Directed multicut is $W[1]$-hard, even for four terminal pairs[*]

Marcin Pilipczuk[†]   Magnus Wahlström[‡]

Abstract

We prove that Multicut in directed graphs, parameterized by the size of the cutset, is $W[1]$-hard and hence unlikely to be fixed-parameter tractable even if restricted to instances with only four terminal pairs. This negative result almost completely resolves one of the central open problems in the area of parameterized complexity of graph separation problems, posted originally by Marx and Razgon [SIAM J. Comput. 43(2):355–388 (2014)], leaving only the case of three terminal pairs open. The case of two terminal pairs was shown to be FPT by Chitnis et al. [SIAM J. Comput. 42(4):1674–1696 (2013)].

Our gadget methodology also allows us to prove $W[1]$-hardness of the Steiner Orientation problem parameterized by the number of terminal pairs, resolving an open problem of Cygan, Kortsarz, and Nutov [SIAM J. Discrete Math. 27(3):1503-1513 (2013)].

1 Introduction

The study of cuts and flows is one of the central areas of combinatorial optimization, with decades of intensive research on polynomial-time exact and approximate algorithms. Since the seminal work of Marx [23], it has also become one of the most dynamic research directions in parameterized complexity, under the name of *graph separation problems.*

The key contribution of Marx' paper [23] lies in the notion of an important separator, which is a kind of greedy cut that is useful in some graph cut problems. Marx showed that the number of important separators of size up to k in a graph is bounded as a function of k alone, and gave applications to a number of graph separation problems, in particular an FPT algorithm for Multiway cut [23].[1] (An improved bound was given by Chen et al. [4], with a simple but influential proof method.) These techniques also led to FPT algorithms for Directed Feedback Vertex Set [5] and eventually for Almost 2-SAT [30], which is a powerful generic problem encapsulating a number of other well-studied problems, such as Odd Cycle Transversal and Vertex Cover parameterized by the excess above a maximum matching. Further study of the graph separation problems in the realm of parameterized complexity revealed a plethora of other algorithmic techniques: the technique of shadow removal [28] lead to an FPT algorithm for the directed version of Multiway Cut [8], the framework of randomized contractions [6] lead to an FPT algorithm for Minimum Bisection [11], whereas the idea to guide a branching algorithm by the solution to the LP relaxation [12] lead not only to the fastest known algorithm for Odd Cycle Transversal [22], but also can be cast into a very general CSP framework [31].

In tandem with these developments, we have gained a growing understanding of the structure of bounded-size cuts in graphs. Starting again with the proof that bounded size important separators are bounded in number, we wish to highlight two further main contributions in this vein. First, Marx et al. [27] showed that all inclusion-minimal st-cuts of size at most k in a graph G (for fixed vertices s, t) are contained in (essentially)

---

[1]In the Multiway Cut problem, the input consists of a graph G, a set $T \subseteq V(G)$ of terminals, and an integer p; the goal is to delete at most p edges (in the edge-deletion variant) or nonterminal vertices (in the node-deletion variant) so that every terminal lies in a different connected component of the resulting graph, that is, to separate all pairs of terminals.

a subgraph of G of treewidth at most $f(k)$. This statement is attractive in its simplicity, although algorithmic applications of it tend to have a bad running time dependency on k (since $f(k)$ is already exponential, and a treewidth dynamic programming algorithm would normally have running time exponential in $f(k)$). Second, the FPT algorithm for MINIMUM BISECTION [11] involves a technically intricate but very powerful decomposition of the input graph in a tree-like manner into "nearly unbreakable" parts (see the paper for details). However, these two results apply only to undirected graphs, and for directed graphs we have a much weaker grasp of the structure.

A central milestone in the above developments was the discovery in 2010 of FPT algorithms for MULTICUT, a robust generalization of MULTIWAY CUT, parameterized by the size of the cutset only [3, 28]. In this problem we are given a graph G, a family $\mathcal{T}$ of k terminal pairs, and an integer p; the goal is to delete at most p nonterminal vertices from the graph such that all terminal pairs are separated.[2] Marx and Razgon [28] also proved that a similar result is unlikely in directed graphs, showing that DIRECTED MULTICUT is $W[1]$-hard parameterized by the cutset only. In contrast, DIRECTED MULTIWAY CUT is FPT by the same parameter [8]. Marx and Razgon [28] asked about the complexity of DIRECTED MULTICUT when you restrict also the number of terminal pairs, either assuming it is constant or by adding it to the parameter. Since then, this has become one of the most important open problems in the study of graph separation problems in parameterized complexity. It was positively resolved for DAGs (i.e., the problem is FPT on directed acyclic graphs parameterized by both the cutset and the number of terminals) [19], but otherwise little progress has been made. In particular, it has been repeated in the survey of Marx [26] on future directions of parameterized complexity, where also the study of cut problems in directed graphs has been identified as an important research direction. Similarly, the question of the structure of bounded-size directed cuts (e.g., the existence of a directed version of [27]) has been floating around the community.

Our results.   We almost completely resolve the question of Marx and Razgon [28]:

Theorem 1.1. DIRECTED MULTICUT *is $W[1]$-hard when parameterized by p, the size of the cutset, even if restricted to instances with only four terminal pairs. Furthermore, assuming the Exponential Time Hypothesis (ETH), there is no algorithm solving n-vertex instances of* DIRECTED MULTICUT *with four terminal pairs in time $f(p)n^{o(p/\log p)}$ for any computable function f.*

Since DIRECTED MULTICUT is polynomial-time solvable for one terminal pair, and reduces to DIRECTED MULTIWAY CUT for two terminal pairs [8], only the case of three terminal pairs remains open.

The Exponential Time Hypothesis (ETH) [18], now a standard lower bound assumption in parameterized complexity and moderately-exponential algorithms (cf. [21]), essentially asserts that there exists no algorithm verifying satisfiability of 3-CNF SAT formulae in time subexponential in the number of variables. The ETH lower bound of Theorem 1.1 implies that a brute-force solution running in time $n^{\mathcal{O}(p)}$ is close to optimal.

Furthermore, we observe that our gadgets and reduction outline, after minor modifications, gives also a lower bound for the STEINER ORIENTATION problem, answering an open question of [9]. In the STEINER ORIENTATION problem, given a mixed graph G with k terminal pairs $\mathcal{T}$, one asks for an orientation of all undirected edges of G, such that for every terminal pair $(s, t) \in \mathcal{T}$ there is a path from s to t in the oriented graph.

Theorem 1.2. STEINER ORIENTATION *is $W[1]$-hard when parameterized by k, the number of terminal pairs. Furthermore, assuming ETH, there is no algorithm solving n-vertex instances of* STEINER ORIENTATION *in time $f(k)n^{o(k/\log k)}$ for any computable function f.*

Cygan, Kortsarz, and Nutov [9] showed an $n^{\mathcal{O}(k)}$-time algorithm; Theorem 1.2 implies that there is little hope for a fixed-parameter algorithm, and that the running time of the algorithm of Cygan, Kortsarz, and Nutov is close to optimal.

We also show a structural result for directed st-cuts of bounded size. A graph G is *k-cut-minimal* (for arc-cuts) if it contains a source vertex s and a sink vertex t, and every other vertex is incident to at least one arc that participates in an inclusion-minimal st-cut of size at most k. (For example, one can reduce an

---

[2]In this paper we use consistently the node-deletion variant of MULTICUT; note that it is equivalent to the edge-deletion setting in the directed case, and strictly more general in the undirected case.
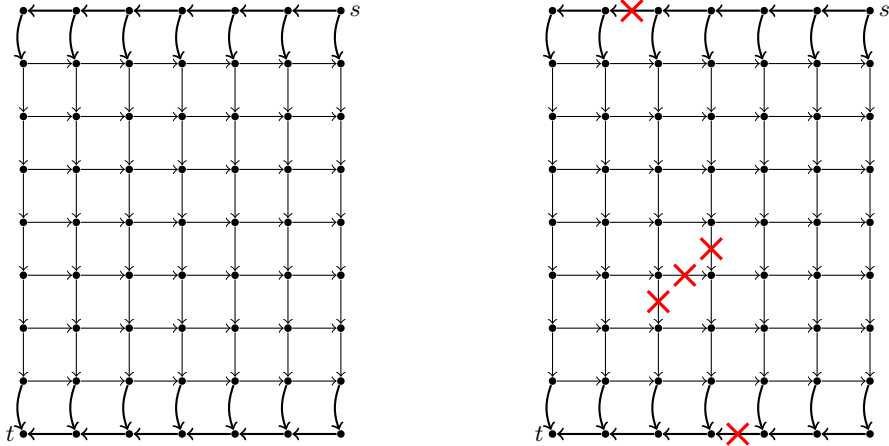
Figure 1: An example of 5-cut-minimal graph with unbounded treewidth of the underlying undirected graph. The right panel shows a minimal cut of size five; observe that every edge of the graph participates in a similar cut, except for a few edges in the four corners of the graph, which can be easily covered by other small cuts.

arbitrary graph to a k-cut-minimal one by repeatedly bypassing every vertex that is not incident to any such arcs.) We have the following.

Theorem 1.3. *Every k-cut-minimal graph has directed treewidth at most $f(k)$, where $f(k) = 2^{\mathcal{O}(k^2)}$.*

We note that this result is likely not as tight as it could be, both in terms of the choice of width measure and the function $f(k)$. However, as discussed below, our results also show that DIRECTED MULTICUT is essentially hard already for four terminal pairs and directed pathwidth two, so for the purpose of the DIRECTED MULTICUT problem, a sharpening of Theorem 1.3 is not likely to help (even though it may be useful for other problems).

