# Parameterized Algorithms for Node Connectivity Augmentation Problems

Zeev Nutov
The Open University of Israel, Ra'anana, Israel

## Abstract

A graph $G$ is $k$-out-connected from its node $s$ if it contains $k$ internally disjoint $sv$-paths to every node $v$; $G$ is $k$-connected if it is $k$-out-connected from every node. In connectivity augmentation problems, the goal is to augment a graph $G_0 = (V, E_0)$ by a minimum costs edge set $J$ such that $G_0 \cup J$ has higher connectivity than $G_0$. In the $k$-Out-Connectivity Augmentation ($k$-OCA) problem, $G_0$ is $(k-1)$-out-connected from $s$ and $G_0 \cup J$ should be $k$-out-connected from $s$; in the $k$-Connectivity Augmentation ($k$-CA) problem $G_0$ is $(k-1)$-connected and $G_0 \cup J$ should be $k$-connected. The parameterized complexity status of these problems was open even for $k = 3$ and unit costs. We will show that $k$-OCA and 3-CA can be solved in time $9^p \cdot n^{O(1)}$, where $p$ is the size of an optimal solution. Our paper is the first that shows fixed-parameter tractability of a *$k$-node-connectivity* augmentation problem with *high values of* $k$. We will also consider the $(2, k)$-Connectivity Augmentation ($(2, k)$-CA) problem where $G_0$ is $(k-1)$-edge-connected and $G_0 \cup J$ should be both $k$-edge-connected and 2-connected. We will show that this problem can be solved in time $9^p \cdot n^{O(1)}$, and for unit costs approximated within 1.892.

**2012 ACM Subject Classification** Theory of computation → Design and analysis of algorithms

**Keywords and phrases** node connectivity augmentation, fixed-parameter tractability

**Digital Object Identifier** 10.4230/LIPIcs.ESA.2024.92

## 1 Introduction

In network design problems, the goal is to find a minimum cost subgraph that satisfies given connectivity requirement. Most of these problems are NP-hard, hence parameterized and approximation algorithms are of interest. A natural question then is whether the problem is **fixed-parameter tractable** w.r.t a parameter $p$, namely, if it can be solved in time $f(p) \cdot N^{O(1)}$, where $N$ is the input size. A related question is what approximation ratio can be achieved within this time bound. One of the most studied problems is the Steiner Tree problem, where we seek a minimum cost subtree that spans a given set of terminals. Already in the 70's, Dreyfus and Wagner [14] showed that this problem can be solved in time $3^q \cdot n^{O(1)}$, where $q$ is the number of terminals and $n = |V|$ is the number of nodes in the graph; for improvements over this running time see [21]. The Dreyfus-Wagner algorithm extends to the Directed Steiner Tree problem, in which the goal is to find a minimum cost directed tree that contains a path from a root node $s$ to every terminal.

Graphs in this paper are assumed to be undirected and may have parallel edges, unless stated otherwise. While there was a large progress in the study of parameterized complexity of edge-connectivity problems [22, 29, 5, 1, 16], many papers mention that very little is known about their much harder node-connectivity counterparts. We will consider the "simplest" type of node-connectivity problems, that however have a rich history, when the goal is to increase the node connectivity from $k-1$ to $k$ from a given node to other nodes, or between all nodes. A graph $G = (V, E)$ is **$k$-out-connected from** $s$ if it contains $k$ internally disjoint $sv$-paths for every $v \in V \setminus \{s\}$, and $G$ is **$k$-connected** if it is $k$-out-connected from every node and $|V| \geq k + 1$. In the $k$-Out-Connected Subgraph problem the goal is to find a minimum cost spanning subgraph that is $k$-outconnected from $s$, while in the $k$-Connected Subgraph problem the spanning subgraph should be $k$-connected. These two problems are trivially fixed-parameter tractable w.r.t. an optimal solution size, since any feasible solution has at least $kn/2$ edges. Therefore, it is reasonable to choose as a parameter the number of *non-zero* cost edges in an optimal solution, a number that may be between 1 and $\Theta(kn)$. This leads to the augmentation versions of these problems, where the goal is to augment a graph $G_0 = (V, E_0)$ (of cost zero) by a minimum costs edge set $J$ such that $G_0 \cup J$ has larger connectivity than $G_0$. In this work, we will consider the problem of increasing the connectivity only by 1. Formally, these augmentation problems are as follows.

$k$-Out-Connectivity Augmentation ($k$-OCA)
*Input:* A $(k-1)$-out-connected from $s$ graph $G_0 = (V, E_0)$ and an edge set $E$ with costs $\{c_e : e \in E\}$.
*Output:* A minimum cost edge set $J \subseteq E$ such that $G_0 \cup J$ is $k$-out-connected from $s$.

$k$-Connectivity Augmentation ($k$-CA)
*Input:* A $(k-1)$-connected graph $G_0 = (V, E_0)$ and an edge set $E$ with costs $\{c_e : e \in E\}$.
*Output:* A minimum cost edge set $J \subseteq E$ such that $G_0 \cup J$ is $k$-connected.

These are the optimization versions of $k$-OCA and $k$-CA. In the decision versions, we are also given a parameter $p$, and ask whether there exists a feasible solution $J \subseteq E$ of size $|J| \leq p$ and cost $c(J) \leq \mathsf{opt}$, where $\mathsf{opt}$ is the optimal solution cost.

Let us briefly review the parameterized and approximation status of these problems. The directed version of $k$-Out-Connected Subgraph admits a polynomial time algorithm [20], and this implies approximation ratio 2 for the undirected version. On the other hand, the approximability status of $k$-Connected Subgraph is somewhat complicated; the problem admits ratio $\lceil \frac{k+1}{2} \rceil$ for $2 \leq k \leq 7$ [26, 3, 12, 27], $4 + \epsilon$ for any constant $k$ and $\epsilon > 0$ [40, 9][1], and $O\left(\log k \log \frac{n}{n-k}\right)$ for any $k$ [34]. The augmentation version $k$-CA admits better approximation ratios for $k \geq 8$: ratio 4 for $n \geq 3k - 5$ and $O\left(\log \frac{n}{n-k}\right)$ for any $k$ [35, 34]. When $(V, E)$ is a complete graph with unit costs (so any edge can be added by a cost of 1), the problem of augmenting an arbitrary graph $G_0$ to be $k$-connected can be solved in time $f(k) \cdot n^{O(1)}$ [24], and for a $(k-1)$-connected $G_0$ admits a polynomial time algorithm [43]; however, no such results are known for $k$-OCA, see [32]. See also a survey in [36].

Let $k$-Edge-Connectivity Augmentation be the edge-connectivity version of $k$-CA; note that it is equivalent to the edge connectivity version of $k$-OCA. By the cactus model of Dinitz, Karzanov, and Lomonosov [11], for $k$-Edge-Connectivity Augmentation, the case of $k-1$ odd and even is equivalent to the case $k-1 = 1$ (so called Tree Augmentation problem) and $k - 1 = 2$ (so called Cactus Augmentation problem), respectively. For approximation algorithms for these two problems see recent papers [8, 41, 42].

