Parameterized algorithms for connectivity, separation, and modification problems in graphs

Summary of the PhD dissertation

Shaohua Li

Institute of Informatics, University of Warsaw

1 Introduction

Graph problems are ubiquitous in computer science. Graphs are one of the most natural models that represent the networks in real-life world and have numerous applications in different disciplines. Computer scientists are perusing faster algorithms to solve graph problems, both in practice and in theory. On the other side, there are many graph problems which are resistant to efficient algorithms. NP-completeness theory provides some clue on these problems [29, 74, 54]. If a problem is shown to be in the class of NP-complete problems, any efficient or polynomial-time algorithms for this problem imply that every NP-complete problem admits polynomial-time algorithms. In other words, there are probably no efficient algorithms for this problem. Although NP-hardness imply strong restrictions of algorithms for problems, people are still interested in how fast a problem can be solved and where the limitations of algorithms are. Exact algorithms for NP-hard problems focus mostly on reducing the exponential part of the running time as much as possible [49]. Approximation algorithms for NP-hard (optimization) problems aim to find efficient algorithms, classically polynomial-time algorithms at the cost of the optimality of the solution. Approximation algorithms try to find an *approximate solution* such that the distance between the approximate solution and the optimal solution is within a provable guarantee [114, 115].

Recently parameterized algorithms for NP-hard problems have received a lot of attention, which focus on both the input instance and the parameter. More formally, a *parameterized problem* is a language $L \subseteq \Sigma^* \times \mathbb{N}$, where Σ is a fixed finite alphabet. An input instance of a parameterized problem is of the form $(x, k) \in \Sigma^* \times \mathbb{N}$ and k is called the *parameter*. If a parameterized problem can be solved in time bounded by $f(k)|x|^c$, where $|x|$ is the size of the input instance, k is the parameter, $f : \mathbb{N} \to \mathbb{N}$ is a computable function, and c is a universal constant, then we say this problem is *fixed-parameter tractable* (FPT). If a parameterized problem can be solved in time bounded by $f(k)|x|^{f(k)}$, where $|x|$ is the size of the input instance, k is the parameter and $f : \mathbb{N} \to \mathbb{N}$ is a computable function, then we say this problem can be solved in XP time. A parameterized problem admits a *kernel* of size $g(k)$ for some computable function g if there is a polynomial-time procedure that reduces an arbitrary instance I of this problem with parameter k to an equivalent instance I' with size and parameter value bounded by $g(k)$. We refer to the following books for a deeper introduction to parameterized algorithms [41, 48, 30, 51].

In this thesis, we study a few graph problems, mostly concerning connectivity and separation in graphs.

2 Independent Feedback Vertex Set

The first part of this thesis is devoted to the Independent Feedback Vertex Set problem, which is a variant of the classic Feedback Vertex Set problem. Given a graph G, a feedback vertex set of G is a set of vertices $S \subseteq V(G)$ such that $G \setminus S$ is a forest. The Feedback Vertex Set problem (FVS) asks to find a feedback vertex set of the minimum size. This problem is a classic NP-hard problem which has been studied extensively in many fields of complexity and algorithms [1]. In the context of parameterized complexity of the Feedback Vertex Set problem, there is a long line of work improving the upper bound of the FPT algorithm for the standard parameterization of the solution size [15, 19, 21, 39, 40, 61, 72, 78, 68, 88] (i.e., the input consists of a graph G and a parameter k, and the goal is to find a feedback vertex set of size at most k or show that no such set exists). At the same time, many variants of Feedback Vertex Set received significant attention, including Subset FVS [36, 69, 95], Group FVS [33, 59, 69, 84], or Simultaneous FVS [104].

In this part, we focus on the parameterized version of the Independent Feedback Vertex Set problem (IFVS). The formal definition of this problem is as follows.

Independent Feedback Vertex Set (IFVS)
Input: An undirected graph G and an integer k.
Question: Is there a feedback vertex set S of size at most k such that no two vertices of S are adjacent in G.

Misra et al. gave the first FPT algorithm running in time $\mathcal{O}(5^k n^{\mathcal{O}(1)})$ and an $\mathcal{O}(k^3)$ kernel for IFVS [103]. Agrawal et al. presented an improved FPT algorithm running in time $\mathcal{O}^*(4.1481^k)$ for IFVS [3]. In this part, we propose a faster FPT algorithm.

Theorem 1. *The* Independent Feedback Vertex Set *problem, parameterized by the solution size, can be solved in* $\mathcal{O}^*((1+\varphi^2)^k) \leq \mathcal{O}^*(3.619^k)$ *time, where* $\varphi = \frac{1+\sqrt{5}}{2} < 1.619$ *is the golden ratio.*

We remark here that the exponential function of the time bound of Theorem 1 matches the one of the algorithm of Kociumaka and Pilipczuk [78] for the classic Feedback Vertex Set problem. Since Feedback Vertex Set trivially reduces to Independent Feedback Vertex Set (subdivide each edge once), any (deterministic) improvement to the base of the exponential function of Theorem 1 would give a similar improvement for Feedback Vertex Set. Although Iwata and Kobayashi already gave a faster FPT algorithm for Feedback Vertex Set problem [68], they use a totally different method which is involved in some sense. We believe it still makes sense if one can show an algorithm for Feedback Vertex Set which is faster than the algorithm of Kociumaka and Pilipczuk through some method different from the one of Iwata and Kobayashi.

On the technical side, we follow the standard approach of iterative compression as in [3] to reduce to a "disjoint" version of the problem. Here, our approach diverges from the one of [3]. We follow a modified measure for the subsequent branching process, somewhat inspired by the work of Kociumaka and Pilipczuk [78]. With a number of new notions (generalized W-degree, potential nice vertices and tents) and some new reduction rules, we get a clean branching algorithm for the "disjoint" version of the problem. This allows us to get an improved and also simplified algorithm for the Independent Feedback Vertex Set problem.

3 Multi-budgeted cut

The second part of this thesis is devoted to the Multi-budgeted cut problem and the multi-budgeted variants of Directed Feedback Arc Set and Skew Multicut. Graph separation problems are important topics in both theoretical area and applications. Although the famous minimum cut problem is known to be polynomial-time solvable, many well-known variants are NP-hard, which are intensively studied from the point of view of approximation [2, 20, 45, 56, 55, 73] and, what is more relevant here, parameterized complexity.

The notion of important separators, introduced by Marx [98], turned out to be fundamental for a number of graph separation problems such as Multiway Cut [98], Directed Feedback Vertex Set [22], or Almost 2-CNF SAT [108]. Further work, concerning mostly undirected graphs, resulted in a wide range of involved algorithmic techniques: applications of matroid techniques [86, 85], shadow removal [27, 101], randomized contractions [24], LP-guided branching [34, 60, 70, 66], and treewidth reduction [100], among others.

From the above techniques, only the notion of important separators and the related technique of shadow removal generalizes to directed graphs, giving FPT algorithms for Directed Feedback Arc Set [22], Directed Multiway Cut [27], and Directed Subset Feedback Vertex Set [26]. As a result, the parameterized complexity of a number of important graph separation problems in directed graphs remains open, and the quest to investigate them has been put on by Marx in a survey from 2012 [99]. Since the publication of this survey, two negative answers have been obtained. Pilipczuk and Wahlström showed that Directed Multicut is W[1]-hard even for four terminal pairs (leaving the case of three terminal pairs open) [106], while Lokshtanov et al. [96] showed intractability of Directed Odd Cycle Transversal.

