Egerváry Research Group
on Combinatorial Optimization

Technical reportS

TR-TR-2005-15. Published by the Egrerváry Research Group, Pázmány P. sétány 1/C, H–1117, Budapest, Hungary. Web site: www.cs.elte.hu/egres . ISSN 1587–4451.

Alternating paths revisited IV: packings and 2-packings of A-paths

Gyula Pap

December 2005

Alternating paths revisited IV: packings and 2-packings of A-paths *

Gyula Pap **

Abstract

We construct a combinatorial algorithm to find a maximum packing of fully node-disjoint $\mathcal{A}$-paths. In doing this, we discover relations with fractional packings, 2-packings, and b-packings.

1 Introduction

Path-packing problems considered in this paper are common generalizations of non-bipartite matching and node-disjoint s–t paths. Mader's theorem [6] on fully node-disjoint $\mathcal{A}$-paths is one of the most general results in this area. Chudnovsky et al. [1] proved a slight generalization of Mader's theorem, their result concerns packing non-zero A-paths. For this result, the author [7] gave a short proof – in fact for a slightly stronger result on non-returning A-paths. All these results are direct generalizations of the Berge-Tutte formula for non-bipartite matching. Edmonds [3] constructed a polynomial time algorithm for non-bipartite matching, which uses an augmentation structure called the alternating forest. A desired combinatorial algorithm for path-packing could be a direct generalization of Edmonds' matching algorithm. Such an algorithm is not known, the reason for this is probably that the alternating forest structure is not easy to generalize to the framework of path-packing. Note that two distinct approaches have already resulted in polynomial time algorithms for path-packing. First, L. Lovász [5] noticed that Mader's path-packing problem reduces to the general framework of matriod matching, and he made use of general results from that framework. However, some structural properties of path-packing are lost due to the reduction to this general setting. Recently, Chudnovsky et al. [2] constructed another algorithm which works directly with path-packings on the stage. They set up a long list of feasible path-packings in the search of an augmentation. This provides an algorithmic proof for an Edmonds-Gallai type decomposition. The algorithm proposed in this paper also reveals an odd-ear-decomposition of odd components in

*Research is supported by OTKA grants T 037547 and TS 049788, by European MCRTN Adonet, Contract Grant No. 504438 and by the Egerváry Research Group of the Hungarian Academy of Sciences.

**Dept. of Operations Research, Eötvös University, Pázmány P. s. 1/C, Budapest, Hungary H-1117. The author is a member of the Egerváry Research Group (EGRES). e-mail: `gyuszko@cs.elte.hu`

this decomposition. Most steps can be considered as local manipulations rather than a global augmentation structure. Our algorithm considers a slightly broader class of path-packing problems. This is necessary so that our reduction principle stays within this class, in order to use induction.

2 Notation

We consider undirected graphs $G = (V, E)$ with a reference orientation. That is, the edges are ordered pairs, but we consider paths, walks, neighbours, contractions, subgraphs as if we would have undirected edges. The deletion of a edge set $F \subseteq E$, resp. an node set $U \subseteq V$ is defined by $G - F := (V, E - F)$, resp. $G - U := (V - U, \{ab \in E : a, b \notin U\})$. For a set $Z \subseteq V$, the induced subgraph is defined by $G[Z] = (Z, E[Z]) := G - (V - Z)$. We obtain the contracted graph G/Z by indentifying nodes in Z by a new node $\{Z\}$, deleting edges in $E[Z]$, and keeping possible parallel edges. For a set $F \subseteq E$ we define $V(F) := \{u \in V : \exists uv \in F\}$. If $(V(F), F)$ is connected, then we also define the contracted graph G/F by identifying the set of nodes in $V(F)$ by a new node $\{F\}$, deleting the edges in F, and keeping possible parallel edges and loops on the new node.

3 Packings in p-graphs — Definitions

The most important notion in this paper is a **permutation labeled graph** or **p-graph**, for short. A p-graph comes in the form of G, A, ω, π, where G is a graph, A is a set of nodes, π are edge-labels. This notion provides a generalization of some well-known packing problems – matching, node-disjoint $\mathcal{A}$-paths, non-zero A-paths. The motivation for this version is that important reduction principles used by our algorithm stay within the concept of a p-graph, but does not stay within well-known previous concepts. The precise definition of a p-graph is formulated as follows.

Let $G = (V, E)$ be an undirected graph with node-set V, edge-set E with a reference orientation. Let $A \subseteq V$ be a fixed set of **terminals**. Let Ω be an arbitrary **set of "potentials"** and let $\mathbf{jj}, \mathbf{JJ}$ be called **Jolly Joker** (some imaginary labels). Let $\omega : A \to \Omega$ define the **potential of origin** for the terminals. Let $\pi : E \to S(\Omega) \cup \{\mathbf{JJ}\}$ where $S(\Omega)$ is the set of all permutations of Ω. For an edge $ab = e \in E$, let $\pi(e, a) := \pi(e)$ and $\pi(e, b) := \pi^{-1}(e)$ be the **mapping of potential** on edge ab.

(We use $\circ$ for the composition of permutations. We define $\mathbf{JJ}^{-1} := \mathbf{JJ} \circ \pi := \pi \circ \mathbf{JJ} := \mathbf{JJ}$ and $\mathbf{JJ}(\omega) := \pi(\mathbf{jj}) := \mathbf{jj}$ for any $\pi \in S(\Omega) \cup \{\mathbf{JJ}\}$ and for any $\omega \in \Omega \cup \{\mathbf{jj}\}$.)

