



HAL
open science

Kernelization for Graph Packing Problems via Rainbow Matching

Stéphane Bessy, Marin Bougeret, Dimitrios M. Thilikos, Sebastian Wiederrecht

► **To cite this version:**

Stéphane Bessy, Marin Bougeret, Dimitrios M. Thilikos, Sebastian Wiederrecht. Kernelization for Graph Packing Problems via Rainbow Matching. SODA 2023 - 34th ACM/SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, Jan 2023, Florence, Italy. pp.3654-3663, 10.1137/1.9781611977554.ch139 . hal-04042995

HAL Id: hal-04042995

<https://hal.science/hal-04042995v1>

Submitted on 17 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Kernelization for Graph Packing and Hitting Problems via Rainbow Matching

Stéphane Bessy^{1,2} Marin Bougeret¹ Dimitrios M. Thilikos^{1,3}

Sebastian Wiederrecht^{4,5}

Abstract

We introduce a new kernelization tool, called *rainbow matching technique*, that is appropriate for the design of polynomial kernels for packing problems and their hitting counterparts. Our technique capitalizes on the powerful combinatorial results of [Graf, Harris, Haxell, SODA 2021]. We apply the rainbow matching technique on four (di)graph packing or hitting problems, namely the TRIANGLE-PACKING IN TOURNAMENT problem (TPT), where we ask for a packing of k directed triangles in a tournament, DIRECTED FEEDBACK VERTEX SET IN TOURNAMENT problem (FVST), where we ask for a (hitting) set of at most k vertices which intersects all triangles of a tournament, the INDUCED 2-PATH-PACKING (I2PP) where we ask for a packing of k induced paths of length two in a graph and INDUCED 2-PATH HITTING SET problem (I2PHS), where we ask for a (hitting) set of at most k vertices which intersects all induced paths of length two in a graph. The existence of a sub-quadratic kernels for these problems was proven for the first time in [Fomin, Le, Lokshtanov, Saurabh, Thomassé, Zehavi. *ACM Trans. Algorithms*, 2019], where they gave a kernel of $\mathcal{O}(k^{3/2})$ vertices for the two first problems and $\mathcal{O}(k^{5/3})$ vertices for the two last. In the same paper it was questioned whether these bounds can be (optimally) improved to linear ones. Motivated by this question, we apply the rainbow matching technique and prove that TPT and FVST admit (almost linear) kernels of $k^{1+\frac{\mathcal{O}(1)}{\sqrt{\log k}}}$ vertices and that I2PP and I2PHS admit kernels of $\mathcal{O}(k)$ vertices.

Keywords: Kernelization, Parameterized algorithms, Rainbow matching, Graph packing problems, Graph covering problems, Graph hitting problems.

¹LIRMM, Univ de Montpellier, CNRS, Montpellier, France.

²Supported by the ANR project DIGRAPHS (ANR-19-CE48-0013).

³Supported by the ANR projects ESIGMA (ANR-17-CE23-0010) and the French-German Collaboration ANR/DFG Project UTMA (ANR-20-CE92-0027).

⁴Discrete Mathematics Group, Institute for Basic Science, Daejeon, South Korea.

⁵The research of Sebastian Wiederrecht was supported by the ANR project ESIGMA (ANR-17-CE23-0010) and the Institute for Basic Science (IBS-R029-C1).

Contents

1	Introduction	1
2	Definitions	4
2.1	Basic concepts	4
2.2	Tools about rainbow matchings	5
3	The rainbow matching technique	7
3.1	Overview of the kernelization algorithm	7
3.2	Origin of the invariants in a partial decomposition.	7
3.3	Invariants in a partial decomposition	9
3.4	Description of one round of the kernel	9
3.5	Applicability of the rainbow coloring technique	11
4	Linear kernel for I2PP	11
4.1	Notation	11
4.2	Preliminary phase: greedy localization	11
4.3	Nice pair, buckets, partial decomposition, and auxiliary graph	12
4.4	Analysis of the two cases: rainbow matching or small vertex cover	15
4.5	Analysis of the overall kernel	19
5	Linear kernel for I2PHS	20
6	An (almost) linear kernel for TPT	22
6.1	Notations	22
6.2	Preliminary phase: greedy localization	23
6.3	Nice pairs, buckets, and partial decomposition	23
6.4	Intervals: demand definition and basic properties	27
6.5	The small total demand property	30
6.6	Auxiliary graph and bucket allocation	34
6.7	Operations on buckets	38
6.8	Analysis the two cases: rainbow matching or small vertex cover	44
6.9	Analysis of the overall kernel	47
7	Almost linear kernel for FVST	49
8	Conclusion	51

1 Introduction

A *parameterized problem* Π is a subset of $\Sigma^* \times \mathbb{N}$ for some finite alphabet Σ . Given an instance (x, k) of parameterized problem Π we typically refer to $|x|$ as the *size* of the problem and to k as the *parameter* of the problem and the general question of parameterized computation is whether Π admits an algorithm able to decide, given an instance $(x, k) \in \Sigma^* \times \mathbb{N}$, whether $(x, k) \in \Pi$ in $f(k) \cdot |x|^{\mathcal{O}(1)}$ time. When this is indeed possible, then we say that Π is *Fixed Parameter Tractable* or, in short FPT. Parameterized computation was introduced by Downey and Fellows in their pioneering work in [1, 15–18] and currently constitutes a fully developed discipline of Theoretical Computer Science (see [10, 19, 21, 38] for textbooks).

Kernelization algorithms. A particularly vibrant field of parameterized computation is *kernelization*. A *kernelization algorithm* for a parameterized problem $\Pi \subseteq \Sigma^* \times \mathbb{N}$ is a polynomial algorithm able to reduce every instance $(x, k) \in \Sigma^* \times \mathbb{N}$ to an *equivalent one* whose size depends *exclusively* on the parameter k . If this size is bounded by a (polynomial) function $f(k)$, then we say that Π admits a (polynomial) kernel of size $f(k)$. The design of kernelization algorithms for parameterized problems has been a prominent topic of parameterized computation, mainly because it can be seen as a way to formalize preprocessing: when some parameterization of an NP-hard problem admits a kernel of size $f(k)$, then we may solve it by first applying, as a preprocessing step, the corresponding kernelization algorithm and then apply brute force techniques on a problem instance where the size $|x|$ of the problem has been radically reduced. Clearly, for small values of k , this approach becomes particularly promising, especially when the problem in question admits a polynomial kernel. Unfortunately, not all parameterized problems are amenable to polynomial kernels and an extensive theory of kernelization has been developed so to either provide algorithmic techniques for the derivation of polynomial kernels (see e.g., [24, 28, 34, 35]) or to develop complexity-theoretic lower bound tools for kernelization [6, 12, 14, 20, 25, 31, 32] (see [23] for a dedicated textbook).

Greedy localization. A wide family of parameterized problems that have extensively studied from the kernelization point of view are *(di)graph packing problems*. Such problems are defined on graphs or digraphs and, typically, the question is whether an input (di)graph G contains some collection of k pairwise disjoint copies of some fixed (induced) sub(di)graph H . A general approach for such problem is the *greedy localization technique* (see [11, 39]). This technique consists in finding, using some greedy approach, a maximal collection \mathcal{C} of pairwise disjoint copies of H in G . If \mathcal{C} has already at least k elements, we may safely output a positive answer to the problem. If not, then we know that the $|H|(k - 1) = \mathcal{O}(k)$ vertices of the (di)graphs in \mathcal{C} should *cover* every possible solution of the problem. Based on this last covering property, the challenge is to design a polynomial time procedure that may discard all but a polynomial number of vertices from G so that the remaining (di)graph is an equivalent instance. This procedure varies depending on the definition of the problem in question. Moreover, in case a polynomial kernel exists, a particular challenge towards deriving kernels of low polynomial size is to

maximize the set of discarded vertices so that the size of the resulting (di) graph, that is the output of the kernelization algorithm, is bounded by a low-polynomial function.

Rainbow matching technique. In this paper, we propose a framework for tackling the above challenge that we call *the rainbow matching technique*. Our technique capitalizes on the deep combinatorial results of Graf and Haxell in [27] (see also [26]). In fact, in [Subsection 2.2](#), we derive

the following “rainbow-matching” outcome of the main result of [27] asserting that *there exists a polynomial time algorithm that, given an edge-multicolored graph G (by p colors), either outputs a matching of G carrying all p colors or outputs a non-empty set of colors $C \subseteq \{1, \dots, p\}$ and a vertex set X of size $\mathcal{O}(|C|)$ that intersects all edges colored by the colors of C (Corollary 1).*

For the purposes of our technique, we build an auxiliary graph based on the maximal solution \mathcal{C} yielded by the greedy localization routine. We then consider a multi-coloring of the auxiliary graph based on the maximal solution \mathcal{C} . Depending of the outcome of the above algorithm, we either obtain an equivalent instance or we give a way to reorganize the parts of the (di)graph that are not in \mathcal{C} in *buckets* in a way that will permit a recursive application of the above procedure until an equivalent instance is created. We provide a generic description of our technique in Section 3.

The rainbow matching technique seems to naturally apply to packing problems, where a maximum disjoint collection of a specific (di)graph H is seeking in the input graph. We illustrate the technique on two packing problems. As a side effect, we notice that the technique also provides equivalent kernel for the corresponding dual problems of the considered ones. These hitting problems consist in finding a minimum set which intersects all the copies of H in the input graph. In all, we apply the rainbow matching technique to four problems that we describe in the following.

Packing induced paths of length two. The first problem where our technique is applied is the following.

INDUCED 2-PATH-PACKING (I2PP)	Parameter: k .
Input: (G, k) where G is a graph and $k \in \mathbb{N}$	
Question: Does G contain k pairwise disjoint induced paths of length two?	

As in the case of TPT, the above problem admits a kernel on $\mathcal{O}(k^2)$ vertices because of the results of Abu Khzam in [2, 3]. The first time a sub-quadratic kernel was given for I2PP was the one on $\mathcal{O}(k^{5/3})$ vertices by Fomin, Le, Lokshtanov, Saurabh, Thomassé, and Zehavi in [22]. Again, an open question that appeared in [22] is whether this bound can be improved to a linear one. As a second application of our technique, we prove that this is indeed the case.

Induced 2-paths Hitting Set. Given a graph \hat{G} , an *induced 2-path hitting set* of G is a set X which intersect all the induced 2-paths of G . In other words, $G \setminus X$ does not contain any induced 2-path. Then, the dual version of the previous problem, on which we apply the rainbow matching technique, is the following.

INDUCED 2-PATHS HITTING SET (I2PHS)	Parameter: k .
Input: (G, k) where G is a graph and $k \in \mathbb{N}$	
Question: Is there an induced 2-paths hitting set of G of size at least k ?	

Here again, this problem admits a kernel on $\mathcal{O}(k^2)$ vertices because of the results of Abu Khzam in [4]. And the first sub-quadratic kernel for I2PHS was on $\mathcal{O}(k^{5/3})$ vertices given by Fomin, Le, Lokshtanov, Saurabh, Thomassé, and Zehavi in [22]. Here also, using our technique, we prove that I2PHS admits a linear kernel.

Packing directed triangles in tournament. The third problem we consider is the following .

TRIANGLE-PACKING IN TOURNAMENT (TPT)

Parameter: k

Input: (T, k) where T is a tournament and $k \in \mathbb{N}$

Question: Does T contain k pairwise disjoint directed triangles?

Recall that a directed graph $T = (V, E)$ is a *tournament*, if for every distinct $x, y \in V$, either $(x, y) \in E$ or $(y, x) \in E$ holds (but not both).

The NP-completeness of TPT follows from the results of [8]. Moreover, given that certain patterns are excluded from its inputs, it can be solved in polynomial time [7]. For (non) approximability results on the TPT problem see [5, 9, 29, 36]. Notice that TPT can directly be reduced to the 3-HITTING SET problem. For this problem, Abu Khzam gave, in [2, 3], a kernel on $\mathcal{O}(k^2)$ vertices obtained by only removing vertices. This directly implies that TPT admits a kernel on $\mathcal{O}(k^2)$ vertices. On the negative side, Bessy, Bougeret, and Thiebaut proved in [5] that TPT does not admit a kernel of (total bit) size $\mathcal{O}(k^{2-\epsilon})$, unless $\text{NP} \subseteq \text{co-NP/Poly}$. They also proved that TPT admits a kernel of $\mathcal{O}(z)$ vertices, when its input instances are accompanied with a *feedback arc set*¹ of T of size z and that TPT restricted to sparse tournaments² admits a kernel of $\mathcal{O}(k)$ vertices (i.e., of total bit-size $\mathcal{O}(k \log k)$). Towards breaking the quadratic bound in the general case, Fomin, Le, Lokshtanov, Saurabh, Thomassé, and Zehavi gave in [22] a kernel for TPT on $\mathcal{O}(k^{3/2})$ vertices. The open question of [22] is whether a kernel on $\mathcal{O}(k)$ vertices exists for the general TPT. In this paper, we use the rainbow matching technique in order to give a kernel for TPT on $k^{1 + \frac{\mathcal{O}(1)}{\sqrt{\log k}}}$ vertices.

Feedback vertex set in tournament. The last problem on which we apply the rainbow matching technique is the dual version of the previous one and is described below. A triangle hitting set in a tournament T is a set X intersecting all the triangle of T . Equivalently, $T \setminus X$ does not contain any triangle and so, by a classical argument, is acyclic. Consequently, such a set X is called a *feedback vertex set* of T .

FEEDBACK VERTEX SET IN TOURNAMENT (FVST)

Parameter: k

Input: (T, k) where T is a tournament and $k \in \mathbb{N}$

Question: Does T contain a feedback vertex set of size at most k ?

Here again, this problem admits a kernel on $\mathcal{O}(k^2)$ vertices because of the results of Abu Khzam in [4]. And the first sub-quadratic kernel for FVST was on $\mathcal{O}(k^{3/2})$ vertices given by Fomin, Le, Lokshtanov, Saurabh, Thomassé, and Zehavi in [22]. Like for TPT, our technique leads us to obtain a kernel on $k^{1 + \frac{\mathcal{O}(1)}{\sqrt{\log k}}}$ vertices for FVST.

Organization of the paper.

The definitions of the basic concepts that we use are given in Section 2. In the same section we present the combinatorial base of the rainbow coloring lemma (Corollary 1) as well as the proof of how this is derived by the results of [27]. In Section 3, we proceed with a generic description of our technique. An overview is presented in Subsection 3.1, while the main combinatorial assumptions and invariants are presented in Subsection 3.2 and Subsection 3.3. The description of the main

¹Given a tournament $T = (V, E)$ we say that an edge set $F \subseteq E$ is a *feedback arc set* of T if the removal of F from T results to an acyclic digraph.

²A tournament $T = (V, E)$ is *sparse* if it contains a feedback arc set that is a matching.

algorithmic routine of the technique is given in [Subsection 3.4](#). We then present specialization of the technique for I2PP in [Section 4](#), for I2PHS in [Section 5](#), for TPT in [Section 6](#) and for FVST in [Section 7](#). We chose to present the almost linear kernels for TPT and FVST after the linear ones for I2PP and I2PHS, as the application of the technique for the tournament problems is more technical. We conclude the paper in [Section 8](#) with some remarks and open problems.

2 Definitions

We denote by \mathbb{N} the set of non-negative integers and by \mathbb{R}^+ the set of all non-negative reals. Given two integers p and q , the set $[p, q]$ refers to the set of every integer r such that $p \leq r \leq q$. For an integer $p \geq 1$, we set $[p] = [1, p]$, $[c]_0 = [c] \cup \{0\}$, and $\mathbb{N}_{\geq p} = \mathbb{N} \setminus [0, p - 1]$.

For a set S , we denote by 2^S the set of all subsets of S and, given an integer $r \in [|S|]$, we denote by $\binom{S}{r}$ the set of all subsets of S of size r . Given two sets A, B and a function $f : A \rightarrow B$, for a subset $X \subseteq A$ we use $f(X)$ to denote the set $\{f(x) \mid x \in X\}$.

Finally, for every set of subsets X_i , $i \in [c]$, and any subset of indexes $I \subseteq [c]$, we denote $X_I = \bigcup_{i \in I} X_i$. And whenever we refer to a *partition of a set X into c sets* we refer to an (ordered) set $\mathcal{X} = \{X_\ell \mid \ell \in [c]\}$ where $V(\mathcal{X}) = X$ and where we allow that some of the X_ℓ 's is an empty set.

2.1 Basic concepts

Parameterized algorithms and kernels. A *parameterized problem* Π is a subset of $\Sigma^* \times \mathbb{N}$ for some finite alphabet Σ . A parameterized problem $\Pi \subseteq \Sigma^* \times \mathbb{N}$ is *fixed parameter tractable* (in short **fpt**) if there is an algorithm that, given an instance $(x, k) \in \Sigma^* \times \mathbb{N}$, decides whether $(x, k) \in \Pi$ or not in time $f(k) \cdot p(|x|)$, where $f : \mathbb{N} \rightarrow \mathbb{N}$ is some function and $p : \mathbb{N} \rightarrow \mathbb{N}$ is a polynomial in the input size. The notion of *kernelization* is formally defined as follows.

Definition 1 (Kernelization). *Let $\Pi \subseteq \Sigma^* \times \mathbb{N}$ be a parameterized problem and g be a computable function. We say that Π admits a kernel of size g if there exists an algorithm \mathcal{K} , called kernelization algorithm, or, in short, a kernel that given $(x, k) \in \Sigma^* \times \mathbb{Z}^+$, outputs, in time polynomial in $|x| + k$, a pair $(x', k') \in \Sigma^* \times \mathbb{Z}^+$ such that*

- $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and
- $\max\{|x'|, k'\} \leq g(k)$.

When $g(k) = k^{\mathcal{O}(1)}$ or $g(k) = \mathcal{O}(k)$ then we say that Π admits a polynomial or linear kernel respectively. We refer to $g(k)$ as the size of the kernel produced by the kernelization algorithm.

When dealing with graphs algorithmic problems, the produced kernels are graphs, and we often refer to their size by specifying their number of vertices. For instance, we will say that a graph problem will admit a kernel with $\mathcal{O}(k)$ vertices (implicitly implying that the total size of the kernel is $\mathcal{O}(k^2)$).

Graphs, digraphs and tournaments. All graphs in this paper are finite. We use the term *graph* when we refer to an undirected graph without loops (ie. an edge with a unique endpoint) neither parallel edges (ie. distinct edges with the same endpoints). Also we use the term *multigraph* when we allow for loops and parallel edges. Directed graphs are called *digraphs*.

Given a (multi) (di)graph we denote by $V(G)$ and $E(G)$ the set of its vertices and edges respectively. Given a (multi) (di)graph G and a set $X \subseteq V(G)$, we denote by $G[X]$ the sub(di)graph of G induced by X . Similarly, if $X \subseteq E(G)$, we use $G[X]$ in order to denote the graph (V_X, X) , where V_X is the set containing all the endpoints of the edges in X . Given a graph (resp. digraph) G and two disjoint subsets of vertices A and B , we say that $e \in E(G)$ is an edge (resp. arc) between A and B iff $|e \cap A| = |e \cap B| = 1$. In a digraph, whenever $(x, y) \in E(G)$, we say that x *dominates* y . We also say that a digraph D is a *tournament* if, for every $\{x, y\} \in \binom{V(D)}{2}$, either $(x, y) \in E(D)$ or $(y, x) \in E(D)$, but not both. If $\mathcal{S} = \{S_1, \dots, S_\ell\}$ is a collection of vertex sets or subgraphs of some (multi) (di)graph, we denote by $V(\mathcal{S})$ the set $\bigcup_{i \in [\ell]} S_i$ or $\bigcup_{i \in [\ell]} V(S_i)$ respectively. Given a (multi) (di)graph G and $A \subseteq V(G)$, we denote by $G \setminus A$ the (multi) (di)graph $G[V(G) \setminus A]$ obtained after the deletion of all vertices of A .

2.2 Tools about rainbow matchings

Multigraph colourings. Let G be a multigraph. Recall that for $e \in E(G)$ we say that it is a *loop* of G if $|e| = 1$, otherwise we say that it is an *ordinary edge*. A p -*multiedge coloring* of G is a surjective function $\chi: E(G) \rightarrow [p]$ that associates to each edge of G a color in $[p]$ such that no two parallel edges of G receive the same color. We call the pair (G, χ) a p -*edge colored multigraph* and, given a set of colors $C \subseteq [p]$, we define $(G, \chi)[C] = G[\chi^{-1}(C)]$.

Given a p -edge colored multigraph (G, χ) , a *rainbow matching* of (G, χ) is a set $X \subseteq E(G)$ such that

- X is a matching of G , i.e., every two distinct edges of X are vertex-disjoint and
- $|X| = p$, and $\chi(X) = [p]$.

For a subset $C \subseteq [p]$ of colors, recall that a set X of vertices of G is a vertex cover of $(G, \chi)[C]$ iff for every $e \in E(G)$ where $\chi(e) \in C$, it holds that $e \cap X \neq \emptyset$. We denote by $\text{vc}_C(G, \chi)$ the minimum size of a vertex cover of $(G, \chi)[C]$.

The purpose of this section is to prove the following lemma.

Lemma 1. *Let (G, χ) be a p -edge colored multigraph and $\varepsilon \in (0, 1)$. If for every subset of colors $C \subseteq [p]$ we have $\text{vc}_C(G, \chi) > (4 + \varepsilon)(|C| - 1)$, then G contains a rainbow matching. Moreover there is an algorithm that, with input (G, χ) , either outputs a rainbow matching of (G, χ) or a non-empty subset C of $[p]$ and a vertex cover X of $(G, \chi)[C]$ such that $|X| \leq (4 + \varepsilon)(|C| - 1)$. This algorithm runs in time $\mathcal{O}(|V(G)|^8 \cdot p^{f(\varepsilon)})$, for some function $f: \mathbb{N} \rightarrow \mathbb{N}$.*

To prove [Lemma 1](#), we only need a light version of Graf and Haxell's Theorem, appeared as Theorem 4 in [\[27\]](#), which we present here (whose seminal version appeared in [\[30\]](#)).

Let G be a graph enhanced with a partition $\mathcal{V} = (V_1, \dots, V_p)$ of its vertex set. An *independent transversal* of G and \mathcal{V} is an independent set S of G satisfying $|S \cap V_i| = 1$ for every $i \in [p]$. For an integer r , the graph G is r -*claw-free for \mathcal{V}* if no vertex of G has r independent neighbors in distinct sets V_i . Finally, a set X of G is a *dominating set* of G if for every vertex v of G there exists a vertex x of X such that vx is an edge of G .

Theorem 1 (Graf and Haxell [\[27\]](#)). *For any fixed $r \in \mathbb{N}$ and $\varepsilon \in (0, 1)$ there exists an algorithm that takes as input a graph G and a partition $\mathcal{V} = (V_1, \dots, V_p)$ of its vertex set such that G is r -claw-free for \mathcal{V} and produces*

- either an independent transversal of G and \mathcal{V} , or
- a subset I of $\{1, \dots, p\}$ and a set X of vertices of G such that X is a dominating set of $G[\bigcup_{i \in I} V_i]$ and $|X| \leq (2 + \varepsilon)(|I| - 1)$.

Moreover, the algorithm runs in $\mathcal{O}(|V(G)|^4 \cdot p^{f(r, \varepsilon)})$ time, for some function $f: \mathbb{N} \times (0, 1) \rightarrow \mathbb{N}$.

Proof of [Lemma 1](#). Consider a p -edge colored multigraph G and some $\varepsilon \in (0, 1)$. We build an “extended line-graph” G' of G in order to apply [Theorem 1](#) on it. See [Figure 1](#) for an example of construction. The graph G' is defined as follows:

$$V(G') = \{(e, \chi(e)) \mid e \in E(G)\} \quad \text{and} \quad E(G') = \{\{(e, \chi(e)), (e', \chi(e'))\} \mid e \cap e' \neq \emptyset\}.$$

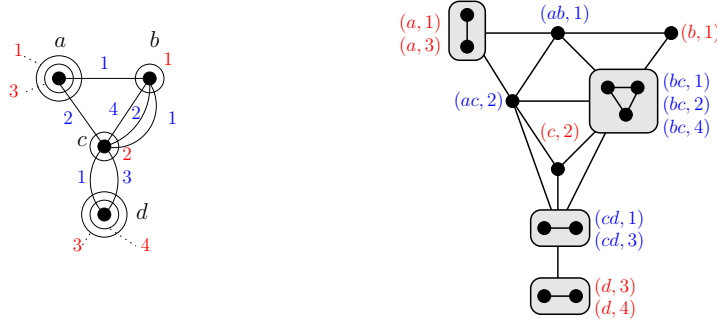


Figure 1: A p -edge-colored multigraph G on the left where the sets of colors are shown in red for the loops and in blue for the ordinary edges of G . On the right, the corresponding extended line-graph of G , where the elements in red and in blue respectively correspond to loops and ordinary edges of G .

Notice that the size of G' is polynomial in the size of G and p , in particular $|V(G')| = |E(G)| \leq \mathcal{O}(|V(G)|^2 \cdot p)$. Notice also that G' is 3-claw free, independently from the chosen partition of \mathcal{V} . Indeed the neighborhood of every vertex $(e, \chi(e))$ is either one clique (if e is an loop) or the union of two cliques (if e is an ordinary edge).

Now, consider the partition of the vertex set of G' given by the colors on the vertices of G' , that is $V_i = \{(e, c_0) \in V(G') \mid c_0 = i\}$, for $i \in [p]$. We then apply on G' the algorithm from [Theorem 1](#) with $r = 3$ and $\varepsilon/2$ as parameters. Clearly, this algorithm runs in time $\mathcal{O}(|V(G)|^8 \cdot p^{f(\varepsilon)})$, for some function $f: \mathbb{N} \rightarrow \mathbb{N}$.

If we obtain an independent transversal of G' , then it corresponds to a rainbow matching of G .

Suppose now that the algorithm outputs a subset I of $[p]$ and a set X of vertices of G' such that X is a dominating set of $G'[\bigcup_{i \in I} V_i]$ and $|X| \leq (2 + \varepsilon/2)(|I| - 1)$. Let Y be the vertices of G that are involved in an element of X , that is $x \in Y$ if there exists c_0 such that $(\{x\}, c_0) \in X$ or there exist c_0 and $y \in V(G)$ such that $(\{x, y\}, c_0) \in X$. We have $|Y| \leq 2|X|$ and we can claim that no edge e of $G \setminus Y$ has a color from I . Indeed, towards a contradiction, assume first that $e = \{x\}$ and $\chi(e) = c_0 \in I$. As X is a dominating set of $G'[\bigcup_{i \in I} V_i]$, there exists an element w of X adjacent to $(\{x\}, c_0)$ in G' . If w corresponds to a vertex of G then, by construction of G' , we have $w = (\{x\}, c')$ for a color $c' \neq c_0$. So, the vertex x belongs to Y , a contradiction. Now, if w corresponds to an edge $e = \{u, v\}$ of G , by construction of G' , we have $u = x$ or $v = x$ and again the vertex x belongs to Y , a contradiction. Finally, assume that $e = \{x, y\}$ and $\chi(e) \in I$. Similarly, there exists an element

w of X adjacent to xy in G' . If w corresponds to a vertex of G , then it must be x or y , while if w corresponds to an edge of G , then this edge must have a common endpoint with xy . In both cases, we have $x \in Y$ or $y \in Y$, again a contradiction. So, $Y' := Y \cap V((G, \chi)[I])$ is a vertex cover of $(G, \chi)[I]$ and $|Y'| \leq (4 + \varepsilon)(|I| - 1)$. \square

The *rainbow coloring technique* will be based on the following restatement of [Lemma 1](#).

Corollary 1. *There exists some function $f: \mathbb{N} \rightarrow \mathbb{N}$ such that for every $\varepsilon > 0$ there is an algorithm that, with input a p -edge-coloring graph (G, χ) , outputs either a rainbow matching of (G, χ) or finds a non-empty subset C of $[p]$ and a vertex cover X of $(G, \chi)[C]$ such that $|X| < (4 + \varepsilon)|C|$. Moreover, this algorithm runs in time $\mathcal{O}(|V(G)|^8 \cdot p^{f(\varepsilon)})$.*

3 The rainbow matching technique

Both our kernelization algorithms for I2PP and TPT will be based on the the rainbow matching technique that we next describe in a generic form.