Saurabh posed the question of parameterized complexity of a weighted variant of Directed Feedback Arc Set during an open problem session at Recent Advancements in Parameterized Complexity school (December 2017), where given a directed edge-weighted graph G, an integer k, and a target weight w, the goal is to find a set $X \subseteq E(G)$ such that $G - X$ is acyclic and X is of cardinality at most k and weight at most w. Consider a similar problem Weighted st-cut: given a directed graph G with positive edge weights and two distinguished vertices $s, t \in V(G)$, an integer k and a target weight w, decide if G admits an st-cut of cardinality at most k and weight at most w. The parameterized complexity of this problem parameterized by k is open even if G is restricted to be acyclic, while with this restriction the problem can easily be reduced to Directed Feedback Arc Set (add an arc (t, s) of prohibitively large weight).

The Weighted st-cut problem becomes similar to another directed graph cut problem, identified in [25], namely Chain ℓ-SAT. While this problem is originally formulated in CSP language, the graph formulation is as follows: given a directed graph G with a partition of edge set $E(G) = P_1 \uplus P_2 \uplus \ldots \uplus P_m$ such that each P_i is an edge set of a simple path of length at most ℓ, an integer k, and two vertices $s, t \in V(G)$, find an st-cut $C \subseteq E(G)$ such that $|\{i | C \cap P_i \neq \emptyset\}| \leq k$. This problem can easily be seen to be equivalent to minimum st-cut problem (and thus polynomial-time solvable) for $\ell \leq 2$, but is NP-hard for $\ell \geq 3$ and its parameterized complexity (with k as a parameter) remains an open problem.

Although the parameterized complexity of two aforementioned problems: weighted st-cut problem (in general digraphs, not necessary acyclic ones) and Chain ℓ-SAT are still open, we make some progress towards answering this question. We define a *multi-budgeted* variant of a number of cut problems (including the minimum cut problem) and show its fixed-parameter tractability. In this variant, the edges of the graph are colored with ℓ colors, and the input specifies separate budgets for each color. More formally, we primarily consider the following problem.

Multi-budgeted cut

Input: A directed graph G, two disjoint sets of vertices $X, Y \subseteq V(G)$, an integer ℓ, and for every $i \in \{1, 2, \ldots, \ell\}$ a set $E_i \subseteq E(G)$ and an integer k_i.

Question: Is there a set of arcs $C \subseteq \bigcup_{i=1}^{\ell} E_i$ such that there is no directed $X - Y$ path in $G \setminus C$ and for every $i \in [\ell]$, $|C \cap E_i| \leq k_i$.

We observe that Multi-budgeted cut for $\ell = 2$ reduces to Weighted st-cut as follows. Let $(G, X, Y, E_1, E_2, k_1, k_2)$ be a Multi-budgeted cut instance for $\ell = 2$. First, observe that we may assume that $E_1 \cap E_2 = \emptyset$, as we can replace every edge $e \in E_1 \cap E_2$ with two copies $e_1 \in E_1 \setminus E_2$ and $e_2 \in E_2 \setminus E_1$. Second, construct an equivalent Weighted st-cut instance (G', s, t, k, w) as follows. To construct G', first add two vertices s, t to G and edges $\{(s, x) | x \in X\}$ and $\{(y, t) | y \in Y\}$ of prohibitively large weight. Assign also prohibitively large weight to every edge $e \in E(G) \setminus (E_1 \cup E_2)$. Assign weight $(k_1 + 1)k_2 + 1$ to every edge $e \in E_1$. For every edge $e \in E_2$, add $k_1 + 1$ copies of e to G' of weight 1 each. Finally, set $k := (k_1 + 1) \cdot k_2 + k_1$ as the cardinality bound and $w := k_1((k_1 + 1)k_2 + 1) + (k_1 + 1)k_2$ as the target weight. The equivalence of the instances follows from the fact that the cardinality bound allows to pick in the solution at most k_2 bundles of $k_1 + 1$ copies of an edge of E_2, while the weight bound allows to pick only k_1 edges of E_1.

Thus, Multi-budgeted cut for $\ell = 2$ corresponds to the case of Weighted st-cut where the weights are integral and both target cardinality and weight are bounded in parameter.[1] This connection was our primary motivation to study the multi-budgeted variants of the cut problems.

Contrary to the classic minimum cut problem, we note that Multi-budgeted Cut becomes NP-hard for $\ell \geq 2$. We show that Multi-budgeted Cut is FPT when parameterized by $k = k_1 + ... + k_\ell$ and ℓ. For this problem, our branching strategy is as follows. A standard application of the Ford-Fulkerson algorithm gives a minimum XY-cut C of size λ and λ edge-disjoint $X - Y$ paths $P_1, P_2, \ldots, P_\lambda$. If C is a solution, then we are done. Similarly, if $\lambda > k$, then there is no solution. Otherwise, we branch which colors of the sought solution should appear on each paths P_j; that is, for every $i \in [\ell]$ and $j \in [\lambda]$, we guess if $P_j \cap E_i$ contains an edge of the sought solution, and in each guess assign infinite capacities to the edges of wrong color. If this change increased the size of a maximum flow from X to Y, then we can charge the branching step to this increase, as the size of the flow cannot exceed k. The critical insight is that if the size of the minimum flow does not increase (i.e., $P_1, \ldots, P_\lambda$ remains a maximum flow), then a corresponding minimum cut is necessarily a solution. As a result, we obtain the following.

Theorem 2. Multi-budgeted Cut *admits an FPT algorithm with running time bound* $\mathcal{O}(2^{k^2\ell} \cdot k \cdot (|V(G)| + |E(G)|))$ *where* $k = \sum_{i=1}^{\ell} k_i$.

The charging of the branching step to a flow increase appears also in the classic argument for bound of the number of important separators [22] (see also Chapter 8 of [31]). This motivates us to define multi-budgeted variants of Directed Feedback Arc Set and Skew Multicut.

The Directed Feedback Arc Set problem is a classic problem that played major role in the development of parameterized complexity. In this problem, given a directed graph G and an integer k, the problem is to decide if there exists an arc set S of size at most k such that $G - S$ has no directed cycles. In a similar way we define the problem Multi-budgeted Directed Feedback Arc Set as follows.

Multi-budgeted Directed Feedback Arc Set

Input: A directed graph G, an integer ℓ, and for every $i \in \{1, 2, \ldots, \ell\}$ a set $E_i \subseteq E(G)$ and an integer k_i.

Question: Is there a set of arcs $S \subseteq \bigcup_{i=1}^{\ell} E_i$ such that there is no directed cycle in $G - S$ and for every $i \in [\ell]$, $|S \cap E_i| \leq k_i$.

The first FPT algorithm for the Directed Feedback Arc Set problem is given by Chen et al. [22]. In their algorithm, they use iterative compression and reduce the Directed Feedback Arc Set compression problem to the Skew Edge Multicut problem. They propose a pushing lemma for Skew Edge Multicut and solve Skew Edge Multicut through enumerating important cuts. We show that for the multi-budgeted variant, a similar strategy enumerating multi-budgeted important cuts works. Formally, the Multi-budgeted Skew Edge Multicut problem is defined as follows.

