Fast Exact Algorithms for Survivable Network Design with Uniform Requirements

Akanksha Agarwal[1], Pranabendu Misra[1], Fahad Panolan[1], and Saket Saurabh[1,2]

[1] Department of Informatics, University of Bergen, Norway.
{Akanksha.Agrawal|Pranabendu.Misra|Fahad.Panolan}@uib.no
[2] The Institute of Mathematical Sciences, HBNI, Chennai, India. saket@imsc.res.in

Abstract. We design exact algorithms for the following two problems in survivable network design: (i) designing a minimum cost network with a desired value of edge connectivity, which is called Minimum Weight λ-connected Spanning Subgraph and (ii) augmenting a given network to a desired value of edge connectivity at a minimum cost which is called Minimum Weight λ-connectivity Augmentation. Many well known problems such as Minimum Spanning Tree, Hamiltonian Cycle, Minimum 2-Edge Connected Spanning Subgraph and Minimum Equivalent Digraph reduce to these problems in polynomial time. It is easy to see that a minimum solution to these problems contains at most $2\lambda(n-1)$ edges. Using this fact one can design a brute-force algorithm which runs in time $2^{\mathcal{O}(\lambda n(\log n+\log \lambda)}$. However no better algorithms were known. In this paper, we give the first single exponential time algorithm for these problems, i.e. running in time $2^{\mathcal{O}(\lambda n)}$, for both undirected and directed networks. Our results are obtained via well known characterizations of λ-connected graphs, their connections to linear matroids and the recently developed technique of dynamic programming with representative sets.

1 Introduction

The survivable network design problem involves designing a cost effective communication network that can survive equipment failures. The failure may be caused by any number of things such as a hardware or software breakage, human error or a broken link between two network components. Designing a network which satisfies certain connectivity constraints, or augmenting a given network to a certain connectivity are important and well studied problems in network design. In terms of graph theory these problems correspond to finding a spanning subgraph of a graph which satisfies given connectivity constraints and, augmenting the given graph with additional edges so that it satisfies the given constraints, respectively. Designing a minimum cost network which connects all the nodes, is the well-known Minimum Spanning Tree(MST) problem. However such a network fails on the failure of a single link. This leads to the question of designing a minimum cost network which can survive one or more link failures. Such a network must be λ-connected, in order to survive $\lambda - 1$ link failures (we use the term λ-connected to represent λ-edge connected). This problem is NP-hard (for $\lambda \geq 2$), and a 2-approximation algorithm is known [18]. In the special case when the weights are 1 or ∞, i.e. we wish to find a minimum spanning λ-connected subgraph, a $1 + \frac{2}{\lambda+1}$ approximation may be obtained in polynomial time [6]. The above results also hold in the case of directed graphs. The case of $\lambda = 1$ for digraphs, known as Minimum Strong Spanning Subgraph(MSSS), is NP-hard as it is a generalization of the Hamiltonian Cycle. Further, the Minimum Equivalent Graph(MEG) problem reduces to it in polynomial time.

Adding a minimum number of edges to make the graph satisfy certain connectivity constrains is known as minimum augmentation problem. Minimum augmentation find application in designing survivable networks [11, 15] and in data security [13, 16]. Watanabe and Nakamura [25] gave a polynomial time algorithm for solving the λ-edge connectivity augmentation in an undirected graph, where we want to add minimum number of edges to the graph to make it λ-edge connected. Frank gave a polynomial time algorithm for the same problem in directed graphs [10]. However in the weighted case, or when the augmenting set must be a subset of a given set of links, the problem becomes NP-Hard problem. Even the restricted case of augmenting the edge connectivity of a graph from $\lambda - 1$ to λ remains NP-hard [1]. A 2-approximation may be obtained for these problems, by choosing a suitable weight function and applying the algorithm of [18]. We refer to [1, 4, 17, 19] for more details, other related problems and further applications. A few results are also known in the frameworks of parameterized complexity and exact exponential time algorithms. Marx and Végh gave an FPT algorithm for computing a minimum cost set of at most k links, which augments the connectivity of an undirected graph from $\lambda - 1$ to λ [21]. Basavaraju et al. [2] improved the running time of this algorithm and, also gave an algorithm for another variant of this problem. Bang-Jensen and Gutin [1, Chapter 12] obtain an FPT algorithm for a variant of MSSS in unweighted graphs. The first exact algorithms for MEG and MSSS, running in time $\mathcal{O}(2^{\mathcal{O}(m)} \cdot n^{\mathcal{O}(1)})$, where m is the number of edges in the graph, were given in by Moyles and Thompson [22] in 1969. Only recently, Fomin et al. [9] gave the first single-exponential algorithm for MEG and MSSS, i.e. with a running time of $2^{\mathcal{O}(n)}$. For the special case of Hamiltonian Cycle, a $\mathcal{O}(2^n)$ time algorithm is known [14, 3] for digraphs from 1960s. It was recently improved to $\mathcal{O}(1.657^n)$ for undirected graphs [5], and to $\mathcal{O}(1.888^n)$ for bipartite digraphs [8] (but these are randomized algorithms). For other results and more details we refer to Chapter 12 of [1].

In this paper we consider the problem of designing an exact algorithm for finding a minimum weight spanning subgraph of a given λ-connected (di)graph.

Minimum Weight λ-connected Spanning Subgraph **Input:** A graph G (or digraph D), and a weight function w on the edges(or the arcs). **Output:** A minimum weight spanning λ-connected subgraph.

One can observe that such a subgraph contains at most $\lambda(n-1)$ edges ($2\lambda(n-1)$ arcs for digraphs). Hence a solution can be obtained by enumerating all possible subgraphs with at most these many edges and testing if it is λ-connected. However such an algorithm will take $2^{\mathcal{O}(\lambda n(\log n+\log \lambda))}$ time. One may try a more clever approach, by using the observation that we can enumerate all possible minimal λ-connected graphs in $2^{\mathcal{O}(\lambda n)}$ time. Then we test if any of these graph is isomorphic to a subgraph of the input graph. However, subgraph isomorphism requires $2^{\lambda n(\log n+\log \lambda)}$ unless the Exponential Time Hypothesis fails [7]. In this paper, we give the first single exponential algorithm for this problem that runs in time $2^{\mathcal{O}(\lambda n)}$. As a corollary, we also obtain single exponential time algorithm for the minimum weight connectivity augmentation problem.

MINIMUM WEIGHT λ-CONNECTIVITY AUGMENTATION **Input:** A graph G (or a digraph D), a set of links $L \subseteq V \times V$ (ordered pairs in case of digraphs), and a weight function $w : L \to \mathbb{N}$. **Output:** A minimum weight subset L' of L such that $G \cup L$ (or $D \cup L$) is λ-connected

Our Methods and Results. We extend the algorithm of Fomin et al. for finding a MINIMUM EQUIVALENT GRAPH [9], to solve MINIMUM WEIGHT λ- CONNECTED SUB-DIGRAPH, exploiting the structural properties of λ-connected (di)graphs. A digraph D is λ-connected if and only if for some $r \in V(D)$, there is a collection $\mathbb{I}$ of λ arc disjoint in-branchings rooted at r and a collection $\mathbb{O}$ of λ arc disjoint out-branchings rooted at r. Then computing a $\mathbb{I}$ and a $\mathbb{O}$ with the largest possible intersection yields a minimum weight λ-connected spanning sub-digraph. We show that the solution can be embedded in a linear matroid of rank $\mathcal{O}(\lambda n)$, and then compute the solution by a dynamic programming algorithm with representative sets over this matroid.

Theorem 1. *Let D be a λ-edge connected digraph on n vertices and $w : A(D) \to \mathbb{N}$. Then we can find a minimum weight λ-edge connected subgraph of D in $2^{\mathcal{O}(\lambda n)}$ time.*

For the case of undirected graphs, no equivalent characterization is known. However, we obtain a characterization by converting the graph to a digraph with labels on the arcs, corresponding to the undirected edges. Then computing a solution that minimizes the number of labels used, gives the following theorem.

Theorem 2. *Let G be a λ-edge connected graph on n vertices and $w : E(G) \to \mathbb{N}$. Then we can find a minimum weight λ-edge connected subgraph of G in $2^{\mathcal{O}(\lambda n)}$ time.*

For the problem of augmenting a network to a given connectivity requirement, at a minimum cost, we obtain the following results by applying the previous theorems with suitably chosen weight functions.

Theorem 3. *Let D be a digraph (or a graph) on n vertices, $L \subseteq V(D) \times V(D)$ be a collection of links with weight function $w : L \to \mathbb{N}$. For any integer λ, we can find a minimum weight $L' \subseteq L$ such that $D' = (V(D), A(D) \cup L')$ is λ-edge connected, in time $2^{\mathcal{O}(\lambda n)}$.*

Due to space constraints, notations, standard definitions and other preliminaries have been moved to the appendix. The definitions and notation related to matriods and representative sets may also be found in [9].

2 Directed Graphs

In this section, we give a single exponential exact algorithm, that is of running time $2^{\mathcal{O}(\lambda n)}$, for computing a minimum weight spanning λ-connected subgraph of a λ connected n-vertex digraph. We first consider the unweighted version of the problem and it will be clear that the same algorithm works for weighted version as well. In a digraph D, we define $\mathsf{Out}_D(v) = \{(v, w) \in A(D)\}$ and $\mathsf{In}_D(v) = \{(u, v) \in A(D)\}$ to be the set of *out-edges* and *in-edges* of v respectively. We begin with the following characterization of λ-connectivity in digraphs.

Lemma 1 ($*$[3]). *Let D be a digraph. Then D is λ-connected if and only if for any $r \in V(D)$, there is a collection of λ arc disjoint in-branchings rooted at r, and a collection of λ arc disjoint out-branchings rooted at r.*

Let D be the input to our algorithm, which is a λ-connected digraph on n vertices. Let us fix a vertex $r \in V(D)$. By Lemma 1, any minimal λ-connected subgraph of D is a union of a collection $\mathbb{I}$ of λ arc disjoint in-branchings and a collection $\mathbb{O}$ of λ arc disjoint out-branchings which are all rooted at vertex r. The following lemma relates the size of such a minimal subgraph to the number of arcs which appear in both $\mathbb{I}$ and $\mathbb{O}$ and it follows easily from Lemma 1. Here, $A(\mathbb{I})$ denotes the set of arcs which are present in some $I \in \mathbb{I}$ and $A(\mathbb{O})$ denotes the set of arcs which are present in some $O \in \mathbb{O}$.

