k}{n}$ [19] |
| $0, 1$ | general | $O(k \ln n)$ [21], $\Omega(2^{\ln^{1-\varepsilon} n})$ [28] | $O(k \ln n)$ [21], $\Omega(2^{\ln^{1-\varepsilon} n})$ [28] |
| $0, 1$ | rooted | $O(\ln n)$ [21], $\Omega(\ln n)$ [26] | $O(\ln n)$ [21], $\Omega(\ln n)$ [26] |
| $0, 1$ | $k$-CS | $\min\{\mathsf{opt} + k^2/2, 2\mathsf{opt}\})$ [13] | in P [10] |

Table 1: Approximation ratios and hardness results for SND problems.

general costs (SND) and of $0, 1$-costs (NCA) separately. The approximability of various SND problems (prior to our work) is summarized in Table 1.

**SND–arbitrary costs:** Frank and Tardos [11] gave a polynomial time algorithm for rooted uniform requirements $r(s, v) = k$ for all $v \in V \setminus \{s\}$. Ravi and Williamson [30] gave a 3-approximation algorithm for $\{0, 1, 2\}$-SND, and the ratio was improved to 2 by Fleisher et al. [7]. As was mentioned, SND is unlikely to admit a polylogarithmic approximation [18]; a recent hardness result of Chakraborty, Chuzhoy, and Khanna [1] shows that SND with requirements in $\{0, k\}$ is $k^{\Omega(1)}$-hard to approximate. Recently, it was shown in [22] that directed SND problems can be reduced to their corresponding undirected variants with large connectivity requirements; one of the consequences of the result of [22] is that the Rooted SND with requirements in $\{0, k\}$ is at least as hard to approximate as the notorious Directed Steiner Tree problem, for $k \geq n/2$. The reduction of [22] does *not* preserves *metric costs*, and indeed, Cheriyan and Vetta [3] showed that (undirected) SND with metric costs admits an $O(\ln k)$-approximation algorithm. However, no sublinear approximation algorithm is known for SND with general requirements and costs. Even for the much easier Directed Steiner Forest problem, the best ratio known in terms of $n$ is $O\left(n^{4/5+\varepsilon}\right)$ [6]. Chakraborty, Chuzhoy, and Khanna [1] initiated recently the study of approximation algorithms for SND problems when the parameter $k$ is not too large. The currently best known ratios for SND with arbitrary costs are $O(k^3 \ln n)$ for arbitrary requirements [4], and $O(k \ln k)$ for Rooted SND [25]. The most famous variant of SND is the $k$-Connected Subgraph problem. The currently best known ratio for directed/undirected $k$-Connected Subgraph is $O\left(\ln k \cdot \ln \frac{n}{n-k}\right)$ [27].

**NCA–0, 1-costs:** While most of the "positive" literature on SND problems with general costs is from the recent 2 years, $0,1$-costs NCA problems were extensively studied already in the 90's. For example, the complexity status of $k$-Connected Subgraph with $0,1$-costs is among the oldest open problems in network design, see [16, 17, 13, 14, 26, 23] (however, the directed case is solvable in polynomial time [10]). In [16, 17] Jordán gave an $\mathsf{opt} + k/2$ approximation for the problem of increasing the connectivity by 1, and for a long time it was not known that the problem is in P, nor that it is NP-hard. Recently, Vegh [31] obtained a polynomial time algorithm for this case of increasing the connectivity by 1, but the complexity status of the more general case is still open. Jordán's algorithm [16, 17] was generalized by Jackson and Jordán [13] who gave an algorithm that computes a solution of size roughly $\mathsf{opt} + k^2/2$ for the general $k$-Connected Subgraph with $0,1$-costs. Another very interesting result of Jackson and Jordán [14] shows that the problem can be solved exactly in time $2^{f(k)} poly(n)$.

For general requirements, NCA admits an $O(k \ln n)$-approximation algorithm [21], and is unlikely to admit a polylogarithmic approximation [28]. For rooted requirements an $O(\ln n)$-approximation is known, and for $k = \Omega(n)$ this is tight [26].

## 2 The algorithm (proof of Theorem 1.1)

Here we prove Theorem 1.1, which is restated for the convenience of the reader.

**Theorem 1.1** *Both* NCA *and* SPNCA *admit the following approximation ratios:*

- $O(k \ln^2 k)$ *for arbitrary requirements (improving* $O(k \ln n)$*);*
- $O(\ln^2 k)$ *for rooted requirements (improving* $O(\ln n)$*).*

*Here* $k = \max_{u,v \in V} r(u,v)$ *is the maximum requirement.*

Theorem 1.1 is proved in several steps, and relies on Theorem 1.2, which is proved in the next section. We start with the following known fact that is proved using standard flow-cut techniques.