A **walk** in G is a sequence of nodes and edges, say $W = (v_0, e_0, v_1, e_1, \ldots, e_{k-1}, v_k)$ where $e_i = v_i v_{i+1}$ or $e_i = v_{i+1} v_i$ for all $0 \le i \le k - 1$. W is called an **A-walk** in G if $v_0, v_k \in A$ and $v_j \notin A$ (for $j \neq 0, k$). $\chi_W \in \mathbb{N}^V$ denotes the **traversing multiplicity vector** of walk W, defined by $\chi_W(v) := |\{j : v_j = v\}|$. A walk W is called a **path** if $\chi_W \le \mathbf{1}$. We will usually use letters P, R for paths. For an A-walk let $\pi(W) := \pi(e_0, v_0) \circ \pi(e_1, v_1) \circ \ldots \circ \pi(e_{k-1}, v_{k-1})$ define the **mapping of potentials on** W. W is called **non-returning** if $\pi(W)(\omega(v_0)) \neq \omega(v_k)$. (Hence, an empty A-walk (having a single node and no edge) is not considered to be non-returning. Notice, if W traverses any edge with label **JJ**, then W is non-returning.)

A family $\mathcal{P}$ of fully node-disjoint non-returning A-paths is called a **packing**. $\nu = \nu(G) = \nu(G, A, \omega, \pi)$ denotes the **maximum cardinality of a packing**. Also, a "node-capacited packing problem" can be defined. Consider a function $b \in \mathbb{N}^V$ of **node capacities**. A family $\mathcal{W}$ of A-walks (we allow walks to be taken multiply) is called a b-**packing** if $\sum_{W \in \mathcal{W}} \chi_W \leq b$. Let $\nu_b = \nu_b(G) = \nu_b(G, A, \omega, \pi)$ denotes the maximum cardinality of a b-packing. $b = \mathbf{1}$ defines packings, $b = \mathbf{2}$ defines 2-packings.

Remarks.

1. A **subgraph** G' **of** G **induces a p-graph** with $A' := A \cap V(G'), \omega' := \omega|_{A'}, \pi' := \pi|_{E(G')}$. We sometimes only refer to this p-graph as G'.

2. For a packing $\mathcal{P}$, let $V(\mathcal{P})$ denote the nodes traversed by some path in $\mathcal{P}$, and let $A(\mathcal{P}) := A \cap V(\mathcal{P})$. Terminals in $A - A(\mathcal{P})$ are **exposed by** $\mathcal{P}$.

3. Notice, the above definitions do not actually depend on the reference orientation. To reassign the opposite reference orientation to an edge e, just redefine $\pi(e)$ by its inverse. This way we retain the same mapping on any walk W.

4. Let us remark that allowing Jolly Jokers does not give a real generalization. We replace edges with label **JJ** by some "general" permutations – for this we possibly need to use a larger groundset. Details of this construction are left to the reader.

4 Min-max Theorems for packings

b-packing reduces to packings by splitting vertices $v \in V$ into $b(v)$ copies. The main result on packings and 2-packings are Theorems 4.1,4.2,4.3. Let us immediately point out that Theorem 4.2 can be derived from Theorem 4.1 by the principle of splitting nodes into 2 copies, while Theorem 4.3 is just a reformulation of Theorem 4.2.

We define the notions on the dual side of the min-max formula. For a set $F \subseteq E$ of edges, let $A^F := A \cup V(F)$. F is called A-**balanced** if ω can be extended to a function $\omega^F : A^F \to \Omega$ such that each edge $ab \in F$ is ω^F-**balanced** – i.e. $\pi(ab, a)(\omega^F(a)) = \omega^F(b)$. (Notice, no edge in F may have label **JJ**. Some easy examples for A-balenced edge sets are: edge set of a returning A-path, or a forest without **JJ**.) Let $c_{odd}(G, A)$ be the number of components in G having an odd number of nodes in A – these will be called **odd components of** G, A. Moreover, we define $c_1(G, A)$ to be the number of nodes in A which are isolated nodes of G – these will be called **isolated nodes of** G, A. (An isolated node has no loop-edge.) Clearly, $c_1(G, A) \leq c_{odd}(G, A)$. We give the below min-max formulae for ν, ν_2, respectively.

Theorem 4.1. *In a p-graph the maximum cardinality of a packing is determined by*

$$\nu(G, A, \omega, \pi) = \min_{F,X} \; |X| + \frac{1}{2}\left(|A^F - X| - c_{odd}(G - F - X, A^F - X)\right) \; , \qquad (1)$$

where the minimum is taken over an A-balanced edge-set F and a set $X \subseteq V$.

Theorem 4.2. *In a p-graph the maximum cardinality of a 2-packing is determined by*

$$\nu_2(G, A, \omega, \pi) = \min_{F,X} \; 2|X| + |A^F - X| - c_1(G - F - X, A^F - X)\ , \tag{2}$$

where the minimum is taken over an A-balanced edge-set F and a set $X \subseteq V$.

In Theorem 4.2 we do not count odd components to determine a maximum 2-packing, this indicates that 2-packings are simpler than packings. There is a similar relation between matchings and 2-matchings, the latter admitting a reduction to bipartite matching, Kőnig's Theorem. The following theorem is in fact a reformulation of Theorem 4.2, here we formulate a Kőnig-type condition for 2-packings.

Theorem 4.3. *In a p-graph the maximum cardinality of a 2-packing is determined by*

$$\nu_2(G, A, \omega, \pi) = \min \; ||c||\ , \tag{3}$$

where $||c|| := \sum_{v \in V} c(v)$ and the minimum is taken over ***2-covers*** *c, i.e. vectors $c \in \{0, 1, 2\}^V$ such that $c \cdot \chi_W \geq 2$ for any non-returning A-walk.*

To see that these theorems give good characterizations notice that A-balance may be checked by depth-first-search. Checking whether a vector c is a 2-cover may be done as follows. Delete all nodes with a "2"-entry, delete all but one nodes with a "1"-entry. Check whether in the remaining graph there is a non-returning A-walk.