3.1 Overview of the kernelization algorithm

To that end, let us consider a generic H -packing problem (where $|H| = 3$) where given an input (G, k) , the objective is to decide if there exist k vertex disjoint sets $P_i \subseteq V(G)$ such that for any i , $G[P_i]$ is isomorphic to H . The rainbow matching technique that we introduce in this article can be summarized as follow. Given an input (G, k) :

1. Maintain a partition (W, B, C) of $V(G)$ with size and structure invariants, called a *partial decomposition*, where $B \cup C$ is small (typically $\mathcal{O}(k)$), and W is the large part where we want to select an appropriate subset.
2. At each round, apply the following rule:
 - define an auxiliary edge-colored multigraph graph $(G, \chi)\langle W, B, C \rangle$ with vertex set W .
 - applies [Corollary 1](#) on $(G, \chi)\langle W, B, C \rangle$:
 - if there exists a rainbow matching M in $(G, \chi)\langle W, B, C \rangle$, stop the kernel and output $G[V(M) \cup B \cup C]$ (we selected $V(M)$ from W)
 - otherwise, use the subset of colors X and its small vertex cover $T \subseteq W$ to compute a new partial decomposition (W', B', C')

We point out that the partial decomposition may contain other information than (W, B, C) , as it is the case of example for TPT here, but for the sake of simplicity we stick to the triplet (W, B, C) in this generic presentation. As invariants of a partial decomposition we used for I2PP and TPT have a lot of similarities, we now explain the common ideas behind these invariants.

3.2 Origin of the invariants in a partial decomposition.

We start with a greedy localization phase where we compute a maximal H -packing $\mathcal{P} = \{H_1, \dots, H_{i_0}\}$, and we assume that $i_0 < k$ as otherwise we get a **yes**-instance. We define $C^0 = V(\mathcal{P})$ and $W^0 = V(G) \setminus C^0$. Observe that there is no copy of H inside W^0 , and that $|C^0| = \mathcal{O}(k)$. The goal

is to select a subset $S \subseteq W^0$ and to output $G[S \cup C^0]$. We want S to be small, typically $|S| = \mathcal{O}(k)$, and safe, in the sense that any packing \mathcal{P} in G can be restructured into a packing \mathcal{P}' such that $|\mathcal{P}'| \geq |\mathcal{P}|$ and $V(\mathcal{P}') \subseteq S \cup C^0$. This will imply that if (G, k) is a yes-instance, then $(G[S \cup C^0], k)$ is also a yes-instance, and thus that these instances are equivalent as the other implication is straightforward.

Then, we define an auxiliary graph $(G, \chi) \langle W^0, \emptyset, C^0 \rangle$ as follows (we used the $\langle W^0, \emptyset, C^0 \rangle$ notation to match notation of Section 3.1, as $B = \emptyset$ at the beginning). Let $V((G, \chi) \langle W^0, \emptyset, C^0 \rangle) = W^0$, and for any $c \in C^0$ and any $\{u, v\} \subseteq W^0$ such that $G[\{c, u, v\}]$ is isomorphic to H , add to $(G, \chi) \langle W^0, \emptyset, C^0 \rangle$ edge $e = \{u, v\}$ and set color $\chi(e) = c$. Notice that C^0 both denotes a subset of vertices in G , and the set of colors of $(G, \chi) \langle W^0, \emptyset, C^0 \rangle$ and this is a convention that we follow all over the paper. Now, apply Corollary 1 on $(G, \chi) \langle W^0, \emptyset, C^0 \rangle$, and let us discuss the two possible outcomes of this corollary.

Case where a rainbow matching exists. We now consider the case where we have a rainbow matching M in $(G, \chi) \langle W^0, \emptyset, C^0 \rangle$. In this case, we stop and return $G[V(M) \cup C^0]$. Observe first that $V(M)$ is small, as $|V(M)| = 2|C^0| = \mathcal{O}(k)$. Let us now examine why it is safe. Consider an H -packing \mathcal{P} in G . We restructure \mathcal{P} as follows (see Figure 2). For any $H_i \in \mathcal{P}$ we define a corresponding H'_i as follows (and we define $\mathcal{P}' = \{H'_i \mid H_i \in \mathcal{P}\}$):

- If $H_i \subseteq C^0$, define $H'_i = H_i$.
- Otherwise, we know that $H_i \cap C^0 \neq \emptyset$ by the maximality of the greedy localization. Let $c \in H_i \cap C^0$, and $e_c = \{u_c, v_c\}$ the edge of M such that $\chi(e_c) = c$. Restructure H_i into $H'_i = \{c, u_c, v_c\}$.

We can observe that the $\{H'_i\}$ are vertex disjoint as M is a matching, and thus that $V(M)$ is safe.

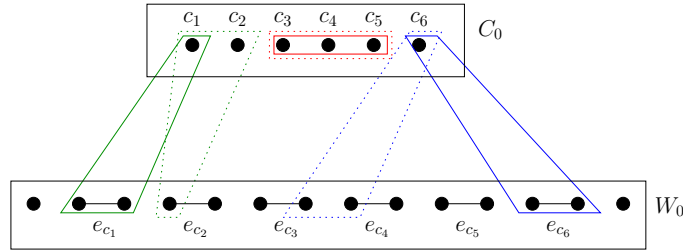


Figure 2: An example showing how to restructure H'_1, H'_2, H'_3 (in dashed lines) to H_1, H_2, H_3 (in plain lines) using the rainbow matching $\{e_{c_i}, 1 \leq i \leq 6\}$.

Case where a small vertex cover exists. Suppose there exists a subset of colors $X \subseteq C^0$ and a vertex cover T of $(G, \chi) \langle W^0, \emptyset, C^0 \rangle[X]$ such that $|T| \leq (4 + \varepsilon)|X|$. Let $B = T \cup X$, $W = W^0 \setminus T$, and $C = C^0 \setminus X$. Observe that as T is a vertex cover of $(G, \chi) \langle W^0, \emptyset, C^0 \rangle[X]$, for any set $H' \subseteq B \cup W$ such that $G[H']$ is isomorphic to H , we have $|H' \cap W| \leq 1$. This crucial property will help us to control how copies of H can be packed in $G[W \cup B]$, and thus motivates the invariants required in the following (informal) definition of partial decomposition.

3.3 Invariants in a partial decomposition

We say that (W, B, C) is a *partial decomposition* (with respect to (C^0, W^0)) of G if

1. W, B, C is a partition of $V(G)$
2. (W, B) is a *nice pair*: $W \subseteq W^0$ and for any $H' \subseteq B \cup W$ such that $G[H']$ is isomorphic to H , $|H' \cap W| \leq 1$
3. (size invariant) B is small (typically $|B| = \mathcal{O}(|C^<|)$, where $C^< = C^0 \setminus C$)

As C^0 and W^0 will be fixed (there are only computed once at the beginning), we voluntarily do not mention “with respect to (C^0, W^0) ” in the remainder of the article. Observe that the tuple (W, B, C) that we obtained at this end of Section 3.2 is a partial decomposition. Properties 1 and 2 listed above are common to both problems, whereas the notion of size to ensure that B is small is ad-hoc (but with the same objective to guarantee that $|B| + |C| = \mathcal{O}(|C^0|)$, as B and C will be part of the output of the kernel). The property of being a nice pair will allow us to provide a structural description of B : in both problems we will partition B into *buckets* B_i , and obtain a description on how copies of H in $W \cup B$ intersect the buckets.

3.4 Description of one round of the kernel

We now consider an arbitrary round of the kernel, where we have our current partial decomposition (W, B, C) . The objective is now to apply again Corollary 1, but in a more general setting than in Section 3.2 where we had $B = \emptyset$.

Let us now discuss how to define the auxiliary graph $(G, \chi)\langle W, B, C \rangle$. We start as before by defining $V((G, \chi)\langle W, B, C \rangle) = W$, and for any $c \in C$, and any $\{u, v\} \subseteq W$ such that $G[\{c, u, v\}]$ is isomorphic to H , adding to $(G, \chi)\langle W, B, C \rangle$ edge $e = \{u, v\}$ and set $\chi(e) = c$. Again, notice that according to the context, C may denote a subset of G , of a subset of our colors in $(G, \chi)\langle W, B, C \rangle$. The crux of this approach is to add to $(G, \chi)\langle W, B, C \rangle$ some loops in W and a coloring of these loops (using a fresh set of colors D , which is disjoint from C), such that:

- i. $|D| = \mathcal{O}(|B|)$
- ii. if there exists a rainbow matching M^D only for loops of colors in D (meaning that $M^D = \{v_d, d \in D\}$ and $|M^D| = |D|$), then for any packing \mathcal{P} of $G[W \cup B]$, there exists a packing \mathcal{P}' such that $|\mathcal{P}'| = |\mathcal{P}|$ and $V(\mathcal{P}') \subseteq B \cup V(M^D)$
- iii. if there exists a subset of colors $X^D \subseteq D$ and a vertex cover T of $(G, \chi)\langle W, B, C \rangle[X^D]$ such that $|T| \leq 2(4 + \varepsilon)|X^D|$, then we can find in polynomial time a non-empty subset $T' \subseteq T$ such that we can add T' to B without violating the size invariant of a partial decomposition. In other words, in this case we define $W' = W \setminus T'$, $B' = B \cup T'$, $C' = C$, and we want that (W', B', C') remains a partial decomposition.

The way we can define such colored loops to obtain the three previous properties depends on the problem that we consider and thus will be specified in each application of the technique. As we guess that, at a first sight, Property iii. may look like it comes out of nowhere, let us now explain why properties i. to iii. are sufficient to obtain the kernel. Suppose that we defined an auxiliary graph $(G, \chi)\langle W, B, C \rangle$ satisfying the previous properties, and that we apply Corollary 1 on $(G, \chi)\langle W, B, C \rangle$. Let us discuss again the two possible outcomes of this Corollary.

Case where a rainbow matching exists. Consider now the case where there is a rainbow matching M in $(G, \chi)\langle W, B, C \rangle$. In this case, we stop and return $G[V(M) \cup B \cup C]$. Let us partition $M = M^C \cup M^D$, where M^C (resp. M^D) are edges whose color is in C (resp. D). Observe first that the kernel output is small, as $|V(M)| = |D| + 2|C| = \mathcal{O}(|B| + |C|)$ (by Property i.), and thus the output has vertex size $\mathcal{O}(|B| + |C|) = \mathcal{O}(|C^<| + |C|) = \mathcal{O}(|C^0|) = \mathcal{O}(k)$ (by Property 3). Let us now examine why it is safe. Consider an H -packing \mathcal{P} in G . We restructure \mathcal{P} as follows (see Figure 3). For any $H_i \in \mathcal{P}$ we define a corresponding H'_i as follows (and we define $\mathcal{P}' = \{H'_i \mid H_i \in \mathcal{P}\}$):

- If $H_i \subseteq C$, define $H'_i = H_i$.
- If $H_i \not\subseteq C$, and $H_i \cap C \neq \emptyset$, choose $c \in H_i \cap C$, and define $e_c = \{u_c, v_c\}$ the edge of M such that $\chi(e_c) = c$. Restructure H_i into $H'_i = \{c, u_c, v_c\}$.
- Now, it only remains to restructure $\mathcal{P}_{(W \cup B)} = \{H_i \in \mathcal{P} \mid H_i \subseteq W \cup B\}$. By Property ii., there exists a packing $\mathcal{P}'_{(W \cup B)}$ such that $|\mathcal{P}'_{(W \cup B)}| = |\mathcal{P}_{(W \cup B)}|$ and $V(\mathcal{P}'_{(W \cup B)}) \subseteq B \cup V(M^D)$.

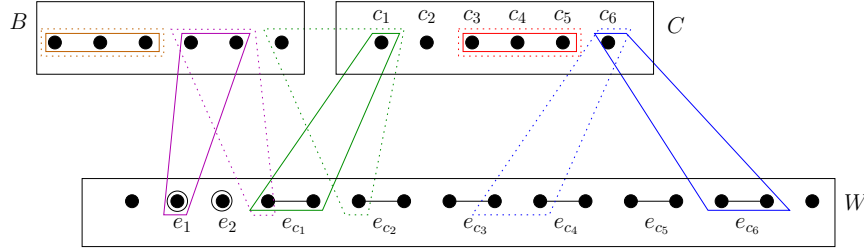


Figure 3: The color set of $(G, \chi)\langle W, B, C \rangle$ is $C \cup D$, where here $C = \{c_1, \dots, c_6\}$, and $D = \{d_1, d_2\}$ (where $D \cap C = \emptyset$). In this example we assume that there exists a rainbow matching M in $(G, \chi)\langle W, B, C \rangle$. We have $M = M^C \cup M^D$, where $M^C = \{e_{c_i}, 1 \leq i \leq 6\}$ and $M^D = \{e_1, e_2\}$. Here, we restructure $H'_1, H'_2, H'_3, H'_4, H'_5$ (in dashed lines) to H_1, H_2, H_3, H_4, H_5 (in plain lines) according to M .

We can observe that the $\{H'_i\}$ are vertex disjoint as M is a matching, and thus that $V(M)$ is safe. Observe that the matching was computed without considering the complex (and unknown) structure of “bad” copies of H that use vertices of both B and C . In other words, the way we organize the previous restructuration allows us to forget these bad copies in G .

Case where a small vertex cover exists. Suppose there exists a non-empty subset of colors $X \subseteq C \cup D$ and a vertex cover of T of $(G, \chi)\langle W, B, C \rangle[X]$ such that $|T| \leq (4 + \varepsilon)|X|$. Let $X^C = X \cap C$ and $X^D = X \cap D$. Notice first that we cannot simply proceed as in Section 3.2 (where X^D was empty) and define $W' = W \setminus T$, $C' = C \setminus X^C$ and $B' = B \cup (T \cup X^C)$. Indeed, if we want the size property 3 to be respected, we need that what we add to B is linear in what we remove from C , or more formally that $|T \cup X^C| = \mathcal{O}(|X^C|)$. However, we only know that $|T| \leq (4 + \varepsilon)|X| = (4 + \varepsilon)(|X^D| + |X^C|)$, and thus if $|X^C|$ is small compared to $|X^D|$ (typically $|X^C| = 0$), we don't have the property we need. This is where we use (for both problems) the following win/win trick.

Case 1: if $|X^D| \leq |X^C|$. In this case, the previous inequality gives us $|T| \leq (4 + \varepsilon)|X| = 2(4 + \varepsilon)|X^C|$, and we can define $W' = W \setminus T$, $C' = C \setminus X^C$ and $B' = B \cup (T \cup X^C)$ while preserving

size Property 3. Notice that the crucial property ensuring that (W', B') is still a nice pair (which is required to obtain that (W', B', C') remains a partial decomposition) is that there is no H' such that $G[H']$ is isomorphic to H where $|H' \cap W'| = 2$ and $|H' \cap X^c| = 1$, which holds because T is a vertex cover of $(G, \chi) \langle W, B, C \rangle [X]$.

Case 2: if $|X^D| > |X^C|$. In this case, the previous inequality gives us $|T| \leq 2(4 + \varepsilon)|X^D|$, and we use Property iii. to find in polynomial time a non-empty subset $T' \subseteq T$ such that (W', B', C') remains a partial decomposition, where $W' = W \setminus T'$, $B' = B \cup T'$, $C' = C$. Thus, Case 2 corresponds to a case where we discover that a certain part $T' \subseteq W$ is small, and can be added to the buckets.

Notice that in both Cases 1 and 2, we obtain a new “smaller” partial decomposition where either $|C'| < |C|$ (because X^C is non-empty, in Case 1) or $|W'| < |W|$ (in Case 2). Thus the algorithm terminates.

3.5 Applicability of the rainbow coloring technique

The rainbow matching technique consists in applying the algorithm of Section 3.1. Adapting this technique to a particular problem requires to find a way to define colors for buckets that respects Properties i., ii., and iii., together with a notion of size used in Property 3. In Section 4 and Section 6, we present how this technique can be applied for I2PP and TPT respectively.

4 Linear kernel for I2PP

4.1 Notation

Given a graph G we refer to a path in G of length 2 as a *2-path* of G . We say that a 2-path of G is *induced* if there is no edge in the graph between its endpoints. We call a P_3 an induced 2-path. When we refer to a P_3 in a graph, we will see it as a subset of vertices rather than an induced subgraph. An *induced P_3 -packing* of G is a set $\mathcal{P} = \{V_i, i \in [x]\}$ of vertex-disjoint induced P_3 's.

INDUCED 2-PATH-PACKING (I2PP)

Parameter: k .

Input: (G, k) where G is a graph and $k \in \mathbb{N}$

Question: Is there an induced P_3 -packing of size at least k ?

In this section we prove the following theorem.

Theorem 2. *There exists some function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that for every $\varepsilon > 0$, there exists an algorithm that, given an instance (G, k) of I2PP outputs a set $A \subseteq V(G)$ such that $(G[A], k)$ is an equivalent instance where $|A| \leq (243 + \varepsilon)k$. Moreover this algorithm runs in time $\mathcal{O}(|V(G)|^{f(\varepsilon)})$. In other words, I2PP admits a kernel of a linear number of vertices.*

4.2 Preliminary phase: greedy localization

Given an input (G, k) of I2PP, we start by a greedy localization phase, that is, by finding, in polynomial time, an inclusion-wise maximal induced 2-path-packing of G . Let C^0 be the set of the vertices in the paths of such a packing. If $|P^0| \geq k$ we can directly answer that (G, k) is a yes-instance, therefore we may suppose that $|P^0| < k$, implying that $|C^0| \leq 3(k - 1)$. The vertices of C^0 will be referred as (distinct) *colors*. Observe that as the considered induced 2-path-packing is inclusion-wise maximal, the graph $W^0 := G \setminus C^0$ does not contain P_3 as an induced subgraph,

consequently it is the disjoint union of a set, say $\mathcal{W}^0 = \{W_1^0, \dots, W_{i_0}^0\}$, such that, for $i \in [i_0]$, the graph $G[W_i^0]$ is a clique. Clearly, $V(G) = C^0 \cup V(\mathcal{W}^0)$ and, for any $\{i, i'\} \in \binom{[i_0]}{2}$, there is no edge between W_i^0 and $W_{i'}^0$. Such a couple (C^0, \mathcal{W}^0) will be referred as a *greedy localized pair for the input* (G, k) .

This completes the initialization phase, and the tuple (C^0, \mathcal{W}^0) will be given as initial input of our kernelization algorithm.

4.3 Nice pair, buckets, partial decomposition, and auxiliary graph

In all this section we consider that we are given an input (G, k) of I2PP, and a greedy localized pair (C^0, \mathcal{W}^0) for input (G, k) . Recall that $\mathcal{W}^0 = \{W_1^0, \dots, W_{i_0}^0\}$ and $W^0 = V(\mathcal{W}^0)$.

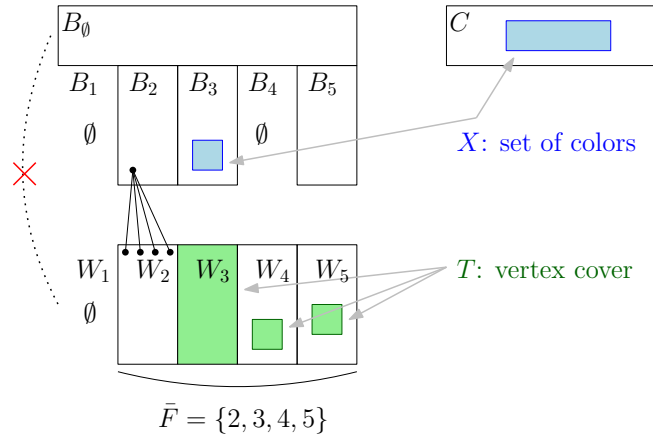


Figure 4: Example of a bucket decomposition and a partial decomposition. The vertex depicted in B_2 is adjacent to all vertices of W_2 , and not adjacent to any other vertex in W . We depict the case where the kernelization algorithm does not find a rainbow matching in $(G, \chi)\langle W, B, C \rangle$ and therefore it finds a subset of colors X (depicted in blue) along with a small set of vertices T (depicted in green) such that T is a vertex cover of $(G, \chi)\langle W, B, C \rangle[X]$.

Given two disjoint subsets W and B of $V(G)$, we say that the pair (W, B) is a *nice pair* of G if $W \subseteq W^0$ and if every induced 2-path of $B \cup W$ contains at most one vertex in W .

Bucket decompositions. Given two disjoint subsets W and B of $V(G)$, a *bucket decomposition* (see Figure 4) of the pair (W, B) is defined by the following three partitions:

- a partition $\{B_{\neq \emptyset}, B_{\emptyset}\}$ of B
- a partition $\{B_1, \dots, B_{i_0}\}$ of $B_{\neq \emptyset}$, (we call the sets of this partition *buckets*) and
- a partition $\{W_1, \dots, W_{i_0}\}$ of W .

(recall that we allow empty sets in partitions)
such that

1. for $i \in [i_0]$ $W_i = W \cap W_i^0$

2. $B_\emptyset = \{v \in B \mid N_G(v) \cap W = \emptyset\}$ and $B_{\neq\emptyset} = \{v \in B \mid N_G(v) \cap W \neq \emptyset\}$
3. for any $i \in [i_0]$ and $v \in B_i$, $N_G(v) \cap W = W_i$ (the neighborhood in W of any vertex of a bucket B_i is exactly the vertex set of its “corresponding” clique $G[W_i]$).

We denote $F = \{i \in [i_0] \mid W_i = \emptyset\}$, $\bar{F} = [i_0] \setminus F$. Observe that if $W_i = \emptyset$, then $B_i = \emptyset$, as vertices in B_i belong to $B_{\neq\emptyset}$ and should have $N(v) \cap W \neq \emptyset$. The contrapositive is not true as we may have $W_i \neq \emptyset$ and $B_i = \emptyset$.

Lemma 2. *Given W and B disjoint subsets of G ,*

1. (W, B) is a nice pair iff it admits a bucket decomposition.
2. if (W, B) is nice pair, then for any induced 2-path P in G such that $P \subseteq W \cup B$, $P \cap W \neq \emptyset$, there exists a unique $i \in [i_0]$ and vertices u, v, w such that $P = \{u, v, w\}$, $u \in B$, $v \in B_i$ and $w \in W_i$. Informally, P must have exactly one vertex in one of the remaining cliques, one in its corresponding bucket, and the last one anywhere in B .

Proof. We start with the proof of the first item.

(\Rightarrow) Suppose (W, B) is a nice pair. Consider a vertex $v \in B_{\neq\emptyset}$. There exists $i \in [i_0]$ such that $N(v) \cap W_i \neq \emptyset$. Let $u \in N(v) \cap W_i$. If there existed $w \in W_i$ such that $w \notin N(v)$, then, as W_i is a clique, we would have that $P = \{v, u, w\}$ is an induced 2-path such that $|P \cap W| \geq 2$, a contradiction. This implies that $N(v) \cap W \supseteq W_i$. Let us now prove that $N(v) \cap W = W_i$. Suppose, towards a contradiction, that there exists $j \neq i$ and $u' \in N(v) \cap W_j$. Then, as there is no edges between W_j and W_i , we obtain that $P = \{v, u, u'\}$ is an induced 2-path such that $|P \cap W| \geq 2$, a contradiction. Thus, we obtain the partition $\{B_1, \dots, B_{i_0}\}$ of B_\emptyset , where $B_i = \{i \in B_{\neq\emptyset} \mid N(v) = W_i\}$ (some of the B_i 's may be empty).

(\Leftarrow) Let (W, B) be a pair admitting a bucket decomposition. Let us prove the following structural property: for any P induced 2-path of $G[W \cup B]$ such that $P \cap W \neq \emptyset$, there exists a unique $i \in [i_0]$ and vertices u, v, w such that $P = \{u, v, w\}$, $u \in B$, $v \in B_i$, and $w \in W_i$. We cannot have $P \subseteq W$ as W is a union of cliques (as $W_i \subseteq W_i^0$), implying that $P \cap B \neq \emptyset$. This implies that there exists an edge $\{v, w\} \subseteq P$ such that $v \in B$ and $w \in W$. By Property 2, we know that $v \notin B_\emptyset$, implying that there exists $i \in [i_0]$ such that $v \in B_i$. Property 3 implies that $w \in W_i$. Let us now consider the third vertex u of P . Firstly, u cannot be in W_i as, by Property 3, this would imply that $\{v, u\}$ is an edge, and thus P would be a triangle. Secondly, u cannot be in a W_j , where $j \neq i$, as by Property 3, $N(v) \cap W = W_i$. This implies that $u \in B$ and concludes the proof of the structural property. The fact that (W, B) is a nice pair is now immediate.

The second item immediately follows. Indeed, by the first part of the result, any nice pair (W, B) admits a bucket decomposition, which implies the structural property as seen previously. \square

Given a nice pair (W, B) , we will refer to B_\emptyset , $B_{\neq\emptyset}$, F and \bar{F} as defined in the bucket decomposition. Informally, F will denote the set of indexes of cliques that will survive during the course of the kernelization algorithm. Recall that, using our notations, $B_{\bar{F}} = \bigcup_{i \in \bar{F}} B_i$ and observe that $B_{\bar{F}} = B_{\neq\emptyset}$.

We fix some $\varepsilon > 0$ and set $c_1 = 4 + \varepsilon$ (c_1 is suited so to permit the application of Corollary 1). Given a nice pair (W, B) , we define the *size* of (W, B) as

$$s(W, B) = \frac{|B_\emptyset|}{1 + 2c_1} + |B_{\neq\emptyset}|.$$

Definition 2. We say that a tuple (W, B, C) is a partial decomposition iff:

Partition requirements:

Let $C^< = C^0 \setminus C$ (which we will see as the colors already treated by previous applications of the rule of the kernelization algorithm)

1. there is a partition $V(G) = W \cup B \cup C$,
2. $C \subseteq C^0$, and
3. (W, B) is a nice pair.

Size requirement:

1. $s(W, B) \leq (1 + 2c_1)|C^<|$.

Moreover, we will say that the partial decomposition is clean if it satisfies the following extra condition:

1. for every vertex $c \in C$ the graph $G[W \cup \{c\}]$ contains an induced 2-path.

The initial partial decomposition we will consider (which will be (W_0, \emptyset, C_0)) as well as partial decompositions which will be produced by our kernelization procedure will not necessarily be clean partial decompositions. However the simple following lemma allows to clean a partial decomposition.

Lemma 3 (Cleaning Lemma). *Let (W, B, C) be a partial decomposition and X be set the of vertices x of C such that $G[W \cup \{x\}]$ does not contain any induced 2-path. Then $(W, B \cup X, C \setminus X)$ is a clean partial decomposition.*

Proof. First let us check that $(W, B \cup X, C \setminus X)$ satisfies the requirements of a partial decomposition. Properties 1 and 2 are clearly satisfy, and by choice of X , no vertex of X is contained in an induced 2-path with two vertices of W , so $(W, B \cup X)$ is also a nice pair and property 3 is satisfies. To conclude that $(W, B \cup X, C \setminus X)$ is a partial decomposition, let us check it fulfills the size requirement. Indeed each time one vertex of X is added to B , $s(W, B)$ increases by at most 1, whereas $(1 + 2c_1)|C^<|$ increases by $1 + 2c_1$. So, the size requirement 1 is still valid. By repeating this counting argument for every vertex of X , we can conclude that $(W, B \cup X, C \setminus X)$ is a partial decomposition. Moreover, it is clear that every vertex of $C \setminus X$ forms an induced 2-path with two vertices of W and that $(W, B \cup X, C \setminus X)$ is clean. \square

Definition 3 (Auxiliary graph). *Let (W, B, C) be a partial decomposition. Let $p = |C| + \sum_{i \in [i_0]} |B_i|$. We define the p -edge-colored multigraph $(\tilde{G}, \chi)(W, B, C)$ where the vertex set of \tilde{G} is W and the edges of \tilde{G} , as well as their colors, are defined as follows. For any $i \in \bar{F}$ and any $u \in B_i$ and for any $v \in W_i$, we add an edge $e = \{v\}$ and we set $\chi(e) = u$. Moreover, for any $u \in C$ and for any v and w in W such that $\{u, v, w\}$ is an induced 2-path, we add an edge $e = \{v, w\}$ and we set $\chi(e) = u$.*

Observe that an edge $\{v, w\}$ of $(\tilde{G}, \chi)\langle W, B, C \rangle$ can be either inside a W_i (in this case $\{v, w\}$ is also an edge in G as $G[W_i]$ is a clique) or between W_i and $W_{i'}$ for $i \neq i'$ (in this case $\{v, w\}$ is a non-edge in G). Notice that if $W = \emptyset$, then $(\tilde{G}, \chi)\langle W, B, C \rangle$ is the empty graph and we consider that it admits a rainbow matching $M = \emptyset$. Notice also that C or $B_{\neq \emptyset}$ may be empty. If $C = B_{\neq \emptyset} = \emptyset$ then $(\tilde{G}, \chi)\langle W, B, C \rangle$ has no edges and we consider that it admits a rainbow matching $M = \emptyset$ (but this case will not occur), however cases where one of the two sets $C, B_{\neq \emptyset}$ is empty will occur in the kernel.