Multi-budgeted Skew Edge Multicut

Input: A directed graph G, an integer ℓ, for every $i \in \{1, 2, \ldots, \ell\}$ a set $E_i \subseteq E(G)$ and an integer k_i, and a sequence $(s_i, t_i)_{i=1}^{q}$ of terminal pairs.

Question: Is there a set of arcs $C \subseteq \bigcup_{i=1}^{\ell} E_i$ such that there is no directed path from s_i to t_j for any $i \geq j$ in $G - C$ and for every $i \in [\ell]$, $|C \cap E(i)| \leq k_i$?

We observe that our branching algorithm can be merged with the classical procedure of enumerating important separators, yielding a bound (as a function of k and ℓ) and enumeration procedure of naturally defined multi-budgeted important separators. This in turn allows us to generalize our FPT algorithm to Multi-budgeted Skew Multicut and Multi-budgeted Directed Feedback Arc Set.

[1] For a reduction in the other direction, replace every arc e of weight $\omega(e)$ with one copy of color 1 and $\omega(e)$ copies of color 2, and set budgets $k_1 = k$ and $k_2 = w$.

Theorem 3. Multi-budgeted Skew Multicut *and* Multi-budgeted Directed Feedback Arc Set *admit FPT algorithms with running time bound* $2^{\mathcal{O}(k^3\ell\log(k\ell))}(|V(G)|+|E(G)|)$ *where* $k=\sum_{i=1}^{\ell}k_i$.

4 Two Disjoint Shortest Paths Problem with transition restrictions

The third part of this thesis is devoted to Two Disjoint Shortest Paths Problem on graphs with transition restrictions. Finding disjoint paths with specified endpoints in a given graph is a well-known problem in graph theory and combinatorial optimization. Given a graph $G=(V,E)$ and k vertex pairs $(s_1,t_1),...,(s_k,t_k)$, the k Disjoint Paths Problem (k-DPP) asks whether there exist k pairwise vertex-disjoint (or edge-disjoint) paths $P_1,....,P_k$ such that P_i starts from s_i and ends at t_i for $i=1,...,k$. If G is a digraph, k-DPP is NP-hard even when $k=2$ [52]. k-DPP is NP-complete if k is part of the input, even when G is a planar undirected graph [102]. Robertson and Seymour gave an $O(n^3)$-time algorithm for k-DPP in general undirected graphs for every constant k [109]. Later Kawarabayashi et al. gave an $O(n^2)$-time algorithm for the same problem [75]. Chudnovsky et al. showed that there is a polynomial time algorithm for k-Vertex-Disjoint Paths Problem for every fixed k if G is a semicomplete digraph [28]. Here a digraph is semicomplete if for all distinct vertices u,v, at least one of uv, vu is an edge.

Researchers also studied k-DPP from the view of parameterized complexity [32, 93, 110]. Cygan et al. gave an FPT algorithm parameterized by k with running time $2^{2^{O(k^2)}}\cdot n^{O(1)}$ for k-Vertex-Disjoint Paths Problem when G is a directed planar graph [32]. Given a tree decomposition of width at most w for the undirected graph G, k-DPP can be solved in time $2^{O(w\log w)}$ using dynamic programming techniques on tree decompositions [110], and Lokshtanov et al. showed that there is no $2^{o(w\log w)}$ time algorithm for k-DPP assuming ETH [93].

It is natural to generalize k-DPP to k-DSPP (k-Disjoint Shortest Paths Problem) with an exceptional requirement that every disjoint path is also a shortest one. More formally, given a directed graph $G=(V,E)$, a length function $w:E\to\mathbb{R}_{\geq 0}$ and k pairs of vertices $((s_1,t_1),...,(s_k,t_k)$ in G, the k-Disjoint Shortest Paths Problem asks to find k disjoint (vertex-disjoint or edge-disjoint) paths $P_1,....,P_k$ in G such that P_i is a shortest path from s_i to t_i for $i=1,...,k$. Eilam-Tzoreff showed that 2-DSPP in an undirected graph is polynomial-time solvable [42]. Bérczi and Kobayashi showed that 2-DSPP is NP-hard in general directed graph but polynomial-time solvable when every directed cycle has positive length [9].

In routing problems on graphs, we sometimes need to express constraints on the permitted walks that are stronger than what the standard graph model allows for. For example, in a road network, there can be a crossroad where drivers are not allowed to turn left. In this case, many walks in the underlying graph would denote routes that a driver is not allowed to use. To overcome this limitation, Kotzig introduced forbidden-transition graphs in [80]. In a directed graph G, a transition is an ordered pair of adjacent edges such that the head of the first edge is the tail of the second edge. A transition system T is a set of transitions in G. We say that a path P is T-compatible if every two consecutive edges of P form a transition of T. For notational clarity, it is sometimes useful to refer to the transitions $T(v)$ of a specific vertex $v\in V(G)$, that is, $T(v)=\{\{e_1,e_2\}\in T\mid \text{head}(e_1)=\text{tail}(e_2)=v\}$.

In this thesis we generalize the polynomial-time algorithm of Bérczi and Kobayashi to graphs with transition restrictions. Suppose that a prescribed transition system $T=\{T(v)\mid v\in V(G)\}$ is given, we study Directed Two Disjoint Shortest Paths Problem (2-DSPP) with transition restrictions. The formal definition is as follows.

Directed Two Disjoint Shortest Paths Problem (2-DSPP) with transition restrictions

Input: A directed graph $G=(V,E)$ with transition system T, a length function $w:E\to\mathbb{R}_{\geq 0}$ and two pairs of vertices (s_1,t_1),(s_2,t_2) in G.

Task: Find two disjoint (vertex-disjoint or edge-disjoint) paths P_1 and P_2 in G such that for both $i=1,2$, path P_i is a shortest path (even in the graph G with no transition restrictions) from s_i to t_i and P_i is also T-compatible.

We show that finding two vertex-disjoint (edge-disjoint) T-compatible paths P_1 and P_2 in a digraph G such that P_i is a shortest path (even in the graph G with no transition restrictions) from s_i to t_i for $i = 1, 2$ can be solved in polynomial time. Roughly speaking, we show that transition restrictions are not a barrier for using the same strategy as that in [9]. Formally, we show the following theorem.

Theorem 4. *If the length of every directed cycle is positive, both edge-disjoint and vertex-disjoint variants of* 2-DSPP WITH TRANSITION RESTRICTIONS *can be solved in polynomial time.*

Corollary 1. *If the length of every edge is positive, both edge-disjoint and vertex-disjoint variant of* 2-DSPP WITH TRANSITION RESTRICTIONS *can be solved in polynomial time.*

On the technical side of this algorithm, we basically follow the strategy of Bérczi and Kobayashi [9], which reduces the edge-disjoint case of 2-DSPP to finding a path in a graph $\mathcal{G}$ constructed from the input graph G. In the edge-disjoint case of 2-DSPP WITH TRANSITION RESTRICTIONS, we just need to delete edges of $\mathcal{G}$ which correspond to forbidden transitions of G with respect to T and it suffices to find the path in the remaining subgraph of $\mathcal{G}$. In graphs without transition restrictions, the vertex-disjoint case of 2-DSPP can be reduced to the edge-disjoint case of 2-DSPP. However, in graphs with transition restrictions, the situation is a bit different. By adding parallel edges, we suffices to keep the transition information. By a careful analysis, we show that the vertex-disjoint case of 2-DSPP WITH TRANSITION RESTRICTIONS can also be solved in polynomial time.