Proof. (of "max ≤ min" for Theorems 4.1 and 4.2.) Consider an A-balanced edge-set F and a set $X \subseteq V$. Let $\mathcal{K}$ be the family of components of $G - X - F$. Suppose W is a non-returning A-walk disjoint from X. Then there is at least one component $K \in \mathcal{K}$ such that at least two of the nodes of walk W are in $V(K) \cap (A^F - X)$ (i.e. $\chi_W \cdot \chi_{V(K)\cap(A^F - X)} \geq 2$). □

Proof. (of "max ≤ min" for Theorem 4.3.) Consider a 2-packing $\mathcal{W}$ and a 2-cover c.

$$2||c|| = c \cdot \mathbf{2} \geq c \cdot \sum_{W \in \mathcal{W}} \chi_W = \sum_{W \in \mathcal{W}} c \cdot \chi_W \geq \sum_{W \in \mathcal{W}} 2 = 2|\mathcal{W}|.$$

□

A pair X, F of a set $X \subseteq V$ and an A-balanced set $F \subseteq E$ is called a **verifying pair with respect to a packing** $\mathcal{P}$ if equality $|\mathcal{P}| = |X| + \frac{1}{2}\big(|A^F - X| - c_{odd}(G - F - X, A^F - X)\big)$ holds. Let us call a pair X, F a **2-packing verifying pair with respect to a 2-packing** $\mathcal{W}$ if equality $|\mathcal{W}| = 2|X| + |A^F - X| - c_1(G - F - X, A^F - X)$ holds. Let us call a 2-cover c a **2-packing verifying 2-cover with respect to a 2-packing** $\mathcal{W}$ if equality $|\mathcal{W}| = ||c||$ holds.

Proof of equivalence of Theorem 4.2 and Theorem 4.3. Let us call the expression on the right hand side as the **value of** X, F, resp. c.

Claim 4.4. *Given a pair X, F, one can construct a 2-cover c with a smaller or equal value.*

Proof. Let $c(v) := 2$ for $v \in X$ and let $c(v) := 1$ for nodes $v \in V(F)$ s.t. $\{v\}$ is not an isolated node of $G - X - F$. □

Claim 4.5. *Given a 2-cover c, one can construct a pair X, F with a smaller or equal value.*

Proof. $X := \{v \in V : c(v) = 2\}$, $Y := \{v \in V : c(v) = 1\}$. Let B be the union of components of $G - X - Y$ which have at least one node in A. Let $F := \{ab \in E[V - X] : a \in B \text{ or } b \in B\}$, which is easily seen to be A-balanced. We claim that $\{v \in A^F - X : v \text{ is not isolated in } G - F - X\} \subseteq Y$. Thus

$$||c|| \geq 2|X| + |A^F - X| - c_1(G - F - X, A^F - X).$$

□

Remark. We only need to consider pairs X, F in Theorems 4.1, 4.2 for which $X \cap V(F) = \emptyset$. This can be easily seen, since the deletion of an edge in F incident with a node in X does not increase the dual value.

5 A remark on $2b$-packings

In this section we consider the problem of determining a maximum packing in a node-capacitated p-graph when all the nodes have even capacities.

Theorem 5.1. *Consider an non-negative integer vector $b \in \mathbb{N}^V$. In a p-graph the maximum cardinality of a $2b$-packing is determined by*

$$\nu_{2b}(G, A, \omega, \pi) = \min \ cb \ , \tag{4}$$

where the minimum is taken over 2-covers c (see Theorem 4.3).

(A sketch of the proof of this theorem goes as follows. We reduce $2b$-packing to Theorem 4.1 by splitting nodes into $2b$ copies. A structural theorem holds similar to the one proved by Chudnovsky et al. [1], implying that a unique optimum dual F, X exists which is canonically determined by the input p-graph. Hence this canonical F, X obeys the symmetry of permuting copies of the same node. This can be transformed into a 2-cover which is constant over the copies of any fixed node. The pre-image of this 2-cover solves Theorem 5.1.)

To determine ν_{2b}, we will make use of the observation that the right hand side of (4) reduces to linear optimization over a polytope for which separation is polytime solvable. For this notice that if a fractional vector c has the property that every non-returning A-walk has c-length at least 2, then cb will also be an upper bound on ν_{2b}. A polyhedral description of fractional 2-covers can be given as follows:

$$c(v) \geq 0 \qquad \text{for all } v \in V \tag{5}$$

$$c(\chi_W) \geq 2 \qquad \text{for all non-returning walks } W \tag{6}$$

So the right hand side of (4) is equal to the minimum of cb over (5)-(6). It is quite easy to see that one can separate inequalities (6) in time polynomial in V, E and Ω. (To check whether a vector c obeys (6), one runs $|A| \cdot |A| \cdot (|\Omega| - 1)$ shortest paths algorithms in an expanded graph.) By Theorem 5.1, the optimum is attained by a $0, 1, 2$-vector c. By trying $0, 1, 2$ for entries of c, and fixing them one by one so that the optimum value does not decrease, one can also find an integer optimum c.

6 Contraction of dragons

A path P is called a **half-A-path** if it starts in a terminal $s \in A$, ends in a node $t \in V$ and $V(P) \cap A = \{s\}$. We say P **ends in t with potential** $\pi(P)(\omega(s))$.

Consider a node $v \in V$ and a potential $\omega_0 \in \Omega \cup \{\mathbf{jj}\}$. We say a node v **is ω_0-reachable** (or ω_0 is reachable at v), if there is a pair $\mathcal{P}, P_v$ such that P_v is a half-A-path ending in v with ω_0, and $\mathcal{P}$ is a packing of ν non-returning A-paths each of which is fully node-disjoint from P_v. We say a node is **reachable** if it is ω_0-reachable for some $\omega_0 \in \Omega \cup \{\mathbf{jj}\}$. v is called **uniquely reachable** if it is ω_0-reachable only with a single element $\omega_0 \neq \mathbf{jj}$. Otherwise – if v is $\mathbf{jj}$-reachable or there are at least two different elements of Ω which are reachable at v, then v is called **multiply reachable**. The definition implies that a reachable terminal is uniquely reachable.

We call a p-graph G a **dragon** if $|A| = 2\nu + 1$ and every node is reachable. A p-graph is called **critical** if it is a dragon such that every non-terminal is multiply reachable. (The notion of criticals is analogue to the notion used in [1]. The notion of dragons should be considered as a weak version of criticality.)

Let us use the expression **odd cycle** for p-graphs s.t. $G = (V, E)$ is an odd cycle, $A = V$, and all the edges in E give one-edge non-returning A-walks (which are in fact non-returning A-paths except for 1-edge odd cycles). A p-graph with $V = \{a, b\}$, $E = \{ab\}$, $A = \{a\}$ is called a **rod**.