Discussion. In the presence of the lower bound of Theorem 1.1, whose proof in this manuscript takes less than five pages including a figure and is not very technically complex, the natural question is why the lower bound was so elusive in the past five years, despite significant effort.

The idea to represent an n-wise choice in the reduction (e.g., the choice of a vertex of a clique or an image of a vertex of the pattern graph in the case of the SUBGRAPH ISOMORPHISM problem) as a choice of a single-vertex cut on a n-vertex bidirectional path connecting a terminal pair, seems quite natural, especially in the light of the FPT algorithm for DIRECTED MULTICUT in DAGs [19] and a deeper study of where this algorithm uses the assumption of acyclicity. However, a gadget to check an arbitrary binary relation between two such choices was elusive; we stumbled upon a "correct" construction by investigating the structural question that led to Theorem 1.3 (the idea being that a sufficiently powerful structural result could have been useful in designing an efficient algorithm).

A few words about directed width measures may be in order (for more, see, e.g., [1, 20]). Two approaches are possible. On the one hand, one can simply consider the underlying undirected graph and ignore the arc orientations; e.g., if the underlying undirected graph of a directed graph G has bounded treewidth, then many problems are solvable on G via standard dynamic programming. On the other hand, several width notions specific to directed graphs have been proposed (to name a few: directed pathwidth; directed treewidth; DAG-width; Kelly-width). Of these, directed treewidth is the most general and directed pathwidth the most restrictive. It has been noted that these directed width notions are rather more difficult to use in algorithms than the undirected variants (e.g., [16, 15]), but on the other hand they are more permissive (e.g., they all include DAGs as a constant-width base case).

Our gadgets are based around a counterexample against the first of these approaches. Figure 1 shows a graph which is 5-cut-minimal (in fact, where *every* arc participates in a minimal *st*-cut of bounded size), but

whose underlying undirected graph is a grid (and thus has unbounded treewidth). Our hardness reduction uses this graph to route one commodity horizontally and one vertically, allowing us to control which pairs of flows can be killed at unit cost.

Also note that the graph in Figure 1 is acyclic; in fact, the reduction of Theorem 1.1 outputs an instance of integer-weighted DIRECTED MULTICUT of directed pathwidth 2. Standard reductions (replacing a vertex of weight w by w unit-weight copies) creates an instance of unweighted DIRECTED MULTICUT with directed pathwidth bounded polynomially in the parameter. Hence DIRECTED MULTICUT remains $W[1]$-hard, even for four terminal pairs, if we parameterize by both the size of the cutset and the directed pathwidth of the input graph. By the discussion above, this leaves little room for generalizing the FPT-algorithm for DAGs [19] to broader classes.

Organization and notation. We show the reduction for Theorem 1.1 in Section 2 and for Theorem 1.2 in Section 3. We prove Theorem 1.3 in Section 4.

We use standard graph notation, see e.g. [2]. Both our hardness reductions start from the $W[1]$-hard PARTITIONED SUBGRAPH ISOMORPHISM problem, parameterized by the number of edges of the pattern graph. An input to PARTITIONED SUBGRAPH ISOMORPHISM consists of two graphs G and H with $|E(H)| = k$, where $V(G)$ is partitioned into $|V(H)|$ color classes, one for each vertex of H: $V(G) = \biguplus_{i \in V(H)} V_i$; the goal is to check if there exists a homomorphism $\phi : V(H) \to V(G)$ such that $\phi(i) \in V_i$ for every $i \in V(H)$. The $W[1]$- and ETH-hardness of PARTITIONED SUBGRAPH ISOMORPHISM has been shown by Marx:

Theorem 1.4 (Cor. 6.3 of [24]). PARTITIONED SUBGRAPH ISOMORPHISM *is* $W[1]$-*hard when parameterized by* $k = |E(H)|$. *Furthermore, assuming ETH, there is no algorithm solving n-vertex instances of* PARTITIONED SUBGRAPH ISOMORPHISM *in time* $f(k)n^{o(k/\log k)}$ *for any computable function* f.

Both our reductions work in polynomial time and, given a PARTITIONED SUBGRAPH ISOMORPHISM instance (G, H), output an instance with the corresponding parameter bounded linearly in $k = |E(H)|$ and size bounded polynomially in the input size.

In both reductions, given an input instance (G, H) of PARTITIONED SUBGRAPH ISOMORPHISM, we denote $V(G) = \bigcup_{i \in V(H)} V_i$, $k = |E(H)|$, and $\ell = |V(H)|$. Furthermore, without loss of generality we assume that H does not have any isolated vertices and thus $\ell \leq 2k$. Indeed, if some color class V_i of $V(G)$ is empty, we can output a trivial no-instance, and otherwise we can delete all isolated vertices from H and their corresponding color classes in G without changing the answer to the problem. We also fix some total order $<$ on the set $V(H)$ and, by some potential padding, assume that $|V_i| = n$ and $V_i = \{v_a^i : 1 \leq a \leq n\}$ for every $i \in V(H)$.

For discussion on the output of the reduction of Theorem 1.1, we need to recall the definition of directed pathwidth; we follow here Kreutzer and Ordyniak [20].

Definition 1.1. Let G be a directed graph. A *directed path decomposition* is a pair (P, β) where $P = v_1, v_2, \ldots, v_s$ is a directed path and $\beta : V(P) \to 2^{V(G)}$ is a mapping satisfying:

1. For every $v \in V(G)$ the set $\beta^{-1}(v)$ is connected in P and nonempty.

2. For every arc $(u, v) \in E(G)$ there are indices $1 \leq i \leq j \leq s$ such that $u \in \beta(v_i)$ and $v \in \beta(v_j)$.

The *width* of (P, β) is $\max_{1 \leq i \leq s} |\beta(v_i)|$, and the *directed pathwidth* of G is the minimum width among all its directed path decompositions. For every $1 \leq i \leq s$, the set $\beta(v_i)$ is called a *bag* at node v_i.

Note that directed pathwidth of a nonempty directed acyclic graph G equals 1: we can take vertices of G one-by-one in any topological ordering of G.

2 Hardness for Directed Multicut

We reduce an input PARTITIONED SUBGRAPH ISOMORPHISM instance (G, H) to a node-weighted variant of DIRECTED MULTICUT, where every non-terminal vertex has a weight being a positive integer, and the goal is to

find a cutset of total weight not exceeding the budget p. We fix some integer constant M; in fact it suffices to set $M = 2$, but it helps to think of it as a sufficiently large integer constant. We use three levels of weight: there will be *light vertices*, of weight 1, *heavy vertices*, of various integer weights being multiples of M, usually depending on the degree of the corresponding vertex of H, and *undeletable vertices*, of weight $p+1$. Observe that it is easy to reduce the weighted variant to the original one by replacing every vertex of weight w with w unit-weight vertices. Thus, it suffices to show the reduction to the weighted variant, with only four terminal pairs.

Construction. Let us now describe the construction of the (weighted) DIRECTED MULTICUT instance $(G', \mathcal{T}, p)$. We start by setting budget $p = (6M + 1)k$. We also introduce eight terminals, arranged in four terminal pairs:

$$(s^x_{0 \to n}, t^x_{0 \to n}), \quad (s^y_{0 \to n}, t^y_{0 \to n}),$$
$$(s^<_{n \to 0}, t^<_{n \to 0}), \quad (s^>_{n \to 0}, t^>_{n \to 0}).$$

For every $i \in V(H)$, we introduce a bidirected path on $2n + 1$ vertices

$$z^i_0, \hat{z}^i_1, z^i_1, \hat{z}^i_2, z^i_2, \ldots, \hat{z}^i_n, z^i_n,$$

called henceforth the *z-path for vertex* i, and denoted by Z^i. Similarly, for every ordered pair (i, j) where $ij \in E(H)$, we introduce two bidirected paths on $2n + 1$ vertices

$$x^{i,j}_0, \hat{x}^{i,j}_1, x^{i,j}_1, \hat{x}^{i,j}_2, x^{i,j}_2, \ldots, \hat{x}^{i,j}_n, x^{i,j}_n;$$
$$y^{i,j}_0, \hat{y}^{i,j}_1, y^{i,j}_1, \hat{y}^{i,j}_2, y^{i,j}_2, \ldots, \hat{y}^{i,j}_n, y^{i,j}_n.$$

We call these paths *the x-path and the y-path for the pair* (i, j), and denote them by $X^{i,j}$ and $Y^{i,j}$. All vertices z^i_a, $x^{i,j}_a$, $y^{i,j}_a$ are undeletable, while all vertices $\hat{z}^i_a$, $\hat{x}^{i,j}_a$, $\hat{y}^{i,j}_a$ are heavy: vertex $\hat{z}^i_a$ has weight $M \cdot \deg_H(i)$ and the vertices $\hat{x}^{i,j}_a$ and $\hat{y}^{i,j}_a$ have weight M each.

Note that so far we have created $4k + \ell$ paths, each having $2n + 1$ vertices. Furthermore, if we are to delete one heavy vertex from each of these paths, the total cost would be

$$M \cdot \left(4k + \sum_{i \in V(H)} \deg_H(i) \right) = 6kM.$$

For every pair (i, j) with $ij \in E(H)$, and every $0 \le a \le n$, we add arcs $(x^{i,j}_a, z^i_a)$ and $(z^i_a, y^{i,j}_a)$. Furthermore, we attach terminals to the paths as follows.