5 Cluster Editing parameterized above modification-disjoint P_3-packings

The fourth part of this thesis is devoted to CLUSTER EDITING PARAMETERIZED ABOVE MODIFICATION-DISJOINT P_3-PACKINGS. CORRELATION CLUSTERING is a well-known problem motivated by research in computational biology [8] and machine learning [5]. This problem aims at partitioning data points into groups or clusters according to their similarity. In this thesis, we study this problem from the view of graph theory. A graph H is called a *cluster graph* if H is a union of vertex-disjoint cliques. Given a graph $G = (V, E)$, the CLUSTER EDITING problem asks for a *cluster editing set* S such that $G \triangle S = (V, E \triangle S)$ is a cluster graph. Here $E \triangle S$ is the symmetric difference of E and S, i.e. $E \triangle S = (E \setminus S) \cup (S \setminus E)$. The optimization version of CLUSTER EDITING asks for a cluster editing set of minimum size, which is shown to be NP-hard [111]. Given a natural number k and a graph $G = (V, E)$, the parameterized version of CLUSTER EDITING asks if there exists a cluster editing set S such that $|S| \leq k$. A number of results were obtained for the parameterized version of CLUSTER EDITING and some of its variants [11, 13, 14, 16, 37, 46, 58, 62, 63, 79, 107, 50]. The current fastest FPT algorithm runs in time $O(1.62^k + n + m)$ [11] and it admits a kernel of $2k$ vertices [18, 23].

The interest in CLUSTER EDITING is not merely theoretical. Indeed, parameterized techniques are implemented in standard clustering tools [105, 116]. Although practitioners report that in particular the parameterized data-reduction techniques are effective [13, 12], the parameter k is not very small in several real-world data sets [10, 13, 113]. For instance, Böcker et al. [10, Table 2] considered 26 graphs derived from biological data with 91 to 100 vertices on which the average number of needed edits is 315, despite noting that the CLUSTER EDITING model outperformed other clustering models.

A technique to deal with such large parameters is *parameterization above lower bounds*. Herein, the parameter is of the form $\ell = k - h$ where h is a lower bound on the solution size, usually computable in polynomial time, and ℓ is the *excess* of the solution size above the lower bound. Research into parameterizations above lower bounds has been active and fruitful for several parameterized problems, not only on the theory-side (see [97, 35, 57, 94, 81], for example) but also in practice, as algorithms based on that approach yielded quite efficient implementations for VERTEX COVER [4] and among the most efficient ones for FEEDBACK VERTEX SET [67, 77]. For CLUSTER EDITING we are aware of only one research work considering parameterizations above lower bounds: Van Bevern, Froese, and Komusiewicz [113] studied edge-modification problems parameterized above the lower bound from packings of forbidden induced subgraphs and showed that CLUSTER EDITING parameterized by the excess above the size of a given packing of *vertex-disjoint*

P_3s is fixed-parameter tractable. Observe that a graph is a cluster graph if and only if it does not contain any P_3, a path on three vertices, as an induced subgraph. Consequently, one needs to perform at least one edge deletion or insertion per element of the packing.

As the P_3s in the above packing are vertex-disjoint, the value by which the packing can decrease the parameter is limited. In the previous example with 315 edits, subtracting the resulting lower bound would reduce the parameter by at most 33. In their conclusion, van Bevern et al. [113] asked whether CLUSTER EDITING is fixed-parameter tractable when parameterized above a stronger lower bound, the size of a modification-disjoint packing of P_3s. Here, a packing $\mathcal{H}$ of induced P_3s in G is *modification-disjoint* if every two P_3s in $\mathcal{H}$ do not share edges or non-edges, that is, they share at most one vertex. The formal problem definition is as follows.

> CLUSTER EDITING ABOVE MODIFICATION-DISJOINT P_3 PACKING (CEAMP)
>
> **Input:** A graph $G = (V, E)$, a packing $\mathcal{H}$ of modification-disjoint induced P_3s of G, and a non-negative integer ℓ.
>
> **Question:** Is there a cluster editing set, i.e. a set of vertex pairs $S \subseteq \binom{V}{2}$ so that $G \triangle S$ is a union of disjoint cliques, with $|S| - |\mathcal{H}| \leq \ell$?

We also say that a set S as above is a *solution*.

At Shonan Meeting no. 144 [71] Christian Komusiewicz re-iterated the question of van Bevern et al. [113] and it was also asked in Vincent Froese's dissertation [53]. In this thesis, we answer this question negatively by showing the following.

Theorem 5. CLUSTER EDITING ABOVE MODIFICATION-DISJOINT P_3 PACKING *is NP-hard even for* $\ell = 0$ *and when each vertex in the input graph is incident with at most 23* P_3*s of* $\mathcal{H}$.

In other words, given a graph G and a packing $\mathcal{H}$ of modification-disjoint P_3s in G, it is NP-hard to decide if one can delete or insert exactly one edge per element of $\mathcal{H}$ to obtain a cluster graph.

On the technical side, we reduce a 3-SAT instance Φ to an equivalent instance $(G, \mathcal{H}, 0)$ of CEAMP in polynomial time. The intuition of the reduction is to use "cliques" as building blocks and try to connect them by packed P_3s such that we can merge or separate these "cliques" by editing exactly one edge or non-edge for every packed P_3 (thus ensuring that $\ell = 0$). To be more precise, the building blocks are *proto-clusters*, which are connected components of the graph obtained by removing the edges of all packed P_3s. On the top level, we design a graph called *merging model*, which is a guide to show which clusters have the potential to be merged or separated. On the lower level, we need a number of tricks to "implement" this merging model, including some algebraic tricks to "pad" the proto-clusters and "P_3-repacking" tricks.

Our NP-hardness result implies that CEAMP is probably not FPT or even in XP unless $P = NP$. This motivates us to study a more restrictive variant of CEAMP in which every vertex is incident with **at most 2** packed P_3s. Call a modification-disjoint P_3 packing *two-restricted* if each vertex is in at most two packed P_3s. The problem CLUSTER EDITING ABOVE TWO-RESTRICTED MODIFICATION-DISJOINT P_3 PACKING (CEATMP) is defined in the same way as CEAMP except that the input packing $\mathcal{H}$ is two-restricted.

> CLUSTER EDITING ABOVE TWO-RESTRICTED MODIFICATION-DISJOINT P_3 PACKING (CEATMP)
>
> **Input:** A graph $G = (V, E)$, a packing $\mathcal{H}$ of modification-disjoint induced P_3s of G such that every vertex $v \in V(G)$ is incident with **at most 2** P_3s of $\mathcal{H}$, and a nonnegative integer ℓ.
>
> **Question:** Is there a cluster editing set, i.e. a set of vertex pairs $S \subseteq \binom{V}{2}$ so that $G \triangle S$ is a union of disjoint cliques, with $|S| - |\mathcal{H}| \leq \ell$?

It turns out that the complexity of the problem indeed drops when making the packing two-restricted.