Claim 6.1. *Odd cycles and rods are dragons.* □

A crucial lemma is the following, saying that the min-max formula holds for dragons. This Lemma will only be proved in the end by using our algorithm. So, we will first give a partial proof of the algorithm and this Lemma, which will be completed in Section 11. The Lemma is stated at this point to "filter" the concept of the algorithm.

Lemma 6.2 (A dragon has a special dual). *Suppose a G is a dragon with exactly its nodes in V_1 being uniquely reachable, say $v \in V_1$ is $\omega'(v)$-reachable. Let $F := \{e \in E[V_1] : e$ is ω'-balanced$\}$. Then $2\nu = |V_1| - c(G - F, V_1)$.*

The notion "reachability" is in fact motivated by the goal to define the contraction of dragon subgraphs. Consider a set $Z \subseteq V$ such that $G[Z]$ is dragon. We wish to define the contraction of Z in such a way that the following property holds. The contracted p-graph should be defined on the contracted graph G/Z with terminals $A/Z := A - Z + \{Z\}$, and ω_Z, π_Z should be defined in such a way that any A/Z-path P in G/Z is non-returning if and only if it has an **expansion in** G. An expansion of P is a packing $\mathcal{P}$ s.t. $\mathcal{P}/Z = \{P\}$, and moreover $\mathcal{P} \cap G[Z]$ is a packing in $G[Z]$ of size $\nu(G[Z]) = (|A \cap Z| - 1)/2$.

Definition 6.3 (Contraction of a dragon). Consider a set $Z \subseteq V$ such that $G[Z]$ is dragon. We define the contracted p-graph on G/Z as follows.

1. Let Z_1 be the uniquely reachable nodes in $G[Z]$, say $a \in Z_1$ is ω_a-reachable.
2. Let $A/Z := A - Z + \{Z\}$. Let $\Omega' := \Omega + \bullet$ for some new element $\bullet \notin \Omega$.
3. Let $\omega_Z(s) := \omega(s)$ for all $s \in A/Z - \{Z\}$, and let $\omega_Z(\{Z\}) := \bullet$.
4. We define $\pi_Z(e)$ by the following case splitting.
   (a) If e is disjoint from Z, then we define $\pi_Z(e)$ by extending $\pi(e)$ to Ω' by mapping $\bullet$ to $\bullet$.
   (b) For an edge ab with $a \in Z_1$, $b \notin Z$ we label its image $\{Z\}b$ such that $\pi_Z(\{Z\}b)(\{Z\}) = \pi(ab)(\omega_a)$.
   (c) For an edge ab with $a \in Z - Z_1$, $b \notin Z$ we define let $\pi_Z(\{Z\}b) := \mathbf{JJ}$.

Remarks.

1. Let us interpret the definition in (b). We only care about which A-walks will be non-returning, and which will be returning. So, for some $a \in A$, the only thing we need to know about a permutation label $\pi(ab)$ is its evaluation at $\pi(ab)(\omega(a))$.

2. Clearly, the contraction of a dragon Z does not have an effect on the part of G disjoint from Z. That is, the p-graph $G[V - Z]$ is the same as $G/Z[V - Z]$. We define the **contraction of a node-disjoint family $\mathcal{Z}$ of dragons** $G/\mathcal{Z}, A/\mathcal{Z}, \omega_{\mathcal{Z}}, \pi_{\mathcal{Z}}$ by contracting the dragons in $\mathcal{Z}$ one-by-one. This is legal, since the contraction of a dragon does not change the others from being a dragon. Whether an $A/\mathcal{Z}$-walk is non-returning does not depend on the order in which we contract the members of $\mathcal{Z}$. Hence the notation $G/\mathcal{Z}, A/\mathcal{Z}, \omega_{\mathcal{Z}}, \pi_{\mathcal{Z}}$ stands for the contracted p-graph in any order of contractions.

3. In the following claims we formulate precisely the property of expansion we wished the contraction of a node-disjoint family of dragons would have.

Claim 6.4 (Expansion of a path). *Consider a non-returning $A/\mathcal{Z}$-path P in $G/\mathcal{Z}$. Then P has an **expansion** in G, i.e. there is a packing $\mathcal{P}$ in G with the following properties. $\mathcal{P}/\mathcal{Z} = P$, and the paths in $\mathcal{P}$ cover all the terminals in the pre-image of P.*

Proof. Directly follows from Definition 6.3 . □

Claim 6.5 (Expansion of a packing). *From any packing in $G/\mathcal{Z}$ one can construct a packing in G which exposes the same number of terminals.*

Proof. Directly follows from Claim 6.4. □

Claim 6.6 (Pre-image of a dragon). *Consider a dragon Z_1 in $G/\mathcal{Z}$. Then the pre-image of Z_1 is dragon. Moreover, $\mathcal{Z}/Z_1 := \{Z : Z \in \mathcal{Z}, \{Z\} \notin Z_1\} \cup \{$the pre-image of $Z_1\}$, is a finer node-disjoint family of dragons.*

Proof. The first claim directly follows from Claim 6.4. The last claim follows from the first. □

7 Sequences of contractions and the 3-Way Lemma

Consider a packing $\mathcal{P}$ in G and a dragon Z in G. We say $\mathcal{P}$ is **equipped with** Z if $\mathcal{P}$ consists of some paths disjoint from $V(Z)$ and exactly $\nu(G[Z]) = (|A \cap V(Z)| - 1)/2$ paths inside Z. A **sequence of contractions** is a sequence

$$\begin{aligned}
&(\mathcal{Z}_1, G_1, \mathcal{P}_1, \mathcal{R}_1, S_1)\\
&(\mathcal{Z}_2, G_2, \mathcal{P}_2, \mathcal{R}_2, S_2)\\
&\ \cdots\\
&(\mathcal{Z}_m, G_m, \mathcal{P}_m, \mathcal{R}_m, S_m)\\
&(\mathcal{Z}_{m+1}, G_{m+1}, \mathcal{P}_{m+1})
\end{aligned} \tag{7}$$

with $m \geq 0$, and the below properties 1,2,3. These are the key properties of a sequence of contractions used by our algorithm.