4.4 Analysis of the two cases: rainbow matching or small vertex cover

First, let us look at the case where the reduction rule produces a rainbow matching.

Lemma 4 (Case of rainbow matching). *Let (W, B, C) be a clean partial decomposition. Suppose that the colored multigraph $(\tilde{G}, \chi)\langle W, B, C \rangle$ admits a rainbow matching M . Let $A = V(M) \cup B \cup C$, and $G' = G[A]$. Then,*

1. (G, k) and (G', k) are equivalent instances of I2PP,
2. $|A| \leq 3(1 + 2c_1)^2 k$.

Proof. Equivalence property. First, if $W = \emptyset$, then $(\tilde{G}, \chi)\langle W, B, C \rangle$ is the empty graph then $M = \emptyset$ and $G' = G$, implying the equivalence. Let us now assume that $W \neq \emptyset$. Let us assume that (G, k) is yes-instance and prove that (G', k) is as well (the other direction is straightforward as G' is an induced subgraph of G). Let $\mathcal{P}^* = \{P_i^*, i \in [k]\}$ be an induced P_3 -packing in G of size k .

Notice that the only vertices of G not belonging to G' are in W . Let us partition $\mathcal{P}^* = \mathcal{P}_1^* \cup \mathcal{P}_W^*$, where $\mathcal{P}_W^* = \{P \in \mathcal{P}^* \mid P \cap W \neq \emptyset\}$. Our objective is to restructure \mathcal{P}^* into another induced 2-path-packing $\mathcal{P} = \mathcal{P}_1^* \cup \mathcal{P}_W$ such that $|\mathcal{P}_W| = |\mathcal{P}_W^*|$. To that end, we will associate to each $P_i^* \in \mathcal{P}_W^*$ a set $P_i \in \mathcal{P}_W$ such that

1. for any $P_i \in \mathcal{P}_W$, P_i is an induced 2-path in G'
2. for any P_i, P_j in \mathcal{P}_W , $P_i \cap P_j = \emptyset$ (paths in \mathcal{P}_W are vertex-disjoint)
3. for any i , $P_i \cap (V(G) \setminus W) \subseteq P_i^* \cap (V(G) \setminus W)$ (roughly speaking outside W , path P_i^* uses more vertices than P_i)

Observe that Properties 2 and 3 imply that paths in \mathcal{P} are vertex-disjoint. In particular, if, towards a contradiction, a path $P \in \mathcal{P}_1^*$ intersected a path $P_i \in \mathcal{P}_W$, then, by Property 3, we would also have $P \cap P_i^* \neq \emptyset$, which is a contradiction. Let us now define the P_i 's.

Let us partition $\mathcal{P}_W^* = \mathcal{P}_{CW}^* \cup \mathcal{P}_{\bar{C}W}^*$, where $\mathcal{P}_{CW}^* = \{P \in \mathcal{P}_W^* \mid P \cap C \neq \emptyset\}$ and $\mathcal{P}_{\bar{C}W}^* = \{P \in \mathcal{P}_W^* \mid P \cap C = \emptyset\}$. As paths in $\mathcal{P}_{\bar{C}W}^*$ use a vertex in W , and no vertex in C , it intersects B and by Property 2 of Lemma 2, for any $P_i^* \in \mathcal{P}_{\bar{C}W}^*$ there exists a unique l_i and vertices u_i, v_i, w_i such that $u_i \in B$, $v_i \in B_{l_i}$, and $w_i \in W_{l_i}$. Thus, we can partition $\mathcal{P}_{\bar{C}W}^* = \bigcup_{l \in \bar{F}} \mathcal{P}_{\bar{C}W}^{(*,l)}$, where $\mathcal{P}_{\bar{C}W}^{(*,l)} = \{P_i^* \in \mathcal{P}_{\bar{C}W}^* \mid l_i = l\}$. Observe that

- any $P_i^* \in \mathcal{P}_{\bar{C}W}^{(*,l)}$ uses exactly one vertex (w_i) in W_l
- any $P_i^* \in \mathcal{P}_{\bar{C}W}^{(*,l)}$ uses at least one vertex (v_i) in B_l , implying $|\mathcal{P}_{\bar{C}W}^{(*,l)}| \leq |B_l|$

Let us now prove the following property used to restructure paths in $\mathcal{P}_{\bar{C}W}^{(*,l)}$.

Property (P₁): For any $P_i^* \in \mathcal{P}_{\bar{C}W}^{(*,l)}$, if we replace w_i by any $w'_i \in W_l$, then $\{u_i, v_i, w'_i\}$ is still an induced 2-path.

Indeed, by Property 3 of a bucket decomposition, $N(v_i) \cap V(W) = W_l$, implying that $\{v_i, w'_i\} \in E(G)$.

Let us now prove that $\{u_i, w_i\} \in E(G)$ iff $\{u_i, w'_i\} \in E(G)$, which will imply that $\{u_i, v_i, w'_i\}$ is an induced 2-path. If $u_i \in B_\emptyset$, then, by Property 2 of a bucket decomposition, we have that $\{u_i, w_i\} \notin E(G)$ and $\{u_i, w'_i\} \notin E(G)$. If $u_i \in B_{l'}$, for $l' \neq l$, then, by Property 3 of a bucket decomposition, we have $\{u_i, w_i\} \notin E(G)$ and $\{u_i, w'_i\} \notin E(G)$. Finally, if $u_i \in B_l$, then, by Property 3 of a bucket decomposition, we have $\{u_i, w_i\} \in E(G)$ and $\{u_i, w'_i\} \in E(G)$.

Let us now define \mathcal{P}_W . For any color c of $(\tilde{G}, \chi)\langle W, B, C \rangle$, let $M(c)$ be the edge of color c in M . Recall that a color c can either belong to B_l for some $l \in \bar{F}$ (in which case $|M(c)| = 1$ and $V(M(c)) \subseteq W_l$) or belong to C (in which case $|M(c)| = 2$ and $V(M(c)) \subseteq W_{\bar{F}}$). Moreover, as the partial decomposition (W, B, C) is clean, for every vertex $c \in C$ there exists an edge of $(\tilde{G}, \chi)\langle W, B, C \rangle$ with color c and then M also contains an edge with color c and $M(c)$ is well defined. Thus, for any $l \in \bar{F}$ and any $P_i^* \in \mathcal{P}_{\bar{C}W}^{(*,l)}$ (such that $P_i^* = \{u_i, v_i, w_i\}$ with $v_i \in B_l$ and $w_i \in W_l$), we define $P_i = \{u_i, v_i, M(v_i)\}$. For any $P_i^* \in \mathcal{P}_{\bar{C}W}^*$, let $u_i \in P_i^* \cap C$, and let $P_i = \{u_i\} \cup M(u_i)$.

Let us now prove Properties 1, 2, and 3. We start with Property 1. For any $l \in \bar{F}$ and any $P_i^* \in \mathcal{P}_{\bar{C}W}^{(*,l)}$, P_i is an induced 2-path as $M(v_i) \in W_l$ and according to Property (P₁), and $P_i \subseteq V(G')$ as $M(v_i) \in V(G')$ and $\{u_i, v_i\} \subseteq B$. For any $P_i^* \in \mathcal{P}_{\bar{C}W}^*$, P_i is an induced 2-path as $M(u_i)$ is an edge of $G(S)$ of color u_i . Moreover, $P_i \subseteq V(G')$ as $M(u_i) \subseteq V(G')$ and $u_i \in C$. Property 3 is direct from the definition, and Property 2 is verified as, inside W , all used vertices are defined using the matching and, outside W , we have Property 3.

Size requirement. Now our objective is to upper bound $|V(G')|$. Let us assume first that $W \neq \emptyset$, and thus that $(\tilde{G}, \chi)\langle W, B, C \rangle$ is not the empty graph. Let us start with $|V(M)|$. Recall that the set of colors in $(\tilde{G}, \chi)\langle W, B, C \rangle$ is $C \cup \bigcup_{i \in \bar{F}} B_i = C \cup B_{\neq \emptyset}$. As edges of colors in C has size two and edges of colors in $B_{\neq \emptyset}$ have size one, and M contains one edge of each color, we get $|V(M)| = 2|C| + |B_{\neq \emptyset}|$.

We are ready to bound the size of G' :

$$\begin{aligned}
|V(G')| &= |B| + |C| + |V(M)| \\
&= |B_\emptyset| + |B_{\neq \emptyset}| + |C| + 2|C| + |B_{\neq \emptyset}| \\
&\leq |B_\emptyset| + |B_{\neq \emptyset}|(1 + 2c_1) + 3|C| \\
&\leq (1 + 2c_1)s(W, B) + 3|C| \\
&\leq (1 + 2c_1)^2|C^<| + 3|C|, \text{ according to the size requirement 1} \\
&\leq (1 + 2c_1)^2(|C^<| + |C|) \\
&= (1 + 2c_1)^2|C^0| \\
&\leq 3(1 + 2c_1)^2k
\end{aligned}$$

If $W = \emptyset$, then $(\tilde{G}, \chi)\langle W, B, C \rangle$ is the empty graph and $M = \emptyset$ and the above equations still hold. \square

Now, let us pay attention to the cases where the reduction rule provides a small vertex cover. Notice that for these cases, we do not necessarily need the considered partial decompositions to be clean.

Lemma 5 (Case 1 of small vertex cover). *Let (W, B, C) be a partial decomposition. Suppose that there is a non-empty set of colors X such that $(\tilde{G}, \chi)\langle W, B, C \rangle[X]$ admits a vertex cover T such that $|T| \leq c_1|X|$ (see Figure 4). Let $X^B = X \cap B$ and $X^C = X \cap C$. Let $S(X^B) = \{i \in \bar{F} \mid X^B \cap B_i \neq \emptyset\}$ be the set of buckets containing a color in X^B and $W(X^B) = \bigcup_{i \in S(X^B)} W_i$.*

- (Case 1:) If $|X^B| \leq |X^C|$, let $W' = W \setminus T$, $B' = B \cup T \cup X^C$, and $C' = C \setminus X^C$. Then (W', B', C') is a partial decomposition.

Proof. Partition requirements. The only non-trivial property is that (W', B') is a nice pair. Now we will prove it using the definition of nice pair (rather than providing a bucket decomposition of (W', B')). Let $Z = T \cup X^C$. Recall that $W' = W \setminus T$ and $B' = B \cup Z$. We clearly have that $W' \cap B' = \emptyset$, $W' \subseteq W^0$ and thus we only have to prove that for any induced 2-path P of $G[W' \cup B']$, $|P \cap W'| \leq 1$. Notice first that, as T is a vertex cover of $(\tilde{G}, \chi)\langle W, B, C \rangle[X]$, it is in particular a vertex cover for all edges of colors X^C , implying that for any edge e with color $c \in X^C$, $T \cap V(e) \neq \emptyset$. This implies that there is no induced 2-path P where $|P \cap X^C| = 1$ and $|P \cap W'| = 2$, as $P \cap W'$ would be an edge (of a color in X^C) not covered by T . Moreover, there is no induced 2-path P where $|P \cap T| = 1$ and $|P \cap W'| = 2$ as this would imply $P \subseteq W$, and $G[W]$ is a disjoint union of cliques. These last two observations imply that there is no induced 2-path P where $|P \cap Z| = 1$ and $|P \cap W'| = 2$. Moreover, as (W, B) was a nice pair, there is also no induced 2-path P where $|P \cap B| = 1$ and $|P \cap W| = 2$. Therefore, there is no induced 2-path P where $|P \cap B| = 1$ and $|P \cap W'| = 2$, as $W' \subseteq W$. This implies that (W', B') is a nice pair.

Size requirements. Let us prove that $s(W', B') \leq (1+2c_1)|C'^{<}|$. Recall first that $C'^{<} = C^0 \setminus C'$ and $C' = C \setminus X^C$, implying that $C'^{<} = C^{<} \cup X^C$. Notice also that $|T| \leq c_1|X| = c_1(|X^B| + |X^C|) \leq 2c_1|X^C|$, as we are in Case 1. Observe first that $B_\emptyset \subseteq B'_\emptyset$, as $W' \subseteq W$. This implies that there exists $Z_1 \subseteq Z$ such that $B'_\emptyset = B_\emptyset \cup Z_1$.

We remark that $B_{\neq \emptyset} \subseteq B'_{\neq \emptyset}$ is not true as a vertex v in a B_i may move to B'_\emptyset if $W_i \subseteq T$ (for example the case for vertices in B_3 in Figure 4). We have

$$\begin{aligned} s(W', B') &= \frac{|B'_\emptyset|}{1+2c_1} + |B'_{\neq \emptyset}| \\ &= \frac{|B_\emptyset| + |Z_1|}{1+2c_1} + |B'_{\neq \emptyset}| \\ &\leq \frac{|B_\emptyset|}{1+2c_1} + |Z_1| + |B'_{\neq \emptyset}| \end{aligned}$$

Now, observe that

$$\begin{aligned} B'_\emptyset \cup B'_{\neq \emptyset} &= B \cup Z \\ \Rightarrow B_\emptyset \cup Z_1 \cup B'_{\neq \emptyset} &= B_\emptyset \cup B_{\neq \emptyset} \cup Z \\ \Rightarrow Z_1 \cup B'_{\neq \emptyset} &= B_{\neq \emptyset} \cup Z \end{aligned}$$

Thus, we get

$$\begin{aligned}
s(W', B') &\leq \frac{|B_\emptyset|}{1+2c_1} + |B_{\neq\emptyset}| + |Z| \\
&\leq \frac{|B_\emptyset|}{1+2c_1} + |B_{\neq\emptyset}| + |T| + |X^C| \\
&\leq \frac{|B_\emptyset|}{1+2c_1} + |B_{\neq\emptyset}| + (1+2c_1)|X^C| \\
&= s(W, B) + (1+2c_1)|X^C| \\
&\leq (1+2c_1)|C^<| + (1+2c_1)|X^C|, \text{ as } (W, B, C) \text{ is a partial decomposition} \\
&= (1+2c_1)|C'^<|
\end{aligned}$$

□

Let us now prove that the vertex cover and colors we add in the kernel output in Case 2 are small.

Lemma 6 (Case 2 of small vertex cover). *Let (W, B, C) be a partial decomposition. Suppose that there is a non-empty set of colors X such that $(\tilde{G}, \chi)\langle W, B, C \rangle[X]$ admits a vertex cover T such that $|T| \leq c_1|X|$ (see Figure 4). Let $X^B = X \cap B$ and $X^C = X \cap C$. Let $S(X^B) = \{i \in \bar{F} \mid X^B \cap B_i \neq \emptyset\}$ be the set of buckets containing a color in X^B and $W(X^B) = \bigcup_{i \in S(X^B)} W_i$.*

- (Case 2:) If $|X^B| > |X^C|$, let $W' = W \setminus W(X^B)$, $B' = B \cup W(X^B)$, and $C' = C$. Then (W', B', C') is a partial decomposition.

Proof. Let us first prove the following claim.

Claim 1. *We have $|W(X^B)| \leq 2c_1|B_{S(X^B)}|$.*

Proof of Claim 1. Notice first that, as T is a vertex cover of $(\tilde{G}, \chi)\langle W, B, C \rangle[X]$, then for any $i \in S(X^B)$ and any color $v \in X^B \cap B_i$, as $(\tilde{G}, \chi)\langle W, B, C \rangle[X]$ contains all edges $\{u\} \in W_i$ of color v , it implies that $T \supseteq W_i$. This implies that $T \supseteq W(X^B)$. Moreover, by Corollary 1, $|T| \leq c_1|X| = c_1(|X^B| + |X^C|) \leq 2c_1|X^B|$, as we are in Case 2. As $|W(X^B)| \leq |T|$ and $|X^B| \leq |B_{S(X^B)}|$, we obtain the result of the claim. □

Let us now prove that (W', B', C') is a partial decomposition.

Partition requirements. The only non-trivial property is that (W', B') is a nice pair, which we will prove by providing a bucket decomposition of (W', B') . Let us define the following partitions which, informally, correspond to the result of moving (for any $i \in S(X^B)$) each W_i to B_i :

- the partition $\{W'_i \mid i \in [i_0]\}$ of W' , where $W'_i = W_i$ if $i \notin S(X^B)$ and $W'_i = \emptyset$ otherwise,
- the partition $\{B'_i \mid i \in [i_0]\}$ of B' , where $B'_i = B_i$ if $i \notin S(X^B)$ and $B'_i = \emptyset$ otherwise,
- $B'_\emptyset = B_\emptyset \cup W(X^B) \cup B_{S(X^B)}$

Let us check that these partitions verify conditions of a bucket decomposition. The only non-trivial part is to verify that all vertices v added to B_\emptyset have $N(v) \cap W' = \emptyset$. If $v \in W(X^B)$, then $N(v) \cap W' = \emptyset$, as there are no edges between the W_i 's. If $v \in B_{S(X^B)}$, where $v \in B_i$ for some

$i \in S(X^B)$, then, as $N(v) \cap W = W_i$ and $W_i \subseteq W(X^B)$, we also get $N(v) \cap W' = \emptyset$. Thus, Property 1 of Lemma 2 implies that (W', B') is a nice pair.

Size requirements. Let us prove that $s(W', B') \leq (1 + 2c_1)|C'^{<}$. Recall first that $C'^{<} = C^{<}$, as $C' = C$. As (W, B, C) is a partial decomposition, we have $s(W, B) \leq (1 + 2c_1)|C^{<}$, and thus it only remains to prove that $s(W', B') \leq s(W, B)$. By definition we have $s(W, B) = \frac{|B_\emptyset|}{1+2c_1} + |B_{\neq\emptyset}|$. Recall that $B_{\neq\emptyset} = B_{\bar{F}}$, and observe that there is a partition $\{B_{S(X^B)}, B_{\bar{F} \setminus S(X^B)}\}$ of $B_{\bar{F}}$. This implies that $s(W, B) = \frac{|B_\emptyset|}{1+2c_1} + |B_{S(X^B)}| + |B_{\bar{F} \setminus S(X^B)}|$. Now, observe that the bucket decomposition of (W', B') is exactly the partition of W' and B' defined above and that $B'_{\neq\emptyset} = B_{\bar{F} \setminus S(X^B)}$. This implies that

$$\begin{aligned} s(W', B') &= \frac{|B'_\emptyset|}{1 + 2c_1} + |B'_{\neq\emptyset}| \\ &= \frac{|B_\emptyset| + |W(X^B)| + |B_{S(X^B)}|}{1 + 2c_1} + |B_{\bar{F} \setminus S(X^B)}| \\ &\leq \frac{|B_\emptyset| + (1 + 2c_1)|B_{S(X^B)}|}{1 + 2c_1} + |B_{\bar{F} \setminus S(X^B)}|, \text{ by Claim 1} \\ &= s(W, B) \end{aligned}$$

□

4.5 Analysis of the overall kernel

We are now ready to define the unique, other than the cleaning phases, rule for our kernelization. This rule summarizes the different cases arising in the previous lemmas.

Definition 4 (Reduction Rule for I2PP). *Given a clean partial decomposition (W, B, C) , with the associated partition for B and W denoted by $(B_\emptyset, B_1, \dots, B_{i_0}, W_1, \dots, W_{i_0})$ with $F = \{i \in [i_0] \mid W_i = \emptyset\}$ and $\bar{F} = [i_0] \setminus F$, let us define the output $R(W, B, C)$ of the rule R as follows:*

- Compute using Corollary 1 (with a suitable ϵ precised later) if there exists a rainbow matching M in the p -edge-colored multigraph $(\tilde{G}, \chi)\langle W, B, C \rangle$ (where $p = |C| + \sum_{i \in [i_0]} |B_i|$ - notice that $p = \mathcal{O}(|V(G)|)$).
- If M exists, then return $V(M) \cup B \cup C$.
- Otherwise, let X be the non-empty set of colors such that $(\tilde{G}, \chi)\langle W, B, C \rangle[X]$ admits a vertex cover T such that $|T| \leq c_1|X|$ (see Figure 4). Let $X^B = X \cap B$ and $X^C = X \cap C$. Let $S(X^B) = \{i \in \bar{F} \mid X^B \cap B_i \neq \emptyset\}$ be the set of buckets containing a color in X^B and $W(X^B) = \bigcup_{i \in S(X^B)} W_i$.
- If $|X^B| \leq |X^C|$ (case 1), let $W' = W \setminus T$, $B' = B \cup T \cup X^C$, and $C' = C \setminus X^C$. Return (W', B', C') .
- Otherwise (case 2), let $W' = W \setminus W(X^B)$, $B' = B \cup W(X^B)$, and $C' = C$. Return (W', B', C') .

We then obtain the following.

Lemma 7. *Given a clean partial decomposition (W, B, C) , Rule $R(W, B, C)$ either returns:*

- a set $A \subseteq V(G)$ such that, if $G' = G[A]$, then (G', k) and (G, k) are equivalent instances of I2PP and $|A| \leq 3(1 + 2c_1)^2k$.
- or a partial decomposition (W', B', C') such that $|\bar{F}'| + |C'| < |\bar{F}| + |C|$

Proof. If R finds a rainbow matching M in $(\tilde{G}, \chi)\langle W, B, C \rangle$, then Lemma 4 immediately implies that the set $A = V(M) \cap B \cup C$ verifies the claimed properties. Let us now consider that R does not find a rainbow matching.

If R falls into Case 1, then according to Lemma 5, (W', B', C') is a partial decomposition. Moreover, in Case 1, as $X \neq \emptyset$, either $X^B \neq \emptyset$, implying that $S(X^B) \neq \emptyset$ and thus that all cliques W_i for $i \in S(X^B)$ become empty and therefore $|\bar{F}|$ strictly decreases (and $|C|$ does not increase). Otherwise, $X^C \neq \emptyset$, implying that $|C|$ strictly decreases (and $|\bar{F}|$ does not increase).

If R falls into Case 2, then according to Lemma 6, (W', B', C') is a partial decomposition. Moreover, in Case 2, $X^B \neq \emptyset$, implying as previously that $|\bar{F}|$ strictly decreases (and $|C|$ does not change). \square

Finally, we can prove the kernelization algorithm for I2PP stated in Theorem 2.

Proof of Theorem 2. Given an input (G, k) , we define the kernelization Algorithm B which starts with a greedy localization phase, as explained in Subsection 4.2. Assume that it does not find a packing of size k and therefore computes a greedy localized pair (C^0, W^0) . We then consider the partition (W_0, \emptyset, C_0) of $V(G)$. It is straightforward to check that this partition is a partial decomposition of G . Using Lemma 3, we then obtain the first clean partial decomposition (W, B, C) of the process. Now, a step of Algorithm B will be made of Reduction Rule R for I2PP and a cleaning phase. Algorithm B exhaustively performs steps, obtaining a clean partial decomposition at the end of each step and stopping only when it falls into the matching case.

Let us prove, by induction on $|\bar{F}| + |C|$, that applying exhaustively steps of Algorithm B terminates in polynomial time and outputs an equivalent instance (G', k) where $|V(G')| \leq 3(1 + 2c_1)^2k$. In order to this, notice first that in the case where Rule R applied on a clean partial decomposition (W, C, B) returns a partial decomposition (W', B', C') with $|\bar{F}'| + |C'| < |\bar{F}| + |C|$, we apply a cleaning phase on (W', B', C') to obtain a clean partial decomposition (W'', B'', C'') . Then it is easy to show that we also have $|\bar{F}''| + |C''| < |\bar{F}| + |C|$. Indeed, in the cleaning phase, the set W' is unchanged, so it is for F' , and the size of C' can only decrease. We obtain $|\bar{F}''| + |C''| \leq |\bar{F}'| + |C'| < |\bar{F}| + |C|$ as desired.

Now, we can finish the analysis of the process. If $\bar{F} = C = \emptyset$, then $W = \emptyset$, $(\tilde{G}, \chi)\langle W, B, C \rangle$ is the empty graph, and we consider that R returns the rainbow matching $M = \emptyset$. Thus, according to Lemma 7, the rule R outputs a set $A \subseteq V(G)$ such that, if $G' = G[A]$, then (G', k) and (G, k) are equivalent instances of I2PP and $|A| \leq 3(1 + 2c_1)^2k$. Now, if $|\bar{F}| + |C| > 0$, it is immediate, using induction, by Lemma 7 and the previous remarks about cleaning phases, that B terminates in polynomial time and outputs an equivalent instance (G', k) where $|V(G')| \leq 3(1 + 2c_1)^2k$.

As $c_1 = 4 + \varepsilon$, we conclude that $|A| \leq (243 + 12\varepsilon^2 + 108\varepsilon)k = (243 + \varepsilon')k$ for a suitable ε' as required. \square

5 Linear kernel for I2PHS

In this section we focus on the induced 2-paths hitting set problem, restated below.

INDUCED 2-PATHS HITTING SET (I2PHS)

Parameter: k .

Input: (G, k) where G is a graph and $k \in \mathbb{N}$

Question: Is there an induced 2-paths Hitting Set of G of size at least k ?

We obtain a linear kernel for this problem, which is obtained by the same algorithm that the one designed in the previous section. The only thing that we will have to check is that, when the algorithm stops, we obtain an equivalence instance than the input instance for I2PHS.

Theorem 3. *There exists some function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that for every $\varepsilon > 0$, there exists an algorithm that, given an instance (G, k) of I2PHS outputs a set $A \subseteq V(G)$ such that $(G[A], k)$ is an equivalent instance where $|A| \leq (243 + \varepsilon)k$. Moreover this algorithm runs in time $\mathcal{O}(|V(G)|^{f(\varepsilon)})$. In other words, I2PHS admits a kernel of a linear number of vertices.*

The key tool to obtain the linear kernel for I2PHS is the following lemma which is the analog of Lemma 4 for I2PHS. All the notations and definitions follow previous section.

Lemma 8 (Case of rainbow matching for I2PHS). *Let (W, B, C) be a clean partial decomposition. Suppose that the colored multigraph $(\tilde{G}, \chi)\langle W, B, C \rangle$ admits a rainbow matching M . Let $A = V(M) \cup B \cup C$, and $G' = G[A]$. Then,*

1. (G, k) and (G', k) are equivalent instances of I2PHS,
2. $|A| \leq 3(1 + 2c_1)^2k$.

Proof. The size requirement concerning A follows from Lemma 4. Let us prove that (G, k) and (G', k) are equivalent instances of I2PHS. As G' is an induced subgraph of G , it is clear that if G admits a 2-induced paths hitting set of size at most k , then it is also the case for G' .