- for every pair (i, j) with $ij \in E(H)$, we add arcs $(s^x_{0 \to n}, x^{i,j}_0)$ and $(y^{i,j}_n, t^y_{0 \to n})$;

- for every $i \in V(H)$ we add arcs $(s^y_{0 \to n}, z^i_0)$ and $(z^i_n, t^x_{0 \to n})$;

- for every pair (i, j) with $ij \in E(H)$ and $i < j$ we add arcs $(s^<_{n \to 0}, x^{i,j}_n)$ and $(y^{i,j}_0, t^<_{n \to 0})$;

- for every pair (i, j) with $ij \in E(H)$ and $i > j$ we add arcs $(s^>_{n \to 0}, x^{i,j}_n)$ and $(y^{i,j}_0, t^>_{n \to 0})$.

We refer to Figure 2 for an illustration. Intuitively, the so-far constructed bidirectional paths and terminals require to delete at least one heavy vertex per bidirectional path; the connections between paths ensure that for every $i \in V(H)$, we need to chose one index $1 \le \phi(i) \le n$ and delete vertices $\hat{z}^i_{\phi(i)}$, $\hat{x}^{i,j}_{\phi(i)}$, and $\hat{y}^{i,j}_{\phi(i)}$, that is, cut all paths corresponding to the vertex i at the same place. The choice of the index $\phi(i)$ corresponds to the choice of the image of i in the sought homomorphism.

Let us now introduce gadgets that check the edge relations between the chosen vertices. For every pair (i, j) with $ij \in E(H)$, $i < j$ we introduce an acyclic $n \times n$ grid with vertices $p^{i,j}_{a,b}$ for $1 \le a, b \le n$ and arcs $(p^{i,j}_{a,b}, p^{i,j}_{a+1,b})$ for every $1 \le a < n$ and $1 \le b \le n$, as well as $(p^{i,j}_{a,b}, p^{i,j}_{a,b+1})$ for every $1 \le a \le n$ and $1 \le b < n$.

We call this grid *p-grid for the pair* (i, j), and denote it by $P^{i,j}$. We set the vertex $p_{a,b}^{i,j}$ to be a light vertex if $v_a^i v_b^j \in E(G)$, and undeletable otherwise. Finally, for every $1 \leq a \leq n$ we introduce arcs:

$$(x_a^{i,j}, p_{a,1}^{i,j}), \quad (p_{a,n}^{i,j}, y_{a-1}^{i,j}), \quad (x_a^{j,i}, p_{1,a}^{i,j}), \quad (p_{n,a}^{i,j}, y_{a-1}^{j,i}).$$

Intuitively, after deleting the aforementioned heavy vertices on the x-, y-, and z-paths, for fixed $ij \in E(H)$, $i < j$, there is only one remaining path from $x_n^{i,j}$ (an out-neighbor of $s_{n \to 0}^<$) to $y_0^{i,j}$ (an in-neighbor of $t_{n \to 0}^<$), passing through the $\phi(i)$-th row of the p-grid for the pair (i, j), and there is only one remaining path from $x_n^{j,i}$ (an out-neighbor of $s_{n \to 0}^>$) to $y_0^{j,i}$ (an in-neighbor of $t_{n \to 0}^>$), passing through the $\phi(j)$-th column of the p-grid for the pair (i, j). We can kill both these paths with a single vertex $p_{\phi(i),\phi(j)}^{i,j}$, but only the existence of the edge $v_{\phi(i)}^i v_{\phi(j)}^j$ ensures that this vertex is light, not undeletable.

This concludes the construction of the instance $(G', \mathcal{T}, p)$. We now formally show that the constructed instance is equivalent to the input PARTITIONED SUBGRAPH ISOMORPHISM instance (G, H).

From a homomorphism to a cutset. Let $\phi : V(H) \to [n]$ be such that $i \mapsto v_{\phi(i)}^i \in V_i$ is a homomorphism of H into G. Define

$$X = \{\hat{x}_{\phi(i)}^{i,j}, \hat{y}_{\phi(i)}^{i,j} : (i, j) \in V(H) \times V(H) \text{ s.t. } ij \in E(H)\} \cup$$
$$\cup \{\hat{z}_{\phi(i)}^i : i \in V(H)\} \cup \{p_{\phi(i),\phi(j)}^{i,j} : (i, j) \in V(H) \times V(H) \text{ s.t. } ij \in E(H), i < j\}.$$

The total weight of the vertices in X equals:

$$2k \cdot 2 \cdot M + \sum_{i \in V(H)} M \cdot \deg_H(i) + k = (6M + 1)k = p.$$

Note that the fact that $p_{\phi(i),\phi(j)}^{i,j}$ is light for every $ij \in E(H)$, $i < j$ follows from the assumption that the vertices $i \mapsto v_{\phi(i)}^i$ is a homomorphism. Consequently, X is of weight exactly p. We now show that it is a multicut in $(G', \mathcal{T})$.

We start with a simple observation about the structure of the graph G'. While the x-, y-, and z-paths are bidirected, they — together with the p-grids — are arranged in a DAG-like fashion. That is, there are directed arcs from $X^{i,j}$ to Z^i, from Z^i to $Y^{i,j}$, from $X^{i,j}$ and $X^{j,i}$ to $P^{i,j}$, and from $P^{i,j}$ to $Y^{i,j}$ and $Y^{j,i}$, but all cycles in G' are contained in one x-, y-, or z-path.

Consider first the terminal pair $(s_{0 \to n}^x, t_{0 \to n}^x)$. The out-neighbors of $s_{0 \to n}^x$ are the endpoints $x_0^{i,j}$ for every pair (i, j) with $ij \in E(H)$; the only in-neighbors of $t_{0 \to n}^x$ are the endpoints z_n^i for every $i \in V(H)$. Thus, by the previous observation, the only paths from $s_{0 \to n}^x$ to $t_{0 \to n}^x$ in the graph G' start by going to some vertex $x_0^{i,j}$, traverse $X^{i,j}$ up to some vertex $x_a^{i,j}$, use the arc $(x_a^{i,j}, z_a^i)$ to fall to Z^i, and then traverse Z^i to the endpoint z_n^i. (In particular, there are no paths from $s_{0 \to n}^x$ to $t_{0 \to n}^x$ that contain a vertex of some grid $P^{i,j}$.) However, all such paths for $a \geq \phi(i)$ are cut by the vertex $\hat{x}_{\phi(i)}^{i,j} \in X$, while all such paths for $a < \phi(i)$ are cut by the vertex $\hat{z}_{\phi(i)}^i \in X$. Consequently, the terminal pair $(s_{0 \to n}^x, t_{0 \to n}^x)$ is separated in $H \setminus X$.

A similar argument holds for the pair $(s_{0 \to n}^y, t_{0 \to n}^y)$. By the same reasoning, the only paths between $s_{0 \to n}^y$ and $t_{0 \to n}^y$ in the graph H are paths that start by going to some vertex z_0^i, traverse Z^i up to some vertex z_a^i, use the arc $(z_a^i, y_a^{i,j})$ for some $ij \in E(H)$ to fall to $Y^{i,j}$, and then continue along this y-path to the vertex $y_n^{i,j}$. However, all such paths for $a \geq \phi(i)$ are cut by the vertex $\hat{z}_{\phi(i)}^i \in X$, while all such paths for $a < \phi(i)$ are cut by the vertex $\hat{y}_{\phi(i)}^{i,j} \in X$.

Let us now focus on the terminal pair $(s_{n \to 0}^<, t_{n \to 0}^<)$. Observe that there are two types of paths from $s_{n \to 0}^<$ to $t_{n \to 0}^<$ in the graph H. The first type consists of paths that starts by going to some vertex $x_n^{i,j}$ where $i < j$, traverse $X^{i,j}$ up to some vertex $x_a^{i,j}$, use the arc $(x_a^{i,j}, z_a^i)$ to fall to Z^i, then traverse Z^i up to some vertex z_b^i, use the arc $(z_b^i, y_b^{i,j'})$ for some $j' > i$ to fall to $Y^{i,j'}$, and finally traverse this y-path to the endpoint $y_0^{i,j'}$. However, similarly as in the previous cases, the vertices $\hat{x}_{\phi(i)}^{i,j}, \hat{z}_{\phi(i)}^i, \hat{y}_{\phi(i)}^{i,j'} \in X$ cut all such paths.

The second type of paths use the p-grids in the following manner: the path starts by going to some vertex $x^{i,j}$ where $i < j$, traverse $X^{i,j}$ up to some vertex $x_a^{i,j}$, use the arc $(x_a^{i,j}, p_{a,1}^{i,j})$ to fall to $P^{i,j}$, traverse this p-grid up to a vertex $p_{b,n}^{i,j}$ where $b \geq a$, use the arc $(p_{b,n}^{i,j}, y_{b-1}^{i,j})$ to fall to $Y^{i,j}$, and traverse this path to the endpoint $y_0^{i,j}$. These paths are cut by X as follows: the paths where $a < \phi(i)$ are cut by $\hat{x}_{\phi(i)}^{i,j} \in X$, the paths where $b > \phi(i)$ are cut by $\hat{y}_{\phi(i)}^{i,j}$, while the paths where $a = b = \phi(i)$ are cut by the vertex $p_{\phi(i),\phi(j)}^{i,j} \in X$; note that the $\phi(i)$-th row of the grid is the only path from $p_{\phi(i),1}^{i,j}$ to $p_{\phi(i),n}^{i,j}$. Please observe that the terminal $t_{n \to 0}^<$ cannot be reached from $P^{i,j}$ by going to the other y-path reachable from this p-grid, namely $Y^{j,i}$, as $Y^{j,i}$ has only outgoing arcs to the terminal $t_{n \to 0}^>$ since $j > i$.