**Proposition 2.1** *The family* $\mathcal{C}_J(\mathcal{D})$ *can be computed in polynomial time and* $|\mathcal{C}_J(\mathcal{D})| \leq 2|\mathcal{D}| \leq n(n-1)$.

**Proof:** It is well known that given $\{u,v\} \in \mathcal{D}$, one max-flow computation suffices to find the unique minimal $uv$-tight set $C^u_{uv}$ containing $u$, and the unique minimal $vu$-tight set $C^v_{vu}$

containing $v$. The family $\mathcal{C}_J(\mathcal{D})$ consists from the inclusion minimal members of the family of all such sets $C^u_{uv}, C^v_{vu}$, two sets for every pair $\{u,v\} \in \mathcal{D}$. The statement follows. $\square$

We now describe the lower bound on the solution size of $\mathcal{D}$-NCA and $\mathcal{D}$-SPNCA that we use.

**Definition 2.1** *A node set $T \subseteq V$ is a $\mathcal{C}$-transversal of a set-family $\mathcal{C}$ if $T$ intersects every $C \in \mathcal{C}$. Let $\tau(\mathcal{C})$ be the minimum size of a $\mathcal{C}$-transversal, and let $\tau^*(\mathcal{C})$ be the minimum value of a fractional $\mathcal{C}$-transversal, namely:*

$$\tau^*(\mathcal{C}) = \min\{\sum_{v \in V} x(v) : \sum_{v \in C} x(v) \geq 1 \ \ \forall C \in \mathcal{C}, \ x(v) \geq 0\} \ .$$

Note that $|I| \geq \tau(\mathcal{C}_J(\mathcal{D}))/2 \geq \tau^*(\mathcal{C}_J(\mathcal{D}))/2$ for any feasible solution $I$ for $\mathcal{D}$-NCA. Indeed, by Menger's Theorem, $I$ is a feasible solution to $\mathcal{D}$-NCA if, and only if, for any $uv$-tight set $S$ with $\{u,v\} \in \mathcal{D}$ there is an edge in $I$ from $S$ to $V \setminus (S \cup \Gamma_J(S) \setminus \{v\})$. In particular, $|\delta_I(C)| \geq 1$ must hold for any $C \in \mathcal{C}_J(\mathcal{D})$. Thus the endnodes of the edges in $I$ form a $\mathcal{C}_J(\mathcal{D})$-transversal, so $|I| \geq \tau_J(\mathcal{D})/2$. Note also that in the case of rooted requirements, $\tau^*(\mathcal{C}_J(\mathcal{D})) \geq \tau^*(\mathcal{C}^s_J(\mathcal{D}))$. By a similar argument, the same lower bound is valid for $\mathcal{D}$-SPNCA.

Given a hypergraph $(V, \mathcal{C})$, the greedy algorithm of Lovász [24] computes in polynomial time a $\mathcal{C}$-transversal $T$ of size $\leq H(\Delta(\mathcal{C}))\tau^*(\mathcal{C})$, where $\Delta(\mathcal{C})$ is the maximum degree of the hypergraph and $H(k)$ is the $k$th Harmonic number. Combining with Theorem 1.2 we deduce the following statement.

**Corollary 2.2** *There exists a polynomial time algorithm that given an instance of $\mathcal{D}$-NCA or of $\mathcal{D}$-SPNCA computes a $\mathcal{C}_J(\mathcal{D})$-transversal $T$ such that $|T| \leq \tau^*(\mathcal{C}_J(\mathcal{D})) \cdot H\left((4\ell+1)^2\right)$, where $\ell = \max_{\{u,v\} \in \mathcal{D}} \kappa_J(u,v)$; for Rooted NCA or for Rooted SPNCA, the algorithm computes a $\mathcal{C}^s_J(\mathcal{D})$-transversal $T$ such that $|T| \leq \tau^*(\mathcal{C}^s_J(\mathcal{D})) \cdot H(2\ell+1)$.*

Now we show how to obtain an augmenting edge set from a given transversal. In what follows, note that if $\mathcal{D}$-SPNCA has a feasible solution then $n \geq k+1 = \ell+2$ must hold.