Theorem 6. CLUSTER EDITING ABOVE TWO-RESTRICTED MODIFICATION-DISJOINT P_3 PACKING *can be solved in* $O(n^{2\ell+O(1)})$ *time.*

The main ingredient for the XP algorithm is the following theorem.

Theorem 7. Cluster Editing above two-restricted modification-disjoint P_3 packing *can be solved in polynomial time when* $\ell = 0$.

The basic idea for the polynomial-time algorithm in theorem 7 is as follows. First, we design a few reduction rules to reduce the size of the proto-clusters. Then we show that the reduced instance is equivalent to an instance of 2-SAT, which can be solved in polynomial time.

6 Hardness of Metric Dimension in Graphs of Constant Treewidth

The last part of this thesis is devoted to the metric dimension problem on graphs of constant treewidth. Let G be an unweighted and undirected graph and let $S \subseteq V(G)$. For a vertex $v \in V(G)$, the *distance vector* from v to S is the assignment $S \ni w \mapsto \mathrm{dist}_G(v, w)$, where dist_G denotes the distance in the graph G. The set S is *resolving* if a distance vector to S uniquely determines the source vertex; that is, no two vertices of G have the same distance vector to S. The Metric Dimension problem asks for a resolving set of minimum possible size; such a set is sometimes called the *metric basis* of G. The formal definition of the decision version of Metric Dimension is as follows.

Metric Dimension
Input: An undirected graph G and an integer k.
Question: Is there a resolving set $S \subseteq V(G)$ such that $|S| \leq k$?

Metric Dimension has already been introduced in 1970s [64, 112]. Determining its computational complexity turned out to be quite challenging. It is polynomial-time solvable on trees [64, 112, 76], outerplanar graphs [38], and chain graphs [47], but NP-hard for example on planar graphs [38] or split graphs [44]. From the parameterized complexity point of view, the FPT status of the Metric Dimension parameterized by the solution size has been open for a while and finally resolved in negative by Hartung and Nichterlein [65]. In the search of a tractable structural parameterization, FPT algorithms has been shown for parameters: treelength plus maximum degree [7], vertex cover number [65], max leaf number [43], and modular-width [7].

The above list misses probably the most important graph width measure, namely treewidth. Determining the complexity of Metric Dimension, parameterized by treewidth, remained elusive in the last years, and has been asked a few times [7, 38, 43]. Bonnet and Purohit in a paper presented at IPEC 2019 [17] showed that the problem is W[1]-hard, even with pathwidth parameterization. In this work we strengthed their result by proving para-NP-hardness of this parameterization.

Theorem 8. Metric Dimension*, restricted to graphs of treewidth at most* 24*, is NP-hard.*

Theorem 8 brings us much closer to closing (unfortunately mostly in negative) the question of the complexity of Metric Dimension in graphs of bounded treewidth. The remaining gap is to determine the exact treewidth value where the problem becomes hard: note that it is open if Metric Dimension is polynomial-time solvable on graphs of treewidth 2.

The proof of Theorem 8 starts with a construction of a graph with a separation of order 9 over which *a lot* of information on a partial solution to Metric Dimension is transferred. More formally, similarly as Bonnet and Purohit [17], we use the Multicolored Resolving Set problem as an auxiliary intermediate problem. In this problem, the input graph is additionally equipped with an integer k, a tuple of k disjoint vertex sets $X_1, X_2, \ldots, X_k$, and a set $\mathcal{P}$ of vertex pairs. The goal is to choose a set S consisting of exactly one vertex from each set X_i so that for every $\{u, v\} \in \mathcal{P}$, the pair $\{u, v\}$ is resolved by S, that is, u and v have different distance vectors to S. In our construction, the sets X_i are on one side of the said separation of order 9, while the pairs $\mathcal{P}$ are on the second side. The crux of the construction is to make every distance from a vertex of the separator to a chosen vertex of S count: despite the fact that the separation has constant size, S is of unbounded size, giving $\Omega(|S|)$ distances to work with. Overall, the above gives a relatively clean reduction giving NP-hardness of Multicolored Resolving Set in graphs of constant treewidth. Then, again similarly as in the work of Bonnet and Purohit [17], it takes a lot of effort to turn the above reduction to Multicolored Resolving Set into a reduction to Metric Dimension. While the toolbox remains almost the same as in [17], the application is different as the graph we are working with is significantly different.

7 Articles comprising this thesis

This thesis is based on the following articles and preprints:

- *An improved FPT algorithm for Independent Feedback Vertex Set*, which is a joint work with Marcin Pilipczuk, published at Theory Comput. Syst. 2020 [90]. The extended abstract of the publication was published in the 44th International Workshop on Graph-Theoretic Concepts in Computer Science, WG, 2018 [89].
- *Multi-budgeted Directed Cuts*, which is a joint work with Stefan Kratsch,Dániel Marx, Marcin Pilipczuk and Magnus Wahlström, published at Algorithmica, 2020 [83]. The extended abstract of the publication was published in 13th International Symposium on Parameterized and Exact Computation, IPEC, 2018 [82].
- *The Complexity of Connectivity Problems in Forbidden-Transition Graphs And Edge-Colored Graphs*, which is a joint work with Thomas Bellitto, Karolina Okrasa, Marcin Pilipczuk and Manuel Sorge, published at 31st International Symposium on Algorithms and Computation, ISAAC, 2020 [6].
- *Cluster Editing Parameterized Above Modification-Disjoint P_3-Packings*, which is a joint work with Marcin Pilipczuk and Manuel Sorge, published at 38th International Symposium on Theoretical Aspects of Computer Science, STACS, 2021 [92].
- *Hardness of Metric Dimension in Graphs of Constant Treewidth*, which is a joint work with Marcin Pilipczuk, CoRR, 2021 [91].

References

[1] *Encyclopedia of Optimization, Second Edition.* Springer, 2009.

[2] Amit Agarwal, Noga Alon, and Moses Charikar. Improved approximation for directed cut problems. In David S. Johnson and Uriel Feige, editors, *Proceedings of the 39th Annual ACM Symposium on Theory of Computing, San Diego, California, USA, June 11-13, 2007*, pages 671–680. ACM, 2007.

[3] Akanksha Agrawal, Sushmita Gupta, Saket Saurabh, and Roohani Sharma. Improved algorithms and combinatorial bounds for independent feedback vertex set. In *11th International Symposium on Parameterized and Exact Computation, IPEC 2016, August 24-26, 2016, Aarhus, Denmark*, volume 63 of *LIPIcs*, pages 2:1–2:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[4] Takuya Akiba and Yoichi Iwata. Branch-and-reduce exponential/FPT algorithms in practice: A case study of vertex cover. *Theoretical Computer Science*, 609:211–225, 2016.

[5] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine Learning*, 56(1-3):89–113, 2004.

[6] Thomas Bellitto, Shaohua Li, Karolina Okrasa, Marcin Pilipczuk, and Manuel Sorge. The complexity of connectivity problems in forbidden-transition graphs and edge-colored graphs. In Yixin Cao, Siu-Wing Cheng, and Minming Li, editors, *31st International Symposium on Algorithms and Computation, ISAAC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, volume 181 of *LIPIcs*, pages 59:1–59:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[7] Rémy Belmonte, Fedor V. Fomin, Petr A. Golovach, and M. S. Ramanujan. Metric dimension of bounded tree-length graphs. *SIAM J. Discret. Math.*, 31(2):1217–1243, 2017.