For the converse direction, assume that G' admits a 2-induced paths hitting set X of size at most k , and let us see how to build one for G . By removing vertices from X if necessary, we can assume that X is a induced 2-paths hitting of G' minimal by inclusion. Let us first precise the structure of X . As (W, B, C) is clean, for every vertex c of C there exists at least one edge of $(\tilde{G}, \chi)\langle W, B, C \rangle$ with color c . And as M is a rainbow matching of this graph, there exist v_c and w_c in W such that $v_c w_c$ is an edge of M and so, such that $\{v_c, w_c, c\}$ induces a 2-path P_c of G . Notice that X has to intersect P_c for every vertex c of C . Denote by M_C the set $\bigcup_{c \in C} \{v_c, w_c\}$ and by M_B the set $M \setminus M_C$. By the previous remark, we obtain that $|X \cap (M_C \cup C)| \geq |C|$. Let us focus now on M_B . For any $i \in [i_0]$, for every vertex $b \in B_i$, every vertex of W_i receives color b . So, as M is a rainbow matching of $(\tilde{G}, \chi)\langle W, B, C \rangle$, it has to contain a vertex of W_i with color b . That is M_B contains exactly $|B_i|$ vertices of W_i for every $i \in [i_0]$. Moreover, assume that for $i \in [i_0]$ the set X contains a vertex x of $M_B \cap W_i$. By minimality of X there exists an induced 2-path P of G' with $X \cap V(P) = \{x\}$. As (W, B) is a nice pair, we know that $P = xby$ with $b \in B_i$ and $y \in B \cup C$. In particular, by Lemma 2, for any $x' \in M_B \cap W_i$ the path $x'by$ is also an induced 2-path. Thus, as $b \notin X$ and $y \notin X$ we must have $x' \in X$, and in all, we obtain $M_B \cap W_i \subseteq X$.

Now, we can modify X in order to obtain an induced 2-paths hitting set of G . Denote by J the subset of $[i_0]$ corresponding to the indices of W_i 's intersected by X , that is $J = \{i \in [i_0] : X \cap W_i \neq \emptyset\}$. Let us define $X' = C \cup (X \cap B) \cup \bigcup_{i \in J} B_i$ and show that X' is induced 2-paths hitting set of G and has size no more than $|X|$. For the latter property, using the previous remarks, we

have:

$$\begin{aligned}
|X'| &\leq |C| + |(X \cap B)| + \sum_{i \in J} |B_i| \\
&\leq |X \cap (C \cup M_C)| + |(X \cap B)| + \sum_{i \in J} |M_B \cap W_i| \\
&\leq |X \cap (C \cup M_C)| + |(X \cap B)| + |X \cap M_B| \\
&\leq |X|
\end{aligned}$$

To prove that X' is an induced 2-paths hitting set of G , assume by contradiction that $P = xyz$ is an induced 2-path of $G \setminus X'$. As $C \subseteq X'$ we have $V(P) \subseteq W \cup B$, and as (W, B) is a nice pair, we have for instance $x \in W$, $y \in B$ and $z \in B$. So, by Lemma 2, there exists more precisely $i \in [i_0]$ such that $x \in W_i$ and $y \in B_i$. The path P is not intersected by X' , meaning in particular that $i \notin J$ (as otherwise we would have $B_i \subseteq X'$). So we have $y \notin X$ and $z \notin X$ (as $X' \cap B = X \cap B$) and $W_i \cap X = \emptyset$. However, M contains at least one vertex x' of W_i , with color y for instance. But then $x'y z$ is an induced 2-path of G' not intersected by X , a contradiction. \square

Now, using Lemma 8 instead of Lemma 4 in the proof of Lemma 7, we directly obtain the analog of this latter one for I2PHS.

Lemma 9. *Given a clean partial decomposition (W, B, C) , Rule $R(W, B, C)$ either returns:*

- a set $A \subseteq V(G)$ such that, if $G' = G[A]$, then (G', k) and (G, k) are equivalent instances of I2PHS and $|A| \leq 3(1 + 2c_1)^2 k$.
- or a partial decomposition (W', B', C') such that $|\bar{F}'| + |C'| < |\bar{F}| + |C|$

Now, proof of Theorem 3 works the same than the kernelization process for I2PP, that is Theorem 2. We start by computing a localized pair (C^0, W^0) . The set C^0 induces a packing of induced 2-paths. If there is more than k induced 2-paths in the packing, then G has no induced 2-paths hitting set of size at most k . Otherwise, we consider the initial partial decomposition (W_0, \emptyset, C_0) of $V(G)$. Then, we exhaustively alternate a cleaning phase with an application of the rule R , until this last one falls into the matching case. As in the proof of Theorem 2, using Lemma 9 an induction on $|\bar{F}| + |C|$ shows that this later case appears after a polynomial number of steps. Then we conclude with Lemma 8.

6 An (almost) linear kernel for TPT

6.1 Notations

Given a tournament T , a *triangle* Δ in T is a subgraph on three vertices where each vertex has in-degree and out-degree exactly one, i.e. a directed cycle of length three. A *triangle-packing* $\mathcal{P} = \{\Delta_i, i \in [x]\}$ is a set of vertex-disjoint triangles of T . The *size* of \mathcal{P} is $|\mathcal{P}| = x$.

TRIANGLE-PACKING IN TOURNAMENT (TPT)

Parameter: k

Input: (T, k) where T is a tournament and $k \in \mathbb{N}$

Question: Is there a triangle-packing of size at least k ?

In this section we prove the following theorem.

Theorem 4. *There exists an algorithm that, given an instance (T, k) of TPT outputs a set $S \subseteq V(T)$ such that $(T[S], k)$ is an equivalent instance of (T, k) where, for every δ with $1 < \delta \leq 2$, we have $|S| \leq 6534 \cdot c(\delta) \cdot k^\delta$ (where $c(\delta) = \max(\frac{20}{(2^\delta-2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$). In other words, for any δ with $1 < \delta \leq 2$ TPT admits a kernel with $6534 \cdot c(\delta)k^\delta$ vertices.*

By fixing the suitable value $\delta_0 = 1 + \frac{\sqrt{\log 21}}{\sqrt{\log k}}$ (assuming the non-trivial case where $k \geq 2$) we obtain the following.

Corollary 2. *TPT admits a kernel with $k^{1+\frac{\mathcal{O}(1)}{\sqrt{\log k}}}$ vertices.*

Proof. Let us upper bound $c(\delta_0)$. For any $1 < \delta \leq 2$, we have $c(\delta) \leq c'(\delta)$ where $c'(\delta) = 21^{\frac{1}{\delta-1}}$. Indeed, $\frac{20}{(2^\delta-2)} \leq \frac{21}{\delta-1} \leq 21^{\frac{1}{\delta-1}}$, and on the other hand, $(\frac{21}{\delta})^{\frac{1}{\delta-1}} \leq 21^{\frac{1}{\delta-1}}$. Thus, the vertex size of the kernel given by Theorem 4 for δ_0 is at most

$$\begin{aligned}
6534 \cdot c'(\delta_0)k_0^\delta &= 6534 \cdot 2^{(\sqrt{\log 21}\sqrt{\log k})} k^{\delta_0} \\
&= 6534 \cdot 2^{(\sqrt{\log 21}\sqrt{\log k})} k \cdot 2^{(\sqrt{\log 21}\sqrt{\log k})} \\
&= 6534 \cdot k \cdot 2^{2 \cdot \sqrt{\log 21}\sqrt{\log k}} \\
&= \mathcal{O}(k \cdot 2^{\frac{9.17 \cdot \log k}{\sqrt{\log k}}}) \\
&= \mathcal{O}(k \cdot k^{\frac{9.17}{\sqrt{\log k}}}) \\
&= k^{1+\frac{\mathcal{O}(1)}{\sqrt{\log k}}}
\end{aligned}$$

□

6.2 Preliminary phase: greedy localization

Given an instance (T, k) of TPT, we first greedily compute a maximal set of vertex-disjoint triangles. If we get at least k triangles, then (T, k) is a positive instance of TPT. Otherwise we denote by C^0 the set of vertices contained in the triangles of the greedy packing and by W^0 the set $V(T) \setminus C^0$. We denote by t_0 the size of W^0 . The vertices of W^0 clearly induce an acyclic subtournament of T , and we call such a partition (C^0, W^0) a *greedily localized pair* of T for TPT.

In the remainder of the section, we consider that W^0 is sorted according to its topological ordering³ and we number the elements of W^0 following this order, that is $W^0 = \{w_1, \dots, w_{t_0}\}$ with $(w_i, w_j) \in A(T)$ iff $i < j$. Finally, for any subset W of W^0 and any $i, j \in [t_0]^2$ we write $W_{[i,j]} = \{w_k \in W \mid k \in [i, j]\}$ and $W_{[i,j[} = W_{[i,j]} \setminus \{w_j\}$.

6.3 Nice pairs, buckets, and partial decomposition

Recall that a basic principle of our technique is to maintain a tuple (W, B, C, \dots) called a partial decomposition where (W, B) is a nice pair. To capitalize on the specific problem considered here, we study in this section which structural properties hold in a nice pair.

In the entirety of this section we consider that we are given an input (T, k) of TPT, and a greedily localized pair (C^0, W^0) for our input (T, k) , where $W^0 = \{w_1, \dots, w_{t_0}\}$.

³Notice that the topological ordering of an acyclic tournament is unique.

Definition 5. Given an instance (T, k) of TPT, and two subsets W, B of $V(T)$, we say that $\psi = (W, B)$ is a nice pair if $W \subseteq W^0$ and $B \cap W = \emptyset$, and for any triangle Δ of $T[W \cup B]$ we have $|\Delta \cap W| \leq 1$. Notice that B may contain vertices of C^0 , as well as vertices of $W^0 \setminus W$.

An illustration of next proposition is depicted in [Figure 5](#).

Proposition 1. Let $\psi = (W, B)$ be a nice pair. There exists a unique $S^\psi \subseteq [t_0] \cup \{\infty\}$, where ∞ is a token representing some value greater than t_0 , and a unique partition $\{B_i \mid i \in S^\psi\}$ of B into non-empty sets such that for any $i \in S^\psi$, the set B_i contains every vertex v of B where

- all arcs between $W_{[1, i-1]}$ and v are oriented from $W_{[1, i-1]}$ to v
- all arcs between $W_{[i, t_0]}$ and v are oriented from v to $W_{[i, t_0]}$

Such a partition, along with the choice of S^ψ , is called a bucket decomposition of ψ , and sets B_i are called buckets.

Proof. For any $i \in [t_0] \cup \{\infty\}$, denote by B_i the set containing every vertex of B that is dominated by $W_{[1, i-1]}$ and dominates $W_{[i, t_0]}$. As there is no cycle of length two in T , the B_i are pairwise disjoint. Moreover, let b be a vertex of B . If W dominates b , then we have $b \in B_\infty$. Otherwise, we denote by i_0 the minimum integer j such that b dominates w_j and $w_j \in W$. If there exists $i > i_0$ such that $w_i b \in A(T)$ and $w_i \in W$, then $bw_i w_i$ would be a triangle containing two vertices in W and one in B , which is not possible. Moreover, by definition of i_0 , the set $W_{[1, i_0-1]}$ dominates b .

So, we have $b \in B_{i_0}$, and more generally $(B_i)_{i=1, \dots, t_0, \infty}$ forms a partition of B . To conclude, we just denote by S^ψ the subset of indices i in $[t_0] \cup \{\infty\}$ with $B_i \neq \emptyset$. \square

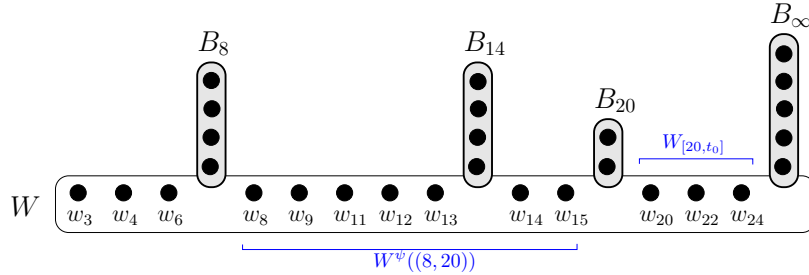


Figure 5: A nice pair $\psi = (W, B)$ and its bucket decomposition with $S^\psi = \{8, 14, 20, \infty\}$. The arcs inside W and from W to the B_i 's all go from left to right, in increasing order of the indices. A bucket interval $I_0 = (8, 20)$ and its associated set of vertices $W^\psi(I)$. Let $I_1 = (14, \infty)$, $I_2 = (20, \infty)$ and $\mathcal{I} = \{I_0, I_1, I_2\}$. We have $I_0 \prec I_1$ and $I_2 \sqsubseteq I_1$.

Informally, in a bucket decomposition, all vertices from the same bucket have the same neighborhood in W . We will use several times the next observation following from the definition of a nice pair, and asserting that any triangle inside $W \cup B$ contains at least two vertices from the buckets. We refer again the reader to [Figure 5](#) for the next definition.

Definition 6 (bucket interval). Let us consider given a nice pair $\psi = (W, B)$. We say that $I = (l_I, r_I)$ is a bucket interval of ψ if $\{l_I, r_I\} \subseteq S^\psi$ and $l_I < r_I$. We denote by $\mathcal{BI}(\psi)$ (or $\mathcal{BI}(W, B)$) the set of bucket intervals of ψ .

Given two bucket intervals I_1 and I_2

- we say that $I_1 \sqsubseteq I_2$ iff $[l_{I_1}, r_{I_1}] \subseteq [l_{I_2}, r_{I_2}]$, and
- $I_1 \prec I_2$ iff $l_{I_1} < l_{I_2} < r_{I_1} < r_{I_2}$.
- Moreover, we define $I_1 \sqcup I_2 = (\min\{l_{I_1}, l_{I_2}\}, \max\{r_{I_1}, r_{I_2}\})$, and
- if $I_1 \prec I_2$, we set $I_1 \cap I_2 = (l_{I_2}, r_{I_1})$

Given any bucket interval $I \in \mathcal{BI}(\psi)$, we write

- $B^\psi(I) = [l_I, r_I] \cap S^\psi$ the indices of buckets in I
- $W^\psi(I) = W_{[l_I, r_I[}$
- $P^\psi(I) = \{I' \in \mathcal{BI}(\psi) \mid I' \sqsubseteq I\}$
- for any $X \subseteq \mathcal{BI}(\psi)$, we write $P_X^\psi(I) = P^\psi(I) \cap X$.

When the nice pair is clear from context, we will drop the ψ from the previous notations.

Observation 1. Given a nice pair (W, B) , for any triangle Δ in $T[W \cup B]$, either $\Delta \subseteq V(B)$, or there exists a bucket interval $I \in \mathcal{BI}(\psi)$ such that

- $|\Delta \cap B_{l(I)}| = |\Delta \cap B_{r(I)}| = 1$, $\Delta \cap W \subseteq W(I)$, and
- for any $v \in W(I)$, $(\Delta \cap B) \cup \{v\}$ is still a triangle.

Recall that in a given round of the rainbow matching technique, if we do not find a rainbow matching then we find a set of vertices $U \subseteq W$ and $X^C \subseteq C^0 \setminus B$ (having some properties corresponding to the hypothesis of [Lemma 10](#)), that we have to add to B , while preserving in particular that what we obtain is still a nice pair. This motivates the next lemma.

Lemma 10. Let (W, B) be a nice pair. Let $U \subseteq W$ and $X^C \subseteq C^0 \setminus B$ such that for any triangle Δ of $T[W \cup X^C]$ we have $|\Delta \cap (W \setminus U)| \leq 1$. Let $W' = W \setminus U$ and $B' = B \cup U \cup X^C$. Then, (W', B') is a nice pair.

Proof. As $X^C \subseteq C^0$ and $W \cap C^0 = \emptyset$, we have that $W' \cap B' = \emptyset$. Let us now suppose, towards a contradiction, that there exists a triangle Δ of $T[W' \cup B']$ such that $|\Delta \cap W'| \geq 2$. This implies that $|\Delta \cap W'| = 2$ and $|\Delta \cap B'| \geq 1$, as we cannot have $|\Delta \cap W'| = 3$, because $W' \subset W^0$ and $T[W^0]$ is acyclic. Let $\Delta \cap B' = \{u\}$ and $\Delta \cap W' = \{v, w\}$. If $u \in X^C$, then this contradicts the hypothesis that for any triangle Δ of $T[W \cup X^C]$ we have $|\Delta \cap (W \setminus U)| \leq 1$. If $u \in U$, then $\Delta \subseteq W$, contradicting the fact that $T[W^0]$ is acyclic. Finally, if $u \in B$, then this contradicts the fact that (W, B) is a nice pair as, for any triangle Δ of $T[W \cup B]$, we should have $|\Delta \cap W| \leq 1$. \square

As buckets will correspond to vertices that we want to keep in our kernel, we need to control their size, motivating the following definition, which is illustrated in [Figure 6](#).

Definition 7 (Bucket partition of a nice pair). Let f be a function from \mathbb{N} to \mathbb{R}^+ and $\psi = (W, B)$ be a nice pair. Let $B^C = B \cap C^0$, and $B^W = B \cap W^0$. For any $i \in S^\psi$, let $B_i^C = B_i \cap B^C$. For any partition (B^{W_1}, B^{W_2}) of B^W and any $i \in S^\psi$, let $B_i^{W_1} = B_i \cap B^{W_1}$, $B_i^{W_2} = B_i \cap B^{W_2}$ and let $S_i = B_i^C \cup B_i^{W_1}$.

A bucket partition of a nice pair $\psi = (W, B)$ is a partition (B^{W_1}, B^{W_2}) of B^W such that

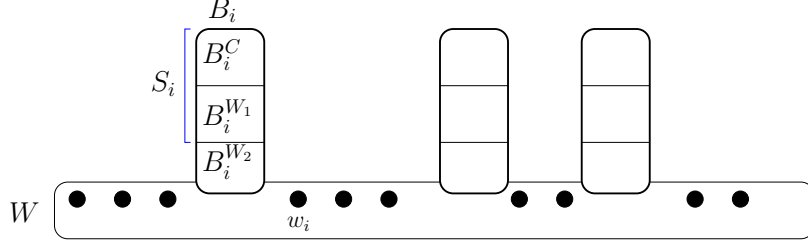


Figure 6: A bucket partition

1. for any $i \in S^\psi$, $|S_i| \geq 1$
2. $|B^{W_1}| \leq 10|B^C|$.

We say that a bucket partition has local size f if for any $i \in S^\psi$, $|B_i^{W_2}| \leq f(|S_i|)$.

Notice that the size condition $|B^{W_1}| \leq 10|B^C|$ required in the bucket partition is a “global” constraint on the size of the B^{W_1} , while the condition size $|B_i^{W_2}| \leq f(|S_i|)$ required by the local size function is a “local” constraint on every bucket. We introduced this local notion of size as a way to control more precisely the number of vertices added to B (and thus to the kernel output), which will be critical when typically adjacent buckets are merged using the *add* operation of Definition 14. However, when the kernel will find a rainbow matching and stop, the only role of these local sizes will be to upper bound the total size of B .

The object that our kernel will manipulate is a partial decomposition, as defined below.

Definition 8 (Partial decomposition). *We say that a tuple $(W, B, C, B^{W_1}, B^{W_2})$ is a partial decomposition if*

1. *there is a partition $V(T) = W \cup B \cup C$*
2. $C \subseteq C^0$
3. (W, B) *is a nice pair*
4. (B^{W_1}, B^{W_2}) *is a bucket partition of (W, B)*

We say that a partial decomposition has local size f if (B^{W_1}, B^{W_2}) has local size f .

Moreover, we will say that a partial decomposition is clean if it satisfies the following extra condition:

1. *for every vertex $c \in C$ the tournament $T[W \cup \{c\}]$ contains a triangle.*

Here again, we have a simple process, called *the cleaning phase*, to obtain a clean partial decomposition from any partial decomposition.

Lemma 11 (Cleaning Lemma). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and X be set the of vertices x of C such that $T[W \cup \{x\}]$ does not contain any triangle. Then $(W, B \cup X, C \setminus X, B^{W_1}, B^{W_2})$ is a clean partial decomposition of T , with the same local size than $(W, B, C, B^{W_1}, B^{W_2})$.*

Proof. Let us first check that $(W, B \cup X, C \setminus X, B^{W_1}, B^{W_2})$ satisfies the requirements of a partial decomposition. Properties 1 and 2 are clearly satisfied, and by choice of X , no vertex of X is contained in a triangle with two vertices of W , so $\psi' = (W, B \cup X)$ is also a nice pair and property 3 is satisfied. Notice that vertices of X can create new buckets or be added in existing buckets of ψ , but in all cases we have $S^{\psi} \subseteq S^{\psi'}$. Finally, in the buckets partition of ψ' , the vertices of X will be added to B^C . That is, with the notations of Definition 7, we have $(B \cup X)^W = B^W$, $(B \cup X)_i^{W_j} = B_i^{W_j}$ for $i \in S^{\psi}$ and $j = 1, 2$ and $(B \cup X)_i^{W_j} = \emptyset$ for $i \in S^{\psi'} \setminus S^{\psi}$ and $j = 1, 2$. It is then straightforward to check that Properties 1. and 2., as well as the local size requirement, from Definition 7 still hold for ψ' . In all, we can conclude that $(W, B \cup X, C \setminus X, B^{W_1}, B^{W_2})$ is a clean partial decomposition with the same local size than $(W, B, C, B^{W_1}, B^{W_2})$. \square

6.4 Intervals: demand definition and basic properties

In this section we introduce the notion of *demand* for a partial decomposition. Informally, a demand is a set of bucket intervals with a value attached to each interval. The value $\text{val}(I)$ attached to interval I depends on the contents of the buckets $\{B_i \mid i \in B(I)\}$ contained in I .

The notion of demand will be used in Subsection 6.6 to define the auxiliary edge-colored multi-graph necessary in our approach. In particular, given a \sqsubseteq -minimal interval I (implying that buckets B_{l_I} and B_{r_I} are consecutive), it will be important that $\text{val}(I)$ upper bounds the size of packing \mathcal{P} where all triangles in \mathcal{P} have one vertex in B_{l_I} , one in B_{r_I} , and one in $W(I)$. Indeed, $\text{val}(I)$ will correspond to the number of vertices that we want to find in $W(I)$ in the rainbow matching we look for. Then, if we indeed find such a rainbow matching, and thus a set $Q_I \subseteq W(I)$ with $|Q_I| = \text{val}(I)$, we must be able to repack such a packing \mathcal{P} (corresponding to a part of an optimal solution) into a packing \mathcal{P}' of same size with $V(\mathcal{P}') \subseteq B_{l_I} \cup B_{r_I} \cup Q_I$ by changing vertices used in $W(I)$ to take instead vertices in Q_I .

The following definition is illustrated Figure 7.

Definition 9 (Block partition). *We call a set $\mathcal{I}^{\max} \subseteq \mathcal{BI}(\psi)$ proper if there do not exist $I_1, I_2 \in \mathcal{I}^{\max}$ such that $I_1 \neq I_2$ and $I_1 \sqsubseteq I_2$.*

Given a proper set $\mathcal{I}^{\max} \subseteq \mathcal{BI}(\psi)$ of bucket intervals we define the block partition of \mathcal{I}^{\max} , denoted $\{Z_1, \dots, Z_t\}$, as follows. Let us order \mathcal{I}^{\max} according to the left points, meaning that $\mathcal{I}^{\max} = \{I_1, \dots, I_x\}$, where $l_i < l_{i+1}$ (and $r_i < r_{i+1}$) for any i .

We find the largest x_1 such that $I_1 \prec I_2 \prec \dots \prec I_{x_1}$ and define $Z_1 = \{I_i \mid x_0 + 1 \leq i \leq x_1\}$ where $x_0 = 0$. Then, we find the largest x_2 such that $I_{x_1+1} \prec I_{x_1+2} \prec \dots \prec I_{x_2}$, and define $Z_2 = \{I_i \mid x_1 + 1 \leq i \leq x_2\}$. We continue until we have a partition $\{Z_\ell, \ell \in [t]\}$ of \mathcal{I}^{\max} (implying $x_t = x$)⁴.

Notice that, for any $\ell \in [t-1]$, $I_{x_\ell} \not\prec I_{x_{\ell+1}}$, implying that $r_{I_{x_\ell}} \leq l_{I_{x_{\ell+1}}}$. We also define the block intervals as $I_\ell^ = \sqcup_{I \in Z_\ell} I$ for any $\ell \in [t]$.*

The next lemma follows from the previous definition.

Lemma 12. *Let ψ be a nice pair, $\mathcal{J} \subseteq \mathcal{BI}(\psi)$ a set of bucket intervals, and let $\mathcal{J}^{\max} \subseteq \mathcal{J}$ be the subset of \sqsubseteq -wise maximal bucket intervals of \mathcal{J} . Let $\{Z_\ell, \ell \in [t]\}$ and $\{I_\ell^*, \ell \in [t]\}$ be the block partition and block intervals of \mathcal{J}^{\max} (which is well defined as \mathcal{J}^{\max} is proper). Then, we have $\mathcal{J} = \bigcup_{\ell \in [t]} \mathcal{P}_{\mathcal{J}}(I_\ell^*)$*

⁴Please note that our relation \prec is not transitive and thus $x_i \neq x_{i+1}$.

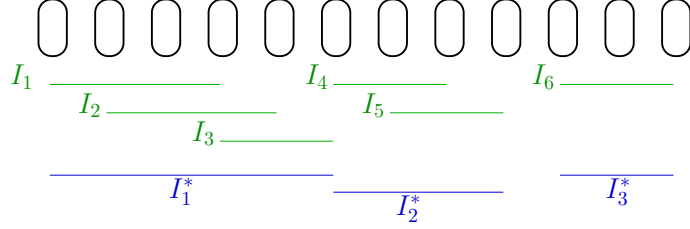


Figure 7: A block partition of $\mathcal{I}^{\max} = \{I_1, \dots, I_6\}$. We have $Z_1 = \{I_1, I_2, I_3\}$, $Z_2 = \{I_4, I_5\}$ and $Z_3 = \{I_6\}$.

Proof. Indeed, we have the following.

$$\mathcal{J} = \bigcup_{I \in \mathcal{J}^{\max}} P_{\mathcal{J}}(I) = \bigcup_{\ell \in [t]} \bigcup_{I \in Z_{\ell}} P_{\mathcal{J}}(I) = \bigcup_{\ell \in [t]} \mathcal{P}_{\mathcal{J}}(I_{\ell}^*)$$

□

The two next definitions introduce the notion of demand for a partial decomposition of a nice pair and are illustrated in Figure 8.

Definition 10. Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and I a bucket interval. We define

- $\Sigma(I) = \sum_{i \in B(I)} |S_i|$
- $m(I) = \sum_{i \in B(I)} |B_i| - |B_{i_0}|$ where $i_0 \in \operatorname{argmax}_{i \in B(I)} |B(i)|$
- $\mu(I) = \min \{\Sigma(I), m(I)\}$
- $t(I) = \begin{cases} \Sigma & \text{if } \Sigma(I) < m(I) \\ m & \text{if } \Sigma(I) > m(I) \\ = & \text{otherwise.} \end{cases}$

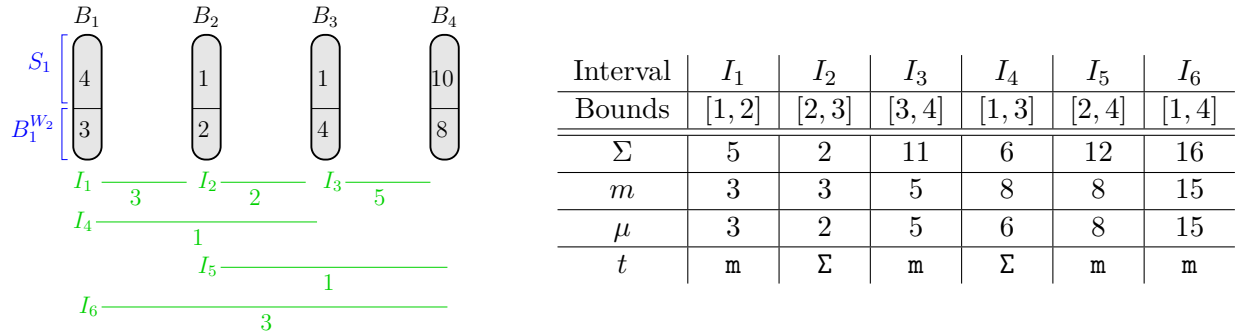


Figure 8: Example of a demand $(\mathcal{I}, \text{val})$. On the left, the buckets are depicted and the cardinalities of S_1 and $B_1^{W_2}$ are indicated. Below, stand the intervals with their value. On the right, for every interval, the values Σ , m , μ and t are computed.