**Proposition 2.3** *There exists a polynomial time algorithm that given an instance of $\mathcal{D}$-NCA or of $\mathcal{D}$-SPNCA and a $\mathcal{C}_J(\mathcal{D})$-transversal $T$, computes a feasible solution $I$ such that $|I| \leq (\ell+2)|T|$.*

**Proof:** If $n \geq \ell+2$ (in particular, in the case of $\mathcal{D}$-SPNCA), then form an edge set $I$ by choosing an arbitrary set $U$ of $\ell+2$ nodes and connecting every node in $T$ to every node in $U$, unless there is already an edge between them. Then $|I| \leq (\ell+2) \cdot |T|$. We claim that $I$ is a feasible solution. Suppose to the contrary that $\kappa_J(u,v) = \kappa_{J \cup I}(u,v) = \ell' \leq \ell$ for some

$\{u,v\} \in \mathcal{D}$. By Menger's Theorem, there exists a partition $X, C, Y$ of $V$ such that $X$ is $uv$-tight, $Y$ is $vu$-tight, $\delta_I(X,Y) = \emptyset$ and $|C| + |\delta_J(X,Y)| = \ell'$. There is $z \in U \setminus C$, and in the case $|C| \leq \ell' - 1$ there is $z \notin C \cup \{u,v\}$. As $T$ is a $\mathcal{C}_J(\mathcal{D})$-transversal there are $x \in X \cap T$ and $y \in Y \cap T$. At least one of the edges $zx, zy$ is in $\delta_I(X,Y)$, which gives a contradiction.

If $n \leq \ell + 1$ (this may happen only in the case of NCA), then let $I$ be a set of new edges that forms a clique on $T$. We have $|T| \leq n \leq \ell + 1$, hence $|I| = |T|(|T|-1)/2 < (\ell+2) \cdot |T|$. A similar argument as in the case $n \geq \ell + 2$ easily gives that $I$ is feasible solution.

In both cases, $I$ can be computed in polynomial time, hence the proof is complete. $\square$

**Proposition 2.4** *There exist a polynomial time algorithm that given an instance of* Rooted $\mathcal{D}$-NCA *or of* Rooted $\mathcal{D}$-SPNCA *and a $\mathcal{C}^s_J(\mathcal{D})$-transversal $T$, computes a feasible solution $I$ such that $|I| \leq 2|T|$.*

**Proof:** In the case of Rooted $\mathcal{D}$-NCA we obtain a feasible solution $I$ of size $|T|$ by connecting every node in $T$ to $s$ by a new edge.

Now let us consider the case of Rooted $\mathcal{D}$-SPNCA. Let $T_0 = \{t \in T : ts \notin E_J\}$ and $I_0 = \{ts : t \in T_0\}$, so $|I_0| = |T_0|$. Let $J' = J \cup I_0$, and let $\mathcal{D}' = \{\{u,s\} \in \mathcal{D} : \kappa_{J'}(u,s) = \kappa_J(u,s)\}$ consist from those pairs in $\mathcal{D}$ that are not "satisfied" by addition of $I_0$ to $J$. Consider an arbitrary $us$-tight set $S$ in $J'$ with $\{u,s\} \in \mathcal{D}'$. It is not hard to verify that $T' = T \setminus T_0$ is a $\mathcal{C}^s_{J'}(\mathcal{D}')$-transversal, hence there is $t \in S \cap T'$. As $ts \in J'$, we must have $u = t$, by the definition of a tight set. Consequently, $\mathcal{D}' = \{\{t,s\} : t \in T'\}$. Hence to obtain a feasible solution, it would be sufficient to add to $I_0$ an edge set $I'$ that increases the connectivity (in $J$ or in $J'$) from every $t \in T'$ to $s$.

We show how to find a set $I(t)$ of at most 2 new edges whose addition increases the $ts$-connectivity by 1. Let $\Pi$ be a set of $\kappa_J(t,s)$ pairwise internally disjoint $ts$-paths (one of these paths is the edge $ts$). If there is a node $a$ that does not belong to any path in $\Pi$, then $I(t) = \{ta, as\} \setminus E_J$. If there is a path of length at least 3 in $\Pi$, say $t - a - b - \cdots - s$, then $I(t) = \{tb, as\} \setminus E_J$. Otherwise, all the paths in $\Pi$ distinct from the edge $ts$ have length 2 and every node belongs to a path in $\Pi$. But then $|V| = \kappa_J(t,s) + 1 \leq \ell$, and thus the problem has no feasible solution $I$ such that $J \cup I$ is a simple graph. Consequently, $I(t)$ as above exists and can be found in polynomial time.

Let $I' = \bigcup_{t \in T'} I(t)$. Then $I = I_0 \cup I'$ is a feasible solution and $|I| \leq |I_0| + |I'| = |T_0| + 2(|T| - |T_0|) \leq 2|T|$. The statement follows. $\square$

From Corollary 2.2 and Propositions 2.3 and 2.4 we obtain the following result:

**Theorem 2.5** *Both* $\mathcal{D}$-NCA *and* $\mathcal{D}$-SPNCA *admit a polynomial time algorithm that computes* *a solution $I$ such that $|I| \leq (\ell+2)H\left((4\ell+1)^2\right) \cdot \tau^*(\mathcal{C}_J(\mathcal{D}))$ for general requirements, and $|I| \leq 2H(2\ell+1) \cdot \tau^*(\mathcal{C}^s_J(\mathcal{D}))$ in the case of rooted requirements, where $\ell = \max_{\{u,v\}\in\mathcal{D}} \kappa_J(u,v)$.*

The following statement relates approximability of NCA to approximability of $\mathcal{D}$-NCA.