[8] Amir Ben-Dor, Ron Shamir, and Zohar Yakhini. Clustering gene expression patterns. *Journal of Computational Biology*, 6(3/4):281–297, 1999.

[9] Kristóf Bérczi and Yusuke Kobayashi. The directed disjoint shortest paths problem. In Kirk Pruhs and Christian Sohler, editors, *25th Annual European Symposium on Algorithms, ESA 2017, September 4-6, 2017, Vienna, Austria*, volume 87 of *LIPIcs*, pages 13:1–13:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[10] S. Böcker, S. Briesemeister, Q.B.A. Bui, and A. Truss. Going weighted: Parameterized algorithms for cluster editing. *Theoretical Computer Science*, 410(52):5467–5480, 2009.

[11] Sebastian Böcker. A golden ratio parameterized algorithm for cluster editing. *Journal of Discrete Algorithms*, 16:79–89, 2012.

[12] Sebastian Böcker and Jan Baumbach. Cluster editing. In Paola Bonizzoni, Vasco Brattka, and Benedikt Löwe, editors, *Proceedings of the 9th Conference on Computability in Europe (CiE 2013)*, volume 7921 of *Lecture Notes in Computer Science*, pages 33–44. Springer, 2013.

[13] Sebastian Böcker, Sebastian Briesemeister, and Gunnar W. Klau. Exact algorithms for cluster editing: Evaluation and experiments. *Algorithmica*, 60(2):316–334, 2011.

[14] Sebastian Böcker and Peter Damaschke. Even faster parameterized cluster deletion and cluster editing. *Information Processing Letters*, 111(14):717–721, 2011.

[15] Hans L. Bodlaender. On disjoint cycles. *Int. J. Found. Comput. Sci.*, 5(1):59–68, 1994.

[16] Hans L. Bodlaender, Michael R. Fellows, Pinar Heggernes, Federico Mancini, Charis Papadopoulos, and Frances A. Rosamond. Clustering with partial information. *Theoretical Computer Science*, 411(7-9):1202–1211, 2010.

[17] Édouard Bonnet and Nidhi Purohit. Metric dimension parameterized by treewidth. In Bart M. P. Jansen and Jan Arne Telle, editors, *14th International Symposium on Parameterized and Exact Computation, IPEC 2019, September 11-13, 2019, Munich, Germany*, volume 148 of *LIPIcs*, pages 5:1–5:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[18] Yixin Cao and Jianer Chen. Cluster editing: Kernelization based on edge cuts. *Algorithmica*, 64(1):152–169, 2012.

[19] Yixin Cao, Jianer Chen, and Yang Liu. On feedback vertex set new measure and new structures. In *Algorithm Theory - SWAT 2010, 12th Scandinavian Symposium and Workshops on Algorithm Theory, Bergen, Norway, June 21-23, 2010. Proceedings*, volume 6139 of *Lecture Notes in Computer Science*, pages 93–104. Springer, 2010.

[20] Chandra Chekuri, Sudipto Guha, and Joseph Naor. The steiner k-cut problem. *SIAM J. Discrete Math.*, 20(1):261–271, 2006.

[21] Jianer Chen, Fedor V. Fomin, Yang Liu, Songjian Lu, and Yngve Villanger. Improved algorithms for the feedback vertex set problems. In *Algorithms and Data Structures, 10th International Workshop, WADS 2007, Halifax, Canada, August 15-17, 2007, Proceedings*, volume 4619 of *Lecture Notes in Computer Science*, pages 422–433. Springer, 2007.

[22] Jianer Chen, Yang Liu, Songjian Lu, Barry O'Sullivan, and Igor Razgon. A fixed-parameter algorithm for the directed feedback vertex set problem. *J. ACM*, 55(5):21:1–21:19, 2008.

[23] Jianer Chen and Jie Meng. A $2k$ kernel for the cluster editing problem. *Journal of Computer and System Sciences*, 78(1):211–220, 2012.

[24] Rajesh Chitnis, Marek Cygan, MohammadTaghi Hajiaghayi, Marcin Pilipczuk, and Michal Pilipczuk. Designing FPT algorithms for cut problems using randomized contractions. *SIAM J. Comput.*, 45(4):1171–1229, 2016.

[25] Rajesh Chitnis, László Egri, and Dániel Marx. List h-coloring a graph by removing few vertices. *Algorithmica*, 78(1):110–146, 2017.

[26] Rajesh Hemant Chitnis, Marek Cygan, Mohammad Taghi Hajiaghayi, and Dániel Marx. Directed subset feedback vertex set is fixed-parameter tractable. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *ICALP (1)*, volume 7391 of *Lecture Notes in Computer Science*, pages 230–241. Springer, 2012.

[27] Rajesh Hemant Chitnis, MohammadTaghi Hajiaghayi, and Dániel Marx. Fixed-parameter tractability of directed multiway cut parameterized by the size of the cutset. *SIAM J. Comput.*, 42(4):1674–1696, 2013.

[28] Maria Chudnovsky, Alex Scott, and Paul Seymour. Disjoint paths in tournaments. *Advances in Mathematics*, 270:582–597, 2015.

[29] Stephen A. Cook. The complexity of theorem-proving procedures. In Michael A. Harrison, Ranan B. Banerji, and Jeffrey D. Ullman, editors, *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA*, pages 151–158. ACM, 1971.

[30] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[31] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[32] Marek Cygan, Dániel Marx, Marcin Pilipczuk, and Michal Pilipczuk. The planar directed k-vertex-disjoint paths problem is fixed-parameter tractable. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 197–206. IEEE Computer Society, 2013.

[33] Marek Cygan, Marcin Pilipczuk, and Michal Pilipczuk. On group feedback vertex set parameterized by the size of the cutset. *Algorithmica*, 74(2):630–642, 2016.

[34] Marek Cygan, Marcin Pilipczuk, Michal Pilipczuk, and Jakub Onufry Wojtaszczyk. On multiway cut parameterized above lower bounds. *TOCT*, 5(1):3, 2013.

[35] Marek Cygan, Marcin Pilipczuk, Michal Pilipczuk, and Jakub Onufry Wojtaszczyk. On multiway cut parameterized above lower bounds. *ACM Transactions on Computation Theory*, 5(1):3:1–3:11, 2013.

[36] Marek Cygan, Marcin Pilipczuk, Michal Pilipczuk, and Jakub Onufry Wojtaszczyk. Subset feedback vertex set is fixed-parameter tractable. *SIAM J. Discrete Math.*, 27(1):290–309, 2013.

[37] Peter Damaschke. Fixed-parameter enumerability of cluster editing and related problems. *Theory of Computing Systems*, 46(2):261–283, 2010.

[38] Josep Díaz, Olli Pottonen, Maria J. Serna, and Erik Jan van Leeuwen. Complexity of metric dimension on planar graphs. *J. Comput. Syst. Sci.*, 83(1):132–158, 2017.

[39] Rodney G. Downey and Michael R. Fellows. Fixed parameter tractability and completeness. In *Complexity Theory: Current Research, Dagstuhl Workshop, February 2-8, 1992*, pages 191–225. Cambridge University Press, 1992.