Lemma 2. *Let D be a λ-connected digraph, r be a vertex in $V(D)$ and $\ell \in [\lambda(n-2)]$. Then a subdigraph D' with at most $2\lambda(n-1) - \ell$ arcs, is a minimal λ-connected spanning subdigraph of D if and only if D' is a union of a collection $\mathbb{I}$ of arc disjoint in-branchings rooted at r, and a collection $\mathbb{O}$ of arc disjoint out-branchings rooted at r such that $|A(\mathbb{I}) \cap A(\mathbb{O})| \geq \ell$ (i.e. they have at least ℓ common arcs).*

By Lemma 2, a minimum λ connected subgraph of D is $\mathbb{I} \cup \mathbb{O}$, where $\mathbb{O} = \{O_1, O_2, \ldots O_\lambda\}$ is a collection of λ arc disjoint out-branchings rooted at r, $\mathbb{I} = \{I_1, I_2, \ldots I_\lambda\}$ is a collection of λ arc disjoint in-branchings rooted at r, and $A(\mathbb{O}) \cap A(\mathbb{I})$ is maximized. To explain the concept of the algorithm let us assume that the number of arcs in a minimum λ connected spanning subdigraph D' is $2\lambda(n-1) - \ell$ and let $A(D') = A(\mathbb{O}) \cup A(\mathbb{I})$, where $\mathbb{O} = \{O_1, O_2, \ldots O_\lambda\}$ is a

[3] Proof of the results marked ($*$) will appear in the full version of the paper.

collection of λ arc disjoint out-branchings rooted at r and $\mathbb{I} = \{I_1, I_2, \dots I_\lambda\}$ is a collection of λ arc disjoint in-branchings rooted at r. Note that $|A(\mathbb{O}) \cap A(\mathbb{I})| = \ell$. The first step of our algorithm is to construct the set $A(\mathbb{O}) \cap A(\mathbb{I})$, and then, given the intersection, we can construct $\mathbb{O}$ and $\mathbb{I}$ in polynomial time. Observe that $A(\mathbb{O})$ and $A(\mathbb{I})$ can intersect in at most $\lambda(n-2)$ arcs. The main idea is to enumerate a subset of potential candidates for the intersection, via dynamic programming. But note that there could be as many as $n^{\mathcal{O}(\lambda n)}$ such candidates, and enumerating them all will violate the claimed running time. So we try a different approach. We first observe that the arcs in a solution, $\mathbb{O} \cup \mathbb{I}$, can be embedded into a linear matroid of rank $\mathcal{O}(\lambda n)$. Then we prove that, it is enough to keep a representative family of the partial solutions in the dynamic programming table. Since, the size of the representative family is bounded by $2^{\mathcal{O}(\lambda n)}$, our algorithm runs in the claimed running time.

Let us delve into the details of the algorithm. Let D_r^- be the digraph obtained from D after removing the arcs in $\mathsf{Out}_D(r)$. Similarly, let D_r^+ be the digraph obtained from D after removing the arcs in $\mathsf{In}_D(r)$. Observe that the arc sets of D_r^- and D_r^+ can be partitioned as follows. $A(D_r^-) = \biguplus_{v \in V(D_r^-)} \mathsf{Out}_{D_r^-}(v)$ and $A(D_r^+) = \biguplus_{v \in V(D_r^+)} \mathsf{In}_{D_r^+}(v)$. We construct a pair of matroids corresponding to each of the λ in-branching in $\mathbb{I}$ and each of the λ out-branching in $\mathbb{O}$. For each in-branching $I_i \in \mathbb{I}$, we have a matroid $\mathcal{M}_{I,1}^i = (E_{I,1}^i, \mathcal{I}_{I,1}^i)$ which is a graphic matroid in D and $E_{I,1}^i$ is a copy of the arc set of D. And similarly, for each out-branching $O_i \in \mathbb{O}$, we have a matroid $\mathcal{M}_{O,1}^i = (E_{O,1}^i, \mathcal{I}_{O,1}^i)$ which is a graphic matroid in D and $E_{O,1}^i$ is again a copy of the arc set of D. Note that the rank of these graphic matroids is $n-1$. Next, for each I_i, we define matroid $\mathcal{M}_{I,2}^i = (E_{I,2}^i, \mathcal{I}_{I,2}^i)$ which is a partition matroid where $E_{I,2}^i$ is a copy of the arc set of D_r^- and $\mathcal{I}_{I,2}^i = \{X \mid X \subseteq E_{I,2}^i, |X \cap \mathsf{Out}_{D_r^-}(v)| \leq 1, \textit{ for all } v \in V(D_r^-)\}$ [4]. Since $\mathsf{Out}_{D_r^-}(r) = \emptyset$ and $|V(D_r^-)| = n$, we have that the rank of these partition matroids, $\mathcal{M}_{I,2}^i$, $i \in [\lambda]$, is $n-1$. Similarly, for each O_i, we define $\mathcal{M}_{O,2}^i = (E_{O,2}^i, \mathcal{I}_{O,2}^i)$ as the partition matroid, where $E_{O,2}^i$ is a copy of the arc set of D_r^+ and $\mathcal{I}_{O,2}^i = \{X \mid X \subseteq E_{O,2}^i, |X \cap \mathsf{In}_{D_r^+}(v)| \leq 1, \textit{ for all } v \in V(D_r^+)\}$ Since $\mathsf{In}_{D_r^+}(r) = \emptyset$ and $V(D_r^+) = n$, we have that the rank of these partition matroids, $\mathcal{M}_{O,2}^i$, $i \in [\lambda]$, is $n-1$. We define two uniform matroids $\mathcal{M}_I$ and $\mathcal{M}_O$ of rank $\lambda(n-1)$, corresponding to $\mathbb{I}$ and $\mathbb{O}$, on the ground sets E_I and E_O, respectively, where E_i and E_O are copies of the arc set of D. We define matroid $\mathcal{M} = (E, \mathcal{I})$ as the direct sum of $\mathcal{M}_I, \mathcal{M}_O$, $\mathcal{M}_{I,j}^i, \mathcal{M}_{O,j}^i$, for $i \in [\lambda]$ and $j \in [2]$, That is,

$$\mathcal{M} = \Big(\bigoplus_{i \in [\lambda], j \in [2]} (\mathcal{M}_{I,j}^i \oplus \mathcal{M}_{O,j}^i) \Big) \oplus \mathcal{M}_I \oplus \mathcal{M}_O$$

Since the rank of $\mathcal{M}_{I,j}^i, \mathcal{M}_{O,j}^i$ where $i \in [\lambda]$ and $j \in [2]$, are $n-1$ each, and rank of $\mathcal{M}_I$ and $\mathcal{M}_O$ is $\lambda(n-1)$, we have that the rank of $\mathcal{M}$ is $6\lambda(n-1)$. We briefly discuss the representation of these matroids. The matroids $\mathcal{M}_{I,1}^i$, $\mathcal{M}_{O,1}^i$

[4] We slightly abuse notation for the sake of clarity, as strictly speaking X and $\mathsf{Out}_{D_G^r}(v)$ are disjoint, since they are subsets of two different copies of the arc set.

for $i \in [\lambda]$ are graphic matroids, which are representable over any field of size at least 2. The matroids $\mathcal{M}^i_{I,2}, \mathcal{M}^i_{O,1}$ are partition matroids with partition size 1, and therefore they are representable over any field of at least 2 as well. Finally, the two uniform matroids, $\mathcal{M}_I$ and $\mathcal{M}_O$, are representable over any field with at least $|A(D)|+1$ elements. Hence, at the start of our algorithm, we choose a representation of all these matroids over a field $\mathbb{F}$ of size at least $|A(D)|+1$. So by Proposition 2, $\mathcal{M}$ is representable over any field of size at least $|A(D)|+1$.

For an arc $e \in A(D)$ not incident to r there are $4\lambda+2$ copies of it in $\mathcal{M}$. Let $e^i_{J,j}$ denotes it's copy in $E^i_{J,j}$, where $i \in [\lambda]$, $j \in [2]$ and $J \in \{I, O\}$. An arc incident to r has only $3\lambda+2$ copies in $\mathcal{M}$. For an arc $e \in \mathsf{In}_D(r)$ we will denote its copies in $E^i_{I,1}, E^i_{O,1}, E^i_{I,2}$ by $e^i_{I,1}, e^i_{O,1}, e^i_{I,2}$, and similarly for an arc $e \in \mathsf{Out}_D(r)$ we will denote its copies in $E^i_{I,1}, E^i_{O,1}, E^i_{O,2}$ by $e^i_{I,1}, e^i_{O,1}, e^i_{O,2}$. And finally, for any arc $e \in A(D)$, let e_I and e_O denote it's copies in E_I and E_O, respectively. For $e \in A(D) \setminus \mathsf{Out}_D(r)$ and $i \in [\lambda]$, let $S^i_{I,e} = \{e^i_{I,1}, e^i_{I,2}\}$. Similarly for $e \in A(D) \setminus \mathsf{In}_D(r), i \in [\lambda]$, let $S^i_{O,e} = \{e^i_{O,1}, e^i_{O,2}\}$. Let $S_e = (\cup_{i=1}^{\lambda} S^i_{I,e}) \bigcup (\cup_{j=1}^{\lambda} S^j_{O,e}) \bigcup \{e_I, e_O\}$. For $X \in \mathcal{I}$, let A_X denote the set of arcs $e \in A(D)$ such that $S_e \cap X \neq \emptyset$.