1. $\mathcal{Z}_0 = \emptyset$, and $\mathcal{Z}_i$ is a node-disjoint family of dragons in G. $G_i = (V_i, E_i) := G/\mathcal{Z}_i$.
2. $G_i[S_i]$ is an odd cycle or a rod, where $S_i \subseteq V_i$. $\mathcal{R}_i$ is a packing in G_i which is equipped with S_i.
3. $\mathcal{P}_{i+1} := \mathcal{R}_i/S_i$, $\mathcal{Z}_{i+1} := \mathcal{Z}_i/S_i$ for $i = 1, \cdots, m$.
4. Each $\mathcal{P}_i, \mathcal{R}_i$ leaves the same number of terminals uncovered.

Clearly, $\mathcal{P}_i$ is a packing in G_i. Notice that in the $m+1$'st line we only have 3 elements, in all other lines we have 5. Notice, the $\bigcup \mathcal{Z}_i$ is laminar. There is a trivial sequence of contractions with $m = 0$ and $\mathcal{Z}_0 = \emptyset$.

The 3-way Lemma 7.2 will be applied sequentially to construct sequences of contractions, this will be shown later. The proof of Theorem 4.1 and the algorithm relies on the following key observation, which provides a tool to construct a verifying pair. We construct from a 2-packing verification in a contraction a packing verification in the original p-graph. This is precisely formulated in the following lemma.

Lemma 7.1 (Constructing a verifying pair). *Suppose we have a sequence of contractions, and a 2-cover c in G_{m+1} with $2|\mathcal{P}_{m+1}| = ||c||$. Then for all i, $\mathcal{P}_i$ is a maximum packing in G_i and one can construct a verifying pair for $\mathcal{P}_i$.*

Lemma 7.2 (The 3-way Lemma). *Consider a p-graph with a packing $\mathcal{P}$. Then at least one of the following alternatives holds:*

1. *There is a packing $\mathcal{R}$ with $|\mathcal{R}| = |\mathcal{P}| + 1$.*
2. *There is a packing $\mathcal{R}$ s.t. $|\mathcal{R}| = |\mathcal{P}|$, and is equipped with a rod or an odd cycle.*
3. *There is a 2-cover c such that $2|\mathcal{P}| = ||c||$. (I.e. a verifying 2-cover for $2 \times \mathcal{P}$)*

In the next section we show how these two lemmas imply a proof and an algorithmi for packing.

8 The algorithmic proof of Theorem 4.1

We have stated useful lemmas 6.5,6.6,7.1,7.2. Now we are in position to describe the algorithmic proof.

Proof. We construct an algorithm "A" with the input of a p-graph, and a packing $\mathcal{P}$. The output is either a larger packing, or a verifying pair for $\mathcal{P}$. Iterative application of "A" provides a constructive proof of Theorem 4.1.

The algorithm "A" starts off with initiating the trivial sequence of contractions, $m = 0$. In a general step, apply Lemma 7.2 to $G_{m+1}, \mathcal{P}_{m+1}$! If alternative 1 holds, then by Claim 6.5 one can construct a packing in G larger than $\mathcal{P}$. If alternative 2 holds, then by Claim 6.6 one can construct a longer sequence of contractions. If alternative 3 holds, then by Claim 7.1 $\mathcal{P}$ is maximum, and a verifying pair can be constructed. The proof of correctness and polynomiality is completed by observing that the length of a sequence of contractions is bounded by $|E| + |V|$. □

Remarks. The above proof will only be algorithmic if we can be deal with the following tasks algorithmically.

1. We need to maintain unique/multiple reachability of nodes in a dragon. We will show that reachability can be maintained for those dragons which appear in a sequence of contractions. In fact dragons in a sequence of contractions have a special decomposition called the "dragon decomposition", Lemma 6.2 will in fact be proved by showing that the dragon decomposition can be maintained throughout a sequence of contractions. A polynomial time algorithm is constructed by maintaining a dragon decomposition for dragons in $\mathcal{Z}_i$.

2. To construct an algorithm we also need an oracle to deal with the p-graph, the permutations. It turns out that it is enough to have an oracle to check whether some A-walk is non-returning. Given such an oracle for $G = G_0$, an oracle for G_i can be constructed by calling the G-oracle. We skip this construction.

9 Proof of the 3-way Lemma

The proof goes by induction on $|V(G)| + |E(G)|$.

First we check whether $c := \chi_{A(\mathcal{P})}$ is a 2-cover. If not, then there exists a non-returning A-walk W starting in $a \in A - A(\mathcal{P})$. If W consists of a single edge e which is disjoint from $V(\mathcal{P})$, then alternative 2 holds if e is a loop, or alternative 1 holds if e is a non-loop edge. If W consists of more than one edge, and its first edge is disjoint from $V(\mathcal{P})$, then clearly, alternative 3 holds.

It remains that there is an edge $ab \in E$ such that

$$a \in A - A(\mathcal{P}),\ b \in V(\mathcal{P}),\ \text{and if } b \in A \text{ then } ab \text{ is a non-returning path.} \tag{8}$$

Say $b \in V(P)$ for some $P \in \mathcal{P}$. If $b \notin A$, then there are 3 A-paths inside $P \cup ab$. By the definition of non-returning A-paths either 2 or 3 of them are non-returning. These are the two cases to be distinguished below. Say edge ab meets path $P \in \mathcal{P}$, b splits P into sections $P = (s_1, P_1, b, P_2, s_2)$

Case 1. $b \in V - A$, $ab + P_1$ is a returning A-path and $ab + P_2$ is a non-returning A-path.