[40] Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity.* Monographs in Computer Science. Springer, 1999.

[41] Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity.* Texts in Computer Science. Springer, 2013.

[42] Tali Eilam-Tzoreff. The disjoint shortest paths problem. *Discret. Appl. Math.*, 85(2):113–138, 1998.

[43] David Eppstein. Metric dimension parameterized by max leaf number. *J. Graph Algorithms Appl.*, 19(1):313–323, 2015.

[44] Leah Epstein, Asaf Levin, and Gerhard J. Woeginger. The (weighted) metric dimension of graphs: Hard and easy cases. *Algorithmica*, 72(4):1130–1171, 2015.

[45] Guy Even, Joseph Naor, Baruch Schieber, and Madhu Sudan. Approximating minimum feedback sets and multicuts in directed graphs. *Algorithmica*, 20(2):151–174, 1998.

[46] Michael R. Fellows, Jiong Guo, Christian Komusiewicz, Rolf Niedermeier, and Johannes Uhlmann. Graph-based data clustering with overlaps. *Discrete Optimization*, 8(1):2–17, 2011.

[47] Henning Fernau, Pinar Heggernes, Pim van 't Hof, Daniel Meister, and Reza Saei. Computing the metric dimension for chain graphs. *Inf. Process. Lett.*, 115(9):671–676, 2015.

[48] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006.

[49] Fedor V. Fomin and Dieter Kratsch. *Exact Exponential Algorithms.* Texts in Theoretical Computer Science. An EATCS Series. Springer, 2010.

[50] Fedor V. Fomin, Stefan Kratsch, Marcin Pilipczuk, Michal Pilipczuk, and Yngve Villanger. Subexponential fixed-parameter tractability of cluster editing. *CoRR*, abs/1112.4419, 2011.

[51] Fedor V Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization: theory of parameterized preprocessing.* Cambridge University Press, 2019.

[52] Steven Fortune, John Hopcroft, and James Wyllie. The directed subgraph homeomorphism problem. *Theoretical Computer Science*, 10(2):111–121, 1980.

[53] Vincent Froese. *Fine-Grained Complexity Analysis of Some Combinatorial Data Science Problems.* PhD thesis, Technische Universität Berlin, 2018.

[54] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

[55] Naveen Garg, Vijay V. Vazirani, and Mihalis Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM J. Comput.*, 25(2):235–251, 1996.

[56] Naveen Garg, Vijay V. Vazirani, and Mihalis Yannakakis. Multiway cuts in node weighted graphs. *J. Algorithms*, 50(1):49–61, 2004.

[57] Shivam Garg and Geevarghese Philip. Raising the bar for vertex cover: Fixed-parameter tractability above A higher guarantee. In Krauthgamer [87], pages 1152–1166.

[58] Jens Gramm, Jiong Guo, Falk Hüffner, and Rolf Niedermeier. Graph-modeled data clustering: Exact algorithms for clique generation. *Theory of Computing Systems*, 38(4):373–392, 2005.

[59] Sylvain Guillemot. FPT algorithms for path-transversal and cycle-transversal problems. *Discrete Optimization*, 8(1):61–71, 2011.

[60] Sylvain Guillemot. FPT algorithms for path-transversal and cycle-transversal problems. *Discrete Optimization*, 8(1):61–71, 2011.

[61] Jiong Guo, Jens Gramm, Falk Hüffner, Rolf Niedermeier, and Sebastian Wernicke. Compression-based fixed-parameter algorithms for feedback vertex set and edge bipartization. *J. Comput. Syst. Sci.*, 72(8):1386–1396, 2006.

[62] Jiong Guo, Iyad A. Kanj, Christian Komusiewicz, and Johannes Uhlmann. Editing graphs into disjoint unions of dense clusters. *Algorithmica*, 61(4):949–970, 2011.

[63] Jiong Guo, Christian Komusiewicz, Rolf Niedermeier, and Johannes Uhlmann. A more relaxed model for graph-based data clustering: s-plex cluster editing. *SIAM Journal on Discrete Mathematics*, 24(4):1662–1683, 2010.

[64] Frank Harary and Robert A Melter. On the metric dimension of a graph. *Ars Combin*, 2(1):191–195, 1976.

[65] Sepp Hartung and André Nichterlein. On the parameterized and approximation hardness of metric dimension. In *Proceedings of the 28th Conference on Computational Complexity, CCC 2013, K.lo Alto, California, USA, 5-7 June, 2013*, pages 266–276. IEEE Computer Society, 2013.

[66] Yoichi Iwata. Linear-time kernelization for feedback vertex set. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80 of *LIPIcs*, pages 68:1–68:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[67] Yoichi Iwata. Linear-time kernelization for feedback vertex set. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *Proceedings of the 44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, volume 80 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 68:1–68:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017.

[68] Yoichi Iwata and Yusuke Kobayashi. Improved analysis of highest-degree branching for feedback vertex set. In Bart M. P. Jansen and Jan Arne Telle, editors, *14th International Symposium on Parameterized and Exact Computation, IPEC 2019, September 11-13, 2019, Munich, Germany*, volume 148 of *LIPIcs*, pages 22:1–22:11. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[69] Yoichi Iwata, Magnus Wahlström, and Yuichi Yoshida. Half-integrality, lp-branching, and FPT algorithms. *SIAM J. Comput.*, 45(4):1377–1411, 2016.

[70] Yoichi Iwata, Magnus Wahlström, and Yuichi Yoshida. Half-integrality, LP-branching, and FPT algorithms. *SIAM J. Comput.*, 45(4):1377–1411, 2016.

[71] Bart M.P. Jansen, Christian Schulz, and Hisao Tamaki. NII shonan meeting report no. 144 parameterized graph algorithms and data reduction, 2019.

[72] Iyad A. Kanj, Michael J. Pelsmajer, and Marcus Schaefer. Parameterized algorithms for feedback vertex set. In *Parameterized and Exact Computation, First International Workshop, IWPEC 2004, Bergen, Norway, September 14-17, 2004, Proceedings*, volume 3162 of *Lecture Notes in Computer Science*, pages 235–247. Springer, 2004.

[73] David R. Karger, Philip N. Klein, Clifford Stein, Mikkel Thorup, and Neal E. Young. Rounding algorithms for a geometric embedding of minimum multiway cut. *Math. Oper. Res.*, 29(3):436–461, 2004.

[74] Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972.

[75] Ken-ichi Kawarabayashi, Yusuke Kobayashi, and Bruce Reed. The disjoint paths problem in quadratic time. *Journal of Combinatorial Theory, Series B*, 102(2):424–435, 2012.

[76] Samir Khuller, Balaji Raghavachari, and Azriel Rosenfeld. Landmarks in graphs. *Discret. Appl. Math.*, 70(3):217–229, 1996.

[77] Krzysztof Kiljan and Marcin Pilipczuk. Experimental evaluation of parameterized algorithms for feedback vertex set. In Gianlorenzo D'Angelo, editor, *Proceedings of the 17th International Symposium on Experimental Algorithms (SEA 2018)*, volume 103 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 12:1–12:12, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[78] Tomasz Kociumaka and Marcin Pilipczuk. Faster deterministic feedback vertex set. *Inf. Process. Lett.*, 114(10):556–560, 2014.