We will consider one of the most common choices of the parameter $p$ – an optimal solution size. Nagamochi [30] showed that the problem of augmenting a tree by a min-size edge set can be solved in time $2^{O(p \log p)} \cdot n^{O(1)}$. Guo and Uhlmann [22] considered both edge an node connectivity versions of 2-CA with unit costs, and showed that they have a kernel of size $O(p^2)$. Marx and Végh [29] showed that $k$-Edge-Connectivity Augmentation with arbitrary costs can be solved in time $2^{O(p \log p)} \cdot n^{O(1)}$, and that the problem of increasing the edge connectivity from 0 to 2 can also be solved within this time bound. Basavaraju et

[1] More precisely, the approximation ratio in [40] is $2\left(2 + \frac{1}{\ell}\right)$, where $\ell \approx \frac{1}{2}(\log_k n - 1)$ is the largest integer such that $2^{\ell-1} k^{\ell+1} \leq n$.

al. [5] used a novel reduction to the Node Weighted Steiner Tree problem to improve the running time for $k$-Edge-Connectivity Augmentation to $9^p \cdot n^{O(1)}$ (see also [39] for a very simple algorithm for Tree Augmentation with running time $16^p \cdot n^{O(1)}$). The reduction of Basavaraju et al. [5] was generalized in [38] to the problem of covering a crossing set family by edges (see definitions below), and also was extended to 2-CA; specifically, the result of [38] implies that 2-CA and 2-OCA can also be solved in time $9^p \cdot n^{O(1)}$. However, the parameterized complexity status of both 3-OCA and 3-CA was open even for unit costs [29, 23].

We will show that the reductions of [5, 38] can be extended to $k$-node-connectivity augmentation problems with *high values of* $k$. But our reduction is not to the Node Weighted Steiner Tree problem, but to the Node Weighted Group Steiner Tree problem. The later can be reduced to the Directed Steiner Tree problem, that can be solved in time $3^q \cdot n^{O(1)}$, where $q$ is the number of terminals. Formally, we will prove the following.

▶ **Theorem 1.** $k$-OCA *can be solved in time* $9^p \cdot n^{O(1)}$.

Algorithms for $k$-OCA are sometimes used as subroutines in algorithms for $k$-CA. Auletta et al. [3] showed that 3-CA can be reduced in polynomial time to 3-OCA. Specifically, this reduction relies on two facts: (i) Any minimally $k$-connected graph has a node $s$ of degree $k$, c.f. [28]; (ii) If a graph is 3-out-connected from a node $s$ that has exactly 3 neighbors then the graph is 3-connected, c.f. [3]. Thus 3-CA is reduced to 3-OCA by "guessing" such $s$ and its 3 neighbors in some minimal 3-CA optimal solution. For $k = 4, 5$, [13] shows that if $k$-OCA admits approximation ratio $\rho$ then $k$-CA admits ratio $\rho + 1$. Thus from Theorem 1 we get the following.

▶ **Corollary 2.** *In time* $9^p \cdot n^{O(1)}$, 3-CA *can be solved optimally, and* 4-CA *and* 5-CA *can be* 2*-approximated.*

Corollary 2 resolves the open question of parameterized complexity of 3-CA [29, 23], by establishing that 3-CA is fixed-parameter tractable.

To prove Theorem 1, we will consider a more general problem of covering a biset family by edges. A **biset** is an ordered pair $\mathbb{A} = (A, A^+)$ such that $A \subseteq A^+$; $\partial\mathbb{A} = A^+ \setminus A$ is the **boundary** of $\mathbb{A}$, $A^* = V \setminus A^+$ is the **co-set** of $\mathbb{A}$, and $\mathbb{A}^* = (V \setminus A^+, V \setminus A)$ is the **co-biset** of $\mathbb{A}$. A biset $\mathbb{A}$ is **proper** if $A \neq \emptyset$ and $A^* \neq \emptyset$. The intersection and the union of bisets $\mathbb{A}, \mathbb{B}$ are defined by $\mathbb{A} \cap \mathbb{B} = (A \cap B, A^+ \cap B^+)$ and $\mathbb{A} \cup \mathbb{B} = (A \cup B, A^+ \cup B^+)$. Two bisets $\mathbb{A}, \mathbb{B}$ **intersect** if $A \cap B \neq \emptyset$, and **cross** if also $A^+ \cup B^+ \neq V$. A biset family $\mathcal{F}$ is **intersecting/crossing** if $\mathbb{A} \cap \mathbb{B}, \mathbb{A} \cup \mathbb{B} \in \mathcal{F}$ whenever $\mathbb{A}, \mathbb{B} \in \mathcal{F}$ intersect/cross. An **edge** $e$ **covers** $\mathbb{A}$ if $e$ goes from $A$ to $A^*$, and an edge set $J$ covers $\mathcal{F}$ if every $\mathbb{A} \in \mathcal{F}$ is covered by some edge in $J$. We will identify a set $A$ with the biset $(A, A)$, so when $A = A^+$ for every $\mathbb{A} \in \mathcal{F}$ we have an ordinary set family, and we use for set families a similar terminology.

Let $G_0 = (V, E_0)$ be $(k-1)$-out-connected from $s$. Let us say that a proper biset $\mathbb{A}$ on $V$ is **tight** if $A \neq \emptyset$, $s \in A^*$ and $|\partial\mathbb{A}| + d_{G_0}(\mathbb{A}) = k - 1$, where $d_{G_0}(\mathbb{A})$ is the number of edges in $G_0$ that cover $\mathbb{A}$. For the family $\mathcal{T}$ of tight bisets the following is known, see [19]:

- $G_0 \cup J$ is $k$-connected iff $J$ covers $\mathcal{T}$ (by Menger's Theorem).
- $\mathcal{T}$ is an intersecting biset family.

Thus the following problem includes $k$-OCA.

| Intersecting Biset Family Cover |
|---|
| *Input:* A graph $G = (V, E)$ with costs $\{c_e : e \in E\}$, an intersecting biset family $\mathcal{F}$ on $V$. |
| *Output:* A minimum cost edge set $J \subseteq E$ that covers $\mathcal{F}$. |

In this problem, $\mathcal{F}$ may not be given explicitly ($|\mathcal{F}|$ may be exponential in $|V|$); instead, we require that some queries on $\mathcal{F}$ can be answered in polynomial time. We will assume that for any edge set $I$ we can find in time polynomial in $n = |V|$ the family $\mathcal{F}^I_{\min}$ and $\mathcal{F}^I_{\max}$ of all inclusion minimal and maximal bisets, respectively, among the bisets in $\mathcal{F}$ not covered by $I$. In Intersecting Biset Family Cover instances that arise from the $k$-OCA problem, this can be done using $n$ max-flow computations. Under this assumption we will prove:

▶ **Theorem 3.** Intersecting Biset Family Cover *can be solved in time* $9^p \cdot n^{O(1)}$.

A node $v$ is a **cut-node** of a graph $G$ if $G \setminus \{v\}$ is disconnected. In addition, we will consider the following problem, that combines both edge and node connectivity augmentation:

$(2,k)$-Connectivity Augmentation ($(2,k)$-CA)
*Input:* A $(k-1)$-edge-connected graph $G_0 = (V, E_0)$ where $k \geq 2$, $Q \subseteq V$, and an edge set $E$ on $V$ with costs $\{c_e : e \in E\}$.
*Output:* A minimum cost edge set $J \subseteq E$ such that $G_0 \cup J$ is $k$-edge-connected and has no cut-node in $Q$.