Observation 1 *Let I be an in-branching in D rooted at r. Then for any $i \in [\lambda]$, $\{e^i_{I,1} \mid e \in A(I)\}$ is a basis in $\mathcal{M}^i_{I,1}$ and $\{e^i_{I,2} \mid e \in A(I)\}$ is a basis in $\mathcal{M}^i_{I,2}$. And conversely, let X and Y be basis of $\mathcal{M}^i_{I,1}$ and $\mathcal{M}^i_{I,2}$, respectively, such that $A_X = A_Y$. Then A_X is an in-branching rooted at r in D.*

Observation 2 *Let O be an out-branching in D. Then for any $i \in [\lambda]$, $\{e^i_{O,1} \mid e \in A(O)\}$ is a basis in $\mathcal{M}^i_{O,1}$ and $\{e^i_{O,2} \mid e \in A(O)\}$ is a basis in $\mathcal{M}^i_{O,2}$. And conversely, let X and Y be basis of $\mathcal{M}^i_{O,1}$ and $\mathcal{M}^i_{O,2}$, respectively, such that $A_X = A_Y$. Then A_X is an out-branching rooted at r in D.*

Observe that any arc $e \in A(D)$ can belong to at most one in-branching in $\mathbb{I}$ and at most one out-branching in $\mathbb{O}$, because both $\mathbb{I}$ and $\mathbb{O}$ are collection of arc disjoint subgraphs of D. Because of Observation 1 and 2, if we consider that each $I_i \in \mathbb{I}$ is embedded into $\mathcal{M}^i_{I,1}$ and $\mathcal{M}^i_{I,2}$ and each $O_i \in \mathbb{O}$ is embedded into $\mathcal{M}^i_{O,1}$ and $\mathcal{M}^i_{O,2}$, then we obtain an independent set Z' of rank $4\lambda(n-1)$ corresponding to $\mathbb{I} \cup \mathbb{O}$ in the matroid $\mathcal{M}$. Further, since the collection $\mathbb{I}$ is arc disjoint, $\{e_I \mid e \in A(\mathbb{I})\}$ is a basis of $\mathcal{M}_I$. And similarly, $\{e_O \mid e \in A(\mathbb{O})\}$ is a basis of $\mathcal{M}_O$. Therefore, $Z = Z' \cup \{e_I \mid e \in A(\mathbb{I})\} \cup \{e_O \mid e \in A(\mathbb{O})\}$ is a basis of $\mathcal{M}$. Now observe that, each arc in the intersection $\mathbb{I} \cap \mathbb{O}$ has six copies in the independent set Z. The remaining arcs in $\mathbb{I} \cup \mathbb{O}$ have only three copies each, and this includes any arc which is incident on r. Now, we define a function $\phi : \mathcal{I} \times A(D) \to \{0,1\}$, where for $W \in \mathcal{I}$ and $e \in A(D)$, $\phi(W,e) = 1$ if and only if exactly one of the following holds. Either, $W \cap S_e = \emptyset$. Or, $\{e_I, e_O\} \subseteq W$ and there exists $t, t' \in [\lambda]$ such that $S^t_{I,e} \subseteq W$ and $S^{t'}_{O,e} \subseteq W$. And for each $i \in [\lambda] \setminus \{t\}$ and $j \in [\lambda] \setminus \{t'\}$, $S^i_{I,e} \cap W = \emptyset$ and $S^j_{O,e} \cap W = \emptyset$. Using function ϕ we define the following collection of independent sets of $\mathcal{M}$. $\mathcal{B}^{6\ell} = \{W \mid W \in \mathcal{I}, |W| = 6\ell, \forall e \in A(D)\ \phi(W,e) = 1\}$ By the definitions of

ϕ, $\mathbb{I}$ and $\mathbb{O}$, $\bigcup_{e\in A(\mathbb{O})\cap A(\mathbb{I})} S_e$ is an independent set of $\mathcal{M}$, which is contained in $\mathcal{B}^{6\ell}$. In fact, for the optimal value of ℓ, the collection $\mathcal{B}^{6\ell}$ contains all possible candidates for the intersection of $\mathbb{O}'$ and $\mathbb{I}'$, where $\mathbb{O}'$ and $\mathbb{I}'$ are collections of arc disjoint in-branchings and arc disjoint out-branchings which form an optimum solution. Our goal is to find one such candidate partial solution from $\mathcal{B}^{6\ell}$. We are now ready to state the following lemma which shows that a representative family of $\mathcal{B}^{6\ell}$ always contains a candidate partial solution which can be extended to a complete solution.

Lemma 3. *Let D be a λ-connected digraph on n vertices, $r \in V(D)$ and $\ell \in [\lambda(n-2)]$. There exists a λ-connected spanning subdigraph D' of D with at most $2\lambda(n-1) - \ell$ arcs if and only if, there exists $\widehat{T} \in \widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$, where $n' = 6\lambda(n-1)$, such that D has λ arc disjoint in-branchings containing $A_{\widehat{T}}$ and λ arc disjoint out-branchings containing $A_{\widehat{T}}$, which are all rooted at r.*

Proof. In the forward direction consider a λ-connected spanning subdigraph D' of D with at most $2\lambda(n-1) - \ell$ arcs. By Lemma 2, D' is union of a collection $\mathbb{I} = \{I_1, I_2, \ldots, I_\lambda\}$ of arc disjoint in-branchings rooted at r, and a collection $\mathbb{O} = \{O_1, O_2, \ldots, O_\lambda\}$ of arc disjoint out-branchings rooted at r such that $|A(\mathbb{I}) \cap A(\mathbb{O})| \geq \ell$. By Observation 1, for all $i \in [\lambda]$, $\{e^i_{I,1} \mid e \in A(I_i)\}$ is a basis in $\mathcal{M}^i_{I,1}$ and $\{e^i_{I,2} \mid e \in A(I_i)\}$ is a basis in $\mathcal{M}^i_{I,2}$. Similarly, by Observation 2, for all $i \in [\lambda]$, $\{e^i_{O,1} \mid e \in A(O_i)\}$ is a basis in $\mathcal{M}^i_{O,1}$ and $\{e^i_{O,2} \mid e \in A(O_i)\}$ is a basis in $\mathcal{M}^i_{O,2}$. Further $\{e_I \mid e \in A(\mathbb{I})\}$ and $\{e_O \mid e \in A(\mathbb{O})\}$ are bases of $\mathcal{M}_I$ and $\mathcal{M}_O$ respectively. Hence the set $Z_{D'} = \{e^i_{I,1}, e^i_{I,2} \mid e \in A(I_i),\ i \in [\lambda]\} \cup \{e^i_{O,1}, e^i_{O,2} \mid e \in A(O_i),\ i \in [\lambda]\} \cup \{e_I \mid e \in A(\mathbb{I})\} \cup \{e_O \mid e \in A(\mathbb{O})\}$ is an independent set in $\mathcal{M}$. Since $|Z_{D'}| = 6\lambda(n-1)$, $Z_{D'}$ is actually a basis in $\mathcal{M}$. Consider $T \subseteq A(\mathbb{I}) \cap A(\mathbb{O})$ with exactly ℓ arcs. Let $T' = \{e^i_{I,1}, e^i_{I,2} \mid e \in T \cap I_i,\ \textit{for some } i \in [\lambda]\} \cup \{e^i_{O,1}, e^i_{O,2} \mid e \in T \cap O_i,\ \textit{for some } i \in [\lambda]\} \cup \{e_I, e_O \mid e \in T\}$. Note that T' is a set of six copies of the ℓ arcs that are common to a pair of an in-branching in $\mathbb{I}$ and an out-branching in $\mathbb{O}$. Therefore, by the definition of $\mathcal{B}^{6\ell}$, $T' \in \mathcal{B}^{6\ell}$. Then, by the definition of representative family, there exists $\widehat{T} \in \widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$, such that $\widehat{Z} = (Z_{D'} \setminus T') \cup \widehat{T}$ is an independent set in $\mathcal{M}$. Note that $|\widehat{Z}| = 6\lambda(n-1)$, and hence it is a basis in $\mathcal{M}$. Also note that $A_{\widehat{T}} \subseteq A_{\widehat{Z}}$.

Claim. 1 $(*)$ *For any $i \in [\lambda]$ and $e \in A(D)$, either $\{e^i_{I,1}, e^i_{I,2}\} \subseteq \widehat{Z}$ or $\{e^i_{I,1}, e^i_{I,2}\} \cap \widehat{Z} = \emptyset$. And further for every $e \in A(D)$ such that $e^i_{I,1} \in \widehat{Z}$ for some $i \in [\lambda]$, $\widehat{Z}$ also contains e_I. Similarly, for any $i \in [\lambda]$ and $e \in A(D)$, either $\{e^i_{O,1}, e^i_{O,2}\} \subseteq \widehat{Z}$ or $\{e^i_{O,1}, e^i_{O,2}\} \cap \widehat{Z} = \emptyset$, and further, for every $e \in A(D)$ such that $e^i_{O,1} \in \widehat{Z}$ for some $i \in [\lambda]$, $\widehat{Z}$ also contains e_O.*

Claim. 2 $(*)$ *For any $i, j \in [\lambda]$, $i \neq j$, either $\{e^i_{I,1}, e^i_{I,2}\} \cap \widehat{Z} = \emptyset$ or $\{e^j_{I,1}, e^j_{I,2}\} \cap \widehat{Z} = \emptyset$. Similarly, for any $i, j \in [\lambda]$, $i \neq j$, either $\{e^i_{O,1}, e^i_{O,2}\} \cap \widehat{Z} = \emptyset$ or $\{e^i_{O,1}, e^i_{O,2}\} \cap \widehat{Z} = \emptyset$.*

Since $\widehat{Z}$ is a basis in $\mathcal{M}$, by Proposition 1, for any $i \in [\lambda], j \in [2]$ and $k \in \{I, O\}$, we have that $\widehat{Z} \cap E^i_{k,j}$ is a basis in $\mathcal{M}^i_{k,j}$. For each $i \in [\lambda]$, let $\widehat{X}^i_1 = \widehat{Z} \cap E^i_{I,1}$ and $\widehat{X}^i_2 = \widehat{Z} \cap E^i_{I,2}$. By Claim 1, $A_{\widehat{X}^i_1} = A_{\widehat{X}^i_2}$ and hence, by Observation 1, $\widehat{I}_i = A_{\widehat{X}^i_1}$ forms an in-branching rooted at r. Because of Claim 2, $\{\widehat{I}_i \mid i \in \lambda\}$ are pairwise arc disjoint as $\widehat{I}_i \cap \widehat{I}_j = \emptyset$ for every $i \neq j \in [\lambda]$. Further $A_{\widehat{T}}$ is covered in arc disjoint in-branchings $\{A_{I_{i,1}} \mid i \in \lambda\}$, as $\widehat{T} \cap E^i_{I,j} \subseteq \widehat{X}^i_j$ for $j \in [2]$. By similar arguments we can show that there exist a collection $\{\widehat{O}_i \mid i \in [\lambda]\}$ of λ out-branchings rooted at r containing $A_{\widehat{T}}$. The reverse direction of the lemma follows from Lemma 2. □

Lemma 4. *Let D be a λ connected digraph on n vertices and $\ell \in [\lambda(n-2)]$. In time $2^{\mathcal{O}(\lambda n)}$ we can compute $\widehat{\mathcal{B}}^{6\ell} \subseteq^{n'-6\ell}_{rep} \mathcal{B}^{6\ell}$ such that $|\widehat{\mathcal{B}}^{6\ell}| \leq \binom{n'}{6\ell}$. Here $n' = 6\lambda(n-1)$.*

Proof. We give an algorithm via dynamic programming. Let $\mathcal{D}$ be an array of size $\ell+1$. For $i \in \{0, 1, \ldots, \ell\}$ the entry $\mathcal{D}[i]$ will store the family $\widehat{\mathcal{B}}^{6i} \subseteq^{n'-6i}_{rep} \mathcal{B}^{6i}$. We will fill the entries in array $\mathcal{D}$ according to the increasing order of index i, i.e. from $0, 1, \ldots, \ell$. For $i = 0$, we have $\widehat{\mathcal{B}}^0 = \{\emptyset\}$. Let $\mathcal{W} = \{\{e_I, e_O, e^i_{I,1}, e^i_{I,2}, e^j_{O,1}, e^j_{O,2}\} \mid i, j \in [\lambda], e \in A(D)\}$ and note that $|\mathcal{W}| = \lambda^2 m$, where $m = |A(D)|$. Given that we have filled all the entries $\mathcal{D}[i']$, where $i' < i+1$, we fill the entry $\mathcal{D}[i+1]$ at step $i+1$ as follows. Let $\mathcal{F}^{6(i+1)} = (\widehat{\mathcal{B}}^{6i} \bullet \mathcal{W}) \cap \mathcal{I}$.