A similar argument holds for the pair $(s_{n \to 0}^>, t_{n \to 0}^>)$. The paths going through $X^{i,j}$, $i > j$, Z^i, and $Y^{i,j'}$, $i > j'$, are cut by vertices $\hat{x}_{\phi(i)}^{i,j}, \hat{z}_{\phi(i)}^i, \hat{y}_{\phi(i)}^{i,j'} \in X$. The paths going through $X^{i,j}$, $i > j$, $P^{j,i}$, and $Y^{i,j}$, are cut by the vertices $\hat{x}_{\phi(i)}^{i,j}, \hat{y}_{\phi(i)}^{i,j}, p_{\phi(j),\phi(i)}^{j,i}$. Again, it is essential that the other y-path reachable from the p-grid for the pair (j, i), namely $Y^{j,i}$, does not have outgoing arcs to the terminal $t_{n \to 0}^>$, but only to the terminal $t_{n \to 0}^<$.

We infer that X is a solution to the DIRECTED MULTICUT instance $(G', \mathcal{T}, p)$.

From a multicut to a homomorphism. Let X be a solution to the DIRECTED MULTICUT instance $(G', \mathcal{T}, p)$. Our goal is to find a homomorphism of H into G.

First, let us focus on heavy vertices in X. Observe that for every pair (i, j), $ij \in E(H)$ the following three paths needs to be cut by X:

- a path from $s_{0 \to n}^x$ to $t_{0 \to n}^x$ that traverses the entire path $X^{i,j}$ up to the vertex $x_n^{i,j}$, and uses the arc $(x_n^{i,j}, z_n^i)$ to reach $t_{0 \to n}^x$,

- a path from $s_{0 \to n}^x$ to $t_{0 \to n}^x$ that starts with using the arc $(x_0^{i,j}, z_0^i)$, and then traverses Z^i up to the vertex z_n^i, and

- a path from $s_{0 \to n}^y$ to $t_{0 \to n}^y$ that starts with using the arc $(z_0^i, y_0^{i,j})$, and then traverses $Y^{i,j}$ up to the vertex $y_n^{i,j}$.

We infer that X needs to contain at least one heavy vertex on every x-, y-, and z-path in H. Recall that the total weight of exactly one heavy vertex from each of these paths is $6kM = p - k$. Thus, we have only k slack in the budget constraint.

We say that a path $X^{i,j}$, $Y^{i,j}$, or Z^i is *normal* if it contains exactly one vertex of X, and *cheated* otherwise. We say that a pair (i, j) for $ij \in E(H)$ is *normal* if each of the paths $X^{i,j}$, $Y^{i,j}$, $X^{j,i}$, $Y^{j,i}$, Z^i, and Z^j is normal. A pair (i, j) is *cheated* if it is not normal. Note that (i, j) is normal if and only if (j, i) is normal.

For every $i \in V(H)$ we fix one $\phi(i) \in [n]$ such that $\hat{z}_{\phi(i)}^i \in X$.

Fix now a normal pair (i, j), $ij \in E(H)$. Assume $i < j$; a symmetrical argument holds for $i > j$ but uses the terminal pair $(s_{n \to 0}^>, t_{n \to 0}^>)$ instead of $(s_{n \to 0}^<, t_{n \to 0}^<)$. Let $\hat{x}_a^{i,j}, \hat{z}_b^i, \hat{y}_c^{i,j} \in X$. Observe that $a \leq b$, as otherwise the path from $s_{0 \to n}^x$ to $t_{0 \to n}^x$ that traverses $X^{i,j}$ up to the vertex $x_{a-1}^{i,j}$, uses the arc $(x_{a-1}^{i,j}, z_{a-1}^i)$, and traverses Z^i up to the endpoint z_n^i is not cut by X, a contradiction. A similar argument for the terminal pair $(s_{0 \to n}^y, t_{0 \to n}^y)$ implies that $b \leq c$. However, if $a < b$, then the path from $s_{n \to 0}^<$ to $t_{n \to 0}^<$ that traverses $X^{i,j}$ up to the vertex $x_a^{i,j}$, uses the arc $(x_a^{i,j}, z_a^i)$, traverses Z^i up to the endpoint z_0^i, and finally uses the arc $(z_0^i, y_0^{i,j})$, is not cut by X, a contradiction. A similar argument implies gives a contradiction if $b < c$.

We infer that for every normal pair (i, j) we have $\hat{z}_{\phi(i)}^i \in X$ and $\hat{x}_{\phi(i)}^{i,j}, \hat{y}_{\phi(i)}^{i,j} \in X$.

Fix now a normal pair (i, j) with $i < j$. Observe that the following paths are not cut by the heavy vertices in X:

- a path from $s_{n \to 0}^<$ to $t_{n \to 0}^<$ that traverses $X^{i,j}$ up to the vertex $x_{\phi(i)}^{i,j}$, uses the arc $(x_{\phi(i)}^{i,j}, p_{\phi(i),1}^{i,j})$, traverses the $\phi(i)$-th row of $P^{i,j}$ up to the vertex $p_{\phi(i),n}^{i,j}$, uses the arc $(p_{\phi(i),n}^{i,j}, y_{\phi(i)-1}^{i,j})$, and traverses $Y^{i,j}$ up to the endpoint $y_0^{i,j}$;

- a path from $s^>_{n\to 0}$ to $t^>_{n\to 0}$ that traverses $X^{j,i}$ up to the vertex $x^{j,i}_{\phi(j)}$, uses the arc $(x^{j,i}_{\phi(j)}, p^{i,j}_{1,\phi(j)})$, traverses the $\phi(j)$-th column of $P^{i,j}$ up to the vertex $p^{i,j}_{n,\phi(j)}$, uses the arc $(p^{i,j}_{n,\phi(j)}, y^{j,i}_{\phi(j)-1})$, and traverses $Y^{j,i}$ up to the endpoint $y^{j,i}_0$.

Consequently, X needs to contain at least one light vertex in the p-grid $P^{i,j}$. Furthermore, if X contains exactly one light vertex in $P^{i,j}$, then, as the only vertex in common of the two aforementioned paths for a fixed choice of normal (i,j), $i < j$, is the vertex $p^{i,j}_{\phi(i),\phi(j)}$, we have that $p^{i,j}_{\phi(i),\phi(j)} \in X$ is a light vertex, and, by construction, $v^i_{\phi(i)} v^j_{\phi(j)} \in E(G)$.

It remains to show that every pair (i,j), $ij \in E(H)$, is normal. Indeed, if this is the case, then, as $p = 6kM + k$, the total weight of exactly one heavy vertex on each path $X^{i,j}$, $Y^{i,j}$, and Z^i is $6kM$, and there are exactly k grids $P^{i,j}$, every grid $P^{i,j}$ contains exactly one vertex of X, and the argumentation from the previous section shows that $i \mapsto v^i_{\phi(i)}$ is a homomorphism, concluding the proof of Theorem 1.1.

Recall that (i,j) is normal if and only if (j,i) is normal. Let c be the number of cheated pairs (i,j), $i < j$. If (i,j), $i < j$, is cheated, then there is a witness for it: one of the paths $X^{i,j}$, $Y^{i,j}$, $X^{j,i}$, $Y^{j,i}$, Z^i, or Z^j is cheated, that is, contains more than one vertex of X. However, note that a cheated path $X^{i,j}$, $Y^{i,j}$, $X^{j,i}$, or $Y^{j,i}$ is a witness only that (i,j) and (j,i) is cheated. Let c_{XY} be the number of cheated x- and y-paths. Furthermore, a cheated path Z^i is a witness that (i,j) is cheated for every $ij \in E(H)$: there are only $\deg_H(i)$ such pairs (i,j). We infer that

$$c \le c_{XY} + \sum_{i \in V(H):Z^i \text{ cheated}} \deg_H(i).$$

On the other hand, a cost of a second heavy vertex in X on an x- or y-path is M, while the cost of a second heavy vertex on Z^i is $M \cdot \deg_H(i)$. Furthermore, recall that if (i,j), $i < j$, is normal, then $P^{i,j}$ contains at least one vertex of X. Thus, the total weight of X is at least

$$6kM + M \cdot c_{XY} + \sum_{i \in V(H):z^i \text{ cheated}} M \cdot deg_H(i) + k - c$$
$$\ge 6kM + M \cdot c + k - c = p + (M-1)c.$$

Consequently, if $M > 1$ and the weight of X is at most p, we have $c = 0$. This finishes the proof of Theorem 1.1.

The resulting digraph can easily be shown to have directed pathwidth 2. Clearly, it cannot have smaller directed pathwidth, as it contains two-vertex cycles. In the other direction, order the vertices of the resulting graphs as follows.

1. All source terminals.

2. The vertices $x^{i,j}_a$, sorted first by the pair (i,j) lexicographically, and then by the subscript a.

3. The vertices z^i_a, sorted first by i, and then by the subscript a.

4. The vertices $p^{i,j}_{a,b}$, sorted first by the pair (i,j) lexicographically, and then by the pair (a,b) lexicographically.

5. The vertices $y^{i,j}_a$, sorted first by the pair (i,j) lexicographically, and then by the subscript a.

6. All sink terminals.

Observe now that we can construct a directed path decomposition of the resulting graph by taking bags consisting of every two consecutive vertices in this order.