We will show that $(2,k)$-CA (resp., with unit costs) can be reduced to the Node Weighted Steiner Tree problem (resp., with unit weights) with the following properties:

**(A)** The neighbors of every terminal induce a clique.

**(B)** Every non-terminal has at most 2 terminal neighbors.

**(C)** There are no edges between the terminals.

Node Weighted Steiner Tree can be solved in time $3^q \cdot n^{O(1)}$ by the Dreyfus-Wagner algorithm [14]. Byrka et al. [7] showed that unit weight instances with properties (A,B,C) admit approximation ratio 1.91, and Angelidakis et al. [2] improved the ratio to 1.892. Thus we get the following.

▶ **Theorem 4.** $(2,k)$-CA *can be solved in time* $9^p \cdot n^{O(1)}$, *and in the case of unit costs admits approximation ratio* 1.892.

In the Crossing Family Cover problem we are given a graph $G_0 = (V, E)$ with edge costs and a symmetric crossing set family $\mathcal{F}$ on $V$, and seek a minimum cost edge set $J \subseteq E$ that covers $\mathcal{F}$. Theorem 4 was known for the Crossing Family Cover problem [38]. We will show that $(2,k)$-CA generalizes Crossing Family Cover, see Section 3.

Theorems 3 and 4 are proved in Sections 2 and 3, respectively.

In the rest of the Introduction we survey some additional related work. As was mentioned, min-cost connectivity problems that have solution size $\Omega(n)$ are trivially fixed-parameter tractable w.r.t. an optimal solution size; this is why our parameter choice is the number of non-zero cost edges in an optimal solution. Several other papers studied the so called "deletion set" parameter, i.e., the number of edges to be removed from the input graph in order to obtain a minimum cost solution. For example, Bang-Jensen et al. [4] show that the $k$-Edge Connected Subgraph problem is fixed-parameter tractable for the combined parameter of $k$ and the size of a deletion set. Gutin et al. [23] show a similar result for $k$-Connected Subgraph with unit costs.

In the more general Survivable Network Design (SND) problem we are given a graph with edge costs and pairwise connectivity requirements $\{r_{uv} : uv \in D\}$ over a set $D \subseteq V \times V$ of demand pairs. The goal is to find a min-cost subgraph that contains $r_{uv}$ internally disjoint paths for all $uv \in D$. In the edge-connectivity version EC-SND the paths are only required to be edge disjoint. EC-SND admits a 2-approximation algorithm [25], and can be solved in $2^{O(p \log p)} \cdot n^{O(1)}$ time [16], where $p$ is the solution size. The status of the node-connectivity version SND with $r_{uv} \in \{0,1,2\}$ is similar, see [18] and [16], respectively. On the other hand SND parameterized by the solution size is W[1]-hard even when $|D| = 2$. SND admits approximation ratio $O(k^3 \log n)$ for arbitrary requirements [10], $O(k^2)$ for rooted requirements, and $O(k \log k)$ for rooted requirements in $\{0,k\}$ [33], where $k$ is the maximum requirement. See also a survey in [37]. For the current status of SND problem on directed graphs, see for example [15] and [37].

## 2 Covering intersecting biset families (Theorem 3)

We will reduce Intersecting Biset Family Cover to the following problem:

Subset Steiner Connected Dominating Set (SS-CDS)
*Input:* A graph $H = (U, I)$, a set $R \subseteq U$ of terminals, and node weights $\{w_v : v \in U \setminus R\}$.
*Output:* A min-weight node set $J \subseteq U \setminus R$ such that $H[J]$ is connected and $J$ dominates $R$.

As was observed in [6, 38], SS-CDS reduces to the Node Weighted Group Steiner Tree problem: given a graph with node weights $w_v$ and a collection $\mathcal{S}$ of subsets (groups) of the node set, find a min-weight subtree that contains a node from every group. Given a SS-CDS instance $(H, R, w)$ obtain an equivalent Node Weighted Group Steiner Tree instance as follows: for every $r \in R$, introduce a group $S_r$ that consists of the neighbors of $r$ in $H$, and then remove $r$. This problem reduces to the Directed Steiner Tree problem with $|R| = |\mathcal{S}|$ terminals, that can be solved in time $3^q \cdot n^{O(1)}$, where $q = |R| = |\mathcal{S}|$.

In fact, we will consider the rooted version Rooted SS-CDS, when we are also given a non-terminal root node $s \in U \setminus R$ and we must have $s \in J$. All algorithms that we use, as well as hardness results, are applicable to Rooted SS-CDS. In the case when property (A) holds, we get the rooted version of the Node Weighted Steiner Tree problem, and then the algorithms of Dreyfus and Wagner [14] and of Angelidakis et al. [2] apply on (and were in fact designed for) the rooted version.

Let $\mathcal{F}$ be an intersecting biset family and $J$ an edge set on $V$. W.l.o.g. we may assume that there is a root node $s \in V \setminus (\cup_{\mathbb{A} \in \mathcal{F}} A^+)$. For $\mathbb{A}, \mathbb{B} \in \mathcal{F}$ we say that $\mathbb{B}$ **contains** $\mathbb{A}$ and write $\mathbb{A} \subseteq \mathbb{B}$ if $A \subseteq B$ and $A^+ \subseteq B^+$. An **$\mathcal{F}$-core** is an inclusion minimal biset in $\mathcal{F}$. Let $\mathcal{C} = \mathcal{C}_{\mathcal{F}}$ denote the set of $\mathcal{F}$-cores. For $\mathbb{C} \in \mathcal{C}$ let $\mathcal{F}(\mathbb{C}) = \{\mathbb{A} \in \mathcal{F} : \mathbb{C} \subseteq \mathbb{A}\}$ denote the set of those members of $\mathcal{F}$ that contain $\mathbb{C}$. Given an edge set $J$ and a node set $A$ we will write $J \subseteq A$ and say that $A$ **contains** $J$ meaning that the set of endnodes of $J$ is contained in $A$.

Now we define a certain "separability relation" on $J \cup \mathcal{C} \cup \{s\}$ and a "separability graph" that represents this relation. This follows [38], where the problem of covering a symmetric crossing set family $\mathcal{F}$ was considered. In [38], two edges $x, y$ are "separated" by a set $A$ if one of $x, y$ is contained in $A$ and the other in $V \setminus A$, but it is less clear how to extend this definition to bisets. Consider for example the biset $\mathbb{A}$ and the 6 edges $x, y, z, e, f, g$ in Fig. 1 . It is reasonable to say that $\mathbb{A}$ separates each one of $x, y$ from each one of $e, f$. But does $\mathbb{A}$ separate $x$ from $z$, or $z$ from $g$? Our answer would be yes in both cases. For disjoint edge sets $X, Y$, we will say that $\mathbb{A}$ **separates** $X$ **from** $Y$ if $X \subseteq A^+$ and $Y \subseteq V \setminus A$. So in Fig. 1, $\mathbb{A}$ separates $\{x, y, z\}$ from $\{e, f, g\}$, because $\{x, y, z\} \subseteq A^+$ and $\{e, f, g\} \subseteq V \setminus A$. Also note that $\mathbb{A}$ does not separate an edge that covers $\mathbb{A}$ from any other edge. In the next definition we will extend this notion of "separability" to sets that include both edges and cores.