To get an insight on why we used the previous definition of μ , one can check that the property announced at the begining of this section holds: let us consider a \sqsubseteq -minimal interval I (implying that buckets B_{l_I} and B_{r_I} are consecutive), and check that $\text{val}(I)$ upper bounds the size of packing \mathcal{P} where all triangles in \mathcal{P} have one vertex in B_{l_I} , one in B_{r_I} , and one in $W(I)$. On one hand, as any such triangle consumes one vertex in B_{l_I} and one in B_{r_I} , we get $|\mathcal{P}| \leq \min(|B_{l_I}|, |B_{r_I}|) = m(I)$. On the other hand, as any triangle in \mathcal{P} cannot use only vertices in W^0 , it must uses at least one vertex from $S_{l_I} \cup S_{r_I}$, implying that $|\mathcal{P}| \leq \Sigma(I)$. We discuss at the end of [Subsection 6.7](#) why taking simply $\mu(I) = \Sigma(I)$ or $\mu(I) = m(I)$ would not be sufficient to get a kernel in $\mathcal{O}(k^\delta)$ for any $\delta > 1$.

Definition 11 (Demand). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition of ψ . Let us define the following polynomial algorithm which, given $(W, B, C, B^{W_1}, B^{W_2})$, computes a set \mathcal{I} of bucket intervals of ψ , and a value $\text{val}(I)$ for any $I \in \mathcal{I}$.*

- start with $X = \emptyset$
- for any $L \in [2, |S^\psi|]$ do
 - for any bucket interval I of ψ where $|B(I)| = L$:
 - let $\text{val}(P_X(I)) = \sum_{I \in P_X(I)} \text{val}(I)$. Notice that at this stage $I \notin P_X(I)$, and in particular we have $\text{val}(P_X(I)) = 0$ if $L = 2$.
 - if $\text{val}(P_X(I)) \leq \mu(I)$ then define $\text{val}(I) = \mu(I) - \text{val}(P_X(I))$, and add I to X (and thus, in this case, $I \in P_X(I)$)
- let $\mathcal{I} = X$
- return $(\mathcal{I}, \text{val})$

We call the pair $(\mathcal{I}, \text{val})$ the demand for $(W, B, C, B^{W_1}, B^{W_2})$. We denote

$$\mathcal{I}_{>0} = \{I \in \mathcal{I} \mid \text{val}(I) > 0\}.$$

For any $X \subseteq \mathcal{I}$, we also denote $\text{val}(X) = \sum_{I \in X} \text{val}(I)$.

Observe that in the example of [Figure 8](#) we have $\mathcal{I}_{>0} = \mathcal{I} = \mathcal{BI}$, but it may be the case that $\mathcal{I}_{>0} \subsetneq \mathcal{I} \subsetneq \mathcal{BI}$. Let us now provide properties on demands.

Lemma 13. *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and $(\mathcal{I}, \text{val})$ be its demand. Then, for any $I \in \mathcal{BI}(W, B)$:*

1. $\text{val}(P_{\mathcal{I}}(I)) = \text{val}(P_{\mathcal{I}_{>0}}(I))$
2. $\text{val}(P_{\mathcal{I}}(I)) \geq \mu(I)$
3. $I \in \mathcal{I} \Leftrightarrow \text{val}(P_{\mathcal{I}}(I)) = \mu(I) \Leftrightarrow \text{val}(P_{\mathcal{I}}(I) \setminus \{I\}) \leq \mu(I)$
4. $I \in \mathcal{I}_{>0} \Leftrightarrow \text{val}(P_{\mathcal{I}}(I) \setminus \{I\}) < \mu(I)$
5. if $t(I) = \Sigma$, then for any $i_0 \in \text{argmax}_{i \in B(I)} |B(i)|$, we have $|S_{i_0}| < \sum_{i \in B(I) \setminus \{i_0\}} |B_i^{W_2}|$
6. if there exists $i_0 \in \text{argmax}_{i \in B(I)} |B(i)|$ such that $|S_{i_0}| < \sum_{i \in B(I) \setminus \{i_0\}} |B_i^{W_2}|$, then we have $t(I) = \Sigma$

7. if $t(I) = \mathfrak{m}$, then for any $i_0 \in \operatorname{argmax}_{i \in B(I)} |B(i)|$, we have $|S_{i_0}| > \sum_{i \in B(I) \setminus \{i_0\}} |B_i^{W_2}|$

Before proving Lemma 13, notice that for every interval I in Figure 8, computing its demand leads to $I \in \mathcal{I}$ and $\operatorname{val}(P_{\mathcal{I}}(I)) = \mu(I)$, illustrating Property 3 above.

Proof. Let $I \in \mathcal{BI}(W, B)$. In what follows, we consider the iteration where the algorithm considers I , and let X denote the variable of the algorithm at the line where it computes $\operatorname{val}(P_X(I))$. Observe first that, no matter whether the algorithm decides to add I in X or not, we have $P_X(I) = P_{\mathcal{I}}(I) \setminus \{I\}$. In particular, we have the following equivalences :

- $I \notin \mathcal{I} \Leftrightarrow P_{\mathcal{I}}(I) = P_X(I) \Leftrightarrow \operatorname{val}(P_X(I)) > \mu(I)$.
- $I \in \mathcal{I} \Leftrightarrow P_{\mathcal{I}}(I) = P_X(I) \cup \{I\} \Leftrightarrow \operatorname{val}(P_X(I)) \leq \mu(I)$.

Now, we can prove Properties 1 to 7.

Property 1 is immediate.

Property 2 holds in the case where $I \notin \mathcal{I}$. In case $I \in \mathcal{I}$, we have $\operatorname{val}(P_{\mathcal{I}}(I)) = \operatorname{val}(P_X(I)) + \operatorname{val}(I) = \mu(I)$ and the property holds.

The first equivalence of Property 3 follows from the proof of Property 2 above. As $\operatorname{val}(P_X(I)) = \operatorname{val}(P_{\mathcal{I}}(I) \setminus \{I\})$, we get the second part of Property 3 from the seminal observation.

Property 4 can be rewritten as $I \in \mathcal{I}_{>0} \Leftrightarrow \operatorname{val}(P_X(I)) < \mu(I)$. Since $I \in \mathcal{I} \Leftrightarrow \operatorname{val}(I) = \mu(I) - P_X(I)$, we get $\operatorname{val}(I) > 0 \Leftrightarrow \operatorname{val}(P_X(I)) < \mu(I)$, implying Property 4.

For Properties 5 and 6, let $i_0 \in \operatorname{argmax}_{i \in B(I)} |B(i)|$.

$$\begin{aligned} t(I) = \Sigma &\Leftrightarrow \sum_{i \in B(I)} |S_i| < \sum_{i \in B(I) \setminus \{i_0\}} |B_i| \\ &\Leftrightarrow \sum_{i \in B(I)} |S_i| < \sum_{i \in B(I) \setminus \{i_0\}} (|S_i| + |B_i^{W_2}|) \\ &\Leftrightarrow |S_{i_0}| < \sum_{i \in B(I) \setminus \{i_0\}} |B_i^{W_2}| \end{aligned}$$

Property 7 is obtained by reversing the above inequalities. □

6.5 The small total demand property

The objective of the section is only to prove Lemma 16. To that end, we prove that the “intersection property” of Lemma 14 implies the “union property” of Lemma 15, which finally implies Lemma 16.

Lemma 14 (The intersection property). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and $(\mathcal{I}, \operatorname{val})$ be its demand. Let I_1, I_2 in $\mathcal{I}_{>0}$ such that $I_1 \prec I_2$. Then, $t(I_1 \cap I_2) = \Sigma$.*

An example of the intersection property can be seen in Figure 8 where $t(I_4 \cap I_5) = t(I_2) = \Sigma$.

Proof. We refer the reader to Figure 9 for an illustration of the notations used in this proof.

Let $I_3 = I_1 \cap I_2$, and observe that $I_3 = [l_{I_2}, r_{I_1}]$. Let $i_0 \in \operatorname{argmax}_{i \in B(I_3)} |B_i|$. Let $I_a = (l_{I_2}, i_0)$ and $I_b = (i_0, r_{I_1})$. Observe that I_a or I_b may not be a bucket interval (when $i_0 = l_{I_2}$ or $i_0 = r_{I_1}$), but at least one of them is a bucket interval. Let $X = \{I_a, I_b\} \cap \mathcal{BI}$.

We now show that, for any $I \in X$, $t(I) = \Sigma$. Without loss of generality, assume $I_a \in X$, which implies $l_{I_2} < i_0$. Suppose, towards a contradiction, that $t(I_a) \in \{=, \mathfrak{m}\}$. Observe that $i_0 \in$

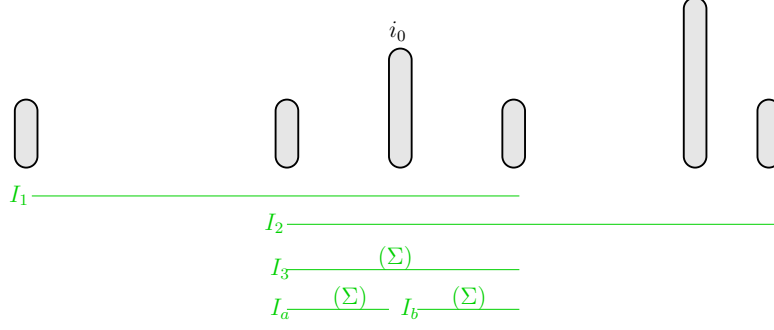


Figure 9: Notations for the intersection property proof.

$\operatorname{argmax}_{i \in B(I_a)} |B_i|$, implying $\mu(I_a) = \sum_{i \in B(I_a)} |B_i| - |B_{i_0}|$. Let $I_{a'} = (i_0, r_{I_2})$. Observe that $I_{a'}$ is a bucket interval as $i_0 \leq r_{I_1} < r_{I_2}$ (as we have $I_1 \prec I_2$). Note that $P_{\mathcal{I}}(I_a) \cap P_{\mathcal{I}}(I_{a'}) = \emptyset$, $P_{\mathcal{I}}(I_a) \subseteq P_{\mathcal{I}}(I_2) \setminus \{I_2\}$, and $P_{\mathcal{I}}(I_{a'}) \subseteq P_{\mathcal{I}}(I_2) \setminus \{I_2\}$. By combining these observations, we obtain

$$\begin{aligned} \operatorname{val}(P_{\mathcal{I}}(I_2) \setminus \{I_2\}) &\geq \operatorname{val}(P_{\mathcal{I}}(I_a)) + \operatorname{val}(P_{\mathcal{I}}(I_{a'})) \\ &\geq \mu(I_a) + \mu(I_{a'}) \text{ by Lemma 13 Property 2} \end{aligned}$$

Let us now prove that $\mu(I_a) + \mu(I_{a'}) \geq \mu(I_2)$. This will imply $\operatorname{val}(P_{\mathcal{I}}(I_2) \setminus \{I_2\}) \geq \mu(I_2)$, which contradicts Lemma 13 Property 4 as $\operatorname{val}(I_2) > 0$.

We now distinguish two cases according to $t(I_{a'})$.

Case 1: $t(I_{a'}) \in \{\Sigma, =\}$.

In this case, $\mu(I_{a'}) = \sum_{i \in B(I_{a'})} |S_i|$ and we have the following.

$$\begin{aligned} \mu(I_a) + \mu(I_{a'}) &= \sum_{i \in B(I_a)} |B_i| - |B_{i_0}| + \sum_{i \in B(I_{a'})} |S_i| \\ &= \sum_{i \in B(I_a) \setminus \{i_0\}} |B_i| + \sum_{i \in B(I_{a'})} |S_i| \\ &\geq \sum_{i \in B(I_a) \setminus \{i_0\}} |S_i| + \sum_{i \in B(I_{a'})} |S_i| \\ &\geq \Sigma(I_2) \\ &\geq \mu(I_2) \end{aligned}$$

Case 2: $t(I_{a'}) = m$.

In this case, $\mu(I_{a'}) = \sum_{i \in B(I_{a'})} |B_i| - |B_{i_1}|$, where $i_1 \in \operatorname{argmax}_{i \in B(I_{a'})} |B_i|$. Notice that $|B_{i_1}| \geq |B_{i_0}|$. Let $i_2 \in \operatorname{argmax}_{i \in B(I_2)} |B_i|$. Observe that $|B_{i_2}| = |B_{i_1}|$, and that in the two cases ($i_1 = i_0$ or $i_1 \neq i_0$) we have $\mu(I_a) + \mu(I_{a'}) = \sum_{i \in B(I_2)} |B_i| - |B_{i_1}| = m(I_2) \geq \mu(I_2)$. This concludes the proof of our claim, and we now assume that for any $I \in X$, $t(I) = \Sigma$.

Let us now prove that $t(I_3) = \Sigma$. If $i_0 = l_{I_2}$ or $i_0 = r_{I_1}$, then $X = \{I_3\}$, and we are done. Suppose now $l_{I_2} < i_0 < r_{I_1}$. Observe that $i_0 \in \operatorname{argmax}_{i \in B(I_3)} |B_i|$ implies $i_0 \in \operatorname{argmax}_{i \in B(I_a)} |B_i|$. As $t(I_a) = \Sigma$ and as $i_0 \in \operatorname{argmax}_{i \in B(I_a)} |B_i|$, Lemma 13 Property 5 implies $|S_{i_0}| < \sum_{i \in B(I_a) \setminus \{i_0\}} |B_i^{W_2}| \leq \sum_{i \in B(I_3) \setminus \{i_0\}} |B_i^{W_2}|$. By Lemma 13 Property 6, this implies $t(I_3) = \Sigma$. \square

Lemma 15 (The union property). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and $(\mathcal{I}, \text{val})$ be its demand. Let $\mathcal{Z} \subseteq \mathcal{I}_{>0}$, where $\mathcal{Z} = \{I_1, \dots, I_x\}$ for some $x \geq 1$, and where $I_i \prec I_{i+1}$ for any $i = 1, \dots, x-1$. Then, we have $\sqcup_{I \in \mathcal{Z}} I \in \mathcal{I}$.*

Proof. We refer the reader to Figure 10 for notations.

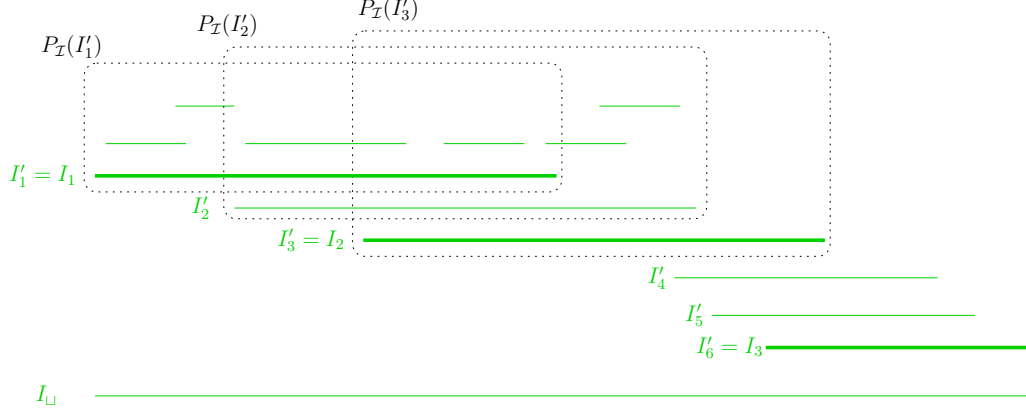


Figure 10: Interval of \mathcal{Z} are represented in bold.

Let $I_\sqcup = \sqcup_{I \in \mathcal{Z}} I$. Let also

$$\mathcal{Z}^{\max} = \{I \in \mathcal{I}_{>0} \mid I \subseteq I_\sqcup, I \neq I_\sqcup \text{ and } I \text{ is } \sqsubseteq\text{-wise maximal}\}.$$

Let $\mathcal{Z}^{\max} = \{I'_1, \dots, I'_y\}$, where intervals are ordered according to their left endpoint. If $x = 1$, the lemma follows immediately, and thus we may assume $x \geq 2$ which implies $y \geq 2$ as well.

Let us prove that we even have $I'_i \prec I'_{i+1}$, for any i . Let $i \in [y-1]$ and let j_0 be the minimum j such that $r_{I'_i} < r_{I'_j}$. Such a j_0 exists, as $i \leq y-1$, and thus $r_{I'_i} < r_{I_\sqcup}$. Moreover, $j_0 > 1$ as otherwise we would have $I'_i \sqsubset I_{j_0}$, a contradiction to the fact that I'_i is \sqsubseteq -wise maximal. Notice first that $I_{j_0} \not\subseteq I'_\ell$, for any $\ell \leq i$. As $I_{j_0-1} \prec I_{j_0}$, we get $l_{I_{j_0}} < r_{I_{j_0-1}}$ and, moreover, by the definition of j_0 , we get $r_{I_{j_0-1}} \leq r_{I'_i}$, implying $l_{I_{j_0}} < r_{I'_i}$. Thus, if, towards a contradiction, we do not have $I'_i \prec I'_{i+1}$ then we had $r_{I'_i} \leq l_{I'_{i+1}}$ and we would have $l_{I_{j_0}} < l_{I'_{i+1}}$, implying that $I_{j_0} \not\subseteq I'_{i+1}$ for any $\ell \geq i+1$. Thus, there would be no $I \in \mathcal{Z}^{\max}$ such that $I_{j_0} \sqsubseteq I$, which is a contradiction as $I_{j_0} \neq I_\sqcup$. This concludes the proof that $I'_i \prec I'_{i+1}$, for any i .

According to Lemma 13, $I_\sqcup \in \mathcal{I} \Leftrightarrow \text{val}(P_{\mathcal{I}}(I_\sqcup) \setminus \{I_\sqcup\}) \leq \mu(I_\sqcup)$, and thus our objective is to prove that $\text{val}(P_{\mathcal{I}}(I_\sqcup) \setminus \{I_\sqcup\}) \leq \mu(I_\sqcup)$.

By definition of \mathcal{Z}^{\max} , we get $P_{\mathcal{I}}(I_\sqcup) \setminus \{I_\sqcup\} = \bigcup_{\ell \in [y]} P_{\mathcal{I}}(I'_\ell)$ (see Figure 10). Thus, $\text{val}(P_{\mathcal{I}}(I_\sqcup) \setminus \{I_\sqcup\}) = \text{val}(\bigcup_{\ell \in [y]} P_{\mathcal{I}}(I'_\ell))$. Let us now prove that $\text{val}(\bigcup_{\ell \in [y]} P_{\mathcal{I}}(I'_\ell)) = s$, where

$$s = \sum_{\ell \in [y]} \text{val}(P_{\mathcal{I}}(I'_\ell)) - \sum_{\ell \in [y-1]} \text{val}(P_{\mathcal{I}}(I'_\ell) \cap P_{\mathcal{I}}(I'_{\ell+1})).$$

Let $I \in \bigcup_{\ell \in [y]} P_{\mathcal{I}}(I'_\ell)$ and let us prove that $\text{val}(I)$ appears exactly one time in s . Let ℓ_1 (resp. ℓ_2) the minimum (resp. maximum) value such that $I \in P_{\mathcal{I}}(I'_\ell)$. Observe that $\text{val}(I)$ appears $\ell_2 - \ell_1 + 1$ times in $\sum_{\ell \in [y]} \text{val}(P_{\mathcal{I}}(I'_\ell))$, and $\ell_2 - \ell_1$ times in $\sum_{\ell \in [y-1]} \text{val}(P_{\mathcal{I}}(I'_\ell) \cap P_{\mathcal{I}}(I'_{\ell+1}))$, and thus exactly 1 time in s . Moreover, as for any $\ell \in [y-1]$, $P_{\mathcal{I}}(I'_\ell) \cap P_{\mathcal{I}}(I'_{\ell+1}) = P_{\mathcal{I}}(I'_\ell \cap I'_{\ell+1})$, we can even rewrite

$$s = \sum_{\ell \in [y]} \text{val}(P_{\mathcal{I}}(I'_\ell)) - \sum_{\ell \in [y-1]} \text{val}(P_{\mathcal{I}}(I'_\ell \cap I'_{\ell+1})).$$

It now remains to prove that $s \leq \mu(I_\sqcup)$.

$$\begin{aligned}
s &= \sum_{\ell \in [y]} \text{val}(P_{\mathcal{I}}(I'_\ell)) - \sum_{\ell \in [y-1]} \text{val}(P_{\mathcal{I}}(I'_\ell \cap I'_{\ell+1})) \\
&= \sum_{\ell \in [y]} \mu(I'_\ell) - \sum_{\ell \in [y-1]} \text{val}(P_{\mathcal{I}}(I'_\ell \cap I'_{\ell+1})), \text{ by Lemma 13 Property 3 as } I'_\ell \in \mathcal{I} \\
&\leq \sum_{\ell \in [y]} \mu(I'_\ell) - \sum_{\ell \in [y-1]} \mu(I'_\ell \cap I'_{\ell+1}), \text{ by Lemma 13 Property 2} \\
&\leq \sum_{\ell \in [y]} \mu(I'_\ell) - \sum_{\ell \in [y-1]} \sum_{i \in B(I'_\ell \cap I'_{\ell+1})} |S_i|, \text{ by Lemma 14 and as we have } I'_\ell \in \mathcal{I}_{>0} \text{ for all } \ell \in [y]
\end{aligned}$$

Let us now distinguish two cases according to $t(I_\sqcup)$.

Case 1: $t(I_\sqcup) \in \{\Sigma, =\}$.

In this case, $\mu(I_\sqcup) = \sum_{i \in B(I_\sqcup)} |S_i|$, and we have the following.

$$\begin{aligned}
s &\leq \sum_{\ell \in [y]} \sum_{i \in B(I'_\ell)} |S_i| - \sum_{\ell \in [y-1]} \sum_{i \in B(I'_\ell \cap I'_{\ell+1})} |S_i|, \text{ by definition of } \mu \\
&= \sum_{i \in B(\bigcup_{\ell \in [y]} I'_\ell)} |S_i|, \text{ as for any } i, \text{ the term } |S_i| \text{ appears exactly once} \\
&= \Sigma(I_\sqcup) \\
&\leq \mu(I_\sqcup) \text{ as we are in case 1}
\end{aligned}$$

Case 2: $t(I_\sqcup) = \mathfrak{m}$.

Let $i_0 \in \text{argmax}_{i \in B(I_\sqcup)} |B_i|$. In this case, $\mu(I_\sqcup) = \sum_{i \in B(I_\sqcup)} |B_i| - |B_{i_0}|$, and according to Lemma 13 Property 7, we have $|S_{i_0}| > \sum_{i \in B(I_\sqcup) \setminus \{i_0\}} |B_i^{W_2}|$.

Let us prove the following property: Let $B^* = (B(I'_1) \setminus \bigcup_{2 \leq \ell \leq y} B(I'_\ell)) \cup (B(I'_y) \setminus \bigcup_{1 \leq \ell \leq y-1} B(I'_\ell))$, then $i_0 \in B^*$. This property means that i_0 is in one of the two ‘‘extreme’’ sides of the union. For example, in Figure 8, where $I'_1 = I_4, I'_2 = I_5$, and $I_\sqcup = I_6$, we have that $t(I_\sqcup) = \mathfrak{m}$, $i_0 = 4$, and $i_0 \in B(I'_2) \setminus B(I'_1)$.

Observe that, as $I'_\ell \prec I'_{\ell+1}$ for any ℓ , it holds that, for all $i \in B(I_\sqcup) \setminus B^*$, there exists ℓ such that $i \in B(I'_\ell) \cap B(I'_{\ell+1})$. Assume, towards a contradiction, that $i_0 \notin B^*$, implying that there exists ℓ_0 such that $i_0 \in B(I'_{\ell_0}) \cap B(I'_{\ell_0+1})$. Therefore $i_0 \in B(I'_{\cap})$, where $I'_{\cap} = I'_{\ell_0} \cap I'_{\ell_0+1}$. By Lemma 14, as I'_{ℓ_0} and I'_{ℓ_0+1} are in $\mathcal{I}_{>0}$ and $I'_{\ell_0} \prec I'_{\ell_0+1}$, we get $t(I'_{\cap}) = \Sigma$. By Lemma 13 Property 5, and as $i_0 \in \text{argmax}_{i \in B(I'_{\cap})} |B_i|$, we get $|S_{i_0}| < \sum_{i \in B(I'_{\cap}) \setminus \{i_0\}} |B_i^{W_2}| < \sum_{i \in B(I_\sqcup) \setminus \{i_0\}} |B_i^{W_2}|$, a contradiction.

Hence the property holds and, without loss of generality, we may assume $i_0 \in B(I'_1) \setminus \bigcup_{2 \leq \ell \leq y} B(I'_\ell)$. Note that the case where $i_0 \in B(I'_y) \setminus \bigcup_{1 \leq \ell \leq y-1} B(I'_\ell)$ is symmetric.

Observe that $i_0 \in \text{argmax}_{i \in B(I'_1)} |B(i)|$ which implies $m(I'_1) = \sum_{i \in B(I'_1) \setminus \{i_0\}} |B_i|$.

As $i_0 \in B(I'_1) \setminus \bigcup_{2 \leq \ell \leq y} B(I'_\ell)$, we have $i_0 \in [l(I'_1), l(I'_2)[$, and thus

$$m(I'_1) = \sum_{i \in [l(I'_1), l(I'_2)[\setminus \{i_0\}} |B_i| + \sum_{i \in [l(I'_2), r(I'_1)]} (|S_i| + |B_i^{W_2}|).$$

We conclude that

$$\begin{aligned}
s &\leq \sum_{\ell \in [y]} \mu(I'_\ell) - \sum_{\ell \in [y-1]} \sum_{i \in B(I'_\ell \cap I'_{\ell+1})} |S_i| \\
&\leq m(I'_1) + \sum_{2 \leq \ell \leq y} \Sigma(I'_\ell) - \sum_{\ell \in [y-1]} \sum_{i \in B(I'_\ell \cap I'_{\ell+1})} |S_i| \\
&= \sum_{i \in [l(I'_1), l(I'_2)] \setminus \{i_0\}} |B_i| + \sum_{i \in [l(I'_2), r(I'_1)]} (|S_i| + |B_i^{W_2}|) + \\
&\quad \sum_{2 \leq \ell \leq y} \sum_{i \in B(I'_\ell)} |S_i| - \sum_{\ell \in [y-1]} \sum_{i \in B(I'_\ell \cap I'_{\ell+1})} |S_i| \\
&= \sum_{i \in [l(I'_1), l(I'_2)] \setminus \{i_0\}} |B_i| + \sum_{i \in [l(I'_2), r(I'_1)]} |B_i^{W_2}| + \sum_{1 \leq \ell \leq y} \sum_{i \in B(I'_\ell)} |S_i| - \sum_{\ell \in [y-1]} \sum_{i \in B(I'_\ell \cap I'_{\ell+1})} |S_i| \\
&\quad \text{by defining } I_1^* = [l(I'_2), r(I'_1)] \text{ and } I_\ell^* = I'_\ell, \text{ for any } 2 \leq \ell \leq y \\
&= \sum_{i \in [l(I'_1), l(I'_2)] \setminus \{i_0\}} |B_i| + \sum_{i \in [l(I'_2), r(I'_1)]} |B_i^{W_2}| + \sum_{i \in B(\bigcup_{\ell \in [y]} I_\ell^*)} |S_i| \\
&\quad \text{as, for any } i, \text{ each term } |S_i| \text{ appears exactly once} \\
&= \sum_{i \in B(I'_1) \setminus \{i_0\}} |B_i| + \sum_{i \in B(I_\sqcup) \setminus B(I'_1)} |S_i| \\
&\leq \sum_{i \in B(I'_1) \setminus \{i_0\}} |B_i| + \sum_{i \in B(I_\sqcup) \setminus B(I'_1)} |B_i| \\
&= m(I_\sqcup) \\
&\leq \mu(I_\sqcup), \text{ as we are in Case 2.}
\end{aligned}$$

□

The following lemma will be used in both cases of our kernelization algorithm (rainbow matching or small vertex cover) in order to prove that what is added in the kernel is small.