**Proposition 2.6** *Suppose that $\mathcal{D}$-NCA admits a polynomial time algorithm that computes a solution of size $\leq \alpha(\ell) \cdot \tau^*(\mathcal{C}_J(\mathcal{D}))$, where $\alpha(\ell)$ is increasing in $\ell$. Then NCA admits a polynomial time algorithm that computes a solution of size $\leq 2\mathsf{opt} \cdot \sum_{\ell=0}^{k-1} \frac{\alpha(\ell)}{k-\ell} \leq 2H(k) \cdot \alpha(k) \cdot \mathsf{opt}$, where opt denotes the optimal solution size for NCA. The same is valid for Rooted NCA, SPNCA, and Rooted SPNCA.*

**Proof:** We consider NCA, and the proof of the other variants is similar. Apply the algorithm for $\mathcal{D}$-NCA as in the proposition sequentially: at iteration $\ell = 0, \ldots, k-1$ add to $J$ an augmenting edge set $I_\ell$ that increases the connectivity between pairs in $\mathcal{D}_\ell = \{\{u,v\} : u,v \in V, \kappa_J(u,v) = r(u,v) - k + \ell\}$ by 1. Note that $\kappa_J(u,v) \leq \ell$ for $\{u,v\} \in \mathcal{D}_\ell$, thus the algorithm assumed in the proposition can be used to produce a solution $I_\ell$ to $\mathcal{D}$-NCA such that $|I_\ell| \leq \alpha(\ell) \cdot \tau^*_J(\mathcal{D}_\ell)$. After iteration $\ell$, we have $\kappa_J(u,v) \geq r(u,v) - k + \ell + 1$ for all $u,v \in V$. Consequently, after $k-1$ iterations, $\kappa_J(u,v) \geq r(u,v)$ holds for all $u,v \in V$. Hence the computed solution for NCA is feasible. We claim that $|I_\ell| \leq 2\mathsf{opt} \cdot \frac{\alpha(\ell)}{k-\ell}$, $\ell = 0, \ldots, k-1$. For that, it is sufficient to show that $\tau^*_J(\mathcal{D}_\ell) \leq 2\mathsf{opt}/(k-\ell)$. For any $C \in \mathcal{C}_J(\mathcal{D}_\ell)$, any feasible solution to NCA has at least $k-\ell$ edges with an endnode in $C$, by Menger's Theorem. Thus

$$\begin{aligned} \mathsf{opt} &\geq \frac{1}{2} \cdot \min\{\sum_{v\in V} x(v) : \sum_{v\in C} x(v) \geq k-\ell \ \ \forall C \in \mathcal{C}_J(\mathcal{D}_\ell),\ x(v) \geq 0\} \\ &= \frac{1}{2}(k-\ell) \cdot \tau^*(\mathcal{C}_J(\mathcal{D}_\ell)) \ . \end{aligned}$$

□

Theorem 1.1 now follows from Theorem 2.5 and Proposition 2.6.

# 3 Maximum degree of hypergraph of minimal tight sets (proof of Theorem 1.2)

Here we prove Theorem 1.2, which is restated for the convenience of the reader.

**Theorem 1.2** *Suppose that $\max_{\{u,v\}\in\mathcal{D}} \kappa_J(u,v) = \ell$ for an instance of $\mathcal{D}$-NCA. Then the maximum degree in the hypergraph $(V, \mathcal{C}_J(\mathcal{D}))$ is at most $(4\ell+1)^2$. For rooted requirements, the maximum degree in the hypergraph $(V, \mathcal{C}^s_J(\mathcal{D}))$ is at most $2\ell+1$.*

To avoid considering "mixed" cuts that contain both nodes and edges, we assume that $\delta_J(u,v)=\emptyset$ for all $\{u,v\}\in\mathcal{D}$. One way to achieve this is to consider the graph $J'=(V',E'_J)$ obtained from $J$ by subdividing every edge $e\in\delta_J(u,v)$ with $\{u,v\}\in\mathcal{D}$ by a new node $a^e_{uv}$. It is easy to see that this transformation preserves the connectivity between the pairs in $\mathcal{D}$, and that there is a bijective correspondence between the minimal tight sets in $J$ and in $J'$; namely, $C\in\mathcal{C}_J(\mathcal{D})$ if, and only if, $C'\in\mathcal{C}_J(\mathcal{D})$, where $C'=C\cup\{a^e_{uv}: u,v\in C, \{u,v\}\in\mathcal{D}\}$. This implies that for every $v\in V$, the degree of $v$ in the hypergraph $(V,\mathcal{C}_J(\mathcal{D}))$ equals the degree of $v$ in the hypergraph $(V,\mathcal{C}_{J'}(\mathcal{D}))$, while the degree of any node $a^e_{uv}$ is at most the degree of each of $u,v$. Summarizing we have:

- $\max\limits_{\{u,v\}\in\mathcal{D}} \kappa_{J'}(u,v)=\max\limits_{\{u,v\}\in\mathcal{D}} \kappa_J(u,v)=\ell$.
- The maximum degree in the hypergraph $(V,\mathcal{C}_{J'}(\mathcal{D}))$ equals the maximum degree in the hypergraph $(V,\mathcal{C}_J(\mathcal{D}))$.

Thus it is sufficient to prove Theorem 1.2 for the case when $\delta_J(u,v)=\emptyset$ for all $\{u,v\}\in\mathcal{D}$. Then $S$ is $uv$-tight if, and only if, $u\in S$, $v\in S^*$, and $|\Gamma_J(S)|=\kappa_J(u,v)$, where we use the notation $S^*=V\setminus(S\cup\Gamma_J(S))$.

The following "sub-modular" and "posi-modular" properties of the function $\Gamma(\cdot)=\Gamma_J(\cdot)$ is well known, see for example [16] and [28].

**Proposition 3.1** *For any $X,Y\subseteq V$ the following holds:*

$$|\Gamma(X)|+|\Gamma(Y)| \geq |\Gamma(X\cap Y)|+|\Gamma(X\cup Y)| \tag{3}$$

$$|\Gamma(X)|+|\Gamma(Y)| \geq |\Gamma(X\cap Y^*)|+|\Gamma(Y\cap X^*)| \tag{4}$$

**Lemma 3.2** *Let $X$ be $xx'$-tight and let $Y$ be $yy'$-tight. If $\Gamma(X)\cap\{y,y'\}=\emptyset$ and $\Gamma(Y)\cap\{x,x'\}=\emptyset$, then at least one of the sets $X\cap Y, X\cap Y^*, Y\cap X^*$ is: $xx'$-tight, or $yy'$-tight, or $x'x$-tight, or $y'y$-tight.*

**Proof:** W.l.o.g. assume that $\kappa_J(x,x')\geq\kappa_J(y,y')$. We now consider several cases, see Figure 1.

If $x\in X\cap Y$ and $x'\in X^*\cap Y^*$ then (see Figure 1(a)):

$$\begin{aligned} 2\kappa_J(x,x') &\geq |\Gamma(X)|+|\Gamma(Y)|\geq|\Gamma(X\cap Y)|+|\Gamma(X\cup Y)| \\ &\geq \kappa_J(x,x')+\kappa_J(x,x')=2\kappa_J(x,x')\ . \end{aligned}$$

Hence equality holds everywhere, so $X\cap Y$ (and also $X\cup Y$) is $xx'$-tight.

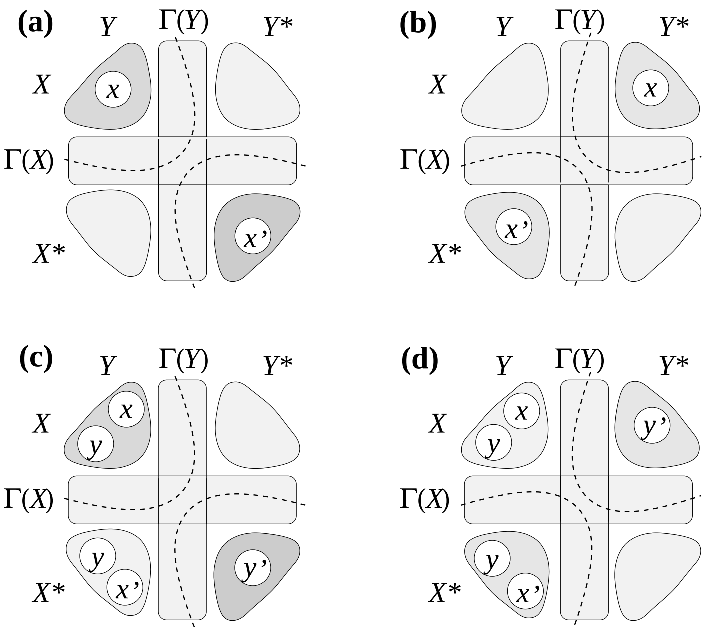


Figure 1: Illustration to the proof of Lemma 3.2. Here the sets $X, \Gamma(X), X^*$ are the "rows" and $Y, \Gamma(Y), Y^*$ are the "columns" of a $3 \times 3$ "matrix".