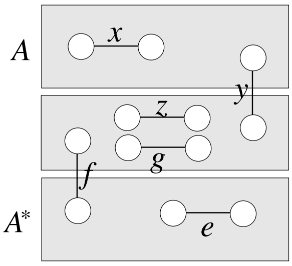

**Figure 1** Illustration to the cases when a biset separates one edge set from the other.

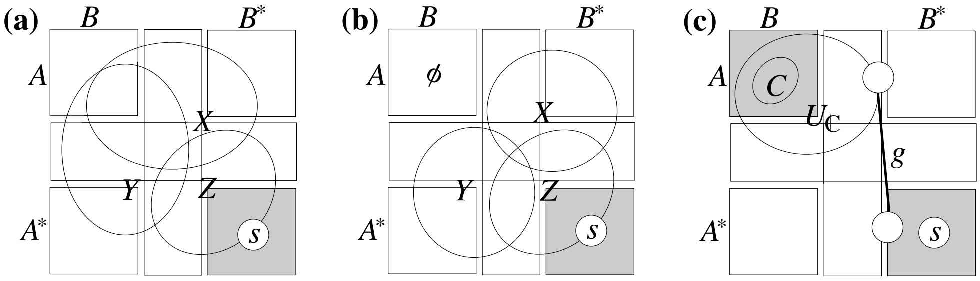

**Figure 2** Illustration to the proof of Lemmas 6 and 7. Dark parts are non-empty, empty parts are marked by $\emptyset$; other parts may or may not be empty.

▶ **Definition 5.** *We say that a biset* $\mathbb{A}$ **separates** $X \subseteq \mathcal{C} \cup J$ **from** $Y \subseteq J \cup \{s\}$ *if* $X \cap Y \cap J = \emptyset$, *every* $\mathcal{F}$*-core in* $X$ *is contained in* $\mathbb{A}$, $J \cap X \subseteq A^+$, *and* $J \cap Y \subseteq V \setminus A$. $X, Y$ **are** $\mathcal{F}$**-separable** *if such* $\mathbb{A} \in \mathcal{F}$ *exists, and* $X, Y$ *are* $\mathcal{F}$**-inseparable** *otherwise. The* **separability graph** $H = (U, I)$ *of* $\mathcal{F}, J$ *has node set* $U = \mathcal{C} \cup J \cup \{s\}$ *and edge set* $I = \{xy : x \in \mathcal{C} \cup J, y \in J \cup \{s\}$ *are* $\mathcal{F}$*-inseparable*$\}$.

Note that $s$ and any $\mathbb{C} \in \mathcal{C}$ are $\mathcal{F}$-separable, and that $\mathcal{C} \cup \{s\}$ is an independent set in the separability graph $H$. We need the following technical lemma.

▶ **Lemma 6.** *If* $\mathbb{A}$ *separates* $X$ *from* $Z$ *and* $\mathbb{B}$ *separates* $Y$ *from* $Z$*, then (see Fig. 2(a,b)):*

**(i)** $\mathbb{A} \cup \mathbb{B}$ *separates* $X \cup Y$ *from* $Z$*.*

**(ii)** *If* $A \cap B = \emptyset$ *then* $\mathbb{A}$ *separates* $X$ *from* $Y$*.*

**Proof.** By the assumption of the lemma we have (see Fig. 2(a)):

- Since $\mathbb{A}$ separates $X$ from $Z$, every $\mathcal{F}$-core in $X$ is contained in $\mathbb{A}$, every edge in $X$ is contained in $A^+$, and every edge in $Z$ is contained in $V \setminus A$.
- Since $\mathbb{B}$ separates $Y$ from $Z$, every $\mathcal{F}$-core in $Y$ is contained in $\mathbb{B}$, every edge in $X$ is contained in $B^+$, and every edge in $Z$ is contained in $V \setminus B$.

Consequently, every $\mathcal{F}$-core in $X \cup Y$ is contained in $\mathbb{A} \cup \mathbb{B}$, every edge in $X \cup Y$ is contained in $A^+ \cup B^+$, and every edge in $Z$ is contained in $V \setminus (A \cup B)$. This implies that $\mathbb{A} \cup \mathbb{B}$ separates $X \cup Y$ from $Z$.

Now suppose that $A \cap B = \emptyset$; see Fig. 2(b). To see that then $\mathbb{A}$ separates $X$ from $Y$, note that every $\mathcal{F}$-core in $X$ is contained in $\mathbb{A}$ and every edge in $X$ is contained in $A^+$ (since $\mathbb{A}$ separates $X$ from $Z$), and that every edge in $Y$ is contained in $V \setminus A$ (since $A \cap B = \emptyset$). ◀

The following key lemma is the technical part of the reduction.

▶ **Lemma 7.** *Let $H = (\mathcal{C} \cup J \cup \{s\}, I)$ be the separability graph of an intersecting biset family $\mathcal{F}$ and an edge set $J$ on $V$. Let $\mathbb{C} \in \mathcal{C}$. Then $J$ covers $\mathcal{F}(\mathbb{C})$ iff $H$ has a $\mathbb{C}s$-path. (Equivalently: $J$ does not cover some $\mathbb{C} \in \mathcal{F}(\mathbb{C})$ iff $H$ has no $\mathbb{C}s$-path.)*

**Proof.** Suppose that $J$ does not cover some $\mathbb{A} \in \mathcal{F}(\mathbb{C})$. Let $\mathcal{C}_{\mathbb{A}}$ be the set of $\mathcal{F}$-cores contained in $\mathbb{A}$ and $J_{\mathbb{A}}$ the set of edges in $J$ contained in $A^+$. Then $H$ has no edge between $\mathcal{C}_{\mathbb{A}} \cup J_{\mathbb{A}}$ and $(\mathcal{C} \cup J \cup \{s\}) \setminus (\mathcal{C}_{\mathbb{A}} \cup J_{\mathbb{A}})$. Consequently, $H$ has no $\mathbb{C}s$-path.

Suppose that $H$ has no $\mathbb{C}s$-path. Let $U_{\mathbb{C}}$ be the set of nodes of the connected component of $H$ that contains $\mathbb{C}$, and let $g \in (J \setminus U_{\mathbb{C}}) \cup \{s\}$. We now will show that $\mathcal{F}$ contains a biset that separates $U_{\mathbb{C}}$ from $g$. Let $f_0, f_1, \ldots$ be an ordering of $U_{\mathbb{C}}$, where $f_0 = \mathbb{C}$, such that in $H$ each $f_i$ with $i \geq 1$ is adjacent to some $f_j$ with $j < i$; since $H[U_{\mathbb{C}}]$ is connected, such an ordering exists. For any $f_i$ there is $\mathbb{A}_i \in \mathcal{F}$ that separates $f_i$ from $g$. By Lemma 6 and since $\mathcal{F}$ is an intersecting family, for $X, Y \subseteq \mathcal{C} \cup J$ and $Z \subseteq J \cup \{s\}$, if $\mathbb{A}$ separates $X$ from $Z$ and $\mathbb{B}$ separates $Y$ from $Z$, then (see Fig. 2(a,b)):

**(i)** $\mathbb{A} \cup \mathbb{B}$ separates $X \cup Y$ from $Z$; moreover, if $\mathbb{A}, \mathbb{B} \in \mathcal{F}$ and $A \cap B \neq \emptyset$ then $\mathbb{A} \cup \mathbb{B} \in \mathcal{F}$.
**(ii)** If $A \cap B = \emptyset$ then $\mathbb{A}$ separates $X$ from $Y$.