We infer that DIRECTED MULTICUT is $W[1]$-hard already for integer-weighted instances, parameterized by total solution weight, for instances with 4 terminals and directed pathwidth 2. This is in sharp contrast to the result that it is FPT for DAGs [19].

3 Hardness for Steiner Orientation

Construction. Given an input PARTITIONED SUBGRAPH ISOMORPHISM instance (G, H), we construct an equivalent STEINER ORIENTATION instance $(G', \mathcal{T})$ as follows.

First, we introduce a number of undirected paths on n vertices: for every $i \in V(H)$ we introduce a path C^i on vertices $c_1^i, c_2^i, \ldots, c_n^i$ and a path D^i on vertices $d_1^i, d_2^i, \ldots, d_n^i$, while for every ordered pair (i, j) where $ij \in E(H)$, we introduce a path $X^{i,j}$ on vertices $x_1^{i,j}, x_2^{i,j}, \ldots, x_n^{i,j}$ and a path $Y^{i,j}$ on vertices $y_1^{i,j}, y_2^{i,j}, \ldots, y_n^{i,j}$. We connect these paths as follows: for every $i \in V(H)$ and $1 \le a \le n$ we add an arc (c_a^i, d_a^i), while for every pair (i, j) with $ij \in E(H)$ and for every $1 \le a \le n$ we add arcs $(x_a^{i,j}, d_a^i)$ and $(c_a^i, y_a^{i,j})$. Furthermore, for every $i \in V(H)$ we add terminal pairs (c_1^i, d_n^i) and (c_n^i, d_1^i) and for every pair (i, j) where $ij \in E(H)$, we add terminal pairs $(x_1^{i,j}, d_n^i)$, $(x_n^{i,j}, d_1^i)$, $(c_1^i, y_n^{i,j})$, and $(c_n^i, y_1^{i,j})$.

For an index $1 \le a \le n$, we say that the path C^i is *oriented towards* c_a^i if for every $1 \le b < n$, the edge $c_b^i c_{b+1}^i$ is oriented from c_b^i to c_{b+1}^i if $b < a$ and from c_{b+1}^i to c_b^i if $b \ge a$. Similarly, we say that the path D^i is *oriented away from* d_a^i if for every $1 \le b < n$, the edge $d_b^i d_{b+1}^i$ is oriented from d_{b+1}^i to d_b^i if $b < a$ and from d_b^i to d_{b+1}^i if $b \ge a$, and similar notions are defined for the paths $X^{i,j}$ and $Y^{i,j}$.

Informally speaking, the introduced paths play the role of $x-$, $y-$, and $z-$ paths from the reduction for DIRECTED MULTICUT. In the currently constructed graph, the only way to satisfy both the terminal pair (c_1^i, d_n^i) and the terminal pair (c_n^i, d_1^i) is to chose a number $1 \le \phi(i) \le n$ and direct the path C^i towards $c_{\phi(i)}^i$ and the path D^i away from $d_{\phi(i)}^i$. The choice of $\phi(i)$ corresponds to the image of $i \in V(H)$ in the sought homomorphism. The additional terminal pairs ensure that the choice is copied to other paths: $X^{i,j}$ needs to be directed towards $x_{\phi(i)}^{i,j}$ and $Y^{i,j}$ needs to be directed away from $y_{\phi(i)}^{i,j}$. In the remainder of the construction, we will not introduce any more edge nor arc incident to a vertex of any path C^i or D^i, and only arcs going out of the paths $X^{i,j}$ and arcs going towards the paths $Y^{i,j}$. In this way we will not introduce any unwanted way of satisfying the terminal pairs introduced so far, guaranteeing the desired behavior.

Let us now proceed to the construction of the p-grids. Similarly as in the case of DIRECTED MULTICUT, for every pair (i, j) with $ij \in E(H)$ and $i < j$ we introduce an acyclic $n \times n$ grid (called again a *p-grid*) with vertices $p_{a,b}^{i,j}$ for $1 \le a, b \le n$ and arcs $(p_{a,b}^{i,j}, p_{a+1,b}^{i,j})$ for every $1 \le a < n$ and $1 \le b \le n$, as well as $(p_{a,b}^{i,j}, p_{a,b+1}^{i,j})$ for every $1 \le a \le n$ and $1 \le b < n$. We connect the grid to the previously constructed vertices as follows: for every $1 \le a \le n$ we introduce arcs

$$(x_a^{i,j}, p_{a,1}^{i,j}), \quad (p_{a,n}^{i,j}, y_a^{i,j}), \quad (x_a^{j,i}, p_{1,a}^{i,j}), \quad (p_{n,a}^{i,j}, y_a^{j,i});$$

please note the lack of shift by one as compared to the DIRECTED MULTICUT construction. Furthermore, we add terminal pairs $(x_n^{i,j}, y_1^{i,j})$ and $(x_n^{j,i}, y_1^{j,i})$.

Observe that, if the paths $X^{i,j}$ and $X^{j,i}$ were directed towards $x_{\phi(i)}^{i,j}$ and $x_{\phi(j)}^{j,i}$ respectively, while the paths $Y^{i,j}$ and $Y^{j,i}$ were directed away from $y_{\phi(i)}^{i,j}$ and $y_{\phi(j)}^{j,i}$ respectively, there are unique paths in the graph satisfying the newly introduced terminal pairs: the one from $x_n^{i,j}$ to $y_1^{i,j}$ needs to traverse the grid for the pair (i, j) along the $\phi(i)$-th row, while the one from $x_n^{j,i}$ to $x_1^{j,i}$ needs to traverse it along the $\phi(j)$-th column. These two paths intersect at the vertex $p_{\phi(i),\phi(j)}^{i,j}$; we finish the construction by encoding the edges of G by the following modification of the vertices $p_{a,b}^{i,j}$.

For every vertex $p_{a,b}^{i,j}$, we call its incident four arcs as follows: the *north* arc goes from $p_{a-1,b}^{i,j}$ (or $x_b^{j,i}$ if $a = 1$) to $p_{a,b}^{i,j}$, the *south* arc goes from $p_{a,b}^{i,j}$ to $p_{a+1,b}^{i,j}$ (or $y_b^{j,i}$ if $a = n$), the *west* arc goes from $p_{a,b-1}^{i,j}$ (or $x_a^{i,j}$ if $b = 1$) to $p_{a,b}^{i,j}$, while the *east* arc goes from $p_{a,b}^{i,j}$ to $p_{a,b+1}^{i,j}$ (or $y_a^{i,j}$ if $b = n$). If $v_a^i v_b^j \in E(G)$, we keep the vertex $p_{a,b}^{i,j}$ intact; otherwise, we split the vertex $p_{a,b}^{i,j}$ into two vertices $p_{a,b,SW}^{i,j}, p_{a,b,NE}^{i,j}$, connected by an undirected edge, with $p_{a,b,SW}^{i,j}$ incident to the south and west arcs and $p_{a,b,NE}^{i,j}$ incident to the north and east arcs. With this construction, we cannot traverse the split vertex $p_{a,b}^{i,j}$ from north to south and from west to east at the same time, forbidding us from choosing $\phi(i) = a$ and $\phi(j) = b$ simultaneously. This implies that $(G', \mathcal{T})$ is a positive instance if and only if (G, H) is a positive instance.

This finishes the description of the constructed STEINER ORIENTATION instance $(G', \mathcal{T})$. Note that the

number of terminal pairs is bounded by $\mathcal{O}(k + \ell) = \mathcal{O}(k)$. We now prove formally that the reduction is a correct hardness reduction for STEINER ORIENTATION.

From a homomorphism to an orientation. Let $\phi : [k] \to [n]$ be such that $i \mapsto v_{\phi(i)}^i$ is a homomorphism of H into G. We start by orienting some of the edges of H as follows. For every $i \in V(H)$, orient the path C^i towards $c_{\phi(i)}^i$ and the path D^i away from $d_{\phi(i)}^i$. Similarly, for every pair (i, j) with $ij \in E(H)$, orient the path $X^{i,j}$ towards $x_{\phi(i)}^{i,j}$ and orient the path $Y^{i,j}$ away from $y_{\phi(i)}^{i,j}$.

Let G'' be the mixed graph obtained so far. For every terminal pair $(s, t) \in \mathcal{T}$, we exhibit a path in G'' from s to t in such a way that no two such paths share an undirected edge. This proves that the remaining undirected edges can be oriented in the desired way.

It is straightforward to observe that for every terminal pair (s, t) involving a c-vertex or a d-vertex (i.e., one introduced in the first part of the construction) there exists a directed path in G'' from s to t consisting of directed arcs only: we may traverse from any endpoint of $X^{i,j}$ or C^i up to $x_{\phi(i)}^{i,j}$ or $c_{\phi(i)}^i$, fall to $d_{\phi(i)}^i$ or $y_{\phi(i)}^{i,j}$, and go along the path D^i or $Y^{i,j}$ to any of its endpoints. Thus, it remains to focus on the pairs $(x_n^{i,j}, y_1^{i,j})$ and $(x_n^{j,i}, y_1^{j,i})$ for $ij \in E(H)$, $i < j$.

For the first pair, we traverse $X^{i,j}$ up to $x_{\phi(i)}^{i,j}$, then traverse along the $\phi(i)$-row of the grid (i.e., using vertices $p_{\phi(i),b}^{i,j}$ for $1 \le b \le n$; if any of these vertices is split, we traverse it from west to east using the intermediate undirected edge), and traverse $Y^{i,j}$ from $y_{\phi(i)}^{i,j}$ up to $y_1^{i,j}$. The path for the second pair is similar, but uses the $\phi(j)$-th column of the grid, and traverses every vertex $p_{a,\phi(j)}^{i,j}$ for $1 \le a \le n$ from north to south.