Suppose P_1 has more than one edge, say its first edge is $s_1 s_1'$. Then $\mathcal{R} := \mathcal{P} - P + (ab + P_2)$ and rod $s_1 s_1'$ solves alternative 2. So, suppose P_1 has exactly one edge, namely $s_1 b$. We define a contracted graph G', A', ω', π' as follows. Let $G' := (G - ab - s_1 b))/\{s_1, a\}$, let p_1 denote the new node, let $A' := A - a - s_1 + b + p_1$ and

$$\omega'(v) := \begin{cases} \omega(v) & \text{for } v \in A' - b - p_1 \\ \pi(ab)(\omega(a)) & \text{for } v = b \\ \bullet & \text{for } v = p_1 \end{cases}$$

and

$$\pi'(uv) := \begin{cases} \pi(uv) & \text{if } u, v \neq p_1 \\ \bullet \mapsto \pi(av)(\omega(a)) & \text{if } uv \text{ is the image of an edge } av \\ \bullet \mapsto \pi(av)(\omega(a)) & \text{if } uv \text{ is the image of an edge } s_1 v\,. \end{cases}$$

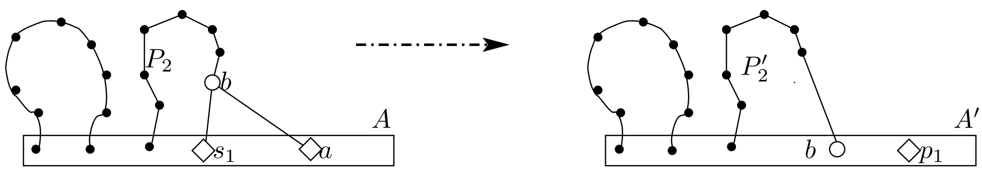

Figure 1: The reduction in **Case 1.**

If alternative 3 holds with c', then one can easily see that $c'(p_1) = 0$. Hence alternative 3 is solved by $c(u) := c'(u)$ (for $u \notin \{s_1, a\}$) and $c(a) := c(s_1) := 0$. We prove that if one of alternatives 1,2 holds for G', then we find a solution also for G. Consider $\mathcal{R}'$, $\mathcal{R}'$ and S', $\mathcal{R}'$ and $u'v'$ in the respective alternative for G'. We find one of the alternatives for G by considering the pre-image of the solution (i.e. $\mathcal{R}'$, $\mathcal{R}'$ and C', $\mathcal{R}'$ and $u'v'$) plus $ab, s_1 b$. To check these we only need to check components incident with p_1, b. So at most two components we need to think about, each of which may be a non-returning A-path in $\mathcal{R}'$, the odd cycle C, or the edge $u'v'$. Subcases are depicted in Figure 2.

Case 2. $b \in A$.

If P had more than one edge, then $\mathcal{P} - P + ab$ and the first edge of P would solve alternative 2. So, suppose $P = s_1 b$. Then $a, b, s_1 \in A$. We define a contracted p-graph G', A', ω', π'. Let $G' := (G - b)/\{s_1, a\}$, let s denote the contracted node $A' := A - a - b - s_1 + s$ and

$$\omega'(v) := \begin{cases} \omega(v) & \text{for } v \in A' - s \\ \bullet & \text{for } v = s \end{cases}$$

and

$$\pi'(uv) := \begin{cases} \pi(uv) & \text{if } u, v \neq s \\ \bullet \mapsto \pi(s_1 v)(\omega(s_1)) & \text{if } uv \text{ is the image of an edge } s_1 v \\ \bullet \mapsto \pi(av)(\omega(a)) & \text{if } uv \text{ is the image of an edge } av\,. \end{cases}$$

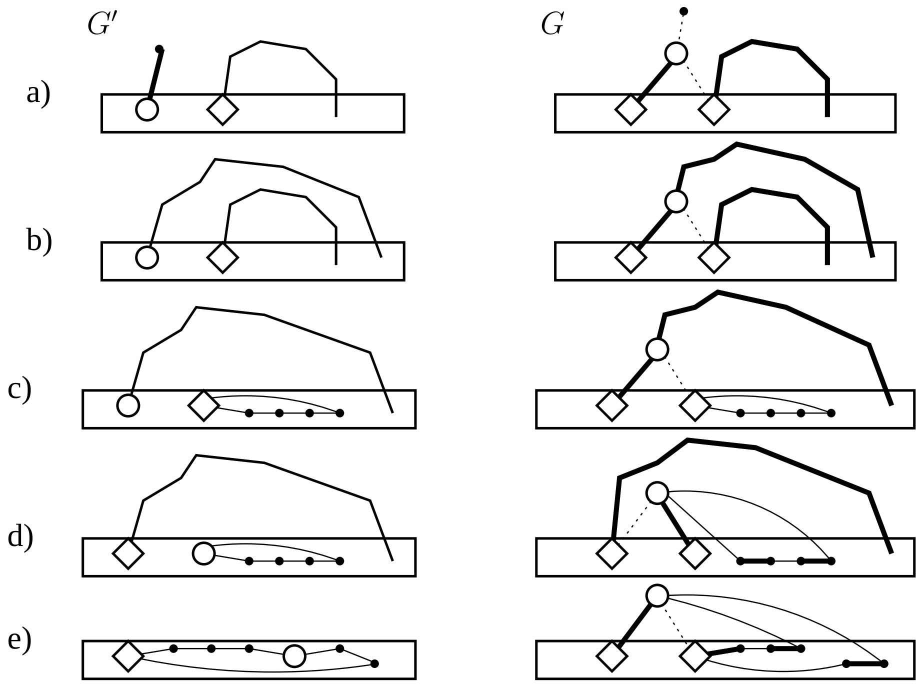

Figure 2: Subcases of Case 1.

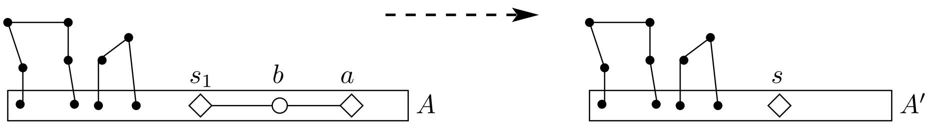

Figure 3: The reduction in **Case 2.**

If alternative 3 holds with c', then one can easily see that $c'(s) = 0$. Hence alternative 3 is solved by $c(u) := c'(u)$ (for $u \notin \{a, b, s_1\}$), $c(a) := c(s_1) := 0$ and $c(b) := 2$. We claim that if one of alternatives 1,2 holds for G', then we find a solution also for G. Subcases are depicted in Figure 4.