In particular, if $\mathbb{A}, \mathbb{B} \in \mathcal{F}$ and $X, Y$ are $\mathcal{F}$-inseparable, then $A \cap B \neq \emptyset$ must hold, and thus $\mathbb{A} \cup \mathbb{B}$ belongs to $\mathcal{F}$ and separates $X \cup Y$ from $Z$. Applying this on $X = \{f_0\}, Y = \{f_1\}, Z = \{g\}$ and $\mathbb{A} = \mathbb{A}_0, \mathbb{B} = \mathbb{A}_1$, we get that $\mathbb{A}_1 \cup \mathbb{A}_2 \in \mathcal{F}$ and separates $\{f_0, f_1\}$ from $g$. By an identical argument applied on $X = \{f_0, f_1\}$ and $Y = \{f_2\}$ we get that $(A_0 \cup A_1) \cup A_2 \in \mathcal{F}$ separates $\{f_0, f_1, f_2\}$ from $g$. By induction we get that $\cup_i A_i \in \mathcal{F}$ and separates $U_{\mathbb{C}}$ from $g$.

Let $\mathbb{A}$ be an inclusion minimal biset in $\mathcal{F}$ that separates $U_{\mathbb{C}}$ from $s$. Note that $\mathbb{A} \in \mathcal{F}(\mathbb{C})$. We claim that $J$ does not cover $\mathbb{A}$. Suppose to the contrary that some $g \in J$ covers $\mathbb{A}$. Note that $g \in J \setminus U_{\mathbb{C}}$, thus as we just proved, there is $\mathbb{B} \in \mathcal{F}$ that separates $U_{\mathbb{C}}$ from $g$, see Fig. 2(c). Note that $\mathbb{C} \subseteq \mathbb{A} \cap \mathbb{B}$, hence $\mathbb{A}, \mathbb{B}$ intersect and thus $\mathbb{A} \cap \mathbb{B} \in \mathcal{F}$. Summarizing, we have:

- $\mathbb{A}$ separates $U_{\mathbb{C}}$ from $s$ and $\mathbb{B}$ separates $U_{\mathbb{C}}$ from $g$.
- $\mathbb{A} \cap \mathbb{B} \neq \emptyset$ and thus $\mathbb{A} \cap \mathbb{B} \in \mathcal{F}$.

By interchanging the roles of $\mathbb{A}, \mathbb{B}$ and $\mathbb{A}^*, \mathbb{B}^*$ in Lemma 6, we get that if $\mathbb{A}$ separates $Z$ from $X$ and $\mathbb{B}$ separates $Z$ from $Y$, then $\mathbb{A} \cap \mathbb{B}$ separates $Z$ from $X \cup Y$. Applying this on $Z = U_{\mathbb{C}}$, $X = \{s\}$, and $Y = \{g\}$, we get that $\mathbb{A} \cap \mathbb{B}$ separates $U_{\mathbb{C}}$ from $\{s, g\}$. As $g$ has both ends in $V \setminus B$ and covers $\mathbb{A}$, it has one end in $A \setminus B$ and the other in $A^* \setminus B$. This implies that $\mathbb{A} \cap \mathbb{B} \subsetneq \mathbb{A}$ (namely, $\mathbb{A} \cap \mathbb{B}$ is strictly contained in $\mathbb{A}$). Since $\mathbb{A} \cap \mathbb{B}$ separates $U_{\mathbb{C}}$ from $g$ and $\mathbb{A} \cap \mathbb{B} \in \mathcal{F}$, we obtain a contradiction to the minimality of $\mathbb{A}$. ◀

From Lemma 7 we get:

▶ **Corollary 8.** *An edge set $J$ covers an intersecting biset family $\mathcal{F}$ iff the separability graph $H$ of $\mathcal{F}, J$ has a subtree that contains $s$ and has leaf set $\mathcal{C}$.*

The reduction of Intersecting Biset Family Cover to Rooted SS-CDS is as follows.

▶ **Definition 9.** *Given an* Intersecting Biset Family Cover *instance $(\mathcal{F}, E, c)$, the corresponding* Rooted SS-CDS *instance $(H, R, w)$ is constructed as follows.*

- *$H$ is the separability graph of $\mathcal{F}, E$.*
- *$R = \mathcal{C}$ and the root is $s$.*
- *For every $e \in E$, the weigh of the node $e$ in $H$ equals to the cost of the edge $e$ in $E$.*

By Corollary 8, $J \subseteq E$ is a feasible solution to the obtained Rooted SS-CDS instance iff $J$ covers $\mathcal{F}$. By the reduction, the weight of $J$ equals the cost of $J$. As was explained at the beginning of this section, Rooted SS-CDS can be solved in time $3^q \cdot n^{O(1)}$, where $q = |R|$. Since $A \cap B = \emptyset$ for any $\mathbb{A}, \mathbb{B} \in \mathcal{C}_{\mathcal{F}}$, we need at least $|\mathcal{C}_{\mathcal{F}}|/2$ edges to cover $\mathcal{F}$, and since $|R| = |\mathcal{C}_{\mathcal{F}}| + 1$, we can find an optimal cover of $\mathcal{F}$ in $3^q \cdot n^{O(1)} = 9^p \cdot n^{O(1)}$ time.

It remains to show that $H$ can be constructed in polynomial time. For that it is sufficient to show that for any $x \in \mathcal{C} \cup E$ and $y \in E \cup \{s\}$ we can check in polynomial time whether there is $\mathbb{A} \in \mathcal{F}$ that separates $x$ from $y$. Recall that we assume that for any edge set $I$ we can find in polynomial time the family $\mathcal{F}^I_{\min}$ and $\mathcal{F}^I_{\max}$ of all inclusion minimal and maximal bisets, respectively, among the bisets in $\mathcal{F}$ not covered by $I$. From Definition 5, it is not hard to verify that there is a biset in $\mathcal{F}$ that separates:

- $\mathbb{C} \in \mathcal{C}$ from $e \in E$ iff $e \subseteq V \setminus C$.
- $e \in E$ from $s$ iff there is $\mathbb{A} \in \mathcal{F}^{\emptyset}_{\max}$ such that $e \subseteq A^+$.
- $uv \in E$ from $u'v' \in E$ iff there is $\mathbb{A} \in \mathcal{F}^I_{\max}$ with $e \subseteq A^+$ and $e' \subseteq V \setminus A$, where $I = \{u's, v's\}$.

This concludes the proof of Theorem 3.