Lemma 16 (The small total demand property). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and $(\mathcal{I}, \text{val})$ be its demand. Let $\mathcal{J} \subseteq \mathcal{I}_{>0}$, and let $\mathcal{J}^{\text{max}} \subseteq \mathcal{J}$ be the subset of \sqsubseteq -wise maximal bucket intervals of \mathcal{J} . Let $\{Z_\ell, \ell \in [t]\}$ and $\{I_\ell^*, \ell \in [t]\}$ be the block partition and block intervals of \mathcal{J}^{max} . Then, we have $\text{val}(\mathcal{J}) \leq \sum_{\ell \in [t]} \mu(I_\ell^*)$.*

Proof. According to Lemma 12 we have, $\mathcal{J} = \bigcup_{\ell \in [t]} \mathcal{P}_{\mathcal{J}}(I_\ell^*)$ and thus, $\text{val}(\mathcal{J}) \leq \sum_{\ell \in [t]} \text{val}(\mathcal{P}_{\mathcal{J}}(I_\ell^*))$. Let $\ell \in [t]$. As $Z_\ell = \{I_{x_{\ell-1}+1}, \dots, I_{x_\ell}\}$, where $I_{x_{\ell-1}+i} \prec I_{x_{\ell-1}+i+1}$ for any i , and all $I \in Z_\ell$ satisfy $\text{val}(I) > 0$, Lemma 15 implies that $I_\ell^* \in \mathcal{I}$. Thus, Lemma 13 Property 3 implies that $\text{val}(\mathcal{P}_{\mathcal{J}}(I_\ell^*)) \leq \mu(I_\ell^*)$, implying in turn that $\text{val}(\mathcal{J}) \leq \sum_{\ell \in [t]} \mu(I_\ell^*)$. □

6.6 Auxiliary graph and bucket allocation

In this section we first define an auxiliary graph based on a demand. Then, we prove in Lemma 17 that if a rainbow matching gives us in particular a matching for all the colored loops of this auxiliary graph, which correspond to what we call here a *bucket allocation*, then we can repack any triangle-packing contained in $W \cup B$ into this bucket allocation.

We refer the reader to Figure 11 for an illustration of the situation.

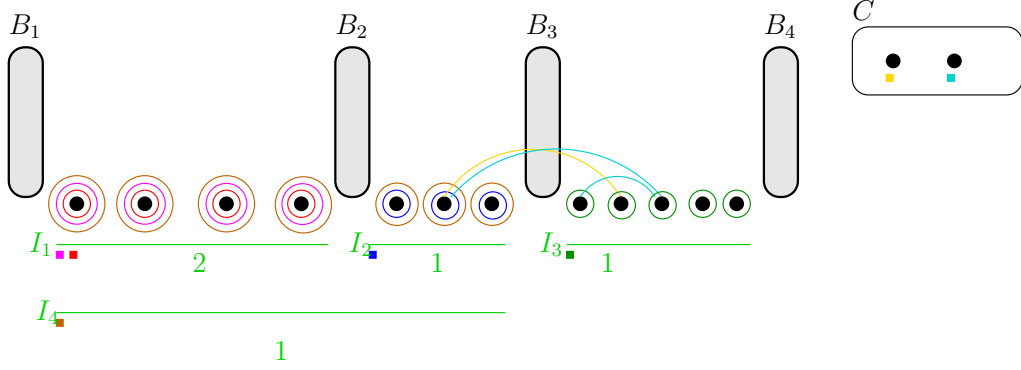


Figure 11: Example of an auxiliary graph. Squares indicate the colors associated to each interval or element of C . For example, D_{I_1} is composed of colors pink and red. Notice that there is a rainbow matching.

Definition 12 (Auxiliary graph). Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and $(\mathcal{I}, \text{val})$ be its demand. Let also $p = |C| + \sum_{I \in \mathcal{I}_{>0}} \text{val}(I)$. We next define the p -edge-colored multigraph $(G, \chi)(W, B, C, B^{W_1}, B^{W_2})$ where the vertex set of G is W and the edges of G , as well as their colors, are defined as follows. We start with $E_1 = E_2 = \emptyset$. Then, for any $c \in C$, for any $\{u, w\} \subseteq W$ such that $\{u, w, c\}$ is a triangle, we add the edge $e = \{u, w\}$ to E_1 and we set $\chi(e) = c$. Moreover, for any $I \in \mathcal{I}_{>0}$, we define a set of new colors D_I where $|D_I| = \text{val}(I)$. Then, for any $u \in D_I$ and $v \in W(I)$, we add to E_2 edge $e = \{v\}$ and we set $\chi(e) = u$. Finally, we denote $\bigcup_{I \in \mathcal{I}_{>0}} D_I$ by D .

Notice that if the partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ is clean, then for any vertex $c \in C$, there exists an edge with color c in E_1 . Notice also that the value of p is a polynomial function of the size $|V(T)|$ of the tournament. In fact, $p = \mathcal{O}(|V(T)|^3)$.

Definition 13 (Bucket allocation). Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and $(\mathcal{I}, \text{val})$ be its demand. A bucket allocation for $(W, B, C, B^{W_1}, B^{W_2})$ is a set $\mathcal{Q} = \{Q_I, I \in \mathcal{I}_{>0}\}$ such that

- for any $I \in \mathcal{I}_{>0}$, $Q_I \subseteq W(I)$
- for any $I \in \mathcal{I}_{>0}$, $|Q_I| = \text{val}(I)$
- for any I_1 and I_2 in $\mathcal{I}_{>0}$, $Q_{I_1} \cap Q_{I_2} = \emptyset$

We also denote $V(\mathcal{Q}) = \bigcup_{I \in \mathcal{I}_{>0}} Q_I$.

The next lemma shows that a bucket allocation allows to repack any packing triangle inside $T[W \cup B]$.

Lemma 17 (Safeness of a bucket allocation for packing). Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and let \mathcal{Q} be a bucket allocation for $(W, B, C, B^{W_1}, B^{W_2})$. Let also \mathcal{P} be a triangle-packing with $V(\mathcal{P}) \subseteq W \cup B$. Then, there exists a triangle-packing \mathcal{P}' such that

- $V(\mathcal{P}') \subseteq V(\mathcal{Q}) \cup B$ and
- $|\mathcal{P}'| = |\mathcal{P}|$

Proof. According to [Observation 1](#), we can partition $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$ such that, for any $\Delta \in \mathcal{P}_1$, $\Delta \subseteq B$ and, moreover, for any $\Delta \in \mathcal{P}_2$, there exists $I \in \mathcal{BI}(W, B)$ such that $|\Delta \cap B_{l_I}| = |\Delta \cap B_{r_I}| = 1$ and $\Delta \cap W \subseteq W_I$. Let $A(\mathcal{P}_2) = \{\Delta \cap B, \Delta \in \mathcal{P}_2\}$ be the set of backward arcs used by triangles in \mathcal{P}_2 .

Let us now define an auxiliary graph H that will help us to describe how to repack \mathcal{P} . The vertex set of H is B and for any bucket interval $I = [l_I, r_I]$ of (W, B) , we define $E_I = \{\{u, v\} \mid u \in B_{l_I} \setminus B_{l_I}^{W_2} \text{ or } u \in B_{r_I} \setminus B_{r_I}^{W_2}\}$, and we set $E(H) = \bigcup_{I \in \mathcal{BI}(W, B)} E_I$. Informally, for any $i \neq i'$, $H[B_i \cup B_{i'}]$ is the complete bipartite graph $K_{|B_i|, |B_{i'}|}$ where we remove all edges between $B_i^{W_2}$ and $B_{i'}^{W_2}$. For any $e \in E(H)$, we denote $I(e)$ the unique bucket interval I such that $e \in E_I$.

Observe that $A(\mathcal{P}_2) \subseteq E(H)$ as for any $\{u, v\} \in A(\mathcal{P}_2)$, where $\{u, v\} = \Delta \cap B$, there exists $I \in \mathcal{BI}(W, B)$ such that $u \in B_{l(I)}$, $v \in B_{r(I)}$. Moreover, we cannot have $u \in B_{l(I)}^{W_2}$ and $v \in B_{r(I)}^{W_2}$, as this would imply $\Delta \subseteq W_0$. Finally, as \mathcal{P}_2 is a packing, $A(\mathcal{P}_2)$ is even a matching in H .

Property Π_1 . Let us prove the following property Π_1 :

Property Π_1 : For any $I \in \mathcal{BI}(W, B)$, $\text{MaxM}(H(I)) \leq \mu(I)$, where MaxM denotes the size of a maximum matching, and $H(I) = H[\bigcup_{i \in B(I)} B_i]$.

Proof of Π_1 : Firstly, as any edge in $H(I)$ uses at least one vertex from a set S_i for $i \in B(I)$, we deduce that $\text{MaxM}(H(I)) \leq \Sigma(I)$. Let $i_0 = \text{argmax}_{i \in B(I)} |B_i|$. Moreover, as for any graph G' and any independent set $X' \subseteq V(G')$ we have $\text{MaxM}(G') \leq |V(G')| - |X'|$, and as B_{i_0} is an independent set in H , we get $\text{MaxM}(H(I)) \leq m(I)$. This implies that $\text{MaxM}(H(I)) \leq \mu(I)$.

Property Π_2 . We proceed by proving property Π_2 which allows us to associate, in a well defined way, a vertex in $V(\mathcal{Q})$ to any arc in $A(\mathcal{P}_2)$. An example of this property can be found in [Figure 12](#).

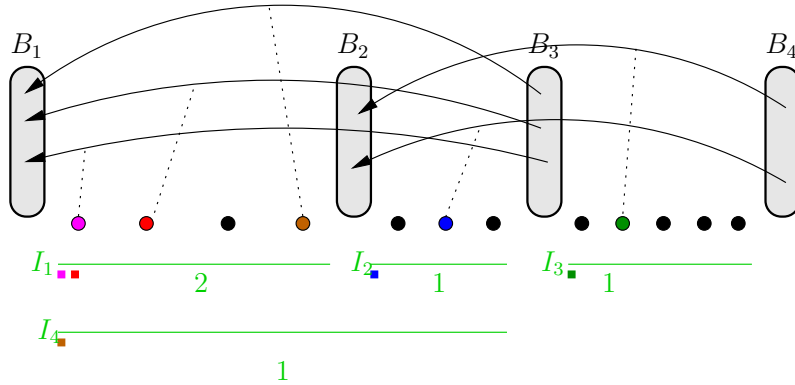


Figure 12: Example of property Π_2 . The bucket allocation is represented by the non-black vertices. Function f , which associates injectively to each backward arc of the initial packing a vertex of the bucket allocation, is depicted using dashed lines.

Property Π_2 : For any matching M in H , there is a function f from M to $V(\mathcal{Q})$ such that f is injective and, for any $e \in M$, $f(e) \in W(I(e)) \cap V(\mathcal{Q})$.

Proof of Π_2 : Let U be a bipartite graph, where $V(U) = (M, \mathcal{Q})$, and for any $e \in M$, $N(e) = W(I(e)) \cap V(\mathcal{Q})$. What remains is the proof that there is a perfect matching f (which associates

to each $e \in M$ a vertex $f(e) \in N(e)$ in U which saturates M . According to Hall's Theorem, it is sufficient to prove that, for any $M' \subseteq M$, $|N(M')| \geq |M'|$.

Let $M' \subseteq M$. Observe that $\bigcup_{e \in M'} [l_{I(e)}, r_{I(e)}]$ is a union of $t \geq 1$ disjoint intervals denoted $\{I'_1, \dots, I'_t\}$, and that M' can be partitioned into $\bigcup_{i \in [t]} M'_i$, such that for any $i \in [t]$, $\bigcup_{e \in M'_i} [l_{I(e)}, r_{I(e)}] = I'_i$.

As an example, consider a matching $M' = \{e_1, e_2, e_3, e_4, e_5\}$ where $I(e_1) = [1, 10], I(e_2) = [3, 4], I(e_3) = [10, 11], I(e_4) = [20, 22], I(e_5) = [21, 23]$, $\bigcup_{e \in M'} [l_{I(e)}, r_{I(e)}] = \{[1, 11], [20, 23]\}$ and $M'_1 = \{e_1, e_2, e_3\}$.

Let $i \in [t]$. Observe that, for any $v \in W(I'_i)$, there exists $e \in M'_i$ such that $v \in W(I(e))$. Thus, for any vertex $v \in \bigcup_{I' \in P_{\mathcal{I}_{>0}}(I'_i)} Q_{I'}$, as v belongs to $W(I'_i)$, and as there exists $e \in M'_i$ such that $v \in W(I(e))$, we get $v \in W(I(e)) \cap Q = N(e)$. This implies $\bigcup_{I' \in P_{\mathcal{I}_{>0}}(I'_i)} Q_{I'} \subseteq N(M'_i)$. As Q is a bucket allocation for $(W, B, C, B^{W_1}, B^{W_2})$, we know that the $Q_{I'}$ are disjoint and $|Q_{I'}| = \text{val}(I')$, implying that

$$\left| \bigcup_{I' \in P_{\mathcal{I}_{>0}}(I'_i)} Q_{I'} \right| = \sum_{I' \in P_{\mathcal{I}_{>0}}(I'_i)} |Q_{I'}| = \sum_{I' \in P_{\mathcal{I}_{>0}}(I'_i)} \text{val}(I') = \text{val}(P_{\mathcal{I}_{>0}}(I'_i)).$$

According to [Lemma 13](#), $\text{val}(P_{\mathcal{I}_{>0}}(I'_i)) \geq \mu(I'_i)$. According to Property Π_1 , $\mu(I'_i) \geq \text{MaxM}(H(I'_i))$. This implies that $|N(M'_i)| \geq \text{MaxM}(H(I'_i))$. Since M' is a matching in H , M'_i is a matching in $H(I'_i)$. This implies $\text{MaxM}(H(I'_i)) \geq |M'_i|$ and therefore $|N(M'_i)| \geq |M'_i|$. To conclude, let us now consider the partition $\{N(M'_i) \mid i \in [t]\}$ of $N(M')$. As the $N(M'_i)$ s are vertex-disjoint, we get

$$|N(M')| = \sum_{i \in [t]} |N(M'_i)| \geq \sum_{i \in [t]} \text{MaxM}(H(I'_i)).$$

As M' is a matching in $\bigcup_{i \in [t]} H(I'_i)$, and as $\sum_{i \in [t]} \text{MaxM}(H(I'_i)) \geq \text{MaxM}(\bigcup_{i \in [t]} H(I'_i))$, we conclude that $|N(M')| \geq |M'|$.

We can now conclude the proof of Lemma. As $A(\mathcal{P}_2)$ is a matching in H , by Property Π_2 , there is a function f from $A(\mathcal{P}_2)$ to $V(\mathcal{Q})$ such that f is injective and, for any $e \in A(\mathcal{P}_2)$, $f(e) \in W(I(e)) \cap V(\mathcal{Q})$. By [Observation 1](#), $e \cup f(e)$ is still a triangle. Thus, we define $\mathcal{P}'_2 = \{e \cup f(e), e \in A(\mathcal{P}_2)\}$ and $\mathcal{P}' = \mathcal{P}_1 \cup \mathcal{P}'_2$. As $V(\mathcal{P}'_2) \cap B = V(\mathcal{P}_2) \cap B$ and as f is injective, \mathcal{P}' is still a triangle-packing, and $|\mathcal{P}'| = |\mathcal{P}|$. Moreover, as $V(\mathcal{P}_1) \subseteq B$ and $V(\mathcal{P}'_2) \subseteq B \cup V(\mathcal{Q})$, we get $V(\mathcal{P}') \subseteq B \cup V(\mathcal{Q})$. \square

Finally, the last lemma of the section indicates how to build a feedback vertex set of $T[W \cup B]$ with the help of a bucket allocation. This will be useful for the kernel of FVST designed in [Section 5](#). We enounce this lemma here, as the notations and the techniques used in its proof are similar, though easier, than in the previous lemma.

Lemma 18 (Safeness of a bucket allocation for hitting). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition and let \mathcal{Q} be a bucket allocation for $(W, B, C, B^{W_1}, B^{W_2})$. Let also X be a feedback vertex set of $T[V(\mathcal{Q}) \cup B]$. Then, there exists X' a feedback vertex set of $T[W \cup B]$ with $|X'| \leq |X|$.*

Proof. We partition X into two sets $X_B = X \cap B$ and $X_{\mathcal{Q}} = X \cap V(\mathcal{Q})$. And we define H being the graph on vertex set B and with edge set $\{uv : uv \in A(T), u \in B_i \setminus X_B, v \in B_j \setminus X_B \text{ with } j < i\}$. Informally, from $T[B]$ we only keep in H the backward arcs between the B_i 's not incident with vertices of X_B . Notice that in particular, there is no edge in H between any two $B_i^{W_2}$, as there where originally part of W_0 .

Property Π'_1 . Let us prove the following property Π'_1 :

Property Π'_1 : For any $I \in \mathcal{BI}(W, B)$, $\text{MinVC}(H(I)) \leq \mu(I)$, where MinVC denotes the size of a minimum vertex cover, and $H(I) = H[\cup_{i \in B(I)} B_i]$.

Proof of Π'_1 : As every edge of H is incident with a vertex of one S_i (which is $B_i \setminus B_i^{W_2}$), the set $\cup_i S_i$ is a vertex cover of H , and then $\text{MinVC}(H(I)) \leq \Sigma(I)$. Similarly, let $i_0 = \text{argmax}_{i \in B(I)} |B_i|$, the set $(\cup_i B_i) \setminus B_{i_0}$ is clearly a vertex cover of H . Then we get $\text{MinVC}(H(I)) \leq m(I)$, and finally conclude that $\text{MinVC}(H(I)) \leq \mu(I)$.

Now, let us consider an arc uv of T such that $uv \in E(H)$. We have $u \in B_i$ and $v \in B_j$ for some $j < i$ and denote by I the bucket interval $[j, i]$. By Observation 1, uvw is a triangle for every vertex w of $W(I)$. In particular, for every $w \in W(I) \cap V(\mathcal{Q})$ we must have $w \in X_{\mathcal{Q}}$.

So, denote by \mathcal{J} the set of bucket intervals whose extremities contains the end of an edge of H (formally, $[j, i] \in \mathcal{J}$ if there exists $uv \in E(H)$ with $u \in B_i$ and $v \in B_j$). Let J^* be a block of the block partition of \mathcal{J} , meaning that J^* is a non-empty, inclusion-wise minimal bucket interval with the property that for every $J \in \mathcal{J}$, either $J \cap J^* = \emptyset$ or $J \subseteq J^*$. By the previous remark, we know that $X_{\mathcal{Q}}$ contains at least $|W(J^*) \cap V(\mathcal{Q})|$ vertices in $W(J^*)$. As \mathcal{Q} is a bucket allocation, we have in particular that $|W(J^*) \cap V(\mathcal{Q})| = |\cup \{Q_J : Q_J \in \mathcal{Q} \text{ and } J \subseteq J^*\}| = \text{val}(P_{\mathcal{I}}(J^*)) \geq \mu(J^*)$ by Lemma 13.

Finally, by Property Pi'_1 , there exists a vertex cover X'_{J^*} of $H(J^*)$ containing at most $\mu(J^*) \leq |W(J^*) \cap X_{\mathcal{Q}}|$ vertices. Considering all the block intervals J_1^*, \dots, J_p^* of \mathcal{J} , we obtain a vertex cover $X'_{\mathcal{Q}} = \cup_j X'_{J_j^*}$ of H such that $|X'_{\mathcal{Q}}| \leq |X_{\mathcal{Q}}|$. In particular, for every arc uv of T with $u \in B_i$ and $v \in B_j$ with $j < i$ we have $u \in X'_{\mathcal{Q}} \cup X_B$ or $u \in X'_{\mathcal{Q}} \cup X_B$.

We now define $X' = X'_{\mathcal{Q}} \cup X_B$. We have $|X'| \leq |X|$ and X' is a feedback vertex set of $T[W \cup B]$. Indeed, let Δ by a triangle of $T[W \cup B]$. If $V(\Delta) \subseteq B$, then $V(\Delta) \cap X_B \neq \emptyset$. Otherwise, as (W, B) is a nice pair, there exist buckets B_i and B_j with $j < i$ such that $\Delta = uvw$ with $u \in B_i$, $v \in B_j$ and $w \in W_{[j, i]}$. But in this case we must have $u \in X'_{\mathcal{Q}}$ or $v \in X'_{\mathcal{Q}}$. \square

6.7 Operations on buckets

Our kernelization algorithm has only one rule that is applied exhaustively. Each application of the this rule (except the last one that finds a rainbow matching) triggers an “add” operation, defined below, where we add new vertices to the buckets. In this section, we define the two variants of the add operation and prove that, given a partial decomposition, these operations output another partial decomposition.

Definition 14. Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition. Let $U \subseteq W$ and $X^C \subseteq C$. For any $j \in \{1, 2\}$, we define $\text{add}_j(W, B, B^{W_1}, B^{W_2}, U \cup X^C)$ as $(W', B', C', B'^{W_1}, B'^{W_2})$ where

- $W' = W \setminus U$
- $B' = B \cup U \cup X^C$
- $C' = C \setminus X^C$
- $B'^{W_j} = B^{W_j} \cup U$ and $B'^{W_{3-j}} = B^{W_{3-j}}$.

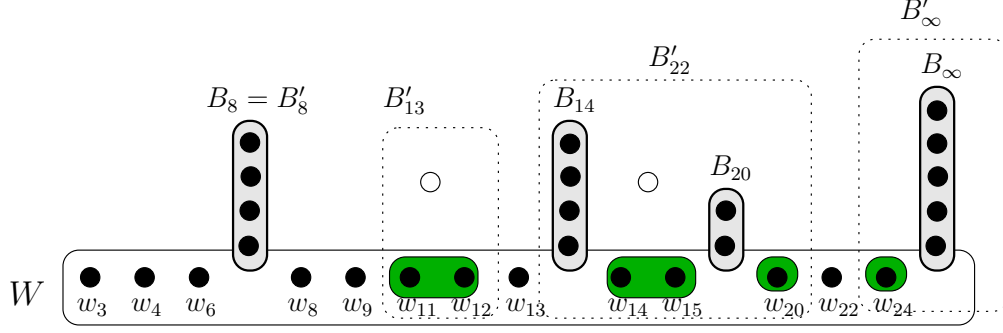


Figure 13: An application of add_1 (or add_2). Vertices of U are inside the green territories and the two vertices of X^C are the two non-filled vertices. We have $S^\psi = \{8, 14, 20, \infty\}$ and $S^{\psi'} = \{8, 13, 22, \infty\}$.

Lemma 19. *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition of local size f . Suppose $f(x) = cx^\delta$ for some constants c and δ , where $\delta > 1$. Let $U \subseteq W$ and $X^C \subseteq C$ such that $|U| \leq 10|X^C|$ and such that, for any triangle Δ of $T[W \cup X^C]$, we have that $|\Delta \cap (W \setminus U)| \leq 1$. Then, $\text{add}_1(W, B, B^{W_1}, B^{W_2}, U \cup X^C)$ is a partial decomposition of local size f .*

Proof. Let $(W', B', C', B'^{W_1}, B'^{W_2}) = \text{add}_1(W, B, B^{W_1}, B^{W_2}, U \cup X^C)$. We begin by proving that $(W', B', C', B'^{W_1}, B'^{W_2})$ is a partial decomposition. Properties 1 and 2 are clear. As, by assumption, for any triangle Δ of $T[W \cup X^C]$, we have that $|\Delta \cap (W \setminus U)| \leq 1$, Lemma 10 implies that $\psi' = (W', B')$ is a nice pair, and thus we obtain Property 3.

Let us now prove Property 4. We start by proving the following structural property of the B'_i (see Figure 13).

For any $i \in S^{\psi'}$, $\exists U' \subseteq U$, $X' \subseteq X^C$ and $S' \subseteq S^\psi$ such that $B'_i = \bigcup_{i \in S'} B_i \cup X' \cup U'$.

We point out that we could even prove a stronger property (for example that buckets in S' are consecutive), but we don't need it in the remainder of the proof.

Let $i \in \psi'$. According to Proposition 1, B'_i is the set of vertices v of B' such that all arcs between $W'_{[1, i-1]}$ and v are oriented from $W'_{[1, i-1]}$ to v and all arcs between $W'_{[i, t_0]}$ and v are oriented from v to $W'_{[i, t_0]}$. Assume that there exists $j \in \psi$ and $v_1 \in B_j$ such that $v_1 \in B'_i$. Now, observe that for any $v_2 \in B_j$ and $w \in W'$, $v_1 w \in A(T)$ iff $v_2 w \in A(T)$, and the same holds for $w v_1$ and $w v_2$. This implies that $v_2 \in B'_i$ and thus $B_j \subseteq B'_i$. As all the vertices we add to B are from $U \cup X^C$, this proves the above structural property.

Next we prove that (B'^{W_1}, B'^{W_2}) is a bucket partition of (W', B') . The fact that (B'^{W_1}, B'^{W_2}) is a partition of B'^W is clear, as $B'^W = B^W \cup U$, $B'^{W_1} = B^{W_1} \cup U$, and $B'^{W_2} = B^{W_2}$. Let $i \in S^{\psi'}$. By the previous structural property, $B'_i = \bigcup_{i \in S'} B_i \cup X' \cup U'$. If $S' \neq \emptyset$, then $|S'_i| \geq \bigcup_{i \in S'} |S_i| \geq 1$. Otherwise, as $X' \cup U' \neq \emptyset$, $U' \subseteq B'^{W_1}$ and $X' \subseteq B'^C$, we get $U' \cup X' \subseteq S'_i$, implying $|S'_i| \geq 1$. Let us prove that $|B'^{W_1}| \leq 10|B'^C|$.

$$\begin{aligned}
|B'^{W_1}| &= |B^{W_1}| + |U| \\
&\leq 10|B^C| + |U| \\
&\leq 10|B^C| + 10|X^C| \\
&= 10|B'^C|.
\end{aligned}$$

This concludes the proof that (B'^{W_1}, B'^{W_2}) is a bucket partition of (W', B') .

Let us finally prove that (B'^{W_1}, B'^{W_2}) has local size f , meaning that for any $i \in S^{\psi'}$, we have $|B'_i{}^{W_2}| \leq f(|S'_i|) = c|S'_i|^\delta$. Let $i \in S^{\psi'}$. By the previous structural property, there exists $S' \subseteq S^\psi$ such that $S'_i \supseteq \bigcup_{j \in S'} S_j$. In particular, we have $|S'_i| \geq \sum_{j \in S'} |S_j|$ as the S_j 's are disjoint. Now, noticing that $B'^{W_2} = B^{W_2}$, we have the following.

$$\begin{aligned}
|B'_i{}^{W_2}| &= \left| \bigcup_{i \in S'} B_i \cap B'^{W_2} \right| + |(X' \cup U') \cap B'^{W_2}| \\
&= \left| \bigcup_{i \in S'} B_i \cap B^{W_2} \right| + 0 \\
&\leq \sum_{i \in S'} c|S_i|^\delta \\
&\leq c \left(\sum_{i \in S'} |S_i| \right)^\delta \\
&\leq c|S'_i|^\delta.
\end{aligned}$$

□

Now, we analyze the add_2 operation.

Lemma 20. *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition of local size f . Suppose $f(x) = cx^\delta$ for some constants c and δ , where $1 < \delta \leq 2$ and $c \geq \max(\frac{20}{(2^\delta - 2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$. Let I be a bucket interval and $U = W(I)$ such that $|U| \leq 10\mu(I)$. Then, $\text{add}_2(W, B, B^{W_1}, B^{W_2}, U)$ is a partial decomposition of local size f .*

Proof. Let $(W', B', C', B'^{W_1}, B'^{W_2}) = \text{add}_2(W, B, B^{W_1}, B^{W_2}, U)$. As a first step we prove that $(W', B', C', B'^{W_1}, B'^{W_2})$ is a partial decomposition. Properties 1 and 2 are clear. By applying Lemma 10 (for $X^C = \emptyset$ here), we get that $\psi' = (W', B')$ is a nice pair, and thus we obtain Property 3.