Case 3. $b \notin A$ and $ab + P_1$, $ab + P_2$ are both non-returning A-paths.

If P_1 or P_2 has more than one edge, then one can see by a similar argument that alternative 3 holds. So, suppose $P \cup ab$ is a claw with center $b \in V - A$ and tips $a, s_1, s_2 \in A$ such that all 3 of the A-paths in the claw are non-returning. Let $G' := (G - E(P) - ab - b)/(V(P) - b + a)$, let s denote the contracted node $A' := A - s_1 - s_2 - a + s$ and

$$\omega'(v) := \begin{cases} \omega(v) & \text{for } v \in A' - s \\ \bullet & \text{for } v = s \end{cases}$$

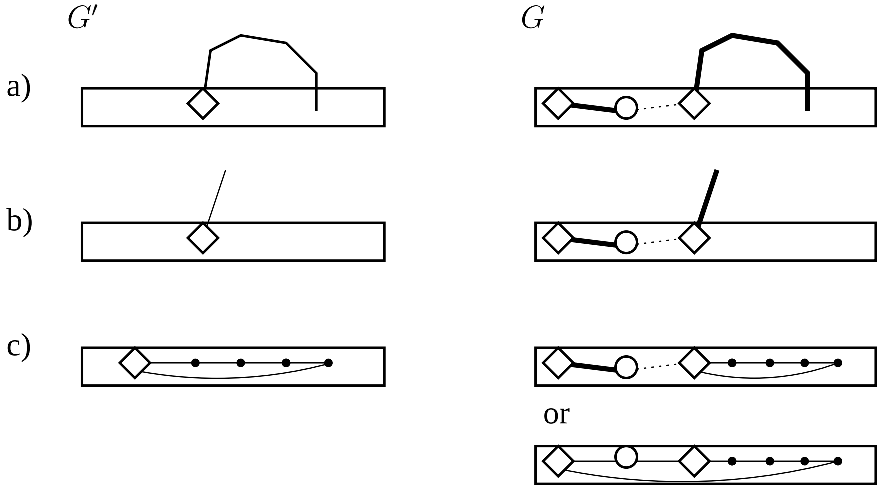

Figure 4: Subcases of Case 2.

and

$$\pi'(uv) := \begin{cases} \pi(uv) & \text{if } u, v \neq s \\ \bullet \mapsto \pi(s_1 v)(\omega(s_1)) & \text{if } uv \text{ is the image of an edge } s_1 v \\ \bullet \mapsto \pi(s_2 v)(\omega(s_2)) & \text{if } uv \text{ is the image of an edge } s_2 v \\ \bullet \mapsto \pi(av)(\omega(a)) & \text{if } uv \text{ is the image of an edge } av\,. \end{cases}$$

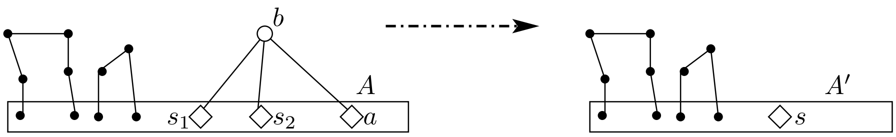

Figure 5: The reduction in **Case 3.**

If alternative 3 holds with c', then one can easily see that $c'(s) = 0$. Hence alternative 3 is solved by $c(u) := c'(u)$ (for $u \notin \{a, b, s_1, s_2\}$), $c(a) := c(s_1) := c(s_2) := 0$ and $c(b) := 2$. We claim that if one of alternatives 1,2 holds for G', then we find a solution also for G. Subcases are depicted in Figure 6.

10 Proof of Lemma 7.1

By Claim 4.5, there is a 2-packing verifying pair X, F_{m+1} for $2 \times \mathcal{P}_{m+1}$. For $i < m+1$ let $\{S_i\}^{(m+1)} \in V_{m+1}$ denote the image of S_i in G_{m+1}. We will construct a pair F_1, X below, and to prove that it is a verifying pair with respect to $\mathcal{P}_1$ we are going to use that for all $i' \geq 1$ the following holds:

$$\{S_{i'}\}^{(m+1)} \in A_{m+1}^{F_{m+1}} - X \text{ and } \{S_{i'}\}^{(m+1)} \text{ is an isolated node of } G_{m+1} - X - F_{m+1}. \tag{9}$$

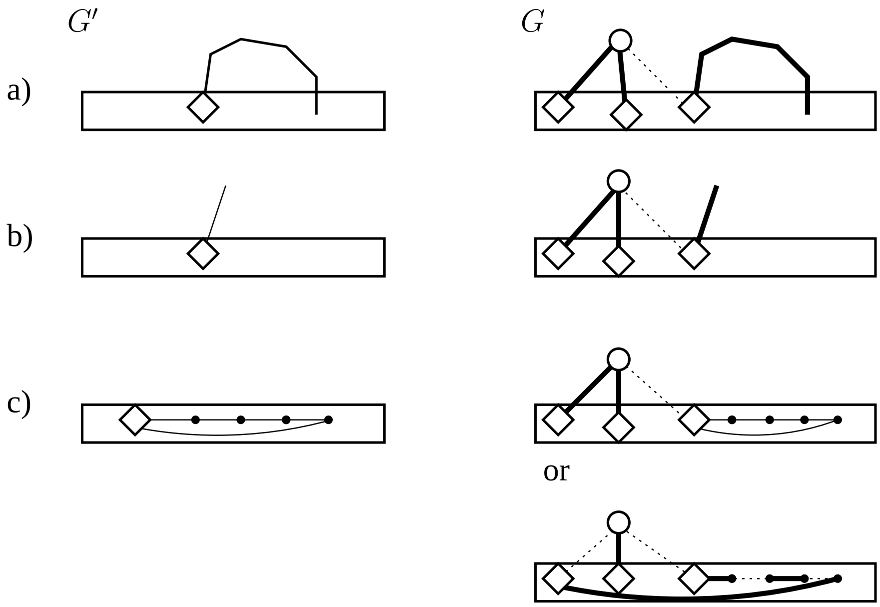

Figure 6: Subcases of Case 3.