## 3 Reduction for (2,*k*)-Connectivity Augmentation (Theorem 4)

Let $(G_0 = (V, E_0), E, c, Q, k)$ be an instance of $(2, k)$-CA with $k \geq 2$. Recall that $G_0$ is $(k-1)$-edge-connected, and we seek a min cost edge set $J \subseteq E$ such that $G_0 \cup J$ is both $k$-edge-connected and has no cut node in $Q$. We will construct an equivalent instance $(H = (U, F), w, R)$ of Rooted SS-CDS that satisfies the following two properties:

**(A)** The neighbors of every terminal induce a clique.

**(B)** Every non-terminal has at most 2 terminal neighbors.

It is easy to see that if property (A) holds, then any subtree of $H$ that contains $R$ can be converted into a subtree of the same cost with leaf set $R$. Thus any SS-CDS instance that satisfies property (A) is equivalent to the Node Weighted Steiner Tree instance with the same graph, weights, and set of terminals.

▶ **Definition 10.** *Given a $(k-1)$-connected graph $G_0 = (V, E_0)$ with $k \geq 2$ and $Q \subseteq V$, we assign capacities to the nodes $q(v) = k - 1$ if $v \in Q$ and $q(v) = \infty$ otherwise. We say that a proper biset $\mathbb{A}$ is* **tight** *if $d_{G_0}(\mathbb{A}) + q(\partial\mathbb{A}) = k - 1$ and denote by $\mathcal{T} = \mathcal{T}_{G_0}$ the family of tight bisets.*

Equivalently, a proper biset $\mathbb{A}$ is tight if either $\partial\mathbb{A} = \emptyset$ and $d_{G_0}(\mathbb{A}) = k - 1$, or $\partial\mathbb{A}$ is a single node in $Q$ and $d_{G_0}(\mathbb{A}) = 0$. Namely, the family of tight bisets is a union $\mathcal{T} = \mathcal{A} \cup \mathcal{B}$ of a set family $\mathcal{A}$ and a biset family $\mathcal{B}$ defined by

$$\mathcal{A} = \{A : d_{G_0}(A) = k - 1\} \qquad \mathcal{B} = \{\mathbb{B} : \partial\mathbb{B} \text{ is a single node in } Q, d_{G_0}(\mathbb{B}) = 0, B, B^* \neq \emptyset\} .$$

Note that since $G_0$ is $(k-1)$-edge-connected, $d_{G_0}(\mathbb{A}) + q(\partial\mathbb{A}) \geq k - 1$ for every proper biset $\mathbb{A}$. From Menger's Theorem it follows that $J$ is a feasible $(2, k)$-CA solution iff $J$ covers the family $\mathcal{T} = \{\mathbb{A} : A, A^* \neq \emptyset, d_{G_0}(\mathbb{A}) + q(\partial\mathbb{A}) = k - 1\}$ of tight bisets.

We need some definitions. The **co-biset** of a biset $\mathbb{A}$ is the biset $\mathbb{A}^* = (V \setminus A^+, V \setminus A)$. A biset family is symmetric if $\mathbb{A}^* \in \mathcal{F}$ whenever $\mathbb{A} \in \mathcal{F}$. Two bisets $\mathbb{A}, \mathbb{B}$ **co-cross** if $A \setminus B^+, B \setminus A^+$ are both non-empty.

▶ **Lemma 11.** *The family $\mathcal{T}$ of tight bisets is symmetric and crossing, and any $\mathbb{A}, \mathbb{B} \in \mathcal{T}$ cross or co-cross. Consequently, $\mathbb{C} \subseteq \mathbb{A}$ or $\mathbb{C} \subseteq \mathbb{A}^*$ holds for any $\mathbb{A} \in \mathcal{T}$ and a $\mathcal{T}$-core $\mathbb{C} \in \mathcal{C}_{\mathcal{T}}$.*

**Proof.** Define a biset function $f(\mathbb{A}) = d_{G_0}(\mathbb{A}) + q(\partial\mathbb{A})$. Since $f(\mathbb{A}) = f(\mathbb{A}^*)$ for any biset $\mathbb{A}$, we get that $\mathcal{T}$ is symmetric. We will show that $\mathcal{T}$ is a crossing family. The functions $f(\mathbb{A})$ is submodular, namely, $f(\mathbb{A}) + f(\mathbb{B}) \geq f(\mathbb{A} \cap \mathbb{B}) + f(\mathbb{A} \cup \mathbb{B})$ holds for any two bisets $\mathbb{A}, \mathbb{B}$; this is

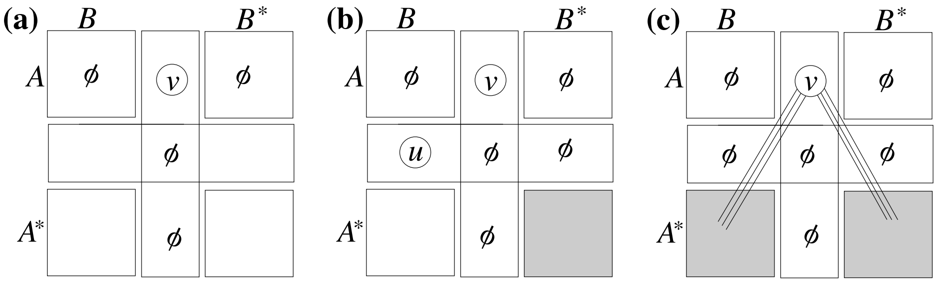

**Figure 3** Illustration to the proof of Lemma 11. Dark parts are non-empty, empty parts are marked by $\emptyset$; other parts may or may not be empty.

so since it is known that each of the functions $d_{G_0}(\mathbb{A})$ and $q(\partial\mathbb{A})$ is submodular. If $\mathbb{A},\mathbb{B}$ cross then $\mathbb{A}\cap\mathbb{B},\mathbb{A}\cup\mathbb{B}$ are both proper bisets and thus $f(\mathbb{A}\cap\mathbb{B}),f(\mathbb{A}\cup\mathbb{B})\geq k-1$. This implies

$$k-1+k-1=f(\mathbb{A})+f(\mathbb{B})\geq f(\mathbb{A}\cap\mathbb{B})+f(\mathbb{A}\cup\mathbb{B})\geq k-1+k-1\ .$$

Thus equality holds everywhere, hence $f(\mathbb{A}\cap\mathbb{B})=f(\mathbb{A}\cup\mathbb{B})=k-1$.

Now we show that any $\mathbb{A},\mathbb{B}\in\mathcal{T}$ cross or co-cross. Suppose to the contrary that there are $\mathbb{A},\mathbb{B}\in\mathcal{T}$ that do not cross nor co-cross. Then $A\subseteq\partial\mathbb{B}$, or $A^*\subseteq\partial\mathbb{B}$, or $B\subseteq\partial\mathbb{A}$, or $B^*\subseteq\partial\mathbb{A}$; say $A\subseteq\partial\mathbb{B}$; see Fig. 3(a). Then $A\cap\partial\mathbb{B}\neq\emptyset$ since $A\neq\emptyset$, and $\partial\mathbb{B}=\{v\}$ for some $v\in Q\cap A$ since $\mathbb{B}$ is tight.

Suppose that $\partial\mathbb{A}\neq\emptyset$, say $\partial\mathbb{A}=\{u\}$ for some $u\in Q\cap B$; see Fig. 3(b). This implies $A^*\cap B^*=A^*\neq\emptyset$. Since $\mathbb{A},\mathbb{B}$ are tight, $G_0$ has no edge between $A^*\cap B^*$ and $\{u,v\}$, contradicting that $G_0$ is connected.