These two paths intersect at $p_{\phi(i),\phi(j)}^{i,j}$. Since $v_{\phi(i)}^i v_{\phi(j)}^j \in E(G)$, this vertex is not split and the two aforementioned paths do not share an undirected edge. As only these two paths are present in the grid for the pair (i, j), and all undirected edges in G'' are contained in such grids, we conclude that $(G', \mathcal{T})$ is a yes-instance.

From an orientation to a homomorphism. Assume that $(G', \mathcal{T})$ is a yes-instance, and let G'' be an oriented graph G' such that for every $(s, t) \in \mathcal{T}$, there is a path from s to t in G''.

Fix a vertex $i \in V(H)$. Recall that no arc leads toward any path C^i or $X^{i,j}$, and no arc leads from any path D^i or $Y^{i,j}$. Furthermore, all arcs leaving C^i lead to D^i or one of the paths $Y^{i,j}$, and all arcs going into D^i start in C^i or in one of the paths $X^{i,j}$. We infer that the path from c_1^i to d_n^i and the path from c_n^i to d_1^i in G'' both need to be completely contained in $G''[C^i \cup D^i]$. Furthermore, such a path for (c_1^i, d_n^i) traverses C^i up to some vertex c_a^i, falls to d_a^i, and continues along D^i, and similarly for the pair (c_n^i, d_1^i). We infer that there exists an index $1 \le \phi(i) \le n$ such that C^i is oriented towards $c_{\phi(i)}^i$, while D^i is oriented away from $d_{\phi(i)}^i$.

We claim that $i \mapsto v_{\phi(i)}^i$ is a homomorphism of H into G. Fix an edge $ij \in E(H)$, $i < j$; we aim to show that the vertex $p_{\phi(i),\phi(j)}^{i,j}$ is not split, which is equivalent to $v_{\phi(i)}^i v_{\phi(j)}^j \in E(G)$.

A similar reasoning as earlier for the path $X^{i,j}$ and terminal pairs $(x_1^{i,j}, d_n^i)$ and $(x_n^{i,j}, d_1^i)$ implies that $X^{i,j}$ is oriented towards $x_{\phi(i)}^{i,j}$. Analogously, we obtain that $X^{j,i}$ is oriented towards $x_{\phi(j)}^{j,i}$, $Y^{i,j}$ is oriented away from $y_{\phi(i)}^{i,j}$, and $Y^{j,i}$ is oriented away from $y_{\phi(i)}^{j,i}$.

The only arcs leaving the path $X^{i,j}$ are arcs going towards D^i and the p-grid for the pair (i, j). Similarly, the only arcs ending in $Y^{i,j}$ start in C^i and in the aforementioned p-grid, and analogous claims hold for the paths $X^{j,i}$ and $Y^{j,i}$. We infer that the paths in H' for terminal pairs $(x_n^{i,j}, y_1^{i,j})$ and $(x_n^{j,i}, y_1^{j,i})$ need to traverse through the p-grid for the pair (i, j).

With the paths $X^{i,j}$, $X^{j,i}$, $Y^{i,j}$, and $Y^{j,i}$ oriented as described, even with keeping the undirected edges in the grid not oriented, the only path for the first terminal pair traverses the $\phi(i)$-th row of the grid from $x_{\phi(i)}^{i,j}$, through $p_{\phi(i),b}^{i,j}$ for $1 \le b \le n$ (split or not), towards $y_{\phi(i)}^{i,j}$, and similarly the only path for the second terminal pair traverses the $\phi(j)$-th column of the grid. If the vertex $p_{\phi(i),\phi(j)}^{i,j}$ is split, these paths traverse the undirected edge $p_{\phi(i),\phi(j),SW}^{i,j} p_{\phi(i),\phi(j),NE}^{i,j}$ in opposite directions, a contradiction.

This concludes the proof of Theorem 1.2.

4 Directed treewidth bound of k-cut-minimal graphs

We now prove the directed treewidth upper bound (Theorem 1.3). For the duration of this section, we will consider arc cuts for convenience, but the result also implies a very similar statement for vertex cuts.

We will need some further preliminaries. For a vertex set U, we let $\delta(U)$ denote the set of arcs leaving U. Let $G = (V, A)$ be a digraph. A set $T \subseteq V$ is *well-linked* if for every pair of equal sized subsets X and Y of T, there are $|X|$ vertex disjoint paths from X to Y in $G - (T \setminus (X \cup Y))$. Well-linked sets are connected to directed treewidth in the following sense.

Theorem 4.1 (Cor. 6.4.24 of [20]). *Let G be a digraph with no well-linked set of cardinality more than k. Then G has directed treewidth $\mathcal{O}(k)$.*

In what follows we will work only with well-linked sets, and thus we omit the definition of directed treewidth. For the definition of directed treewidth, and more on directed width notions, see Kreutzer and Ordyniak [20].

We will show that a k-cut-minimal graph cannot contain an arbitrarily large well-linked set. To illustrate the approach, assume that $T \subseteq V$ is a well-linked set of sufficiently large size ($|T| > f(k)$ for some $f(k)$ yet to be specified). We will identify a set $U \subseteq V$ with $|\delta(U)| \leq k$, such that $|T \cap U|, |T \setminus U| > k$. This contradicts the assumption that T is well-linked, and implies that G has bounded directed treewidth.

The basic engine of the proof is the *anti-isolation lemma*, due to Marx. For this, we need to recall the notion of important separators. (We use the version for directed graphs [8], adapted to our purpose; in particular, we use arc separators instead of vertex separators. The original definition for undirected graphs was [23].)

Let $G = (V, A)$ be a digraph with $s, t \in V$. An *important st-separator* is a minimal s-t-cut $C \subseteq A$ such that there is no s-t-cut $C' \subseteq A$, $C' \neq C$, with $|C'| \leq |C|$ such that every vertex reachable from s in $G - C$ is also reachable from s in $G - C'$.

Lemma 4.2 ([8]). *There are at most 4^k important s-t-separators of size at most k.*

It is important to note that every s-t-cut C can be "pushed" to an important s-t-separator C' with $|C'| \leq |C|$, such that every vertex reachable from s in $G - C$ is also reachable from s in $G - C'$.

We state and prove the anti-isolation lemma. (This is taken from a set of lecture slides of Marx [25]; to the best of our knowledge, no proof has appeared in a formally reviewed publication.) Because the expression will be used several times, define $g(k) = (k+1)4^{k+1}$.

Lemma 4.3 (Anti-isolation [25]). *Let $s, v_1, \ldots, v_r$ be vertices in a digraph $G = (V, A)$, and let $C_1, \ldots, C_r \subseteq A$ be arc sets of size at most k such that for all $i, j \in [r]$ there is a path from s to v_j in $G - C_i$ if and only if $i = j$. Then $r \leq g(k)$.*

Proof. Add a vertex t, and an arc (v_i, t) for every $i \in [r]$. Then for every $i \in [r]$, the cut $C_i \cup \{v_i, t\}$ is an s-t-cut of size at most $k + 1$, and can be pushed to an important separator C'_i of size at most $k + 1$. Note that necessarily $(v_i, t) \in C'_i$. Since there are only at most 4^{k+1} important separators of size at most $k + 1$, and each of them contains at most $k + 1$ arcs (v_j, t), $j \in [r]$, and since every such arc is contained in an important separator, we conclude that $r \leq (k+1)4^{k+1} = g(k)$. $\square$

We will also use the following dual form.

Lemma 4.4. *Let $t, v_1, \ldots, v_r$ be vertices in a digraph $G = (V, A)$, and let $C_1, \ldots, C_r \subseteq A$ be arc sets of size at most k such that for all $i, j \in [r]$ there is a path from v_j to t in $G - C_i$ if and only if $i = j$. Then $r \leq g(k)$.*

Proof. Simply apply the anti-isolation lemma to the reversed graph $G' = (V, \{(v, u) : (u, v) \in A\})$. $\square$

Let us now recall the sunflower lemma. A *sunflower* is a collection $\{X_1, \ldots, X_r\}$ of subsets of a ground set V such that all pairwise intersections are identical, i.e., there is a set $K \subseteq V$ (the *core* of the sunflower) such that for all $i, j \in [r]$, $i \neq j$ we have $X_i \cap X_j = K$. The (pairwise disjoint) sets $X_i \setminus K$ for $i \in [r]$ are called the *petals* of the sunflower. The famous Sunflower Lemma says the following (original lemma is due to Erdős and Rado [13]; the following is from Flum and Grohe [14]).

Lemma 4.5. *Let $\mathcal{H} \subseteq 2^V$ be a collection of subsets of size d of a ground set V. If $|\mathcal{H}| > d!k^d$, then $\mathcal{H}$ contains a sunflower of cardinality more than k.*

Observe that for any minimal st-cut C containing an arc (u,v), the graph $G - C$ contains both an su-path and a vt-path. Using the Erdős-Rado sunflower lemma, we can then sharpen these statements into the following.