Similarly, if $x \in X \cap Y^*$ and $x' \in X^* \cap Y$ then (see Figure 1(b)):

$$\begin{aligned} 2\kappa_J(x,x') &\geq |\Gamma(X)| + |\Gamma(Y)| \geq |\Gamma(X \cap Y^*)| + |\Gamma(X^* \cap Y)| \\ &\geq \kappa_J(x,x') + \kappa_J(x,x') = 2\kappa_J(x,x') \ . \end{aligned}$$

Hence equality holds everywhere, so both $X \cap Y^*, X^* \cap Y$ are $xx'$-tight.

The remaining cases are $x, x' \in Y$ or $x, x' \in Y^*$. We consider the case $x, x' \in Y$, and the proof of the case $x, x' \in Y^*$ is similar. If $x, x' \in Y$ then $x \in X \cap Y$ and $x' \in X^* \cap Y$. We have two cases: $y' \in Y^* \cap X^*$ or $y' \in Y^* \cap X$.

If $y' \in Y^* \cap X^*$ (see Figure 1(c)) then independently of the location of $y$ (in $Y \cap X$ or in $Y \cap X^*$) we have:

$$\begin{aligned} \kappa_J(x,x') + \kappa_J(y,y') &= |\Gamma(X)| + |\Gamma(Y)| \geq |\Gamma(X \cap Y)| + |\Gamma(X \cup Y)| \\ &\geq \kappa_J(x,x') + \kappa_J(y,y') \ . \end{aligned}$$

Hence equality holds everywhere, so $X \cap Y$ is $xx'$-tight (and $X \cup Y$ is $yy'$-tight).

If $y' \in Y^* \cap X$ (see Figure 1(d)) then independently of the location of $y$ (in $Y \cap X$ or in $Y \cap X^*$) we have:

$$\begin{aligned} \kappa_J(x,x') + \kappa_J(y,y') &= |\Gamma(X)| + |\Gamma(Y)| \geq |\Gamma(X \cap Y^*)| + |\Gamma(X^* \cap Y)| \\ &\geq \kappa_J(y',y) + \kappa_J(x',x) = \kappa_J(y,y') + \kappa_J(x,x') \ . \end{aligned}$$

Hence equality holds everywhere, so $X \cap Y^*$ is $y'y$-tight and $X^* \cap Y$ is $x'x$-tight.

This concludes the proof of the lemma. $\square$

**Corollary 3.3** *Let $C_1, C_2 \in \mathcal{C}_J(\mathcal{D})$ such that $C_1$ is $u_1v_1$-tight, $C_2$ is $u_2v_2$-tight, and $C_1 \neq C_2$. Then $(u_1, v_1) \neq (u_2, v_2)$. If in addition $C_1 \cap C_2 \neq \emptyset$ then $\Gamma(C_1) \cap \{u_2, v_2\} \neq \emptyset$ or $\Gamma(C_2) \cap \{u_1, v_1\} \neq \emptyset$.*

**Proof:** The first statement is obvious, as for any $\{u, v\} \in \mathcal{D}$ the minimal $uv$-tight set is unique. For the second statement, if $\Gamma(C_1) \cap \{u_2, v_2\} = \emptyset$ and $\Gamma(C_2) \cap \{u_1, v_1\} = \emptyset$, then by Lemma 3.2 at least one of the sets $C_1 \cap C_2, C_1 \cap C_2^*, C_2 \cap C_1^*$ is $u_1v_1$-tight or is $u_2v_2$-tight. Since $C_1 \cap C_2 \neq \emptyset$ then this set is strictly contained in $C_1$ (if it is $u_1v_1$-tight), or is strictly contained in $C_2$ (if it is $u_1v_1$-tight). This contradicts the minimality of one of $C_1, C_2$. $\square$

For $z \in V$ let $\mathcal{C}(z) = \{C \in \mathcal{C}_J(\mathcal{D}) : z \in C\}$ be the set of members in $\mathcal{C}_J(\mathcal{D})$ containing $z$. Let $q = |\mathcal{C}(z)|$. Construct an auxiliary directed labeled graph $\mathcal{J}(z)$ with labels on the arcs as follows. The node set of $\mathcal{J}(z)$ is $\mathcal{C}(z)$. Add an arc $C'C$ with *label* $(u', v')$ if $C'$ is $u'v'$-tight and $\Gamma(C) \cap \{u', v'\} \neq \emptyset$; from every set of parallel arcs keep only one. By Corollary 3.3, $\mathcal{J}(z)$ has an arc from $C$ to $C'$ or from $C'$ to $C$ (or both), for any $C, C' \in \mathcal{C}(z)$. In what follows, note that any graph on $q$ nodes that has this property (namely, has an arc from $a$ to $b$ or from $b$ to $a$ for any pair $a, b$ of its nodes) has at least $q(q-1)/2$ arcs, and thus has a node of indegree at least $(q-1)/2$.