Suppose that $\partial\mathbb{A}=\emptyset$. Then since $\mathbb{B}$ is a proper biset, none of $A^*\cap B,A^*\cap B^*$ is empty; see Fig. 3(c). Since $\partial\mathbb{B}\neq\emptyset$ and since $\mathbb{B}$ is tight, there is no edge between $B\cap A^*$ and $B^*\cap A^*$. This gives the contradiction $k-1+k-1\leq d_{G_0}(\mathbb{B}\setminus\mathbb{A})+d_{G_0}(\mathbb{A}\cup\mathbb{B})=d_{G_0}(\mathbb{A})=k-1$. ◀

▶ **Lemma 12.** *Fix some $\mathcal{T}$-core $\mathbb{C}_0$ and $s\in C_0$, and let $\mathcal{F}=\{\mathbb{A}\in\mathcal{T}:s\in A^*\}$. Then $\mathcal{F}$ is an intersecting biset family and $J$ is a feasible $(2,k)$-CA solution iff $J$ covers $\mathcal{F}$.*

**Proof.** To prove the lemma it is sufficient to show that if $J$ covers $\mathcal{F}$ then $J$ covers $\mathcal{T}$. Let $\mathbb{A}\in\mathcal{T}$. By Lemma 11, $\mathbb{C}_0\subseteq\mathbb{A}$ or $\mathbb{C}_0\subseteq\mathbb{A}^*$. If $\mathbb{C}_0\subseteq\mathbb{A}$ then $\mathbb{A}^*\in\mathcal{F}$ and if $\mathbb{C}_0\subseteq\mathbb{A}^*$ then $\mathbb{A}\in\mathcal{F}$. Thus $J$ covers $\mathbb{A}$ or $\mathbb{A}^*$, which is equivalent to covering $\mathbb{A}$. ◀

Let $\mathcal{C}=\mathcal{C}_{\mathcal{F}}$ be the family of $\mathcal{F}$-cores. Let $(H=(U,F),R,s,w)$ be a Rooted SS-CDS instance as in Definition 9, where $H$ is the separability graph of $\mathcal{F},E$, $R=\mathcal{C}$, the root is $s$, and for every $e\in E$, the weigh of the node $e$ in $H$ equals to the cost of $e$. By Lemma 11 and Corollary 8, $G_0\cup J$ is a feasible solution to $(2,k)$-CA iff $H[J]$ is connected and $J$ dominates $R$ in $H$. The first part of Theorem 4 now follows from Lemma 12 and Theorem 3.

For the second part of Theorem 4 we will prove the following.

▶ **Lemma 13.** *The* SS-CDS *instance $(H=(U,F),w,R)$ satisfies properties (A,B).*

**Proof.** We prove property (A). Consider an $\mathcal{F}$-core $\mathbb{C}$. Let $uv,xy\in E$ such that each of the pairs $uv,\mathbb{C}$ and $xy,\mathbb{C}$ is $\mathcal{F}$-inseparable. Then $\{u,v\}\cap C\neq\emptyset$ and $\{x,y\}\cap C\neq\emptyset$; say, $u,x\in C$. Let $\mathbb{A}\in\mathcal{F}_s$. If $\mathbb{C}\subseteq\mathbb{A}$ then $u,x\in A$. If $\mathbb{C}\subseteq\mathbb{A}^*$ then $u,x$ in $A^*$. In both cases, $\mathbb{A}$ cannot separate one of $uv,xy$ from the other.

Property (B) follows from the fact that if $\mathbb{C}$ and $uv$ are $\mathcal{F}$-inseparable then $u\in C$ or $v\in C$, and since $C\cap C'=\emptyset$ holds for any two $\mathcal{F}$-cores. ◀

This concludes the proof of Theorem 4.

Now we will show that $(2,k)$-CA generalizes the Crossing Family Cover problem. Let $(\mathcal{F},E,c)$ be an instance of Crossing Family Cover, where $\mathcal{F}$ is a symmetric crossing set family on a groundset $U$, and $E$ is an edge set with costs $\{c_e : e \in E\}$. We will show how to construct an equivalent $(2,3)$-CA instance. A **cactus** is a 2-edge-connected graph in which any two cycles have at most one node in common (equivalently: every block of the graph is a cycle). Dinitz, Karzanov, and Lomonosov [11] showed that the family $\mathcal{F}$ of minimum edge cuts of a graph $G$ on node set $U$ can be represented by the family $\mathcal{T}$ of minimum edge cuts of a cactus $G_0 = (V,E_0)$ and a mapping $\psi : U \longrightarrow V$, such that $\mathcal{F} = \{\psi^{-1}(A) : A \in \mathcal{T}\}$. Dinitz and Nutov [12, Theorem 4.2] (see also [31, Theorem 2.7]) extended this representation by showing that an arbitrary symmetric crossing family $\mathcal{F}$ can be represented by 2-edge cuts and specified 1-node cuts of a cactus.[2] This representation can be stated as follows.

▶ **Theorem 14** ([12]). *Let $\mathcal{F}$ be a crossing family on a groundset $U$. Then there exists a cactus $G_0 = (V,E_0)$, a mapping $\varphi : U \longrightarrow V$, and a set $Q$ of cut-nodes of $G_0$ with $\psi^{-1}(Q) = \emptyset$, such that $\mathcal{F} = \{\varphi^{-1}(A) : A \in \mathcal{A}\} \cup \{\varphi^{-1}(B) : \mathbb{B} \in \mathcal{B}\}$, where*

$$\mathcal{A} = \{A : d_{G_0}(\mathbb{A}) = 2\} \qquad \mathcal{B} = \{\mathbb{B} : \partial\mathbb{A} \textit{ is a single node in } Q, d_{G_0}(\mathbb{B}) = 0, B, B^* \neq \emptyset\} \ .$$

*Furthermore, if for any $A,B \in \mathcal{F}$ the set $(A \setminus B) \cup (B \setminus A)$ is not in $\mathcal{F}$, then $Q = \emptyset$.*

Given a Crossing Family Cover instance $(\mathcal{F},E,c)$ construct a $(2,3)$-CA instance $(G_0 = (V,E_0),E',Q,c')$ as follows.

- $G_0$ and $Q$ (and $\varphi$) are as in Theorem 14.
- For every edge $e = uv$ in $E$ there is an edge $e' = \varphi(u)\varphi(v)$ in $E'$ of cost $c'(e') = c(e)$.

Then $\mathcal{T} = \mathcal{A} \cup \mathcal{B}$ is the family of tight sets of $G_0$, and $J \subseteq E$ covers $\mathcal{F}$ iff $J' \subseteq E$ covers $\mathcal{T}$, where $J' = \{e' : e \in J\}$. Note that in the obtained $(2,3)$-CA instance no edge in $E'$ is incident to a node in $Q$, while in general $(2,3)$-CA instances such edge might exist. This suggests that $(2,3)$-CA strictly generalizes the Crossing Family Cover problem.

## References

1 A. Agrawal, P. Misra, F. Panolan, and S. Saurabh. Fast exact algorithms for survivable network design with uniform requirements. In *WADS*, pages 25–36, 2017.