[79] Christian Komusiewicz and Johannes Uhlmann. Alternative parameterizations for cluster editing. In Ivana Cerná, Tibor Gyimóthy, Juraj Hromkovic, Keith G. Jeffery, Rastislav Královic, Marko Vukolic, and Stefan Wolf, editors, *SOFSEM 2011: Theory and Practice of Computer Science - 37th Conference on Current Trends in Theory and Practice of Computer Science, Nový Smokovec, Slovakia, January 22-28, 2011. Proceedings*, volume 6543 of *Lecture Notes in Computer Science*, pages 344–355. Springer, 2011.

[80] Anton Kotzig. Moves without forbidden transitions in a graph. *Matematický časopis*, 18(1):76–80, 1968.

[81] Stefan Kratsch. A randomized polynomial kernelization for vertex cover with a smaller parameter. *SIAM Journal on Discrete Mathematics*, 32(3):1806–1839, 2018.

[82] Stefan Kratsch, Shaohua Li, Dániel Marx, Marcin Pilipczuk, and Magnus Wahlström. Multi-budgeted directed cuts. In Christophe Paul and Michal Pilipczuk, editors, *13th International Symposium on Parameterized and Exact Computation, IPEC 2018, August 20-24, 2018, Helsinki, Finland*, volume 115 of *LIPIcs*, pages 18:1–18:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[83] Stefan Kratsch, Shaohua Li, Dániel Marx, Marcin Pilipczuk, and Magnus Wahlström. Multi-budgeted directed cuts. *Algorithmica*, 82(8):2135–2155, 2020.

[84] Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 450–459. IEEE Computer Society, 2012.

[85] Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 450–459. IEEE Computer Society, 2012.

[86] Stefan Kratsch and Magnus Wahlström. Compression via matroids: A randomized polynomial kernel for odd cycle transversal. *ACM Transactions on Algorithms*, 10(4):20, 2014.

[87] Robert Krauthgamer, editor. *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*. SIAM, 2016.

[88] Jason Li and Jesper Nederlof. Detecting feedback vertex sets of size k in $O^*(2.7^k)$ time. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 971–989. SIAM, 2020.

[89] Shaohua Li and Marcin Pilipczuk. An improved FPT algorithm for independent feedback vertex set. In Andreas Brandstädt, Ekkehard Köhler, and Klaus Meer, editors, *Graph-Theoretic Concepts in Computer Science - 44th International Workshop, WG 2018, Cottbus, Germany, June 27-29, 2018, Proceedings*, volume 11159 of *Lecture Notes in Computer Science*, pages 344–355. Springer, 2018.

[90] Shaohua Li and Marcin Pilipczuk. An improved FPT algorithm for independent feedback vertex set. *Theory Comput. Syst.*, 64(8):1317–1330, 2020.

[91] Shaohua Li and Marcin Pilipczuk. Hardness of metric dimension in graphs of constant treewidth. *CoRR*, abs/2102.09791, 2021.

[92] Shaohua Li, Marcin Pilipczuk, and Manuel Sorge. Cluster editing parameterized above modification-disjoint p_3-packings. In Markus Bläser and Benjamin Monmege, editors, *38th International Symposium on Theoretical Aspects of Computer Science, STACS 2021, March 16-19, 2021, Saarbrücken, Germany (Virtual Conference)*, volume 187 of *LIPIcs*, pages 49:1–49:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[93] Daniel Lokshtanov, Dániel Marx, and Saket Saurabh. Slightly superexponential parameterized problems. *SIAM J. Comput.*, 47(3):675–702, 2018.

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

[95] Daniel Lokshtanov, M. S. Ramanujan, and Saket Saurabh. Linear time parameterized algorithms for subset feedback vertex set. *ACM Trans. Algorithms*, 14(1):7:1–7:37, 2018.

[96] Daniel Lokshtanov, M. S. Ramanujan, Saket Saurabh, and Meirav Zehavi. Parameterized complexity and approximability of directed odd cycle transversal. *CoRR*, abs/1704.04249, 2017.

[97] Meena Mahajan and Venkatesh Raman. Parameterizing above guaranteed values: Maxsat and maxcut. *Journal of Algorithms*, 31(2):335–354, 1999.

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

[99] Dániel Marx. What's next? Future directions in parameterized complexity. In Hans L. Bodlaender, Rod Downey, Fedor V. Fomin, and Dániel 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.

[100] Dániel Marx, Barry O'Sullivan, and Igor Razgon. Finding small separators in linear time via treewidth reduction. *ACM Transactions on Algorithms*, 9(4):30, 2013.

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

[102] Matthias Middendorf and Frank Pfeiffer. On the complexity of the disjoint paths problem. *Combinatorica*, 13(1):97–107, 1993.

[103] Neeldhara Misra, Geevarghese Philip, Venkatesh Raman, and Saket Saurabh. On parameterized independent feedback vertex set. *Theor. Comput. Sci.*, 461:65–75, 2012.

[104] Neeldhara Misra, Geevarghese Philip, Venkatesh Raman, Saket Saurabh, and Somnath Sikdar. FPT algorithms for connected feedback vertex set. *J. Comb. Optim.*, 24(2):131–146, 2012.

[105] John H. Morris, Leonard Apeltsin, Aaron M. Newman, Jan Baumbach, Tobias Wittkop, Gang Su, Gary D. Bader, and Thomas E. Ferrin. clusterMaker: a multi-algorithm clustering plugin for Cytoscape. *BMC Bioinformatics*, 12(1):436, 2011.

[106] Marcin Pilipczuk and Magnus Wahlström. Directed multicut is $W[1]$-hard, even for four terminal pairs. In Krauthgamer [87], pages 1167–1178.

[107] Fábio Protti, Maise Dantas da Silva, and Jayme Luiz Szwarcfiter. Applying modular decomposition to parameterized cluster editing problems. *Theory of Computing Systems*, 44(1):91–104, 2009.

[108] Igor Razgon and Barry O'Sullivan. Almost 2-sat is fixed-parameter tractable. *J. Comput. Syst. Sci.*, 75(8):435–450, 2009.

[109] Neil Robertson and Paul D Seymour. Graph minors. xiii. the disjoint paths problem. *Journal of combinatorial theory, Series B*, 63(1):65–110, 1995.

[110] Petra Scheffler. *A practical linear time algorithm for disjoint paths in graphs with bounded tree-width.* TU, Fachbereich 3, 1994.

[111] Ron Shamir, Roded Sharan, and Dekel Tsur. Cluster graph modification problems. *Discrete Applied Mathematics*, 144(1-2):173–182, 2004.

[112] Peter J Slater. Leaves of trees. *Congr. Numer*, 14(37):549–559, 1975.

[113] René van Bevern, Vincent Froese, and Christian Komusiewicz. Parameterizing edge modification problems above lower bounds. *Theory of Computing Systems*, 62(3):739–770, 2018.

[114] Vijay V Vazirani. *Approximation algorithms.* Springer Science & Business Media, 2013.

[115] David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms.* Cambridge University Press, 2011.

[116] Tobias Wittkop, Dorothea Emig, Sita Lange, Sven Rahmann, Mario Albrecht, John H. Morris, Sebastian Böcker, Jens Stoye, and Jan Baumbach. Partitioning biological data with transitivity clustering. *Nature Methods*, 7(6):419–420, 2010.