**Lemma 3.4** *For any $u \in V$, there are at most $2\ell + 1$ arcs that have label $(u, v')$ for some $v' \in V$, and there are at most $2\ell + 1$ arcs that have label $(v', u)$ for some $v' \in V$.*

**Proof:** Let $u \in V$. We prove there are at most $2\ell + 1$ arcs that have labels $(u, v')$ for some $v' \in V$; the proof for the other case is similar. Consider all the edges with labels of the form $(u, v')$, say $(u, v_1), \ldots (u, v_t)$, and the corresponding minimal tight sets $C_1, \ldots, C_t$, where $C_i$ is $uv_i$-tight. We claim that $t \leq 2\ell + 1$. For that, consider the subgraph $\mathcal{J}'$ of $\mathcal{J}(z)$ induced by $C_1, \ldots, C_t$. We have that $u$ belongs to the intersection of the sets $C_i$ for $i = 1, \ldots, t$. Thus for every $i \neq j$ we have $v_i \in \Gamma(C_j)$ or $v_j \in \Gamma(C_i)$, by Corollary 3.3. As $\mathcal{J}(z)$ has an arc from $C$ to $C'$ or from $C'$ to $C$ for any $C, C' \in \mathcal{C}(z)$, there is a node $C$ in $\mathcal{J}'$ with indegree at least $(t-1)/2$. Every arc $C_iC$ entering $C$ contributes the node $v_i$ to $\Gamma(C)$; thus $(t-1)/2 \leq \ell$, since the nodes $v_1, \ldots v_t$ are distinct. This implies $t \leq 2\ell + 1$, as claimed. $\square$

In the case of rooted requirements, all labels are of the form $(v', s)$, where $s$ is the root. Hence in this case Lemma 3.4 implies that the total number of minimal tight sets containing $z$ but not $s$ is at most $2\ell + 1$.

**Corollary 3.5** *For any arc with label* $(u, v)$ *there are at most* $4(2\ell + 1)$ *arcs with labels* $(u', v')$ *such that* $\{u', v'\} \cap \{u, v\} \neq \emptyset$*.*

**Proof:** If $\{u', v'\} \cap \{u, v\} \neq \emptyset$, then there are 4 cases: $u' = u$, or $v' = v$, or $u' = v$, or $v' = u$. Namely, the label $(u', v')$ belongs to one of the following 4 types: $(u, v'), (u', v), (v, v'), (u', u)$ By Lemma 3.4, the number of arcs with labels of each one of these types is at most $2\ell + 1$, which implies the statement. $\square$

We now finish the proof of Theorem 1.2. As $\mathcal{J}(z)$ has an arc from $C$ to $C'$, or from $C'$ to $C$ for any $C, C' \in \mathcal{C}(z)$, it has a node $C$ of indegree $\geq (q-1)/2$. Now consider the labels of the arcs entering $C$ in $\mathcal{J}(z)$. By Corollary 3.5, there are at least $(q-1)/(16\ell + 8)$ arcs entering $C$, such that no two arcs have intersecting labels. Each one of these arcs contributes a node to $\Gamma(C)$. Consequently, we must have $(q-1)/(16\ell+8) \leq \ell$, which implies $q \leq 8\ell(2\ell + 1) + 1 = (4\ell + 1)^2$.

The proof of Theorem 1.2 is complete.

# 4 Open problems

- Does SND with arbitrary costs admit a $\rho(k)$-approximation algorithm? The answer is positive for rooted requirements, see [25]. We conjecture the answer is positive for general requirements, motivated also by the results of this paper. As was mentioned, the currently best ratios for SND problems are $O(k^3 \ln n)$ for SND [4], and $O(k \ln k)$ for Rooted SND [25]. Note that the ratio of [4] for SND depends on $n$, while in this paper we showed for 0, 1-costs the ratio $O(k \ln^2 k)$ that does not depend on $n$.
- What versions of SND can be solved exactly and/or well approximated in $t(k)poly(n)$ time? One example of such a problem is $k$-Connected Subgraph with 0, 1-costs [14].
- Does directed/undirected Rooted $\mathcal{D}$-SND with requirements in $\{0, k\}$ admit an approximation scheme similar to the one given in [2] for the Directed Steiner Tree problem?

**Acknowledgment:** I thank anonymous referees for many useful comments.

# References


[1] T. Chakraborty, J. Chuzhoy, and S. Khanna. Network design for vertex connectivity. In *Proceedings of the 40th annual ACM Symposium on Theory of Computing (STOC)*, pages 167–176, 2008.

[2] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, S. Guha, and M. Li. Approximation algorithms for directed Steiner problems. *Journal of Algorithms*, 33:73–91, 1999.