2 H. Angelidakis, D. H. Denesik, and L. Sanità. Node connectivity augmentation via iterative randomized rounding. *CoRR*, abs/2108.02041, 2021. URL: https://arxiv.org/abs/2108.02041.

3 V. Auletta, Y. Dinitz, Z. Nutov, and D. Parente. A 2-approximation algorithm for finding an optimum 3-vertex-connected spanning subgraph. *J. Algorithms*, 32(1):21–30, 1999.

4 J. Bang-Jensen, M. Basavaraju, K. V Klinkby, P. Misra, M. Ramanujan, S. Saurabh, and M. Zehavi. Parameterized algorithms for survivable network design with uniform demands. In *SODA*, pages 2838–2850, 2018.

5 M. Basavaraju, F. V. Fomin, P. A. Golovach, P. Misra, M. S. Ramanujan, and S. Saurabh. Parameterized algorithms to preserve connectivity. In *ICALP, Part I*, pages 800–811, 2014.

6 A. Belgi and Z. Nutov. A polylogarithmic approximation algorithm for 2-edge-connected dominating set. *Information Processing Letters*, 173:106175, 2022.

7 J. Byrka, F. Grandoni, and A. J. Ameli. Breaching the 2-approximation barrier for connectivity augmentation: a reduction to Steiner tree. In *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing, STOC 2020*, pages 815–825, 2020.

[2] A representation identical to the one of [12] was announced later by Fleiner and Jordán [17].

8 F. Cecchetto, V. Traub, and R. Zenklusen. Bridging the gap between tree and connectivity augmentation: unified and stronger approaches. In *Proceedings of the 53rd ACM Symposium on Theory of Computing, STOC 2021*, pages 370–383, 2021.

9 J. Cheriyan and L. A. Végh. Approximating minimum-cost $k$-node connected subgraphs via independence-free graphs. *SIAM J. Comput.*, 43(4):1342–1362, 2014.

10 J. Chuzhoy and S. Khanna. An $O(k^3 \log n)$-approximation algorithm for vertex-connectivity survivable network design. *Theory Comput.*, 8(1):401–413, 2012.

11 E. A. Dinic, A. V. Karzanov, and M. V. Lomonosov. On the structure of a family of minimal weighted cuts in a graph. *Studies in Discrete Optimization*, pages 290–306, 1976.

12 Y. Dinitz and Z. Nutov. A 2-level cactus model for the system of minimum and minimum+1 edge-cuts in a graph and its incremental maintenance. In *Proceedings of the 27th Annual ACM Symposium on Theory of Computing, STOC 1995*, pages 509–518, 1995.

13 Y. Dinitz and Z. Nutov. A 3-approximation algorithm for finding optimum 4,5-vertex-connected spanning subgraphs. *J. Algorithms*, 32(1):31–40, 1999.

14 S. E. Dreyfus and R. A. Wagner. The steiner problem in graphs. *Networks*, 1.3:195–207, 1971.

15 A. E. Feldmann and D. Marx. The complexity landscape of fixed-parameter directed steiner network problems. In *ICALP*, pages 27:1–27:14, 2016.

16 A. E. Feldmann, A. Mukherjee, and E.-Jan van Leeuwen. The parameterized complexity of the survivable network design problem. In *SOSA*, pages 37–56, 2022.

17 T. Fleiner and T. Jordán. Coverings and structure of crossing families. *Mathematical Programming*, 84(3):505–518, 1999.

18 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.

19 A. Frank. Rooted $k$-connections in digraphs. *Discrete Applied Mathematics*, 157(6):1242–1254, 2009.

20 A. Frank and É. Tardos. An application of submodular flows. *Linear Algebra Appl.*, 114/115:329–348, 1989.

21 B. Fuchs, W. Kern, D. Molle, S. Richter, P. Rossmanith, and X. Wang. Dynamic programming for minimum steiner trees. *Theory of Computing Systems*, 41(3):493–500, 2007.

22 J Guo and J. Uhlmann. Kernelization and complexity results for connectivity augmentation problems. *Networks*, 56(2):131–142, 2010.

23 G. Gutin, M. Ramanujan, F. Reidl, and M. Wahlström. Path-contractions, edge deletions and connectivity preservation. *Journal of Computer and System Sciences*, 101:1–20, 2019.

24 B. Jackson and T. Jordán. Independence free graphs and vertex connectivity augmentation. *J. Comb. Theory, Ser. B*, 94(1):31–77, 2005.

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

26 S. Khuller and B. Raghavachari. Improved approximation algorithms for uniform connectivity problems. *J. Algorithms*, 21(2):434–450, 1996.

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

28 W. Mader. Ecken vom grad $n$ in minimalen n-fach zusammenhängenden graphen. *Archive der Mathematik*, 23:219–224, 1972.

29 D. Marx and L. A. Végh. Fixed-parameter algorithms for minimum-cost edge-connectivity augmentation. *ACM Trans. Algorithms*, 11(4):27:1–27:24, 2015.

30 H. Nagamochi. An approximation for finding a smallest 2-edge-connected subgraph containing a specified spanning tree. *Discrete Applied Mathematics*, 126:83–113, 2003.

31 Z. Nutov. *Structures of Cuts and Cycles in Graphs; Algorithms and Applications*. PhD thesis, Technion, Israel Institute of Technology, 1997.

32 Z. Nutov. Approximating rooted connectivity augmentation problems. *Algorithmica*, 44(3):213–231, 2006.

**33** Z. Nutov. Approximating minimum-cost connectivity problems via uncrossable bifamilies. *ACM Transactions on Algorithms*, 9(1):1–16, 2012.

**34** Z. Nutov. Approximating minimum-cost edge-covers of crossing biset-families. *Combinatorica*, 34(1):95–114, 2014.

**35** Z. Nutov. Improved approximation algorithms for minimum cost node-connectivity augmentation problems. *Theory Comput. Syst.*, 62(3):510–532, 2018.

**36** Z. Nutov. The $k$-connected subgraph problem. In T. F. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics, Second Edition, Volume 2: Contemporary and Emerging Applications*. Chapman and Hall/CRC, 2018.

**37** Z. Nutov. Node-connectivity survivable network problems. In T. F. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics, Second Edition, Volume 2: Contemporary and Emerging Applications*. Chapman and Hall/CRC, 2018.

**38** Z. Nutov. 2-node-connectivity network design. In *WAOA*, pages 220–235, 2020.

**39** Z. Nutov. On the tree augmentation problem. *Algorithmica*, 83(2):553–575, 2021.

**40** Z. Nutov. A $4+\epsilon$ approximation for $k$-connected subgraphs. *J. Computer and System Science*, 123:64–75, 2022.

**41** V. Traub and R. Zenklusen. Local search for weighted tree augmentation and steiner tree. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms, SODA 2022*, pages 3253–3272, 2022.

**42** V. Traub and R. Zenklusen. A $(1.5+\epsilon)$-approximation algorithm for weighted connectivity augmentation. In *Proceedings of Annual ACM Symposium on Theory of Computing, STOC 2023*, pages 1820–1833, 2023.

**43** L. A. Végh. Augmenting undirected node-connectivity by one. *SIAM J. Discret. Math*, 25(2):695–718, 2011.