Claim. 1 (∗) *$\mathcal{F}^{6(i+1)} \subseteq^{n'-6(i+1)}_{rep} \widehat{\mathcal{B}}^{6(i+1)}$, for all $i \in \{0, 1, \ldots \ell - 1\}$*

Now the entry for $\mathcal{D}[i+1]$ is $\widehat{\mathcal{F}}^{6(i+1)}$ which is $n' - 6(i+1)$ representative family for $\mathcal{F}^{6(i+1)}$, which is computed as follows. By Theorem 6 we know that $|\widehat{\mathcal{B}}^{6i}| \leq \binom{n'}{6i}$. Hence it follows that $|\mathcal{F}^{6(i+1)}| \leq \lambda^2 m \binom{n'}{6i}$ and moreover, we can compute $\mathcal{F}^{6(i+1)}$ in time $\mathcal{O}(\lambda^2 m n \binom{n'}{6i})$. We use Theorem 6 to compute $\widehat{\mathcal{F}}^{6(i+1)} \subseteq^{n'-6(i+1)}_{rep} \mathcal{F}^{6(i+1)}$ of size at most $\binom{n'}{6(i+1)}$. This step can be done in time $\mathcal{O}(\binom{n'}{6(i+1)} tp^\omega + t \binom{n'}{6(i+1)}^{\omega-1})$, where $t = |\mathcal{F}^{6(i+1)}| = \lambda^2 m \binom{n'}{6i}$. We know from Claim 1 that $\mathcal{F}^{6(i+1)} \subseteq^{n'-6(i+1)}_{rep} \mathcal{B}^{6(i+1)}$, and therefore by Lemma 10 we have $\widehat{\mathcal{B}}^{6(i+1)} = \widehat{\mathcal{F}}^{6(i+1)} \subseteq^{n'-6(i+1)}_{rep} \mathcal{B}^{6(i+1)}$. Finally, we assign the family $\widehat{\mathcal{B}}^{6(i+1)}$ to $\mathcal{D}[i+1]$. This completes the description of the algorithm and its correctness.

Since $\ell \leq n'/6$, we can bound the total running time of this algorithm as $\mathcal{O}\big(\sum_{i=1}^{\ell} \Big(i^\omega \binom{n'}{6(i+1)} + \binom{n'}{6(i+1)}^{\omega-1}\Big) \lambda^2 m \binom{n'}{6i}\big) \leq 2^{\mathcal{O}(\lambda n)}$. □

We have the following algorithm for computing $\mathbb{I}$ and $\mathbb{O}$ given $A(\mathbb{I}) \cap A(\mathbb{O})$. This algorithm extends a given set of arcs to an minimum weight collection of λ arc disjoint out-branchings. This is a simple corollary of [24, Theorem 53.10] and it also follows from the results of Gabow [12].

Lemma 5 (∗). *Let D be a digraph and w be a weight function on the arcs. For any subset X of arcs of D, a vertex r and an integer λ, we can find a minimum weight collection $\mathbb{O}$ of λ arc disjoint out-branchings rooted at r, such that $X \subseteq A(\mathbb{O})$, if it exists, in polynomial time.*

Theorem 4. *Let D be a λ edge connected digraph on n vertices. Then we can find a minimum λ edge connected subgraph of D in $2^{\mathcal{O}(\lambda n)}$ time.*

Proof. Let $n' = 6\lambda(n-1)$. By Lemma 2 we know that finding a minimum subdigraph D' of D is equivalent to finding a collection $\mathbb{I}$ of λ arc disjoint in-branchings and a collection $\mathbb{O}$ of λ arc disjoint out-branchings which are all rooted at a vertex $r \in V(D)$ such that $|A(\mathbb{I}) \cap A(\mathbb{O})|$ is maximized . We fix an arbitrary $r \in V(D)$ and for each choice of ℓ, the cardinality of $|A(\mathbb{I}) \cap A(\mathbb{O})|$, we attempt to construct a solution. By Lemma 3 we know that there exists a λ-connected spanning subdigraph D' of D with at most $2\lambda(n-1) - \ell$ arcs if and only if there exists $\widehat{T} \in \widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$, where $n' = 6\lambda(n-1)$, such that D has a collection $\mathbb{I} = \{I_1, I_2, \ldots, I_\lambda\}$ of arc disjoint in-branchings rooted at r and a collection $\mathbb{O} = \{O_1, O_2, \ldots, O_\lambda\}$ of arc disjoint out-branchings rooted at r such that $A_{\widehat{T}} \subseteq A(\mathbb{I}) \cap A(\mathbb{O})$. Using Lemma 4 we compute $\widehat{\mathcal{B}}^{6l} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$ in time $2^{\mathcal{O}(\lambda n)}$, and for every $F \in \widehat{\mathcal{B}}^{6\ell}$ we check if A_F can be extended to a collection of λ arc disjoint out-branchings rooted at r and a collection of λ arc disjoint in-branchings rooted at r, using Lemma 5. Since $\ell \leq \lambda(n-2)$, the running time of the algorithm is bounded by $2^{\mathcal{O}(\lambda n)}$. $\square$

An algorithm with the same running time can be obtained for the weighted version of the problem using the notion of weighted representative sets in the above, thus proving Theorem 1.

3 Undirected Graphs

In this section, we give an algorithm for computing a minimum λ-connected subgraph of an undirected graph G. As before, we only consider the unweighted version of the problem. While there is no equivalent characterization of λ-connected graphs as there was in the case of digraphs, we show that we can obtain a characterization by converting the graph to a digraph with labels on the arcs. Then, as in the previous section, we embed the solutions in a linear matroid and compute them by a dynamic programming algorithm with representative families. Let D_G be the digraph with $V(D_G) = V(G)$ and for each edge $e = (u, v) \in E(G)$, we have two arcs $a_e = (u, v)$ and $a'_e = (v, u)$ in $A(D_G)$. We label the arcs a_e and a'_e by the edge e, which is called the *type* of these arcs. For $X \subseteq A(D_G)$ let $\mathsf{Typ}(X) = \{e \in E(G) \mid a_e \in X \text{ or } a'_e \in X\}$. The following two lemmata relate λ-connected subgraphs of G with collections of out-branchings in D_G.

Lemma 6 (∗). *Let G be an undirected graph and D_G be the digraph constructed from G as described above. Then G is λ-connected if and only if for any $r \in V(D_G)$, there are λ arc disjoint out-branchings rooted at r in D_G.*

By Lemma 6 we know that G is λ-connected if and only if for any $r \in V(D)$, there is a collection $\mathbb{O}$ of λ arc disjoint out-branchings rooted at r in D_G. Given a collection of out-branchings, we can obtain a λ-connected subgraph of G with at most $\lambda(n-1)$ edges. For an edge $e \in E(G)$ which is not incident on r, the two arcs corresponding to it in D_G may appear in two distinct out-branchings of $\mathbb{O}$, but for an edge e incident on r in G, only the corresponding outgoing arc of r may appear in $\mathbb{O}$. Since there are $\lambda(n-1)$ arcs in total that appear in $\mathbb{O}$ and at least λ of those are incident on r, the number of edges of G such that both the arcs corresponding to it appear in $\mathbb{O}$ is upper bounded by $\frac{\lambda(n-2)}{2}$. So any minimal λ-connected subgraph of G has $\lambda(n-1)-\ell$ edges where $\ell \in [\lfloor\frac{\lambda(n-2)}{2}\rfloor]$.

Lemma 7 (∗). *Let G be an undirected λ-connected graph on n vertices and $\ell \in [\lfloor\frac{\lambda(n-2)}{2}\rfloor]$. G has a λ-connected subgraph G' with at most $\lambda(n-1)-\ell$ edges if and only if for any $r \in V(D_G)$, $D_{G'}$ has λ arc disjoint out-branchings $\mathbb{O} = \{O_1, O_2, \ldots, O_\lambda\}$ rooted at r such that $|\mathsf{Typ}(A(\mathbb{O}))| \leq \lambda(n-1)-\ell$.*

By Lemma 7, a collection $\mathbb{O}$ of out-branchings rooted at some vertex r, that minimizes $|\mathsf{Typ}(A(\mathbb{O}))|$ corresponds to a minimum λ-connected subgraph of G. In the rest of this section, we design an algorithm that finds a collection of arc disjoint out-branchings $\mathbb{O}$ in D_G such that $|\mathsf{Typ}(A(\mathbb{O}))|$ is minimized. The first step of our algorithm is to compute the set of edges of G such that both the arcs corresponding to it appear in the collection $\mathbb{O}$, and then we can extend this to a full solution in polynomial time.