Let us now prove Property 4. Let us first describe the bucket decomposition in ψ' . Informally, all buckets of $B(I)$ are merged, together with U , into a new one (B'_{r_I}) , and all the other buckets remain unchanged. More formally, the following three properties hold

- $S^{\psi'} = (S^\psi \setminus B(I)) \cup \{r_I\}$,
- for any $i \in S^{\psi'} \setminus \{r_I\}$, it holds that
 - $B'_i = B_i$,
 - $B'_i{}^C = B_i$, and
 - for any $j \in [2]$, $B'_i{}^{W_j} = B_i{}^{W_j}$,
- $B'_{r_I} = U \cup \bigcup_{i \in B(I)} B_i$.

The next step is to prove that (B'^{W_1}, B'^{W_2}) is a bucket partition of (W', B') . The fact that (B'^{W_1}, B'^{W_2}) is a partition of B'^W is clear as $B'^W = B^W \cup U$, $B'^{W_1} = B^{W_1}$ and $B'^{W_2} = B^{W_2} \cup U$. Let $i \in S^{\psi'}$. If $i \in S^{\psi'} \setminus \{r_I\}$, then, by the previous property, we get $S'_i = S_i$, implying $|S'_i| \geq 1$, as (B^{W_1}, B^{W_2}) is a bucket partition of (W, B) . If $i = r_I$, then $B'_{r_I} \supseteq \bigcup_{i \in B(I)} B_i$ and, as $B'^{W_1} = B^{W_1}$,

$|S'_{r_I}| \geq \sum_{i \in B(I)} |S'_i| \geq |B(I)| \geq 2$. Moreover, we have $|B'^{W_1}| = |B^{W_1}| \leq 10|B^C| = 10|B'^C|$. This concludes the proof that (B'^{W_1}, B'^{W_2}) is a bucket partition of (W', B') .

Let us finally prove that (B'^{W_1}, B'^{W_2}) has local size f , meaning that, for any $i \in S^{\psi'}$, $|B_i'^{W_2}| \leq f(|S'_i|)$. Recall that (W', B') is a nice pair, implying, because of Proposition 1, that there is a unique bucket decomposition B'_i of B' . Thus, for any $i \in S^{\psi'} \setminus \{r_I\}$, $|B_i'^{W_2}| = |B_i^{W_2}| \leq f(|S_i|) = f(|S'_i|)$.

Let $B(I) = \{i_1, \dots, i_L\}$, where $|S_{i_x}| \geq |S_{i_{x+1}}|$, for any x . Notice that $f(|S'_{r_I}|) = c|S'_{r_I}|^\delta = c(\sum_{\ell \in [1, L]} S_{i_\ell})^\delta$, and that $m(I) = \sum_{i \in B(I)} |B_i| - |B_{i_*}|$ where $i_* = \operatorname{argmax}_{i \in B(I)} |B(i)|$, implying that $m(I) \leq \sum_{\ell \in [2, L]} |B_{i_\ell}| \leq \sum_{\ell \in [2, L]} (|B_{i_\ell}^{W_2}| + |S_{i_\ell}|) \leq \sum_{\ell \in [2, L]} (c|S_{i_\ell}|^\delta + |S_{i_\ell}|)$. It remains to prove that $|B_{r_I}'^{W_2}| - f(|S'_{r_I}|) \leq 0$.

Observe that

$$\begin{aligned} |B_{r_I}'^{W_2}| - f(|S'_{r_I}|) &= \sum_{i \in B(I)} |B_i^{W_2}| + |U| - c \left(\sum_{\ell \in [1, L]} |S_{i_\ell}| \right)^\delta \\ &\leq \sum_{i \in B(I)} |B_i^{W_2}| + 10 \min \{ \Sigma(I), m(I) \} - c \left(\sum_{\ell \in [1, L]} |S_{i_\ell}| \right)^\delta \\ &\leq c|S_{i_1}|^\delta + c \sum_{\ell \in [2, L]} |S_{i_\ell}|^\delta + \\ &\quad 10 \min \left\{ \sum_{i \in B(I)} |S_i|, \sum_{\ell \in [2, L]} (c|S_{i_\ell}|^\delta + |S_{i_\ell}|) \right\} - c \left(\sum_{\ell \in [1, L]} |S_{i_\ell}| \right)^\delta \\ &= \min \{ f_1(S_{i_1}), f_2(S_{i_1}) \}, \end{aligned}$$

where

- $f_1(x) = cx^\delta + c \sum_{\ell \in [2, L]} |S_{i_\ell}|^\delta + 10(x + \sum_{\ell \in [2, L]} |S_{i_\ell}|) - c(x + \sum_{\ell \in [2, L]} |S_{i_\ell}|)^\delta$
- $f_2(x) = cx^\delta + c \sum_{\ell \in [2, L]} |S_{i_\ell}|^\delta + 10 \sum_{\ell \in [2, L]} (c|S_{i_\ell}|^\delta + |S_{i_\ell}|) - c(x + \sum_{\ell \in [2, L]} |S_{i_\ell}|)^\delta$

Let $g : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ be such that $g(x) = (x + \sum_{\ell \in [2, L]} |S_{i_\ell}|)^\delta - x^\delta$ and observe that it is a non-decreasing function, as $\delta > 1$. Therefore

$$f_2(x) = c \sum_{\ell \in [2, L]} |S_{i_\ell}|^\delta + 10 \sum_{\ell \in [2, L]} (c|S_{i_\ell}|^\delta + |S_{i_\ell}|) - cg(x),$$

and thus f_2 a non-increasing function. Let $x_0 = \sum_{\ell \in [2, L]} |S_{i_\ell}|^\delta$. Notice that according to the first property of a bucket partition, for any $i \in S^\psi$, we have $|S_i| \geq 1$, implying that $x_0 \geq L - 1 \geq 1$.

To complete the proof we distinguish two cases according to the value of $|S_{i_1}|$. However, before we proceed with these two cases, we prove a helpful claim.

Claim 1: Let $g_2(x) = c^\delta x^\delta + 21x - (cx + x^{\frac{1}{\delta}})^\delta$. For $x \geq 0$, $\delta > 1$ and $c \geq (\frac{21}{\delta})^{\frac{1}{\delta-1}}$, we have $g_2(x) \leq 0$.

Proof of Claim 1: We have

$$\begin{aligned} (cx + x^{\frac{1}{\delta}})^\delta &= c^\delta x^\delta \left(1 + \frac{x^{\frac{1}{\delta}-1}}{c} \right)^\delta \\ &\geq c^\delta x^\delta \left(1 + \delta \frac{x^{\frac{1}{\delta}-1}}{c} \right) \text{ as } (1+u)^\alpha \geq 1 + \alpha u \text{ for any } \alpha > 1 \text{ and } u > 0 \\ &= c^\delta x^\delta + \delta c^{\delta-1} x^{\frac{1}{\delta}-1+\delta} \end{aligned}$$

Thus,

$$\begin{aligned}
g_2(x) &\leq 21x - \delta c^{\delta-1} x^{\frac{1}{\delta}-1+\delta} \\
&\leq 21x - \delta c^{\delta-1} x \text{ as } \frac{1}{\delta} - 1 + \delta = \frac{\delta^2 - 2\delta + 1}{\delta} + 1 = \frac{(\delta-1)^2}{\delta} + 1 \geq 1 \\
&\leq 0 \quad \text{as } c \geq \left(\frac{21}{\delta}\right)^{\frac{1}{\delta-1}}
\end{aligned}$$

This implies that $g_2(x) \leq 0$ for any $x \geq 0$. This completes the proof of Claim 1.

Now that Claim 1 is proved, let us come back to our two cases.

Case 1 : if $|S_{i_1}| \geq cx_0$. In this case, we will prove that $f_2(|S_{i_1}|) \leq 0$. As f_2 is non-increasing, we get $f_2(|S_{i_1}|) \leq f_2(cx_0)$ therefore, it remains to prove that $f_2(cx_0) \leq 0$. Observe that as $\delta \geq 1$, $(\sum_{\ell \in [2, L]} |S_{i_\ell}|)^\delta \geq \sum_{\ell \in [2, L]} |S_{i_\ell}|^\delta$, implying that $(\sum_{\ell \in [2, L]} |S_{i_\ell}|) \geq x_0^{\frac{1}{\delta}}$. Observe that

$$\begin{aligned}
f_2(cx_0) &\leq cc^\delta x_0^\delta + cx_0 + 20cx_0 - c(cx_0 + x_0^{\frac{1}{\delta}})^\delta \iff \\
\frac{f_2(cx_0)}{c} &\leq c^\delta x_0^\delta + 21x_0 - (cx_0 + x_0^{\frac{1}{\delta}})^\delta = g_2(x_0)
\end{aligned}$$

The assertion in this case now follows from Claim 1.

Case 2 : if $|S_{i_1}| < cx_0$. In this case, we will prove that $f_1(|S_{i_1}|) \geq 0$. Let us first prove that f_1 is convex.

$$\begin{aligned}
\frac{d^2}{dx^2} f_1(x) &= c(\delta)(\delta-1)(x^{\delta-2} - (x + \sum_{\ell \in [2, L]} S_{i_\ell})^{\delta-2}) \\
&\geq 0, \text{ as } \delta \leq 2
\end{aligned}$$

Thus, as $|S_{i_2}| \leq |S_{i_1}| < cx_0$, by the definition of the S_{i_j} , and f_1 is convex, $f_1(|S_{i_2}|) \leq \max(f_1(|S_{i_1}|), f_1(cx_0))$. Let us prove that both these values are lower or equal to zero. Observe that $\frac{f_1(cx_0)}{c} \leq g_2(x_0) \leq 0$ by Claim 1, and thus it remains to prove that $f_1(|S_{i_2}|) \leq 0$.

For any $x_i \geq 0$, let $f(x_2, \dots, x_L) = 2c^\delta x_2^\delta + c \sum_{\ell \in [3, L]} x_\ell^\delta + 10(2x_2 + \sum_{\ell \in [3, L]} x_\ell) - c(2x_2 + \sum_{\ell \in [3, L]} x_\ell)^\delta$. Observe that $f_1(|S_{i_2}|) = f(|S_{i_2}|, |S_{i_3}|, \dots, |S_{i_L}|)$.

Claim 2: For any tuple of integers (x_2, \dots, x_L) where $x_\ell \geq 1$ for any $\ell \in [2, L]$, $f(x_2, \dots, x_L) \leq f(x_2, 0, \dots, 0)$.

Proof of Claim 2: Let (x_2, \dots, x_L) be a tuple of integers where $x_\ell \geq 0$. Let $i_0 \in [3, L]$ such that $x_{i_0} \geq 1$, and $x'_{i_0} = x_{i_0} - 1$. Let $D = f(x_2, \dots, x_{i_0}, \dots, x_L) - f(x_2, \dots, x'_{i_0}, \dots, x_L)$. Our goal is to prove that $D \leq 0$. Repeating the decrementation, this will imply Claim 2.

$$\begin{aligned}
D &= cx_{i_0}^\delta - c(x_{i_0} - 1)^\delta + 10 - c(\alpha^\delta - (\alpha - 1)^\delta) \quad (\text{where } \alpha = 2x_2 + \sum_{\ell \in [3, L]} x_\ell) \\
&\leq c(x_{i_0}^\delta - (x_{i_0} - 1)^\delta - (3x_{i_0})^\delta + (3x_{i_0} - 1)^\delta) + 10 \quad (\text{as } D \text{ is non increasing in } \alpha, \text{ and } \alpha \geq 3x_{i_0})
\end{aligned}$$

For any $y \geq 1$, let $u(y) = y^\delta - (y-1)^\delta - (3y)^\delta + (3y-1)^\delta$. Let us first lower bound two different parts of u . We have

$$\begin{aligned} (y-1)^\delta &= y^\delta \left(1 - \frac{1}{y}\right)^\delta \\ &\geq y^\delta \left(1 - \frac{\delta}{y}\right) \quad (\text{as } (1-u)^\alpha \geq 1 - \alpha u \text{ for any } \alpha > 1 \text{ and } 0 \leq u \leq 1) \\ &= y^\delta - \delta y^{\delta-1} \end{aligned}$$

Moreover, denoting $z = 3y - 1$, we have

$$\begin{aligned} (3y)^\delta &= (z+1)^\delta \\ &= z^\delta \left(1 + \frac{1}{z}\right)^\delta \\ &\geq z^\delta \left(1 + \frac{\delta}{z}\right) \quad (\text{as } (1+u)^\alpha \geq 1 + \alpha u \text{ for any } \alpha > 1 \text{ and } u > 0) \\ &= z^\delta + \delta z^{\delta-1} \\ &= (3y-1)^\delta + \delta(3y-1)^{\delta-1} \end{aligned}$$

Using these two lower bounds, we get

$$\begin{aligned} u(y) &\leq \delta(y^{\delta-1} - (3y-1)^{\delta-1}) \\ &\leq \delta(1 - 2^{\delta-1}) \quad (\text{as } y^{\delta-1} - (3y-1)^{\delta-1} \text{ is decreasing in } y \text{ and } y \geq 1) \end{aligned}$$

This implies

$$\begin{aligned} D &\leq cu(x_{i_0}) + 10 \\ &\leq c\delta(1 - 2^{\delta-1}) + 10 \\ &\leq 0 \quad (\text{as } c \geq \frac{20}{(2^\delta-2)} \geq \frac{10}{\delta(2^{\delta-1}-1)} \text{ for } \delta \geq 1) \end{aligned}$$

Concluding the proof of Claim 2.

Let us now come back to the end of proof of Case 2. We had:

$$\begin{aligned} f_1(|S_{i_2}|) &= f(|S_{i_2}|, |S_{i_3}|, \dots, |S_{i_L}|) \\ &\leq f(|S_{i_2}|, 0, \dots, 0) \text{ by Claim 2} \\ &= 2c^\delta |S_{i_2}|^\delta + 20|S_{i_2}| - c(2|S_{i_2}|)^\delta \\ &= c|S_{i_2}|^\delta (2 - 2^\delta) + 20|S_{i_2}| \quad (\text{which is decreasing in } |S_{i_2}| \text{ as } c \geq \frac{20}{(2^\delta-2)} \text{ and } \delta \geq 1) \\ &\leq c(2 - 2^\delta) + 20 \quad (\text{as } |S_{i_2}| \geq 1) \\ &\leq 0 \quad (\text{as } c \geq \frac{20}{(2^\delta-2)} \text{ and } \delta \geq 1) \end{aligned}$$

This concludes the proof of Case 2. □

Discussion on why a simpler definition of μ is not sufficient. We explain here why taking simply $\mu(I) = \Sigma(I)$ or $\mu(I) = m(I)$ (instead of $\mu(I) = \min(\Sigma(I), m(I))$) would not be sufficient to get a kernel in $\mathcal{O}(k^\delta)$ for any $\delta > 1$. Let us assume that the current partial decomposition is

$(W, B, C, B^{W_1}, B^{W_2})$ where in particular $S^\psi = \{i_1, \dots, i_{k'}\}$ for $k' = \frac{k}{2}$, $B^{W_j} = \emptyset$ for $j \in \{1, 2\}$, implying that buckets only contain vertices of C^0 , and $|B_i| = 1$ for any $i \in S^\psi$. Suppose that $k = 2^x$ for some x .

Suppose first that we define $\mu(I) = \Sigma(I)$. Notice that particular $\text{val}([i_1, i_2]) = \mu([i_1, i_2]) = 2$, and thus that in the auxiliary graph we will have in particular two colors d_1, d_2 such that all vertices of $W([i_1, i_2])$ have one loop of color d_1 and one loop of color d_2 . The kernel now applies its unique reduction rule (see Definition 15). Suppose that we do not find a rainbow matching, and that the set of color we find using Corollary 1 is $X = \{d_1, d_2\}$, and thus that the small associated vertex cover is the set $U = W([i_1, i_2])$ (U must cover all these loops), where $|U| = 5|X| = 10$. In this case, we fall into Case 2, and the add_2 operation will merge B_{i_1}, B_{i_2} and $W([i_1, i_2])$ into a new bucket B'_{i_2} . If we now repeat the same scenario between B'_{i_2} and B_{i_3} , we will now have $\mu(I) = 3$ (where $I = [i_2, i_3]$), and thus $|U| \leq 15$. Thus, if we repeat again until all buckets are merged into a single one, the total number of vertices of W added to buckets during these add_2 operations will be $5(\sum_{i=2}^{k'} i) = \mathcal{O}(k^2)$, and thus we could not obtain a kernel in $\mathcal{O}(k^\delta)$ vertices.

Suppose now that we define $\mu(I) = m(I)$. Notice that particular $\text{val}([i_\ell, i_{\ell+1}]) = \mu([i_\ell, i_{\ell+1}]) = 1$, and thus that for any ℓ in the auxiliary graph we will have in particular one color d_ℓ such that all vertices of $W([i_\ell, i_{\ell+1}])$ have one loop of color d_ℓ . The kernel now applies its unique reduction rule (see Definition 15). Suppose that we do not find a rainbow matching, and that the set of color we find using Corollary 1 is $X = \{d_1\}$, and thus that the small associated vertex cover is the set $U = W([i_1, i_2])$, where $|U| = 5|X| = 5$. In this case, we fall into Case 2, and the add_2 operation will merge B_{i_1}, B_{i_2} and $W([i_1, i_2])$ into a new bucket B'_{i_2} where $|B'_{i_2}| = 7$. If we now repeat the same scenario for $\ell = 3, 5, \dots$ between B_{i_ℓ} and $B_{i_{\ell+1}}$, the add_2 operation will merge $B_{i_\ell}, B_{i_{\ell+1}}$ and $W([i_\ell, i_{\ell+1}])$ into a new bucket $B'_{i_{\ell+1}}$ where $|B'_{i_{\ell+1}}| = 7$. We now have 2^{x-2} buckets $B'_{i_2}, B'_{i_4}, \dots, B'_{i_{k'}}$, each of size 7. The total number of vertices added to B in this first part of the scenario is thus $n_1 = 2^{(x-2)}5$. If we again repeat the same scenario for $\ell = 2, 6, \dots$, the add_2 operation will merge $B'_{i_\ell}, B'_{i_{\ell+2}}$ and we will have $|U| = 5|X| \geq 5 \times 7 \geq 5^2$. Thus, the total number of vertices added to B in this second part of the scenario is $n_2 = 2^{(x-3)}5^2$. If we continue until there is only one bucket, the total number of vertices added in the last part of the scenario will be $n_{x-1} = 5^{(x-1)} = 2^{(x-1)\log(5)} = k^{\log(5)}$, and thus we cannot hope for a kernel in $\mathcal{O}(k^\delta)$ vertices.

6.8 Analysis the two cases: rainbow matching or small vertex cover

Our kernelization algorithm takes as input a partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ of size f where $f(x) = cx^\delta$ for some constants c and δ . Before going to the formal description of the algorithm, let us sketch how it works, and why we need the following lemmas. At each round, the kernelization algorithm first derives from $(W, B, C, B^{W_1}, B^{W_2})$ a clean partial decomposition with same size f using Lemma 11, then builds the auxiliary edge-colored graph $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$ and tries to find a rainbow matching using Corollary 1. If it finds such a matching, then the algorithm stops and concludes using Lemma 21. Otherwise, the algorithm finds a small vertex cover and performs an add operation in order to obtain another tuple $(W', B', C', B'^{W_1}, B'^{W_2})$. Lemma 22 will allow us to ensure that $(W', B', C', B'^{W_1}, B'^{W_2})$ is still a partial decomposition of local size f .

Lemma 21 (Case of rainbow matching). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a clean partial decomposition of local size f , where $f(x) = cx^\delta$ for some positive constants c and δ . Suppose also that the colored multigraph $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$ admits a rainbow matching M . Let $A = V(M) \cup B \cup C$. Then, for any integer k , (T, k) is a yes-instance of TPT iff $(T[A], k)$ is a yes-instance of TPT. Moreover,*

we have $|A| \leq 3(c+1)(33)^\delta(k)^\delta$.

Proof. Let $T' = T[A]$. As T' is an induced tournament of T , direction \Leftarrow is clearly true. Let us now suppose that (T, k) is a yes-instance and prove that (T', k) is a yes-instance. Let \mathcal{P} be a triangle-packing of T of size k and let us prove that there is a triangle-packing \mathcal{P}' of size k in T' . Let us partition $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$, where $\mathcal{P}_1 = \{\Delta \in \mathcal{P} \mid \Delta \subseteq W \cup B\}$, and $\mathcal{P}_2 = \{\Delta \in \mathcal{P} \mid \Delta \cap C \neq \emptyset\}$. For any $\Delta \in \mathcal{P}_2$, let $c(\Delta)$ be a vertex in $\Delta \cap C$.

For any $c \in C$, let $f(c)$ be the edge of color c in M . Notice that $f(c)$ is well defined as $(W, B, C, B^{W_1}, B^{W_2})$ is clean. Moreover, for any $I \in \mathcal{I}_{>0}$ (where $(\mathcal{I}, \text{val})$ is the demand of $(G, \chi)(W, B, C, B^{W_1}, B^{W_2})$) and color $d \in D_I$, let $f(I, d)$ be the edge of color d in M . Let $f(I) = \{f(I, d), d \in D_I\}$. Observe that, for any $c \in C$, by the definition of edges of colors c , $\{c\} \cup f(c)$ is a triangle in T . Let $\mathcal{Q}_2 = \{f(c), c \in C\}$ and $\mathcal{Q}_1 = \{f(I), I \in \mathcal{I}_{>0}\}$. Observe that $M = \mathcal{Q}_1 \cup \mathcal{Q}_2$. As M is a matching, for any $I \in \mathcal{I}_{>0}$, $|f(I)| = |D_I| = \text{val}(I)$; moreover notice that the $f(I)$ are disjoint. As, for any $d \in D_I$, the only vertices in $(G, \chi)(W, B, C, B^{W_1}, B^{W_2})$ that have color d are $W(I)$, we get $f(I) \subseteq W(I)$. This implies that \mathcal{Q}_1 is a bucket allocation for $(W, B, C, B^{W_1}, B^{W_2})$. Thus, according to [Lemma 17](#), there exists a packing \mathcal{P}'_1 such that $V(\mathcal{P}'_1) \subseteq V(\mathcal{Q}_1) \cup B$ and $|\mathcal{P}'_1| = |\mathcal{P}_1|$.

Let us now restructure triangles in \mathcal{P}_2 . For any $\Delta \in \mathcal{P}_2$, we define $g(\Delta) = \{c(\Delta)\} \cup f(c(\Delta))$. Let $\mathcal{P}'_2 = \{g(\Delta), \Delta \in \mathcal{P}_2\}$. Finally, we define $\mathcal{P}' = \mathcal{P}'_1 \cup \mathcal{P}'_2$. Let us prove that \mathcal{P}' is a triangle-packing. For any Δ'_1, Δ'_2 in \mathcal{P}'_1 , $\Delta'_1 \cap \Delta'_2 = \emptyset$, as \mathcal{P}'_1 is a packing. Let us now consider Δ'_1, Δ'_2 in \mathcal{P}'_2 , where $\Delta'_i = g(\Delta_i)$. First, as $\Delta_1 \cap \Delta_2 = \emptyset$, we obtain that $c(\Delta_1) \neq c(\Delta_2)$. Moreover, as M is a matching, $f(c(\Delta_1)) \cap f(c(\Delta_2)) = \emptyset$. As $f(c(\Delta_i)) \subseteq V(\mathcal{Q}_i)$ and $c(\Delta_i) \in C$, we also have $f(c(\Delta_i)) \cap \{c(\Delta_{3-i})\} = \emptyset$, for $i \in [2]$, implying $\Delta'_1 \cap \Delta'_2 = \emptyset$. Consider now the last case where $\Delta'_i \in \mathcal{P}'_i$ for $i \in [2]$. As $\Delta'_1 \subseteq V(\mathcal{Q}_1) \cup B$ and $\Delta'_2 \subseteq V(\mathcal{Q}_2) \cup C$, and as $V(\mathcal{Q}_1), V(\mathcal{Q}_2), B$ and C are pairwise disjoint sets, we get the desired result. Finally, $|\mathcal{P}'| = |\mathcal{P}'_1| + |\mathcal{P}'_2| = |\mathcal{P}_1| + |\mathcal{P}_2| = |\mathcal{P}| \geq k$.

The next step is to prove an upper bound on $|V(T')|$. We start by bounding $|V(M)|$. Recall that $V(M) = V(\mathcal{Q}_1) \cup V(\mathcal{Q}_2)$, implying that $|V(M)| = |\mathcal{Q}_1| + 2 \cdot |C|$. Let us now bound $|\mathcal{Q}_1|$. Let $\mathcal{I}_{>0}^{\max}$ be the set of \sqsubseteq -wise maximal intervals of $\mathcal{I}_{>0}$. Let also $\{Z_1, \dots, Z_t\}$ and $\{I_1^*, \dots, I_t^*\}$ be the block partition and the block intervals of $\mathcal{I}_{>0}^{\max}$. Observe that $r_{I_\ell^*} \leq l_{I_{\ell+1}^*}$, implying that any bucket index $i \in S^\psi$ belongs to at most two $B(I_\ell^*)$'s. Moreover, the only case where i belongs to two $B(I_\ell^*)$'s is when $i = r_{I_\ell^*}$ and $r_{I_\ell^*} = l_{I_{\ell+1}^*}$.

We are now ready to bound $|\mathcal{Q}_1|$.

$$\begin{aligned}
|\mathcal{Q}_1| &= \sum_{I \in \mathcal{I}_{>0}} |f(I)| \\
&= \text{val}(\mathcal{I}_{>0}) \\
&\leq \sum_{\ell \in [t]} \mu(I_\ell^*) \text{ by [Lemma 16](#)} \\
&\leq \sum_{\ell \in [t]} \sum_{i \in B(I_\ell^*)} |B_i| \\
&\leq 2 \sum_{i \in S^\psi} |B_i|, \text{ as any } i \in S^\psi \text{ belongs to at most two } B(I_\ell^*)
\end{aligned}$$

This implies $|V(M)| \leq 2|B| + 2|C|$. We are now ready to bound the size of the kernel output:

$$\begin{aligned}
|V(T')| &= |V(M)| + |B| + |C| \\
&\leq 3|B| + 3|C| \\
&= 3(|B^C| + |B^{W_1}| + |B^{W_2}|) + 3|C| \\
&\leq 3(|B^C| + |B^{W_1}| + c \sum_{i \in S^\psi} (|B_i^C| + |B_i^{W_1}|)^\delta) + 3|C| \\
&\leq 3(|B^C| + |B^{W_1}| + c(\sum_{i \in S^\psi} (|B_i^C| + |B_i^{W_1}|))^\delta) + 3|C| \\
&\leq 3(|B^C| + |B^{W_1}|) + c(|B^C| + |B^{W_1}|)^\delta + 3|C| \\
&\leq 3(c+1)(|B^C| + |B^{W_1}|)^\delta + 3|C| \\
&\leq 3(c+1)(11|B^C|)^\delta + 3|C|, \text{ by 7} \\
&\leq 3(c+1)11^\delta(|B^C| + |C|)^\delta \\
&= 3(c+1)11^\delta(|C^0|)^\delta \\
&\leq 3(c+1)11^\delta(3k)^\delta
\end{aligned}$$

□

The next lemma deals with the other case of the reduction rule, where the auxiliary edge-colored graph contains a small vertex cover for a set of colors. Notice that in this case, we do not necessarily need that the considered partial decomposition is clean.