Lemma 4.6. *Let $G = (V, A)$ be a digraph, with two distinguished vertices $s, t \in V$. Let $F \subseteq A$ be a collection of arcs such that every arc $(u, v) \in F$ participates in some minimal st-cut of size at most k. If $|F| > h(k)$ where $h(k) = 2^{\mathcal{O}(k^2)}$, then there is a minimal st-cut $C \subseteq A$ in G of size at most k that splits F, i.e., there are more than k arcs $(u, v) \in F \setminus C$ such that $G - C$ contains an su-path, and more than k arcs $(u, v) \in F \setminus C$ such that $G - C$ contains a vt-path.*

Proof. For each $(u, v) \in F$, let $C_{uv} \subseteq A$ be a minimal s-t-cut with $(u, v) \in C_{uv}$, with $|C_{uv}| \leq k$. Let S_{uv} denote the set of vertices reachable from s in $G - C_{uv}$, and let T_{uv} denote the set of vertices that reach t in $G - C_{uv}$. From the observation stated before the lemma statement, by the minimality of C_{uv} we have $u \in S_{uv}$ and $v \in T_{uv}$. Also let $U = \{u : (u, v) \in F\}$ and $W = \{v : (u, v) \in F\}$. We first make a claim to lower-bound the size of U and W. (This is standard, but we include a proof for completeness.)

Claim 4.7. *For every vertex v, at most $2g(k)$ arcs incident to v participate in minimal s-t-cuts of size at most k.*

Proof. We bound the number of in-arcs and out-arcs separately. For the in-arcs, consider the graph resulting by adding $k + 1$ arcs (v, t). Note that every minimal s-t-cut that contains an in-arc of v remains a minimal s-t-cut. As in the proof of the anti-isolation lemma, we may push each such cut to an important separator, and this operation will not decrease the set of in-arcs of v contained in the cut. Hence at most $g(k)$ in-arcs of v participate in minimal s-t-cuts of cardinality at most k. The proof bounding out-arcs is entirely analogous. ⌐

We make a further subsidiary claim.

Claim 4.8. *There are at most $2^{\mathcal{O}(k^2)}$ arcs $(u, v) \in F$ such that $|S_{uv} \cap U| \leq 2k$.*

Proof. Consider the collection $\mathcal{S} = \{S_{uv} \cap U : (u, v) \in F\}$. Let $X_1, X_2, \ldots, X_r$ be a sunflower in $\mathcal{S}$ with core K and let C_i be the cut C_{uv} such that $S_{uv} \cap U = X_i$ for $i \in [r]$. Pick a vertex $x_i \in X_i \setminus K$ for every $i \in [r]$. Then, the vertices s and x_i together with cuts C_i satisfy the prerequisities for the anti-isolation lemma. Consequently, $\mathcal{S}$ does not contain a sunflower of cardinality more than $g(k)$.

Hence by the sunflower lemma, it contains at most $d!(g(k))^d$ distinct sets of cardinality d, for each $1 \leq d \leq 2k$. Summing, and ignoring the constant factor, we find that $\mathcal{S}$ contains only $2^{\mathcal{O}(k^2)}$ sets of size at most $2k$. By the degree observation in the previous claim, every $u \in U$ is an endpoint for at most $g(k)$ arcs $(u, v) \in F$; hence there are at most $g(k)2^{\mathcal{O}(k^2)} = 2^{\mathcal{O}(k^2)}$ arcs $uv \in F$ such that $|S_{uv} \cap U| \leq 2k$, as promised. ⌐

The dual bound also holds.

Claim 4.9. *There are at most $2^{\mathcal{O}(k^2)}$ arcs $(u, v) \in F$ such that $|T_{uv} \cap W| \leq 2k$.*

Proof. This proof is identical to the previous one (except using the form Lemma 4.4 instead of the usual anti-isolation lemma). ⌐

Hence, if $|F|$ is large enough, there is at least one arc $(u, v) \in F$ such that $|S_{uv} \cap U|, |T_{uv} \cap W| > 2k$. Naturally, for every $u' \in S_{uv} \cap U$ there is a corresponding arc $(u', v') \in F$ and similarly for T_{uv}. Since C_{uv} contains at most k arcs, this leaves more than k further arcs of each type, and the cut C_{uv} fits the requirements of this lemma. This finishes the proof. □

From here, Theorem 1.3 follows easily. Recall that T is a proposed well-linked set. Since G is k-cut-minimal, every vertex $v \in T$ is incident to some arc that participates in a minimal st-cut of size at most k. This gives us a collection $F_T \subseteq A$ of arcs as in Lemma 4.6, with $|F_T| \geq |T|/2$. If $|F_T| > h(k)$, then the cut C provided by the lemma is a witness that T is not well-linked. Hence $|T|$ is bounded by $2^{\mathcal{O}(k^2)}$, and by Theorem 4.1 the directed treewidth of G is bounded by $f(k)$.

5 Conclusions

We have presented reductions showing $W[1]$- and nearly tight ETH-hardness of DIRECTED MULTICUT and STEINER ORIENTATION.

We would like to conclude with two open problems. The first one is the most natural one: is DIRECTED MULTICUT, parameterized by the size of the cutset, fixed-parameter tractable for *three* terminal pairs? It seems that our main gadget (the p-grid that we use to check an auxiliary binary relation) inherently requires four commodities to work: two that pass through it horizontally and vertically, and two additional for synchronizing the cutpoints of the input and output bidirectional paths. On the other hand, we were not able to extract any combinatorial property of the solutions in the three terminal case that would warrant an FPT algorithm. As a related question, we would also like to ask whether the two terminal case is $WK[1]$-hard parameterized by the cutset, or whether it admits a so-called polynomial Turing kernel (see [17]). It is known not to admit a polynomial kernel under standard complexity-theoretical assumptions [10], but nothing is known about Turing kernels (even conjecturally).

Secondly, we would like to repeat from [7] the question of fixed-parameter tractability of the ℓ-CHAIN SAT problem, parameterized the cutset. In this problem (with a fixed integer ℓ in the problem description) we are given an integer k and a set of n binary variables, unary constraints, and constraints of the form $(x_1 \Rightarrow x_2) \wedge (x_2 \Rightarrow x_3) \wedge \ldots \wedge (x_{\ell-1} \Rightarrow x_\ell)$; one asks to delete at most k constraints to get a satisfiable instance. It is easy to see that this problem can be cast as a cut problem in directed graphs via the natural implication/arc correspondence. While it is reasonable to suspect that a strong structural result for k-cut-minimal graphs may lead to an FPT algorithm for this problem, our lower bound methodology seems irrelevant, as it inherently requires different commodities.

References

[1] S. A. Amiri, L. Kaiser, S. Kreutzer, R. Rabinovich, and S. Siebertz. Graph searching games and width measures for directed graphs. In E. W. Mayr and N. Ollinger, editors, *32nd International Symposium on Theoretical Aspects of Computer Science, STACS 2015, March 4-7, 2015, Garching, Germany*, volume 30 of *LIPIcs*, pages 34–47. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2015.

[2] J. Bang-Jensen and G. Gutin. *Digraphs: Theory, Algorithms and Applications*. Springer, 2008.

[3] N. Bousquet, J. Daligault, and S. Thomassé. Multicut is FPT. In L. Fortnow and S. P. Vadhan, editors, *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 459–468. ACM, 2011.

[4] J. Chen, Y. Liu, and S. Lu. An improved parameterized algorithm for the minimum node multiway cut problem. *Algorithmica*, 55(1):1–13, 2009.

[5] J. Chen, Y. Liu, S. Lu, B. O'Sullivan, and I. Razgon. A fixed-parameter algorithm for the directed feedback vertex set problem. *J. ACM*, 55(5), 2008.

[6] R. H. Chitnis, M. Cygan, M. Hajiaghayi, M. Pilipczuk, and M. Pilipczuk. Designing FPT algorithms for cut problems using randomized contractions. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 460–469. IEEE Computer Society, 2012.

[7] R. H. Chitnis, L. Egri, and D. Marx. List H-coloring a graph by removing few vertices. In H. L. Bodlaender and G. F. Italiano, editors, *Algorithms - ESA 2013 - 21st Annual European Symposium, Sophia Antipolis, France, September 2-4, 2013. Proceedings*, volume 8125 of *Lecture Notes in Computer Science*, pages 313–324. Springer, 2013.

[8] R. H. Chitnis, M. Hajiaghayi, and D. Marx. Fixed-parameter tractability of directed multiway cut parameterized by the size of the cutset. *SIAM J. Comput.*, 42(4):1674–1696, 2013.

[9] M. Cygan, G. Kortsarz, and Z. Nutov. Steiner forest orientation problems. *SIAM J. Discrete Math.*, 27(3):1503–1513, 2013.

[10] M. Cygan, S. Kratsch, M. Pilipczuk, M. Pilipczuk, and M. Wahlström. Clique cover and graph separation: New incompressibility results. *TOCT*, 6(2):6, 2014.

[11] M. Cygan, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. Minimum bisection is fixed parameter tractable. In D. B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 323–332. ACM, 2014.

[12] M. Cygan, M. Pilipczuk, M. Pilipczuk, and J. O. Wojtaszczyk. On multiway cut parameterized above lower bounds. *TOCT*, 5(1):3, 2013.

[13] P. Erdős and R. Rado. Intersection theorems for systems of sets. *Journal of the London Mathematical Society*, s1-35(1):85–90, 1960.

[14] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[15] R. Ganian, P. Hlinený, J. Kneis, D. Meister, J. Obdrzálek, P. Rossmanith, and S. Sikdar. Are there any good digraph width measures? In V. Raman and S. Saurabh, editors, *Parameterized and Exact Computation - 5th International Symposium, IPEC 2010, Chennai, India, December 13-15, 2010. Proceedings*, volume 6478 of *Lecture Notes in Computer Science*, pages 135–146. Springer, 2010.