Fix a vertex r. Let D_G^r denote the digraph obtained from D_G by removing the arcs in $\mathsf{In}_{D_G}(r)$. Observe that $A(D_G^r)$ can be partitioned as follows. $A(D_G^r) = \biguplus_{v \in V(D_G^r)} \mathsf{In}_{D_G^r}(v)$ We construct a pair of a graphic matroid and a partition matroid, corresponding to each of the λ out-branching that we want to find. For each $i \in [\lambda]$, we define a matroid $\mathcal{M}_1^i = (A_1^i, \mathcal{I}_1^i)$ which is a graphic matroid of D_G^r whose ground set A_1^i is a copy of the arc set $A(D_G^r)$. Similarly, for each $i \in [\lambda]$ we define matroid $\mathcal{M}_2^i = (A_2^i, \mathcal{I}_2^i)$, which is a partition matroid on the ground set A_2^i, which is a copy of the arc set $A(D_G^r)$, such that the following holds. $\mathcal{I}_2^i = \{I \mid I \subseteq A_2^i, |I \cap \mathsf{In}_{D_G^r}(v)| \leq 1, \textit{ for all } v \in V(D_G^r)\}$ Next, let $\mathcal{M}_O$ be a uniform matroid of rank $\lambda(n-1)$ on the ground set A_O where A_O is also a copy of $A(D_G^r)$. Finally, we define the matroid $\mathcal{M} = (A_\mathcal{M}, \mathcal{I})$ as the direct sum of $\mathcal{M}_O$ and $\mathcal{M}_1^i, \mathcal{M}_2^i$, for $i \in [\lambda]$, i.e. $\mathcal{M} = \left(\bigoplus_{i \in [\lambda]}(\mathcal{M}_1^i \oplus \mathcal{M}_2^i)\right) \oplus \mathcal{M}_O$. Note that the rank of this matroid is $3\lambda(n-1)$ and it is representable over any field of size at least $|A(D_G^r)|+1$. For an arc $a \in A(D_G^r)$, we denote its copies in A_1^i, A_2^i and A_O by a_1^i, a_2^i and a_O respectively. For a collection $\mathbb{O}$ of λ out-branchings in D_G^r, by $A(\mathbb{O})$ we denote the set of arcs which is present in some $O \in \mathbb{O}$. For $X \in \mathcal{I}$, by A_X we denote the set of arcs $a \in A(D_G^r)$ such that $X \cap \bigcup_{i=1}^{\lambda}\{a_1^i, a_2^i\} \neq \emptyset$. For $e \in E(G)$ and $i \in [\lambda]$, we let $S_e^i = \{(a_e)_1^i, (a_e)_2^i, (a'_e)_1^i, (a'_e)_2^i\}$ and $S_e = \{(a_e)_O, (a'_e)_O\} \cup \left(\bigcup_{i=1}^{\lambda} S_e^i\right)$. We define a function $\psi : \mathcal{I} \times E(G) \to \{0,1\}$, where for $W \in \mathcal{I}, e \in E(G)$, $\psi(W,e) = 1$ if and only if exactly one of the following holds. Either $W \cap S_e = \emptyset$; or, there exists $t, t' \in [\lambda], t \neq t'$, such that, (i) $(a_e)_O, (a'_e)_O \in W$, (ii) $S_e^t \cap W = \{(a_e)_1^t, (a_e)_2^t\}$, (iii) $S_e^{t'} \cap W = \{(a'_e)_1^{t'}, (a'_e)_2^{t'}\}$, and (iv) $\forall i \in [\lambda] \setminus \{t, t'\}$, $S_e^i \cap W = \emptyset$. Now for each $\ell \in [\lfloor\lambda(n-2)/2\rfloor]$, we define the following set. $\mathcal{B}^{6\ell} = \{W \mid W \in \mathcal{I}, |W| = 6\ell \textit{ and } \forall e \in E(G), \psi(W,e) = 1\}$ Observe that for every $W \in \mathcal{B}^{6\ell}$, $|\mathsf{Typ}(A_W)| = \ell$ and, $a_e \in A_W$ if and only if $a'_e \in A_W$. Therefore, any set in this collection corresponds to a potential candidate for the subset of arcs which appear in exactly two out-branchings in $\mathbb{O}$. We are now ready to state the following lemma, which relates the computation of λ out-branchings minimizing types and representative sets.

Lemma 8 (∗). *Let G be a λ-connected undirected graph on n vertices, D_G its corresponding digraph and $\ell \in [\lfloor \frac{\lambda(n-2)}{2} \rfloor]$. There exists a set $\mathbb{O}$ of out-branchings rooted at r, with $|\mathsf{Typ}(A(\mathbb{O}))| \leq \lambda(n-1) - \ell$ in D_G if and only if there exists $\widehat{T} \in \widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$, where $n' = 3\lambda(n-1)$, such that D_G has a collection $\widehat{\mathbb{O}}$ of λ out-branchings rooted at r, $A_{\widehat{T}} \subseteq A(\widehat{\mathbb{O}})$ and $|\mathsf{Typ}(\widehat{\mathbb{O}})| \leq \lambda(\mathsf{n}-1) - \ell$.*

Lemma 9 (∗). *Let G be a λ-connected undirected graph on n vertices, D_G its corresponding digraph and $\ell \in [\lfloor \lambda(n-2)/2 \rfloor]$. In time $2^{\mathcal{O}(\lambda n)}$ we can compute $\widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$ such that $|\widehat{\mathcal{B}}^{6\ell}| \leq \binom{n'}{6\ell}$. Here $n' = 3\lambda(n-1)$.*

Finally, Lemmata 5, 7, 8 and 9 give us the following theorem.

Theorem 5 (∗). *Let G be a λ edge connected graph on n vertices. Then we can find a minimum λ edge connected subgraph of G in $2^{\mathcal{O}(\lambda n)}$ time.*

As before, the above theorem can be extended to prove Theorem 2.

4 Augmentation Problems

The algorithms for Minimum Weight λ-connected Spanning Subgraph may be used to solve instances of Minimum Weight λ-connectivity Augmentation as well. Given an instance (D, L, w, λ) of the augmentation problem, we construct an instance (D', w', λ) of Minimum Weight λ-connected Spanning Subgraph, where $D' = D \cup L$ and w' is a weight function that gives a weight 0 to arcs in $A(D)$ and it is w for the arcs from L. It is easy to see that the solution returned by our algorithm contains a minimum weight augmenting set. A similar approach works for undirected graphs as well, proving Theorem 3.

References

1. Jørgen Bang-Jensen and Gregory Z Gutin. *Digraphs: theory, algorithms and applications.* Springer Science & Business Media, 2008.
2. Manu Basavaraju, Fedor V Fomin, Petr Golovach, Pranabendu Misra, MS Ramanujan, and Saket Saurabh. Parameterized algorithms to preserve connectivity. In *Automata, Languages, and Programming*, pages 800–811. Springer, 2014.
3. Richard Bellman. Dynamic programming treatment of the travelling salesman problem. *Journal of the ACM (JACM)*, 9(1):61–63, 1962.
4. Piotr Berman, Bhaskar Dasgupta, and Marek Karpinski. Approximating transitive reductions for directed networks. In *Proceedings of the 11th International Symposium on Algorithms and Data Structures*, pages 74–85. Springer-Verlag, 2009.
5. Andreas Bjorklund. Determinant sums for undirected hamiltonicity. *SIAM Journal on Computing*, 43(1):280–299, 2014.

6. Joseph Cheriyan and Ramakrishna Thurimella. Approximating minimum-size k-connected spanning subgraphs via matching. *SIAM Journal on Computing*, 30(2):528–560, 2000.
7. Marek Cygan, Fedor V. Fomin, Alexander Golovnev, Alexander S. Kulikov, Ivan Mihajlin, Jakub Pachocki, and Arkadiusz Socala. Tight bounds for graph homomorphism and subgraph isomorphism. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1643–1649, 2016.
8. Marek Cygan, Stefan Kratsch, and Jesper Nederlof. Fast hamiltonicity checking via bases of perfect matchings. In *Proceedings of the forty-fifth annual ACM Symposium on Theory of Computing*, pages 301–310. ACM, 2013.
9. Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient computation of representative families with applications in parameterized and exact algorithms. *J. ACM*, 63(4):29:1–29:60, September 2016.
10. András Frank. Augmenting graphs to meet edge-connectivity requirements. *SIAM Journal on Discrete Mathematics*, 5(1):25–53, 1992.
11. H Frank and W Chou. Connectivity considerations in the design of survivable networks. *Circuit Theory, IEEE Transactions on*, 17(4):486–490, 1970.
12. Harold N Gabow. A matroid approach to finding edge connectivity and packing arborescences. *Journal of Computer and System Sciences*, 50(2):259–273, 1995.
13. Dan Gusfield. A graph theoretic approach to statistical data security. *SIAM Journal on Computing*, 17(3):552–571, 1988.
14. Michael Held and Richard M Karp. A dynamic programming approach to sequencing problems. *Journal of the Society for Industrial and Applied Mathematics*, 10(1):196–210, 1962.
15. SP Jain and Krishna Gopal. On network augmentation. *Reliability, IEEE Transactions on*, 35(5):541–543, 1986.
16. Ming-Yang Kao. Data security equals graph connectivity. *SIAM Journal on Discrete Mathematics*, 9(1):87–100, 1996.
17. Samir Khuller. Approximation algorithms for finding highly connected subgraphs. *Vertex*, 2:2, 1997.
18. Samir Khuller and Uzi Vishkin. Biconnectivity approximations and graph carvings. *Journal of the ACM (JACM)*, 41(2):214–235, 1994.
19. Guy Kortsarz and Zeev Nutov. Approximating minimum cost connectivity problems. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2010.
20. Dániel Marx. A parameterized view on matroid optimization problems. *Theor. Comput. Sci.*, 410(44):4471–4479, 2009.
21. Dániel Marx and László A Végh. Fixed-parameter algorithms for minimum-cost edge-connectivity augmentation. *ACM Transactions on Algorithms (TALG)*, 11(4):27, 2015.
22. Dennis M Moyles and Gerald L Thompson. An algorithm for finding a minimum equivalent graph of a digraph. *Journal of the ACM (JACM)*, 16(3):455–460, 1969.
23. James G Oxley. *Matroid theory*, volume 3. Oxford University Press, USA, 2006.
24. Alexander Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer Science & Business Media, 2003.
25. Toshimasa Watanabe, Takanori Narita, and Akira Nakamura. 3-edge-connectivity augmentation problems. In *Circuits and Systems, 1989., IEEE International Symposium on*, pages 335–338. IEEE, 1989.
26. Virginia Vassilevska Williams. Multiplying matrices faster than coppersmith-winograd. In *ACM symposium on Theory of computing*, pages 887–898, 2012.

5 Appendix A: Preliminaries

We denote the set of natural numbers by $\mathbb{N}$. For $n \in \mathbb{N}$, by $[n]$ we denote the set $\{1, \dots, n\}$. We use the term *universe* to distinguish a set from its subsets. For any two subsets X and Y of a universe U, we use $X \setminus Y$ to denote the subset of X whose elements are not present in Y. For any set U define $\binom{U}{i} = \{X \mid X \subseteq U,\ |X| = i\}$. We say that a family $\mathcal{S} = \{S_1, \dots, S_t\}$ of subsets of a universe U is a *p-family* if each set in $\mathcal{S}$ has cardinality at most p. For two families $\mathcal{S}_1$ and $\mathcal{S}_2$ of a universe U, define $\mathcal{S}_1 \bullet \mathcal{S}_2 = \{S_i \cup S_j \mid S_i \in \mathcal{S}_1, S_j \in \mathcal{S}_2 \textit{ and } S_i \cap S_j = \emptyset\}$. We will use ω to denote the exponent in the running time of matrix multiplication, the current best known bound for which $\omega < 2.373$ [26].