Lemma 22 (Case of a small vertex cover). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a partial decomposition of local size f , where $f(x) = cx^\delta$ for some constants $c > 0$ and δ with $1 < \delta \leq 2$. Suppose that there exists a non-empty subset of colors X such that $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle[X]$ admits a vertex cover U with $|U| \leq 5|X|$. Recall that colors of $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$ are partitioned into $C \cup D$. Let $X^C = X \cap C$ and $X^D = X \cap D$.*

- (Case 1:) *If $|X^D| \leq |X^C|$, then $(W', B', C', B'^{W_1}, B'^{W_2}) = \text{add}_1(W, B, C, B^{W_1}, B^{W_2}, U \cup X^C)$ is still a partial decomposition of local size f where $|W'| + |C'| < |W| + |C|$.*
- (Case 2:) *If $|X^D| > |X^C|$, then we can find, in polynomial time, a non-empty $U' \subseteq U$ such that $(W', B', C', B'^{W_1}, B'^{W_2}) = \text{add}_2(W, B, C, B^{W_1}, B^{W_2}, U')$ is a partial decomposition of local size f , where $|W'| + |C'| < |W| + |C|$.*

Proof. Proof of Case 1. Observe first that $|U| \leq 5(|X^D| + |X^C|) \leq 10|X^C|$, as we are in Case 1. Let us now prove the two conditions required by Lemma 19. As U is a vertex cover of $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle[X]$, for any color $c \in X$ (and thus in X^C) and edge e of color c , $U \cap V(e) \neq \emptyset$. Suppose, towards a contradiction, that there exists a triangle $\Delta \in T[W \cup X^C]$ such that $|\Delta \cap (W \setminus U)| \geq 2$. Since we cannot have $\Delta \subseteq W$, this implies that $\Delta = \{w_1, w_2, c\}$ where $w_i \in W \setminus U$ and $c \in X^C$. It follows from Δ being a triangle that $e = \{w_1, w_2\}$ is an edge of $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$ of color c , which is not intersected by U , a contradiction. Thus, we can now apply Lemma 19 and obtain that $\text{add}_1(W, B, C, B^{W_1}, B^{W_2}, U \cup X^C)$ is a partial decomposition of local size f . Moreover, as $|X^D| \leq |X^C|$ and $X \neq \emptyset$, we have $X^C \neq \emptyset$, implying $|C'| < |C|$ and thus $|W'| + |C'| < |W| + |C|$, as required.

Proof of *Case 2*. Observe first that $|U| \leq 5(|X^D| + |X^C|) \leq 10|X^D|$, as we are in *Case 2*. Let us now find an interval I' such that $|U'| \leq 10\mu(I')$, where $U' = W(I')$, in order to apply [Lemma 20](#). Remember that there is a partition $\{D_I \mid I \in \mathcal{I}_{>0}\}$ of D . For any u in X^D , let $I(u) \in \mathcal{I}_{>0}$ be the unique bucket interval such that $u \in D_I$. Let $\mathcal{I}(X^D) = \{I(u) \mid u \in X^D\}$ and $\mathcal{I}^{\max}(X^D)$ be the set of \sqsubseteq -wise maximal intervals of $\mathcal{I}(X^D)$. Let $\{Z_1, \dots, Z_t\}$ and $\{I_1^*, \dots, I_t^*\}$ be the block partition and block intervals of $\mathcal{I}^{\max}(X^D)$. Moreover, as $U' \neq \emptyset$, we get $|W'| < |W|$, implying $|W'| + |C'| < |W| + |C|$.

For any $\ell \in [t]$, let $U_\ell = W(I_\ell^*)$. Remember that in $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$, for each color $u \in D$, we add all $v \in W(I(u))$ as edges of color u . Thus, as U is a vertex cover of $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle[X]$, for any $u \in X^D$, U must contain $W(I(u))$. Hence, we get

$$\begin{aligned} U &= \bigcup_{u \in X^D} W(I(u)) \\ &= \bigcup_{I \in \mathcal{I}(X^D)} W(I) \\ &= \bigcup_{I \in \mathcal{I}^{\max}(X^D)} W(I) \\ &= \bigcup_{\ell \in [t]} W(I_\ell^*) \\ &= \bigcup_{\ell \in [t]} U_\ell. \end{aligned}$$

Finally, observe that the U_ℓ are disjoint.

Let us now upper bound $|X^D|$.

$$\begin{aligned} |X^D| &= \sum_{I \in \mathcal{I}(X^D)} |X^D \cap D_I| \\ &\leq \sum_{I \in \mathcal{I}(X^D)} |D_I| \\ &= \bigcup_{I \in \mathcal{I}(X^D)} \text{val}(I), \text{ by definition of } (G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle \\ &= \text{val}(\mathcal{I}(X^D)) \\ &\leq \sum_{\ell \in [t]} \mu(I_\ell^*), \text{ by } \a href="#">\text{Lemma 16}. \end{aligned}$$

Hence $|U| \leq 10|X^D|$ implies that $\sum_{\ell \in [t]} |U_\ell| \leq 10 \sum_{\ell \in [t]} \mu(I_\ell^*)$. This, in turn, implies that there exists $\ell \in [t]$ such that $|U_\ell| \leq 10\mu(I_\ell^*)$ and we can find this ℓ in polynomial time by enumerating all values in $[t]$. Thus, we set $U' = U_\ell$ and, as $U_\ell = W(I_\ell^*)$ and $|U_\ell| \leq 10\mu(I_\ell^*)$, it follows from [Lemma 20](#) that $\text{add}_2(W, B, C, B^{W_1}, B^{W_2}, U')$ is a partial decomposition of local size f , as required. \square

6.9 Analysis of the overall kernel

We are now ready to define the unique rule, except from the cleaning phases, for our kernelization algorithm. In the following, we assume that δ with $1 < \delta \leq 2$ is fixed.

Definition 15 (Reduction Rule for TPT). *Given a clean partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ of local size f , where $f(x) = c(\delta)x^\delta$ with $c(\delta) = \max(\frac{20}{(2^\delta-2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$, let us define the output $R(W, B, C, B^{W_1}, B^{W_2})$ of the rule R as follows:*

- *Decide, using [Corollary 1](#) (applied for $\varepsilon = 1$), whether there exists a rainbow matching M in the p -edge-colored multigraph $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$ (where $p = |C| + \sum_{i \in \mathcal{I}_{>0}} \text{val}(I)$ – recall that $p = \mathcal{O}(|V(T)|^3)$).*

- *If the algorithm finds a rainbow matching M , then return $V(M) \cup B \cup C$.*
- *Otherwise, let X be the non-empty set of colors such that $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle[X]$ admits a vertex cover U such that $|U| \leq (4 + \varepsilon)|X|$. Let $X^C = X \cap C$ and $X^D = X \cap D$.*
 - * *(Case 1) If $|X^D| \leq |X^C|$, let*

$$(W', B', C', B'^{W_1}, B'^{W_2}) = \text{add}_1(W, B, C, B^{W_1}, B^{W_2}, U \cup C).$$

Return $(W', B', C', B'^{W_1}, B'^{W_2})$.

- * *(Case 2) Otherwise we have $|X^D| > |X^C|$. In this case compute, in polynomial time, according to [Lemma 22](#), a non-empty $U' \subseteq U$ such that*

$$(W', B', C', B'^{W_1}, B'^{W_2}) = \text{add}_2(W, B, C, B^{W_1}, B^{W_2}, U')$$

is a partial decomposition of local size f . Return $(W', B', C', B'^{W_1}, B'^{W_2})$.

Then, we obtain the following.

Lemma 23. *Given a clean partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ of local size f , where $f(x) = c(\delta)x^\delta$ with $c(\delta) = \max(\frac{20}{(2^\delta-2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$, $R(W, B, C, B^{W_1}, B^{W_2})$ either returns:*

- *a set $A \subseteq V(T)$ such that, if $T' = T[A]$, then (T, k) and (T', k) are equivalent instances of TPT and $|V(T')| \leq c_0 \cdot c(\delta) \cdot k^\delta$, with $c_0 = 6534$.*
- *or a partial decomposition $(W', B', C', B'^{W_1}, B'^{W_2})$ of local size f , where $|W'| + |C'| < |W| + |C|$.*

Proof. If R finds a rainbow matching M in $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2} \rangle$, then [Lemma 21](#) immediately implies that the set $A = V(M) \cup B \cup C$ verifies the claimed properties, as in particular $|V(T')| \leq 3(c(\delta) + 1)33^\delta k^\delta \leq 6 \cdot c(\delta) \cdot 33^2 k^\delta = 6534 \cdot c(\delta) \cdot k^\delta$. Let us now consider that R does not find a rainbow matching.

If R falls into Case 1 or Case 2, then by [Lemma 22](#) we know, as we set $\varepsilon = 1$, that $(W', B', C', B'^{W_1}, B'^{W_2})$ is still a partial decomposition of local size f with $|W'| + |C'| < |W| + |C|$. \square

Finally, we can prove the kernelization algorithm for TPT stated in [Theorem 4](#).

Proof of [Theorem 4](#). Let δ with $1 < \delta \leq 2$, $c(\delta) = \max(\frac{20}{(2^\delta-2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$ and notice that $c(\delta) \geq 1$. Given an input (T, k) , the kernelization algorithm **A** starts by the greedy localization phase⁵. Assume that it does not find a packing of size k , therefore it computes a greedy localized pair

⁵The greedy localization is a greedy packing of triangles, see [Subsection 6.2](#).

(C^0, W^0) of T . Observe that $(W^0, \emptyset, C^*, \emptyset, \emptyset)$ is a partial decomposition of local size f , where $f(x) = c(\delta)x^\delta$. Using Lemma 11, we then obtain the first clean partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ of the process, still with local size f . Now, a step of algorithm A consists in applying the Reduction Rule R for TPT followed by a cleaning phase. Algorithm A exhaustively performs steps, obtaining a clean partial decomposition with size f at the end of each step and stopping only when it falls into the matching case.

Let us prove, by induction on $|W| + |C|$, that this terminates in polynomial time and outputs a set $A \subseteq V(T)$ such that, if $T' = T[A]$, then (T, k) and (T', k) are equivalent instances of TPT and $|V(T')| \leq c_0 \cdot c(\delta) \cdot k^\delta$, with $c_0 = 6534$. In order to obtain this result, first notice that when Rule R applied on a clean partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ returns a partial decomposition $(W', B', C', B'^{W_1}, B'^{W_2})$ with the same size and where $|W'| + |C'| < |W| + |C|$. When applying the cleaning phase, Lemma 11, on $(W', B', C', B'^{W_1}, B'^{W_2})$, we move vertices from C' to B' , leaving unchanged W' (and the sets B'^{W_1} and B'^{W_2} , as well as the size of the decomposition). If we denote by $(W'', B'', C'', B''^{W_1}, B''^{W_2})$ the clean partial decomposition obtain after the cleaning phase, we obtain $|W''| + |C''| \leq |W'| + |C'| < |W| + |C|$.

Now, we can analyze the entire process.

If $W = C = \emptyset$, then $W = \emptyset$, the graph $(G, \chi)\langle W, B, C, B^{W_1}, B^{W_2}\rangle[X]$ is the empty graph, and we consider that R returns the rainbow matching $M = \emptyset$. Thus, according to Lemma 23, the rule R outputs a set A as required. Otherwise, if $|W| + |C| > 0$, it is immediate by induction, using Lemma 23 and the previous remarks concerning cleaning phases, that A terminates in polynomial time and outputs a set $A \subseteq V(T)$ such that, if $T' = T[A]$, then (T, k) and (T', k) are equivalent instances of TPT and $|V(T')| \leq c_0 \cdot c(\delta) \cdot k^\delta$, with $c_0 = 6534$. \square

Let us conclude this section by the following observation on bit-size. According to the $\mathcal{O}(k^{2-\epsilon})$ bit-size lower bound of [5], our kernel of Theorem 4, as well as any subquadratic kernel for TPT, necessarily outputs a tournament T' whose minimum feedback arc set cannot have $m' = \mathcal{O}(k^{2-\epsilon})$ arcs for any $\epsilon > 0$. Indeed, suppose by contradiction that we have a kernel that outputs such a T' , where $|V(T')| = \mathcal{O}(k^{2-\epsilon'})$ and $m' = \mathcal{O}(k^{2-\epsilon})$. We could now use the following classical encoding of T' . Compute first a feedback arc set F' of T' with $|F'| = \mathcal{O}(m')$ using any constant approximation algorithm (for example the PTAS of [33]). Compute the topological ordering σ of $T' \setminus F'$ (meaning when removing arcs of F'), and observe that in the ordering σ , the only backward arcs of T' are F' . Thus, T' is encoded by σ and F' , which requires $|V(T')| \log(|V(T')|) + m'(2 \log(|V(T')|)) = \mathcal{O}(k^{2-\min(\epsilon, \epsilon')})$ bits. This draws a parallel with the Vertex Cover problem, which admits a kernel with $\mathcal{O}(k)$ vertices [37], but the resulting instance cannot have $\mathcal{O}(k^{2-\epsilon})$ edges [13].

7 Almost linear kernel for FVST

In this section we focus on the triangle hitting set problem, restated below. Remind that a set intersecting all the triangle of a tournament is also called a feedback vertex set.

FEEDBACK VERTEX SET IN TOURNAMENT (FVST)

Parameter: k .

Input: (T, k) where T is a tournament and $k \in \mathbb{N}$.

Question: Does T contain a feedback vertex set of size at most k ?

We obtain a almost linear kernel for this problem, which is obtained by the same algorithm

that the one designed in the previous section. The only thing that we will have to check is that, when the algorithm stops, we obtain an equivalence instance than the input instance for FVST. The way we obtain this kernelization algorithm is similar to what is done [Section 5](#), where a kernel for I2PHS is derived from the one for I2PP.

Theorem 5. *There exists an algorithm that, given an instance (T, k) of FVST outputs a set $S \subseteq V(T)$ such that $(T[S], k)$ is an equivalent instance of (T, k) where, for every δ with $1 < \delta \leq 2$, we have $|S| \leq 6534 \cdot c(\delta) \cdot k^\delta$ (where $c(\delta) = \max(\frac{20}{(2^\delta-2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$). In other words, for any δ with $1 < \delta \leq 2$ FVST admits a kernel with $6534 \cdot c(\delta)k^\delta$ vertices.*

As in [Corollary 2](#), by setting the suitable value for δ_0 , we obtain the following.

Corollary 3. *FVST admits a kernel with $k^{1+\frac{\mathcal{O}(1)}{\sqrt{\log k}}}$ vertices.*

Here again, the key tool to obtain the linear kernel for FVST is the following lemma, analog of [Lemma 21](#) for TPT. All the notations and definitions follow previous section.

Lemma 24 (Case of rainbow matching for FVST). *Let $(W, B, C, B^{W_1}, B^{W_2})$ be a clean partial decomposition of local size f , where $f(x) = cx^\delta$ for some positive constants c and δ . Suppose also that the colored multigraph $(G, \chi)(W, B, C, B^{W_1}, B^{W_2})$ admits a rainbow matching M . Let $A = V(M) \cup B \cup C$. Then, for any integer k , (T, k) is a yes-instance of FVST iff $(T[A], k)$ is a yes-instance of FVST. Moreover, we have $|A| \leq 3(c+1)(33)^\delta(k)^\delta$.*

Proof. The size requirement concerning A follows from [Lemma 21](#). Let us prove that (T, k) and $(T[A], k)$ are equivalent instances of FVST. As $T[A]$ is a subtournament of T , it is clear that if T admits a feedback vertex set of size at most k , then it is also the case for $T[A]$.

For the converse direction, assume that $T[A]$ admits a feedback vertex set X of size at most k , and let us see how to build one for T . As in the proof of [Lemma 21](#), we will first analyze the structure of M . For any $c \in C$, let $f(c)$ be the edge of color c in M ($f(c)$ is well defined as $(W, B, C, B^{W_1}, B^{W_2})$ is clean). Moreover, for any $I \in \mathcal{I}_{>0}$ (where $(\mathcal{I}, \text{val})$ is the demand of $(G, \chi)(W, B, C, B^{W_1}, B^{W_2})$) and color $d \in D_I$, let $f(I, d)$ be the edge of color d in M . Let $f(I) = \{f(I, d), d \in D_I\}$. Observe that, for any $c \in C$, by the definition of edges of colors c , $\{c\} \cup f(c)$ is a triangle in T . Let $\mathcal{Q}_2 = \{f(c), c \in C\}$ and $\mathcal{Q}_1 = \{f(I), I \in \mathcal{I}_{>0}\}$. As argued in the proof of [Lemma 21](#), \mathcal{Q}_1 is a bucket allocation for $(W, B, C, B^{W_1}, B^{W_2})$. So, denote by X_1 the set $X \cap (B \cup V(\mathcal{Q}_1))$ and by X_2 the set $X \setminus X_1$. As X is a feedback vertex set of $T[A]$, it is clear that X_1 is a vertex set of $T[B \cup V(\mathcal{Q}_1)]$. Thus by [Lemma 18](#), there exists X'_1 a feedback vertex set of $T[W \cup B]$ with $|X'_1| \leq |X_1|$. Finally, notice that X_2 is, in particular, a feedback vertex set of $T[V(\mathcal{Q}_2)]$ which contains the $|C|$ disjoint triangles $\{f(c), c \in C\}$ and that $|C| \leq |X_2|$. To conclude, we consider the set $X' = C \cup X'_1$. We have $|X'| \leq |X| \leq k$ and as every triangle of T either intersects C or is included in $T[W \cup B]$, X' is a feedback vertex set of T .

Therefore, the two instances (T, k) and $(T[A], k)$ are equivalent. \square

Now, using [Lemma 24](#) instead of [Lemma 21](#) in the proof of [Lemma 23](#), we directly obtain the analog of this latter one for FVST.

Lemma 25. *Given a clean partial decomposition $(W, B, C, B^{W_1}, B^{W_2})$ of local size f , where $f(x) = c(\delta)x^\delta$ with $c(\delta) = \max(\frac{20}{(2^\delta-2)}, (\frac{21}{\delta})^{\frac{1}{\delta-1}})$, $R(W, B, C, B^{W_1}, B^{W_2})$ either returns:*

- a set $A \subseteq V(T)$ such that, if $T' = T[A]$, then (T, k) and (T', k) are equivalent instances of FVST and $|V(T')| \leq c_0 \cdot c(\delta) \cdot k^\delta$, with $c_0 = 6534$.
- or a partial decomposition $(W', B', C', B'^{W_1}, B'^{W_2})$ of local size f , where $|W'| + |C'| < |W| + |C|$.

Now, proof of Theorem 5 works the same than the kernelization process for TPT that is Theorem 4. We start by computing a greedy localized pair (C^0, W^0) . The set C^0 induces a packing of triangles of T . If there is more than k triangles in the packing, then T has no feedback vertex set of size at most k . Otherwise, we consider the initial partial decomposition (W_0, \emptyset, C_0) of $V(G)$. Then, we exhaustively alternate a cleaning phase with an application of the rule R , until this last one falls into the matching case. As in the proof of Theorem 4, using Lemma 25 an induction on $|W| + |C|$ shows that this later case appears after a polynomial number of steps. Then we conclude with Lemma 24.

8 Conclusion

In this paper we introduced the rainbow matching technique in order to derive kernelization algorithms for the TRIANGLE-PACKING IN TOURNAMENT and FEEDBACK VERTEX SET IN TOURNAMENT problems and the INDUCED 2-PATH-PACKING and INDUCED 2-PATHS HITTING SET problems. For the two first problems we derive kernels of $k^{1 + \frac{\mathcal{O}(1)}{\sqrt{\log k}}}$ vertices, while for the two last we derive kernels of $\mathcal{O}(k)$ vertices. We stress that both kernels are producing equivalent instance that are sub(di)graphs of the original input (di)graphs. An interesting project is to investigate for which (di)graph packing or hitting set problems our technique can be used for the derivation of kernels of (almost) linear number of vertices. The general frameworks that encompass all such problems are the r -SET PACKING and the r -HITTING SET problems where, given a hypergraph H whose all hyperedges have r vertices and a non-negative integer k , the questions respectively are whether H contains k pairwise disjoint hyperedges and whether H contains a set of size at most k that intersects all the hyperedges. In [3] and [4], Abu-Khazam proved that both problems admit kernels of $\mathcal{O}(k^{r-1})$ vertices. Any improvement of any of this to a kernel of $\mathcal{O}(k^{r-1-\varepsilon})$ vertices, where the kernel is obtained by only removing vertices, would imply equal size kernels for all packing problems or hitting set problems where the structures that are packed or hitted may be modeled by the hyperedges of the input of the r -SET PACKING or r -HITTING SET problems. It is an open challenge whether our technique can be applied to these general settings.

Finally, we wish to mention that the questions of whether TRIANGLE-PACKING IN TOURNAMENT or FEEDBACK VERTEX SET IN TOURNAMENT admit a kernel of linear number of vertices remain open problems.

References

- [1] Karl A. Abrahamson, Rodney G. Downey, and Michael R. Fellows. Fixed-parameter tractability and completeness. IV. On completeness for W[P] and PSPACE analogues. *Annals of Pure and Applied Logic*, 73(3):235–276, 1995. 1
- [2] Faisal N. Abu-Khzam. A quadratic kernel for 3-set packing. In Jianer Chen and S. Barry Cooper, editors, *Theory and Applications of Models of Computation, 6th Annual Conference, TAMC 2009, Changsha, China, May 18-22, 2009. Proceedings*, volume 5532 of *Lecture Notes in Computer Science*, pages 81–87. Springer, 2009. 2, 3
- [3] Faisal N. Abu-Khzam. An improved kernelization algorithm for r -set packing. *Inf. Process. Lett.*, 110(16):621–624, 2010. 2, 3, 51
- [4] Faisal N. Abu-Khzam. A kernelization algorithm for d-hitting set. *J. Comput. Syst. Sci.*, 76(7):524–531, 2010. 2, 3, 51
- [5] Stéphane Bessy, Marin Bougeret, and Jocelyn Thiebaud. Triangle packing in (sparse) tournaments: Approximation and kernelization. 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 14:1–14:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. 3, 49
- [6] Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. On problems without polynomial kernels. *J. Comput. Syst. Sci.*, 75:423–434, December 2009. 1
- [7] Mao-cheng Cai, Xiaotie Deng, and Wenan Zang. A min-max theorem on feedback vertex sets. *Math. Oper. Res.*, 27(2):361–371, 2002. 3
- [8] Pierre Charbit, Stéphan Thomassé, and Anders Yeo. The minimum feedback arc set problem is np-hard for tournaments. *Comb. Probab. Comput.*, 16(1):1–4, 2007. 3
- [9] Marek Cygan. Improved approximation for 3-dimensional matching via bounded pathwidth local search. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 509–518. IEEE Computer Society, 2013. 3
- [10] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. 1
- [11] Frank K. H. A. Dehne, Michael R. Fellows, Frances A. Rosamond, and Peter Shaw. Greedy localization, iterative compression, modeled crown reductions: New FPT techniques, an improved algorithm for set splitting, and a novel 2k kernelization for vertex cover. In Rodney G. Downey, Michael R. Fellows, and Frank K. H. A. Dehne, editors, *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 271–280. Springer, 2004. 1

- [12] Holger Dell and Dániel Marx. Kernelization of packing problems. In Yuval Rabani, editor, *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 68–81. SIAM, 2012. [1](#)
- [13] Holger Dell and Dániel Marx. Kernelization of packing problems. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete algorithms, SODA'12*, 2012. [49](#)
- [14] Holger Dell and Dieter van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. *J. ACM*, 61(4):23:1–23:27, 2014. [1](#)
- [15] Rod Downey and Michael Fellows. Fixed-parameter tractability and completeness. III. Some structural aspects of the W hierarchy. In *Complexity theory*, pages 191–225. Cambridge Univ. Press, Cambridge, 1993. [1](#)
- [16] Rod Downey and Michael Fellows. Fixed-parameter tractability and completeness. III. Some structural aspects of the W hierarchy. In *Complexity theory*, pages 191–225. Cambridge Univ. Press, Cambridge, 1993. [1](#)
- [17] Rod G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness. I. Basic results. *SIAM J. Comput.*, 24(4):873–921, 1995. [1](#)
- [18] Rod G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness II: On completeness for $W[1]$. *Theoretical Computer Science*, 141(1-2):109–131, 1995. [1](#)
- [19] Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. [1](#)
- [20] Andrew Drucker. New limits to classical and quantum instance compression. *SIAM J. Comput.*, 44(5):1443–1479, 2015. [1](#)
- [21] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2006. [1](#)
- [22] Fedor V. Fomin, Tien-Nam Le, Daniel Lokshtanov, Saket Saurabh, Stéphan Thomassé, and Meirav Zehavi. Subquadratic kernels for implicit 3-hitting set and 3-set packing problems. *ACM Trans. Algorithms*, 15(1):13:1–13:44, 2019. [2](#), [3](#)
- [23] Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization: Theory of Parameterized Preprocessing*. Cambridge University Press, 2019. [1](#)
- [24] Fedor V. Fomin and Saket Saurabh. Kernelization methods for fixed-parameter tractability. In Lucas Bordeaux, Youssef Hamadi, and Pushmeet Kohli, editors, *Tractability: Practical Approaches to Hard Problems*, pages 260–282. Cambridge University Press, 2014. [1](#)
- [25] Lance Fortnow and Rahul Santhanam. Infeasibility of instance compression and succinct pcps for NP. *J. Comput. Syst. Sci.*, 77(1):91–106, 2011. [1](#)
- [26] Alessandra Graf, David G. Harris, and Penny Haxell. Algorithms for weighted independent transversals and strong colouring. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 662–674. SIAM, 2021. [1](#)

- [27] Alessandra Graf and Penny Haxell. Finding independent transversals efficiently. *Combinatorics, Probability and Computing*, 29(5):780–806, may 2020. [1](#), [2](#), [3](#), [5](#)
- [28] Jiong Guo and Rolf Niedermeier. Invitation to data reduction and problem kernelization. *SIGACT News*, 38(1):31–45, 2007. [1](#)
- [29] Venkatesan Guruswami, C. Pandu Rangan, Maw-Shang Chang, Gerard J. Chang, and C. K. Wong. The vertex-disjoint triangles problem. In Juraj Hromkovic and Ondrej Sýkora, editors, *Graph-Theoretic Concepts in Computer Science, 24th International Workshop, WG '98, Smolenice Castle, Slovak Republic, June 18-20, 1998, Proceedings*, volume 1517 of *Lecture Notes in Computer Science*, pages 26–37. Springer, 1998. [3](#)
- [30] Penny E. Haxell. A condition for matchability in hypergraphs. *Graphs Comb.*, 11(3):245–248, 1995. [5](#)
- [31] Danny Hermelin, Stefan Kratsch, Karolina Soltys, Magnus Wahlström, and Xi Wu. A completeness theory for polynomial (turing) kernelization. *Algorithmica*, 71(3):702–730, 2015. [1](#)
- [32] Danny Hermelin and Xi Wu. Weak compositions and their applications to polynomial lower bounds for kernelization. In Yuval Rabani, editor, *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 104–113. SIAM, 2012. [1](#)
- [33] Claire Kenyon-Mathieu and Warren Schudy. How to rank with few errors. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 95–103. ACM, 2007. [49](#)
- [34] Stefan Kratsch. Recent developments in kernelization: A survey. *Bull. EATCS*, 113, 2014. [1](#)
- [35] Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. Kernelization - preprocessing with a guarantee. 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 129–161. Springer, 2012. [1](#)
- [36] Matthias Mnich, Virginia Vassilevska Williams, and László A. Végh. A $7/3$ -approximation for feedback vertex sets in tournaments. In Piotr Sankowski and Christos D. Zaroliagis, editors, *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*, volume 57 of *LIPICs*, pages 67:1–67:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. [3](#)
- [37] George L. Nemhauser and Leslie E. Trotter Jr. Properties of vertex packing and independence system polyhedra. *Mathematical Programming*, 6(1):48–61, 1974. [49](#)
- [38] Rolf Niedermeier. *Invitation to fixed-parameter algorithms*, volume 31 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2006. [1](#)
- [39] Christian Sloper and Jan Arne Telle. An overview of techniques for designing parameterized algorithms. *Comput. J.*, 51(1):122–136, 2008. [1](#)