[3] J. Cheriyan and A. Vetta. Approximation algorithms for network design with metric costs. *SIAM Journal on Discrete Mathematics*, 21(3):612–636, 2007.

[4] J. Chuzhoy and S. Khanna. An $O(k^3 \log n)$-approximation algorithms for vertex-connectivity network design. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 437–441, 2009.

[5] Y. Dodis and S. Khanna. Design networks with bounded pairwise distance. In *Proceedings of the 31st annual ACM Symposium on Theory of Computing (STOC)*, pages 750–759, 2003.

[6] M. Feldman, G. Kortsarz, and Z. Nutov. Improved approximation algorithms for directed steiner forest. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 922–931, 2009. To appear in *Journal of Computer and System Sciences*.

[7] L. Fleischer, K. Jain, and D. P. Williamson. Iterative rounding 2-approximation algorithms for minimum-cost vertex connectivity problems. *Journal of Computer and System Sciences*, 72(5):838–867, 2006.

[8] A. Frank. Augmenting graphs to meet edge-connectivity requirements. *SIAM Journal on Discrete Mathematics*, 5(1):25–53, 1992.

[9] A. Frank. Edge-connection of graphs, digraphs, and hypergraphs. In E. Gyri, G. Katona, and L. Lovász, editors, *More sets, graphs and numbers, Bolyai Mathematical Society Math. Studies 15*, pages 93–142. Springer Verlag, 2006.

[10] A. Frank and T. Jordán. Minimal edge-coverings of pairs of sets. *Journal of Combinatorial Theory Series B*, 65:73–110, 1995.

[11] A. Frank and E. Tardos. An application of submodular flows. *Linear Algebra and its Applications*, 114/115:329–348, 1989.

[12] E. Halperin and R. Krauthgamer. Polylogarithmic inapproximability. In *Proceedings of the 35th annual ACM Symposium on Theory of Computing (STOC)*, pages 585–594, 2003.

[13] B. Jackson and T. Jordán. A near optimal algorithm for vertex connectivity augmentation. In *Proceedings of the 11th Annual International Symposium on Algorithms and Computation (ISAAC)*, pages 313–325, 2000.

[14] B. Jackson and T. Jordán. Independence free graphs and vertex connectivity augmentation. *Journal of Combinatorial Theory Series B*, 94(1):31–77, 2005.

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

[16] T. Jordán. On the optimal vertex-connectivity augmentation. *Journal of Combinatorial Theory Series B*, 63:8–20, 1995.

[17] T. Jordán. A note on the vertex connectivity augmentation. *Journal of Combinatorial Theory Series B*, 71(2):294–301, 1997.

[18] G. Kortsarz, R. Krauthgamer, and J. R. Lee. Hardness of approximation for vertex-connectivity network design problems. *SIAM Journal on Computing*, 33(3):704–720, 2004.

[19] G. Kortsarz and Z. Nutov. Approximating node connectivity problems via set covers. *Algorithmica*, 37:75–92, 2003.

[20] G. Kortsarz and Z. Nutov. Approximating minimum-cost connectivity problems. In T. F. Gonzalez, editor, Chapter 58 in *Approximation Algorithms and Metaheuristics*. Chapman & Hall/CRC, 2007.

[21] G. Kortsarz and Z. Nutov. Tight approximation algorithm for connectivity augmentation problems. *Journal of Computer and System Sciences*, 74(5):662–670, 2008.

[22] Y. Lando and Z. Nutov. Inapproximability of survivable networks. *Theoretical Computer Science*, 410(21-23):2122–2125, 2009.

[23] G. Liberman and Z. Nutov. On shredders and vertex-connectivity augmentation. *Journal of Discrete Algorithms*, 5(1):91–101, 2007.

[24] L. Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13:383–390, 1975.

[25] Z. Nutov. Approximating minimum cost connectivity problems via uncrossable bifamilies. Manuscript. Preliminary version in *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)* 2009, pages 417-426.

[26] Z. Nutov. Approximating rooted connectivity augmentation problems. *Algorithmica*, 44:213–231, 2005.

[27] Z. Nutov. An almost $O(\log k)$-approximation for $k$-connected subgraphs. In *Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 912–921, 2009.

[28] Z. Nutov. Approximating connectivity augmentation problems. *ACM Transactions on Algorithms*, 6(1), 2009.

[29] Z. Nutov. Approximating node-connectivity augmentation problems. In *Proceedings APPROX-RANDOM, Lecture Notes in Computer Science*, volume 5687, pages 286–297, 2009.

[30] R. Ravi and D. P. Williamson. An approximation algorithm for minimum-cost vertex-connectivity problems. *Algorithmica*, 18:21–43, 1997.

[31] L. Végh. Augmenting undirected node-connectivity by one. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC)*, pages 563–572, 2010.