Graphs. We define various terms with respect to undirected graphs. Many of these terms are analogously defined for digraphs, and we only mention those that are defined differently. For a graph G (or digraph D) we use $V(G)$ ($V(D)$) and $E(G)$ ($A(D)$) to denote the vertex set and the edge set (arc set) respectively. In an undirected graph, the *neighbourhood* of a vertex $v \in V(G)$ is defined as the set $N(v) = \{u \in V(G) \mid (u, v) \in E(G)\}$. In a directed graph D, we define $\mathsf{Out}_D(v) = \{(v, w) \in A(D)\}$ and $\mathsf{In}_D(v) = \{(u, v) \in A(D)\}$ to be the set of *out-edges* and *in-edges* of v respectively. The *underlying graph* of a digraph D is the undirected graph obtained from D by removing the direction of every edge. Note that this graph may not be a simple graph. A *path* P is a graph with vertex set $V(P) = \{v_1, v_2, \dots, v_\ell\}$ and edge set $E(P) = \{(v_i, v_{i+1}) \mid 1 \leq i \leq \ell - 1\}$ for some $\ell \in \mathbb{N}$. A *cycle* is a graph (or digraph) with vertex set $V(P) = \{v_1, v_2, \dots, v_\ell\}$ and edge set $E(P) = \{(v_i, v_{i+1}) \mid 1 \leq i \leq \ell - 1\} \cup \{(v_\ell, v_1)\}$. An *acyclic graph* (or a digraph), as the name implies, contains no cycles. A connected acyclic graph is called a *tree*. An undirected acyclic graph which is union of trees is called as a *forest*. An acyclic digraph is called DAG, which is short for "directed acyclic graph". For $r \in V(D)$, an *in-branching* rooted at r is a DAG whose underlying graph is a tree rooted at r and the out-degree of every vertex is 1, except r whose out-degree in 0. Observe that there is a path from any vertex to r in the in-branching. We similarly define *out-branching* rooted at r, where every vertex has in-degree 1, except r whose in-degree is 0. A graph is called *connected* if there is a path between every pair of vertices. Similarly, a digraph is called *strongly connected* if for every ordered pair of vertices, (u, v), there is a path from u to v. Let λ be a natural number. A graph G is called λ-*edge connected* if for all $u, v \in V(G)$, there is a collection of λ edge disjoint paths with u and v as their endpoints. A digraph is called λ-edge connected if for every ordered pair (u, v) of vertices, there is a collection of λ arc disjoint paths from u to v. An *edge cut* in a graph G is a partition $(X, \overline{X})$ of $V(G)$ and, $\delta_G(X)$ denotes edges of G with one endpoint in X and the other in $\overline{X}$. When the graph is clear from context, we simply write $\delta(X)$. Note that in digraphs, $(X, \overline{X})$ and $(\overline{X}, X)$ denote different cuts, and $\delta(X)$ denotes those directed edges with their tail in X and head in $\overline{X}$. In this paper we are mainly concerned with the edge connectivity of graphs and digraphs, we use the terms "cut" and "λ-connected graph" to mean an edge cut and a λ-edge connected graph.

Matroids and Representative Families. In the following we state some of basic definitions related to matroids. We refer the reader to [23] for more details. We also state the definition of representative families and give a result regarding its computation.

Definition 1 (Matroid). *A pair $\mathcal{M} = (E, \mathcal{I})$, where E is a set called ground set and $\mathcal{I}$ is a family of subsets of E, called independent sets, is called a matroid if it satisfies the following properties: (i) $\emptyset \in \mathcal{I}$, (ii) If $A \in \mathcal{I}$ and $A' \subseteq A$ then $A' \in \mathcal{I}$, and (iii) If $A, B \in \mathcal{I}$ and $|A| < |B|$ then there is $x \in B \setminus A$ such that $A \cup \{x\} \in \mathcal{I}$.*

An inclusion-wise maximal set in $\mathcal{I}$ is called as a basis of $\mathcal{M}$. All the bases of a matroid are of same size. The size of a basis is called as the rank of the matroid. Linear matroids are an important subclass of matroids that can be defined using linear independence of vectors over some field. Let A be a matrix over a field $\mathbb{F}$ and E be the set of its columns. We define a matroid $\mathcal{M} = (E, \mathcal{I})$ as follows. A subset $X \subseteq E$ is called an independent set if the set of columns in X are linearly independent in $\mathbb{F}$. The matrix A is called a representation of $\mathcal{M}$.

Definition 2. *Let $\mathcal{M}_1 = (E_1, \mathcal{I}_1), \mathcal{M}_2 = (E_2, \mathcal{I}_2), \ldots, \mathcal{M}_t = (E_t, \mathcal{I}_t)$ be t matroids such that $E_i \cap E_j = \emptyset$, for $i \neq j$ and $i, j \in [t]$. The direct sum $\mathcal{M}_1 \oplus \mathcal{M}_2 \oplus \cdots \oplus \mathcal{M}_t$ of $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_t$ is a matroid $\mathcal{M} = (\biguplus_{i \in [t]} E_i, \mathcal{I})$, where $\mathcal{I}$ is defined as follows. For $X \subseteq E$, $X \in \mathcal{I}$ if and only if $X \cap E_i \in \mathcal{I}_i$ for all $i \in [t]$.*

The following easy proposition is used in later sections.

Proposition 1. *Let $\mathcal{M}_1 = (E_1, \mathcal{I}_1), \mathcal{M}_2 = (E_2, \mathcal{I}_2), \ldots, \mathcal{M}_t = (E_t, \mathcal{I}_t)$ be t matroids such that $E_i \cap E_j = \emptyset$, for $i \neq j$ and $i, j \in [t]$. If a set B is a basis in $\mathcal{M}_1 \oplus \mathcal{M}_2 \oplus \cdots \oplus \mathcal{M}_t$, then $B \cap E_i$ is a basis in $\mathcal{M}_i$ for all $i \in [t]$.*

Proposition 2 (Proposition 3.4 [20], [23]). *Given representations of matroids $\mathcal{M}_1, \mathcal{M}_2, \ldots \mathcal{M}_t$ over a field $\mathbb{F}$ then a representation of their direct sum $\mathcal{M}_1 \oplus \mathcal{M}_2 \oplus \cdots \oplus \mathcal{M}_t$ over $\mathbb{F}$ can be found in polynomial time.*

Definition 3. *Given an undirected graph G, the graphic matroid of G, denoted by $\mathcal{M}_G$ has ground set $E(G)$ and its independent sets are defined as follows. For any $X \subseteq E(G)$, X is an independent set in $\mathcal{M}$ if and only if the graph $(V(G), X)$ is a forest.*

Graphics matroids are representable over any field of size at least 2 [23]. For a directed graph D, we define the graphic matroid with respect to the underlying undirected graph of D. The universe is the arc set $A(D)$, and $X \subseteq A(D)$ is an independent set if and only if they form a forest in the underlying undirected graph.

Definition 4. *A pair $\mathcal{M} = (E, \mathcal{I})$, over n element universe E is a uniform matroid if $\mathcal{I} = \{X \mid X \subseteq E, |X| \leq k\}$, where k is some constant. It is denoted by $U_{n,k}$.*

The uniform matroid $U_{n,k}$ is representable over all fields with at least $n+1$ elements [23]. For the special case of $k=1$, it is representable over all fields.

Definition 5. *A pair $\mathcal{M}=(E,\mathcal{I})$ is a partition matroid if the ground set E is partitioned into t sets $E_1, E_2, \ldots E_t$ for some $t \in \mathbb{N}$, and there are t integers $k_1, k_2, \ldots k_t$ such that $X \subseteq E$ is an independent set in $\mathcal{M}$ if and only if $|X \cap E_i| \leq k_i$, for all $i \in [t]$.*

Observe that the partition matroid is the direct sum of a collection of uniform matroids and hence it is also has a representation over any field where each of those uniform matroids are representable. In the special case, when all the partition sizes are one, then the partition matroid can be represented over any field. Finally, we define representative families and give a result on computing such families over a linear matroid.

Definition 6 (Min/Max q-Representative Family [20]). *Given a matroid $\mathcal{M}=(E,\mathcal{I})$, a p-family $\mathcal{B}$ of E and a non-negative weight function $w:\mathcal{B}\to\mathbb{N}$. We say that $\widehat{\mathcal{B}} \subseteq \mathcal{B}$ is a min (max) q-representative for $\mathcal{B}$ if for every set $Y \subseteq E$ of size at most q, if there is a set $X \in \mathcal{B}$, such that $X \cap Y = \emptyset$ and $X \cup Y \in \mathcal{I}$, then there is a set $\widehat{X} \in \widehat{\mathcal{B}}$ such that $\widehat{X} \cap Y = \emptyset$, $\widehat{X} \cup Y \in \mathcal{I}$ and $w(\widehat{X}) \leq w(X)$ ($w(\widehat{X}) \geq w(X)$). If $\widehat{\mathcal{B}} \subseteq \mathcal{B}$ is a min (max) q-representative for $\mathcal{B}$ then we denote it by $\widehat{\mathcal{B}} \subseteq^q_{minrep} \mathcal{B}$ ($\widehat{\mathcal{B}} \subseteq^q_{maxrep} \mathcal{B}$).*

We drop the 'max'/'min' from the subscript in the above notation if we are not concerned with weights.

Theorem 6 ([9]). *Let $\mathcal{M}=(E,\mathcal{I})$ be a linear matroid of rank $k=p+q$, and matrix $A_{\mathcal{M}}$ be a representation of $\mathcal{M}$ over a field $\mathbb{F}$. Also, let $\mathcal{B}=\{B_1, B_2, \ldots, B_t\}$ be a p-family of independent sets in E and $w:\mathcal{B}\to\mathbb{N}$ be a non-negative weight function. Then, there exists $\widehat{\mathcal{B}} \subseteq^q_{minrep} \mathcal{B}$ ($\widehat{\mathcal{B}} \subseteq^q_{maxrep} \mathcal{B}$) of size at most $\binom{p+q}{p}$. Moreover, $\widehat{\mathcal{B}} \subseteq^q_{minrep} \mathcal{B}$ ($\widehat{\mathcal{B}} \subseteq^q_{maxrep} \mathcal{B}$) can be computed in at most $\mathcal{O}(\binom{p+q}{p} tp^\omega + t\binom{p+q}{p}^{\omega-1})$ operations over $\mathbb{F}$.*

The following lemma, which is used in proving the correctness of our algorithms, follows from the definition of representative sets.