[16] R. Ganian, P. Hlinn, J. Kneis, A. Langer, J. Obdrlek, and P. Rossmanith. Digraph width measures in parameterized algorithmics. *Discrete Applied Mathematics*, 168(0):88 – 107, 2014. Fifth Workshop on Graph Classes, Optimization, and Width Parameters, Daejeon, Korea, October 2011.

[17] D. Hermelin, S. Kratsch, K. Soltys, M. Wahlström, and X. Wu. A completeness theory for polynomial (turing) kernelization. *Algorithmica*, 71(3):702–730, 2015.

[18] R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity? *J. Comput. Syst. Sci.*, 63(4):512–530, 2001.

[19] S. Kratsch, M. Pilipczuk, M. Pilipczuk, and M. Wahlström. Fixed-parameter tractability of multicut in directed acyclic graphs. *SIAM J. Discrete Math.*, 29(1):122–144, 2015.

[20] S. Kreutzer and S. Ordyniak. Width-measures for directed graphs and algorithmic applications. In M. Dehmer and F. Emmert-Streib, editors, *Quantitative Graph Theory: Mathematical Foundations and Applications*. Chapman and Hall/CRC Press, 2014.

[21] D. Lokshtanov, D. Marx, and S. Saurabh. Lower bounds based on the exponential time hypothesis. *Bulletin of the EATCS*, 105:41–72, 2011.

[22] D. Lokshtanov, N. S. Narayanaswamy, V. Raman, M. S. Ramanujan, and S. Saurabh. Faster parameterized algorithms using linear programming. *ACM Transactions on Algorithms*, 11(2):15:1–15:31, 2014.

[23] D. Marx. Parameterized graph separation problems. *Theor. Comput. Sci.*, 351(3):394–406, 2006.

[24] D. Marx. Can you beat treewidth? *Theory of Computing*, 6(1):85–112, 2010.

[25] D. Marx. Important separators and parameterized algorithms, 2011. Lecture slides; available at http://www.cs.bme.hu/~dmarx/papers/marx-mds-separators-slides.pdf.

[26] D. Marx. What's next? Future directions in parameterized complexity. In H. L. Bodlaender, R. Downey, F. V. Fomin, and D. Marx, editors, *The Multivariate Algorithmic Revolution and Beyond - Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday*, volume 7370 of *Lecture Notes in Computer Science*, pages 469–496. Springer, 2012.

[27] D. Marx, B. O'Sullivan, and I. Razgon. Finding small separators in linear time via treewidth reduction. *ACM Transactions on Algorithms*, 9(4):30, 2013.

[28] D. Marx and I. Razgon. Fixed-parameter tractability of multicut parameterized by the size of the cutset. *SIAM J. Comput.*, 43(2):355–388, 2014.

[29] M. Pilipczuk and M. Wahlström. Directed multicut is $W[1]$-hard, even for four terminal pairs. In R. Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1167–1178. SIAM, 2016.

[30] I. Razgon and B. O'Sullivan. Almost 2-SAT is fixed-parameter tractable. *J. Comput. Syst. Sci.*, 75(8):435–450, 2009.

[31] M. Wahlström. Half-integrality, LP-branching and FPT algorithms. In C. Chekuri, editor, *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1762–1781. SIAM, 2014.
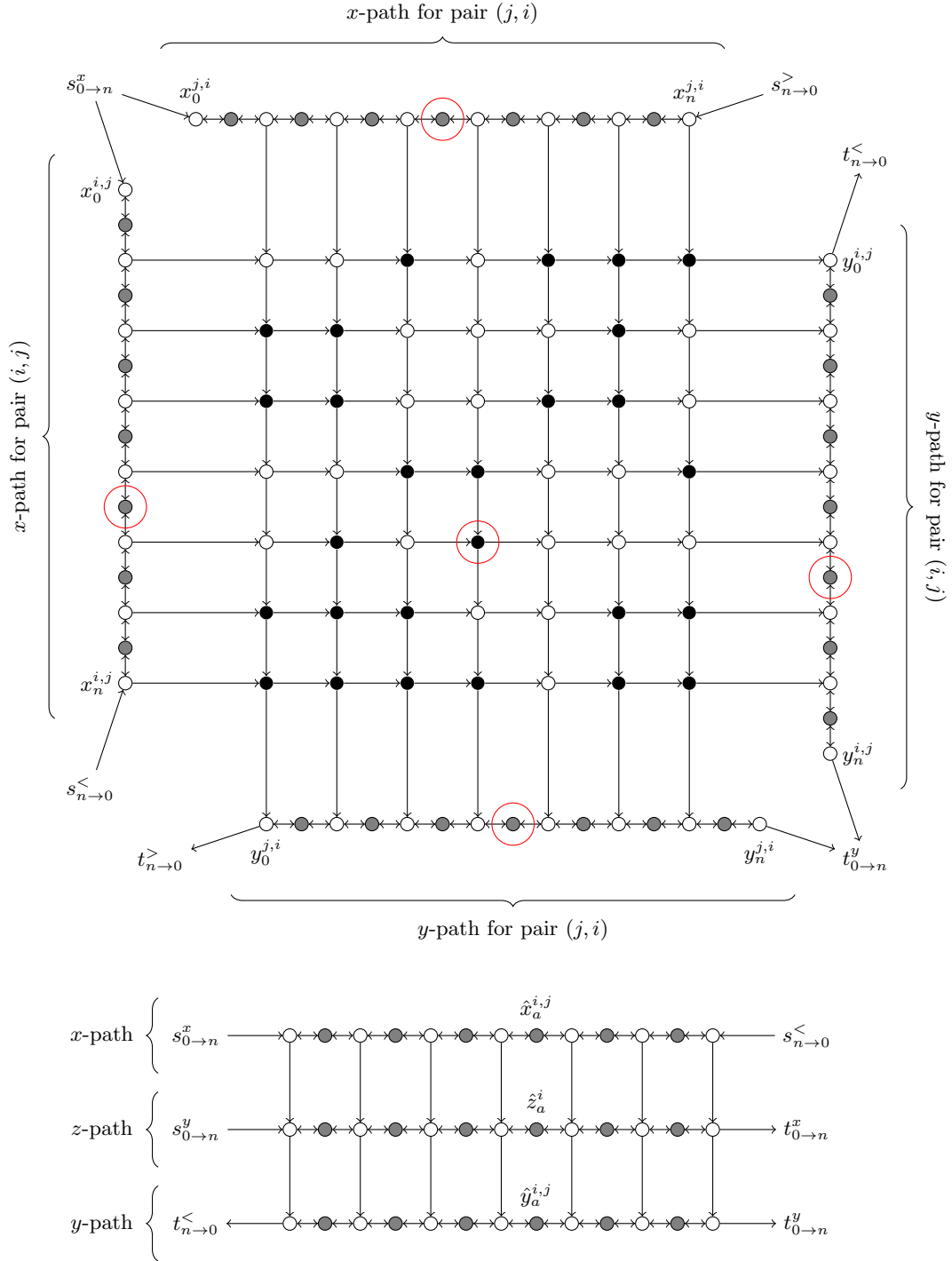
Figure 2: Illustration of the reduction for DIRECTED MULTICUT. Black vertices are light, gray are heavy, and white are undeletable. The top figure illustates a p-grid, together with an intended solution marked by red circles. Here, the vertex $p_{1,1}$ lies in the top-left corner of the grid, the first coordinate describes the row of the grid, and the second one the column. The bottom figure illustrates an x-, z-, and y-path for a pair (i, j) with $ij \in E(H)$ and $i < j$.
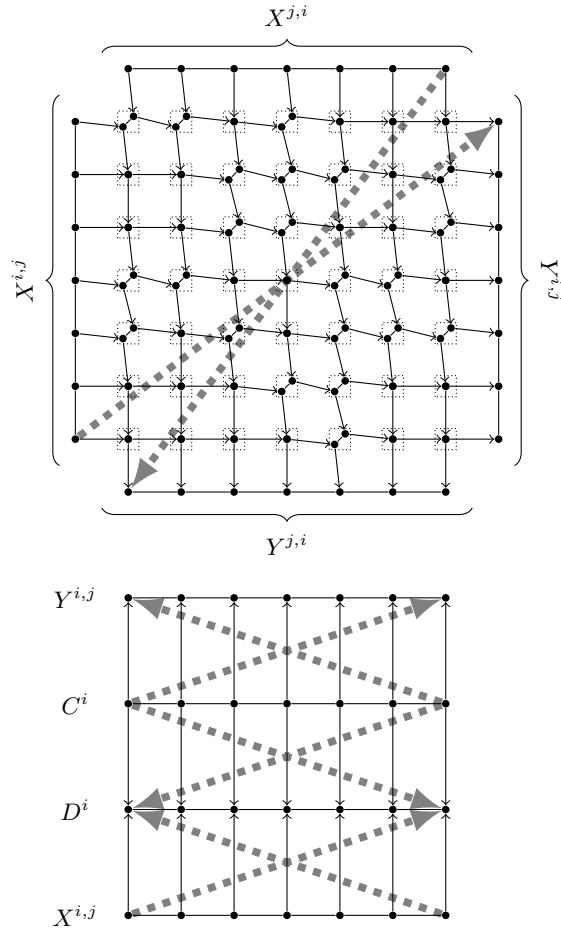
Figure 3: Illustration of the reduction for STEINER ORIENTATION. Thick gray arrows represent terminal pairs. The top figure illustates a p-grid; dashed rectangles represent single entities $p_{a,b}^{i,j}$. The bottom figure illustrates synchronization between paths $X^{i,j}$, $Y^{i,j}$, C^i, and D^i.