Lemma 10 (Lemma 3.3 [9]). *Let $\mathcal{M}=(E,\mathcal{I})$ be a matroid and $\mathcal{B}$ be a family of subsets of E. If $\mathcal{B}' \subseteq^q_{rep} \mathcal{B}$ and $\widehat{\mathcal{B}} \subseteq^q_{rep} \mathcal{B}'$ then $\widehat{\mathcal{B}} \subseteq^q_{rep} \mathcal{B}$.*

Appendix B: Missing Proofs

Proof of Lemma 1. Fix an arbitrary vertex $r \in V(D)$. In the forward direction, by Edmond's disjoint out-branching theorem [24, Corollary 53.1b], there is a collection of λ arc disjoint out-branching rooted at a vertex r. To find the collection of in-branchings, we consider the graph D' which is the "reversed digraph" of D, i.e. $V(D')=V(D)$ and the arc set $A(D')=\{(v,u)|(u,v)\in A(D)\}$. Note that the digraph D' is also λ-connected, and therefore there is a collection of λ arc disjoint out-branchings $O'_1, O'_2, \ldots, O'_\lambda$ rooted at r in D'. From this collection, we construct the λ arc disjoint in-branchings in D as follows. For each $j \in [\lambda]$, let $I_j = \{(u,v) | (v,u) \in A(O'_j)\}$ and note that this is an in-branching rooted at r. This gives the required collection of in-branchings.

The reverse direction is straightforward, e.g. it follows from [24, Corollary 53.1b and Corollary 53.1d]. □

Proof of Claim 1 in Lemma 3. Let us consider the first statement. Recall that $\widehat{Z} = (Z_{D'} \setminus T') \cup \widehat{T}$. As discussed earlier, for any $e \in A_D$ and $i \in [\lambda]$, both $Z_{D'}$ and T' contain both the copy in $\{e^i_{I,1}, e^i_{I,2}\}$, or none of the copies from $\{e^i_{I,1}, e^i_{I,2}\}$. Now $\widehat{T} \in \mathcal{B}^{6\ell}$ also satisfies this condition for every $e \in A(D)$. Hence $\widehat{Z}$ satisfies this condition as well. Now let us consider the second statement. By construction, $Z_{D'}$ contains e_I, e_O for every arc $e \in A(\mathbb{I}) \cup A(\mathbb{O})$. Similarly, T' contains e_I, e_O for every arc $e \in T$. Finally, $\widehat{T}$ contain both e_I, e_O for any arc $e \in A(D)$ if and only if it contains $e^j_{I,1}, e^j_{I,2}$ for some $j \in [\lambda]$. Hence, for any arc e, if $e^j_{I,1} \in \widehat{Z}$ for some $j \in [\lambda]$, then we have $e_I \in \widehat{Z}$ as well. We can similarly show the other two statements. □

Proof of Claim 2 in Lemma 3. Again let us consider the first statement and suppose that it is not true. So there are distinct $i, j \in [\lambda]$ such that, $\{e^i_{I,1}, e^i_{I,2}\} \cap \widehat{Z} \neq \emptyset$ and $\{e^j_{I,1}, e^j_{I,2}\} \cap \widehat{Z} \neq \emptyset$ for an arc $e \in A(D)$. Initially, for any arc f in the collection of arc disjoint in-branchings $\mathbb{I}$, there is exactly one $k \in [\lambda]$ such that $f^k_{I,1}, f^k_{I,2} \in Z_{D'}$ and for any other $k' \in [\lambda]$, $f^{k'}_{I,1}, f^{k'}_{I,2} \notin Z_{D'}$. Further, we have $f_I \in Z_{D'}$. And for any arc not in $A(\mathbb{I})$, no copies of this arc from E_I and $E^k_{I,1}, E^k_{I,2}$ for all $k \in [\lambda]$, is present in $Z_{D'}$. Such a statement also holds true for T' and $\widehat{T}$ as well, as they are both in $\mathcal{B}^{6\ell}$. As $\widehat{Z} = (Z_{D'} \setminus T') \cup \widehat{T}$, it must be the case that $\{e^i_{I,1}, e^i_{I,2}\} \cap (Z_{D'} \setminus T') \neq \emptyset$ and $\{e^j_{I,1}, e^j_{I,2}\} \cap \widehat{T} \neq \emptyset$. But then, $Z_{D'} \setminus T'$ and $\widehat{T}$ have the element e_I in common. This contradicts the fact that $\widehat{T}$ is a representative of T' in $\widehat{\mathcal{B}}^{6\ell}$. Hence, no such pair i, j exists. We can show the second statement in a similar way. □

Proof of Claim 1 in Lemma 4. Let $X \in \widehat{\mathcal{B}}^{6(i+1)}$ and Y be a set of size $n' - 6(i+1)$ such that $X \cup Y \in \mathcal{I}$ and $X \cap Y = \emptyset$. We will prove that there exists some $\widehat{X} \in \mathcal{F}^{6(i+1)}$ such that $\widehat{X} \cup Y \in \mathcal{I}$ and $\widehat{X} \cap Y = \emptyset$. This will prove the claim.

Let $e \in A(D)$ and $i, j \in [\lambda]$ such that $\{e_I, e_O, e^i_{I,1}, e^i_{I,2}, e^j_{O,1}, e^j_{O,2}\} \subseteq X$. Let $X' = X \setminus \{e_I, e_O, e^i_{I,1}, e^i_{I,2}, e^j_{O,1}, e^j_{O,2}\}$ and $Y' = Y \cup \{e_I, e_O, e^i_{I,1}, e^i_{I,2}, e^j_{O,1}, e^j_{O,2}\}$. Note that $X' \in \mathcal{I}$ and $Y' \in \mathcal{I}$, since $X \cup Y \in \mathcal{I}$. But, $X' \in \mathcal{B}^{6i}$, $X' \cup Y' \in \mathcal{I}$ and $|Y'| = n' - 6i$. This implies that there exists $\widehat{X}' \in \widehat{\mathcal{B}}^{6i}$ such that, $\widehat{X}' \cap Y' = \emptyset$ and $\widehat{X}' \cup Y' \in \mathcal{I}$. Therefore, $\widehat{X}' \cup \{e_I, e_O, e^i_{I,1}, e^i_{I,2}, e^j_{O,1}, e^j_{O,2}\} \in \mathcal{I}$ and also $\widehat{X}' \cup \{e_I, e_O, e^i_{I,1}, e^i_{I,2}, e^j_{O,1}, e^j_{O,2}\} \in (\widehat{\mathcal{B}}^{6i} \bullet \mathcal{W})$. This proves the claim. □

Proof of Lemma 5. We define a new weight function w' which gives a weight 0 to any arc which is contained in X and it is same as w for all other arcs. We now apply an algorithm [24, Theorem 53.10] with the weight function w', which returns a minimum weight collection $\mathbb{O}$ of λ arc disjoint out-branchings rooted at r. If $X \subseteq A(\mathbb{O})$, then we return $\mathbb{O}$ as the required solution. Otherwise no such collection exists. □

Proof of Lemma 8. In the forward direction, let $\mathbb{O}$ be a collection of λ out-branchings rooted at r in D_G, such that $|\mathsf{Typ}(A(\mathbb{O}))| \leq \lambda(n-1) - \ell$. For each $i \in [\lambda]$, let O_1^i and O_2^i be the independent sets in $\mathcal{M}_1^i$ and $\mathcal{M}_2^i$ respectively, corresponding to the out-branching $O_i \in \mathbb{O}$. Now consider the set $I = \{a_O, a_1^i, a_2^i \mid a \in A(O_i), i \in [\lambda]\}$ in the matroid $\mathcal{M}$. For each $i \in [\lambda]$, $\{a_1^i \mid a \in A(O_i)\}$ is an independent set in the graphic matroid $\mathcal{M}_1^i$ since O_i corresponds to a tree in the underlying graph. And similarly $\{a_2^i \mid a \in A(O_i)\}$ is an independent set in the partition matroid $\mathcal{M}_2^i$, since O_i is an out-branching. Finally $\{a_O \mid a \in A(\mathbb{O})\}$ is of cardinality $\lambda(n-1)$ and therefore a basis in $\mathcal{M}_O$. Therefore, I is an independent set in the matroid $\mathcal{M}$. Notice that $|I| = 3\lambda(n-1)$, which is equal to the rank of $\mathcal{M}$. Hence, I is a basis in $\mathcal{M}$. Let $P' = \{e \in E(G) \mid a_e, a'_e \in A(\mathbb{O})\}$ and clearly, $|P'| \geq \ell$. Fix an arbitrary subset P of P' with exactly ℓ edges. Let $T = \{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^i, (a'_e)_2^i \mid e \in P, i \in [\lambda]\}$. Observe that $T \in \mathcal{B}^{6\ell}$, and therefore there is a $\widehat{T} \in \widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$ such that $\widehat{I} = (I \setminus T) \cup \widehat{T}$ is an independent set in $\mathcal{M}$ of size $3\lambda(n-1)$, and note that $|\mathsf{Typ}(A_{\widehat{T}})| = \ell$.

We will now prove that there is a collection $\widehat{\mathbb{O}}$ of λ arc disjoint out-branchings, such that $A_{\widehat{T}} \subseteq A(\widehat{\mathbb{O}})$ and $|\mathsf{Typ}(\widehat{\mathbb{O}})| \leq \lambda(n-1) - \ell$.

Claim. 1 *For any arc $a \in A(D_G^r)$ and $i \in [\lambda]$, either $a_1^i, a_2^i \in \widehat{I}$ or $a_1^i, a_2^i \notin \widehat{I}$. Further, if $a_1^i \in \widehat{I}$ for some $i \in [\lambda]$, then $a_O \in \widehat{I}$ as well.*

Proof. Let us consider the first statement. Initially, the statement hold for I by construction. And since $T, \widehat{T} \in \mathcal{B}^{6\ell}$, the statement holds for them as well. This implies that for $\widehat{I} = (I \setminus T) \cup \widehat{T}$ also satisfies this statement. Now we consider the second statement. Initially for any arc $a \in A(\mathbb{O})$ we have $a_O, a_1^i \in I$. And T contains a_O for some arc a if and only if it also contains a_1^i for some $i \in [\lambda]$, and a similar statement holds for $\widehat{T}$. Hence, the second statement also holds for $\widehat{I}$. □

Claim. 2 *For any arc $a \in D_G^r$, and any pair of $i \neq j \in [\lambda]$, either $a_1^i, a_2^i \notin \widehat{I}$ or $a_1^j, a_2^j \notin \widehat{I}$.*

Proof. Suppose that there were some $i \neq j, i, j \in [\lambda]$ and an arc $a \in A(D_G^r)$ such that $a_1^i, a_2^i, a_1^j, a_2^j \in \widehat{I}$. Initially, for any arc b in the collection of arc disjoint out-branchings $\mathbb{O}$, there is exactly one $k \in [\lambda]$ such that $b_1^k, b_2^k \in I$ and for any other $k' \in [\lambda]$, $b_1^{k'}, b_2^{k'} \notin I$. Further, we have $b_O \in I$. And for any arc not in $A(\mathbb{O})$, no copies of this arc from E_O and E_1^k, E_2^k for all $k \in [\lambda]$, is present in I. Such a statement also holds true for T' and $\widehat{T}$ as well, as they are both in $\mathcal{B}^{6\ell}$. As $\widehat{I} = (I \setminus T') \cup \widehat{T}$, it must be the case that $\{a_1^i, a_2^i\} \cap (I \setminus T') \neq \emptyset$ and $\{a_1^j, a_2^j\} \cap \widehat{T} \neq \emptyset$. But then, $I \setminus T'$ and $\widehat{T}$ have the element a_O in common, which contradicts the fact that $\widehat{T}$ is a representative of T, in $\mathcal{B}^{6\ell}$. □

Now since $\widehat{I}$ is a basis of $\mathcal{M}$, we have that $X_j^i = \widehat{I} \cap A_j^i$ is a basis of $\mathcal{M}_j^i$, where $i \in [\lambda]$ and $j \in [2]$. Now by Claim 1, $A_{X_1^i} = A_{X_2^i}$, and hence $\widehat{O}_i = A_{X_1^i}$ is an out-branching rooted at r in D_G. Next, by Claim 2, the collection $\widehat{\mathbb{O}} = \{\widehat{O}_i \mid i \in [\lambda]\}$ is pairwise arc disjoint. And finally we bound the value of $|\mathsf{Typ}(\widehat{\mathbb{O}})|$. Initially $|\mathsf{Typ}(A_I)| = \lambda(n-1) - \ell$, and $|\mathsf{Typ}(A_T)| = |\mathsf{Typ}(A_{\widehat{T}})| = \ell$. Since $T \in \mathcal{B}^{6\ell}$, for any edge e of G, either both or neither of a_e, a'_e lie in T, and a similar statement holds for $\widehat{T}$ as well. Therefore we have $|\mathsf{Typ}(A_{I \setminus T})| = \lambda(n-1) - 2\ell$, and hence $|\mathsf{Typ}(A_{\widehat{I}})| = |\mathsf{Typ}(\widehat{\mathbb{O}})| = \lambda(n-1) - \ell$.

The reverse direction follows from Lemma 7. $\square$

Proof of Lemma 9. We give a dynamic programming based algorithm. Let $\mathcal{D}$ be an array of size $\ell + 1$. For $i \in [\ell + 1]$ the entry $\mathcal{D}[i]$ will store the family $\widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'} \mathcal{B}^{6\ell}$. We will fill the entries in array $\mathcal{D}$ according to the increasing order of index i, where $i \in \{0, 1, \ldots \ell\}$. For $i = 0$, we have $\widehat{B}^0 = \{\emptyset\}$. Let $\mathcal{W} = \big\{\{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^j, (a'_e)_2^j\} \mid i, j \in [\lambda],\ i \neq j,\ \textit{and } e \in E(G) \textit{ is not incident on } r\big\}$, and note that $|\mathcal{W}| \leq \binom{\lambda}{2}m$. Given that we have filled all the entries $\mathcal{D}[i']$ for every $i' < i+1$, we fill the entry at $\mathcal{D}[i+1]$ as described below.

Let $\mathcal{F}^{6(i+1)} = (\widehat{\mathcal{B}}^{6i} \bullet \mathcal{W}) \cap \mathcal{I}$. Observe that for any $X \in \mathcal{F}^{6(i+1)}$, $|\mathsf{Typ}(A_X)| = i+1$. We now have the following claim.

Claim. 3 $\mathcal{F}^{6(i+1)} \subseteq_{rep}^{n'-6(i+1)} \mathcal{B}^{6(i+1)}$*, for all* $i \in \{0, 1, \ldots \ell - 1\}$

Proof. Let $X \in \mathcal{B}^{6(i+1)}$ and Y be a set of size $n' - 6(i+1)$ such that $X \cup Y \in \mathcal{I}$ and $X \cap Y = \emptyset$. We will show that there exists some $X' \in \mathcal{F}^{6(i+1)}$ such that $X' \cup Y \in \mathcal{I}$ and $X' \cap Y = \emptyset$. This will prove the claim.

Let $e \in E(G)$ and $i \neq j \in [\lambda]$ such that $\{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^j, (a'_e)_2^j\} \subseteq X$. Let $X' = X \setminus \{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^j, (a'_e)_2^j\}$ and $Y' = Y \cup \{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^j, (a'_e)_2^j\}$. Note that $X' \in \mathcal{I}$ and $Y' \in \mathcal{I}$, since $X \cup Y \in \mathcal{I}$. But $X' \in \mathcal{B}^{6i}$, $X' \cup Y' \in \mathcal{I}$ and $|Y'| = n' - 6i$, which implies that there exists $\widehat{X}' \in \widehat{\mathcal{B}}^{6i}$ such that, $\widehat{X}' \cup Y' \in \mathcal{I}$. Therefore, $\widehat{X}' \cup \{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^j, (a'_e)_2^j\} \in \mathcal{I}$ and observe that $\widehat{X}' \cup \{(a_e)_O, (a'_e)_O, (a_e)_1^i, (a_e)_2^i, (a'_e)_1^j, (a'_e)_2^j\} \in (\widehat{\mathcal{B}}^{6i} \bullet \mathcal{W})$. $\square$

We fill the entry for $\mathcal{D}[i+1]$ as the following. By Theorem 6 we know that $|\widehat{\mathcal{B}}^{6i}| \leq \binom{n'}{6i}$, and hence it follows that $|\mathcal{F}^{6(i+1)}| \leq \binom{\lambda}{2}m\binom{n'}{6i}$. Moreover, we can compute $\mathcal{F}^{6(i+1)}$ in time $\mathcal{O}(\binom{\lambda}{2}mn\binom{n'}{6i})$. We use Theorem 6 to compute $\widehat{\mathcal{F}}^{6(i+1)} \subseteq_{rep}^{n'-6(i+1)} \mathcal{F}^{6(i+1)}$ of size at most $\binom{n'}{6(i+1)}$ in time $\mathcal{O}(\binom{n'}{6(i+1)}ti^{\omega} + t\binom{n'}{6(i+1)}^{\omega-1})$, where $t = |\mathcal{F}^{6(i+1)}|$. We know from Claim 1 that $\mathcal{F}^{6(i+1)} \subseteq_{rep}^{n'-6(i+1)} \mathcal{B}^{6(i+1)}$, and therefore $\widehat{\mathcal{B}}^{6(i+1)} = \widehat{\mathcal{F}}^{6(i+1)} \subseteq_{rep}^{n'-6(i+1)} \mathcal{B}^{6(i+1)}$. And finally, we assign the family $\widehat{\mathcal{B}}^{6(i+1)}$ to $\mathcal{D}[i+1]$. This completes the description of the algorithm and its correctness. Using the fact that $\ell < n'/6$, the total running time of the algorithm can bounded as follows.

$$\mathcal{O}\Big(\sum_{i=1}^{\ell}\left(\binom{n'}{6(i+1)}ti^{\omega}+t\binom{n'}{6(i+1)}^{\omega-1}\right)\lambda^2 m\binom{n'}{6i}\Big)\leq 2^{\mathcal{O}(\lambda n)}$$

This completes the proof of this lemma. □

Proof of Lemma 6. In the forward direction, observe that since G is λ-connected, D_G is also λ-connected. By Lemma 1, for any $r \in V(D_G)$, there are λ arc disjoint out-branchings rooted at r in D_G.

In the reverse direction, suppose there are λ arc disjoint out-branchings rooted at r in D_G. Therefore for any vertex $v \in V(G)$, there is a collection of λ arc disjoint paths from r to v in D_G. If G is not λ connected, then there is a cut $(X, \overline{X})$ such that $|\delta_G(X)| \leq \lambda - 1$, and we may assume that $r \in X$. But then in D_G, there are at most $\lambda - 1$ arcs which go from X to $\overline{X}$. Therefore, for any vertex $v \in \overline{X}$, there are at most $\lambda - 1$ arc disjoint paths from r to v in D_G. This is a contradiction. This completes the proof of this lemma. □

Proof of Lemma 7. In the forward direction let G' be a λ connected subgraph of G with at most $\lambda(n-1) - \ell$ edges. By Lemma 6, for any $r \in V(D_{G'})$, $D_{G'}$ has λ arc disjoint out-branchings rooted at r. Observe that there are at most $\lambda(n-1) - \ell$ edges in G', therefore the number of different types of edges possible in $D_{G'}$ is at most $\lambda(n-1) - \ell$.

In the reverse direction, consider a vertex $r \in V(D)$ with λ arc disjoint out-branchings $\mathbb{O} = \{O_1, O_2, \ldots, O_\lambda\}$ such that $|\mathsf{Typ}(A(\mathbb{O}))| \leq \lambda(n-1) - \ell$. Consider the graph $G' = (V(D), \mathsf{Typ}(A(\mathbb{O})))$. Observe that G' has at most $\lambda(n-1) - \ell$ edges and is a λ-connected subgraph of G (from Lemma 6). This concludes the proof. □

Proof of Theorem 5. Let $n' = 3\lambda(n-1)$. By Lemma 7 we know that finding a minimum λ-connected spanning subgraph G' of G is equivalent to finding a collection $\mathbb{O}$ of λ arc disjoint out-branchings in D_G rooted at a fixed vertex $r \in V(D)$ such that $\mathsf{Typ}(\mathbb{O})$ is minimized. For each choice of $\ell \in [\lfloor \frac{\lambda(n-2)}{2} \rfloor]$, by Lemma 6 and Lemma 7, we know that there exists a λ-connected spanning subgraph G' of G with at most $\lambda(n-1) - \ell$ arcs if and only if there exists $\widehat{T} \in \widehat{\mathcal{B}}^{6\ell} \subseteq_{rep}^{n'-6\ell} \mathcal{B}^{6\ell}$, such that D_G has a collection $\mathbb{O} = \{O_1, O_2, \ldots, O_\lambda\}$ of out-branchings rooted at r and $A_{\widehat{T}} \subseteq A(\mathbb{O})$. So we apply Lemma 9 to compute $\widehat{\mathcal{B}}^{6\ell}$ in time $2^{\mathcal{O}(\lambda n)}$, and for every $F \in \widehat{\mathcal{B}}^{6\ell}$ we check if A_F can be extended to λ out-branchings rooted at r in D_G by using Lemma 5. We return the graph G' with the least number of edges, among all the graphs computed above, as our solution. Since $\ell \leq \lambda(n-2)/2$, the running time of the algorithm is bounded by $2^{\mathcal{O}(\lambda n)}$. This completes the proof. □