Consider the smallest positive integer i such that for all $i' \geq i$ (9) holds. (If (9) holds for no i', then we consider $i = m + 1$.)

Let $\{Z_{1,m+1}, Z_{2,m+1}, \cdots, Z_{q,m+1}\}$ be the family of dragons in $\mathcal{Z}_{m+1}$ which are the images of at least one of $\{\{S_{i'}\}^{(m+1)} : i \leq i'\}$. Let $\mathcal{Z}^{(i)}_{m+1} = \{Z^{(i)}_{1,m+1}, Z^{(i)}_{2,m+1}, \cdots, Z^{(i)}_{q,m+1}\}$ denote their pre-images in G_i. We claim that each $G_i[Z^{(i)}_{j,m+1}]$ is a dragon – this follows from Claim 6.6. Let $U^{(i)}_{j,m+1} \subseteq Z^{(i)}_{j,m+1}$ denote the set of uniquely reachable nodes of $G_i[Z^{(i)}_{j,m+1}]$. Let $U := \bigcup_j U^{(i)}_{j,m+1}$. By definition, there exists a function ω''_i on U s.t. a node $v \in U$ is uniquely $\omega''_i(v)$-reachable (in $G_i[U^{(i)}_{j,m+1}]$). Let $F_{ij} := \{uv : \exists j, uv \in E_i[U^{(i)}_{j,m+1}]$, and uv is ω''_i-balanced$\}$

$$F_i := \{\text{pre-image of } F_{m+1} \text{ in } G_i\} \cup \bigcup F_{ij}. \tag{10}$$

Claim 10.1. *F_i, X is a verifying pair for $\mathcal{P}_i$.*

Proof. First we show that F_i is A_i-balanced. Consider the set H of nodes in $A^{F_{m+1}}_{m+1} - X$ which are also nodes in V_i – that is, $H = (A^{F_{m+1}}_{m+1} - X) - \{\{S_{i'}\}^{(m+1)} : i \leq i'\}$. To prove that $A^{F_i}_i - X \subseteq U \cup H$, we need to show that there is no edge $ab \in F_i$ with $a \in Z^{(i)}_{j,m+1} - U^{(i)}_{j,m+1} \not\ni b$. This follows from the fact that there is no edge in F_{m+1} with a **JJ** label. Hence we have $A^{F_i}_i - X \subseteq U \cup H$, indeed. By definition, there is a function ω'_{m+1} on $A^{F_{m+1}}_{m+1} - X$ s.t. all edges in F_{m+1} are ω'_{m+1}-balanced. Let us define a function ω'_i by the value of ω''_i on nodes in U, and by the value of ω'_{m+1} on nodes in H. By definition 6.3, all edges in F_i are ω'_i-balanced.

Next we show that the value of F_i, X is equal to $|\mathcal{P}_i|$. By Lemma 6.2 we get

$$|A_i \cap U^{(i)}_{j,m+1}| - 1 = \left|U^{(i)}_{j,m+1}\right| - c\left(G_i[Z^{(i)}_{j,m+1}] - F_{ij}, U^{(i)}_{j,m+1}\right). \tag{11}$$

It is easy to see that $|A_i^{F_i} - X| \le |H| + \sum_j |U_{j,m+1}^{(i)}|$, and $|A_{m+1}^{F_{m+1}} - X| = |H| + q$. Let α denote the number of isolated nodes of $G_{m+1} - X - F_{m+1}$ which are in H. Clearly,

$$c_1(G_{m+1} - X - F_{m+1}, A_{m+1}^{F_{m+1}} - X)) = \alpha + q.$$

Any odd component of some $G_i[Z_{j,m+1}^{(i)}] - F_{ij}$ is also an odd component of $G_i - X - F_i$. Also, the α isolated nodes of H will be isolated nodes also in $G_i - X - F_i$. So

$$c(G_i - X - F_i, A_i^{F_i} - X) \ge \alpha + \sum_j c\left(G_i[Z_{j,m+1}^{(i)}] - F_{ij}, U_{j,m+1}^{(i)}\right).$$

From all these equations/inequalities we get

$$|A_i^{F_i} - X| - c(G_i - X - F_i, A_i^{F_i} - X) \le |H| - \alpha - q + |A_i \cap U| \tag{12}$$

and

$$|A_{m+1}^{F_{m+1}} - X| - c_1(G_{m+1} - X - F_{m+1}, A_{m+1}^{F_{m+1}} - X) = |H| - \alpha \tag{13}$$

Since $|A_i \cap U| - q = |A_i| - |A_{m+1}| = 2|\mathcal{P}_i| - 2|\mathcal{P}_{m+1}|$, finally we get

$$\begin{aligned} &|A_i^{F_i} - X| - c(G_i - X - F_i, A_i^{F_i} - X) \le \\ &\le (|A_{m+1}^{F_{m+1}} - X| - c_1(G_{m+1} - X - F_{m+1}, A_{m+1}^{F_{m+1}} - X)) + 2|\mathcal{P}_i| - 2|\mathcal{P}_{m+1}| \end{aligned} \tag{14}$$

hence we are done. □

We show that Claim 10.1 implies Lemma 7.1. To see this, suppose $i \ge 2$. So, (9) holds for $i, i+1, \cdots, m$, but not for $i-1$. Now, notice that $\{S_{i-1}\}$ is a terminal in G_i which is uncovered by $\mathcal{P}_i$. By Lemma 10.1, F_i, X is a verifying pair for $\mathcal{P}_i$. $\{S_{i-1}\}$ is exposed by $\mathcal{P}_i$, so by a slackness condition $\{S_{i-1}\}$ must be in an odd component of $G_i - X - F_i$. If $\{S_{i-1}\} \in H$, then clearly, (9) also holds for $i-1$. If $\{S_{i-1}\}$ is in one of $G_i[Z_{j,m+1}^{(i)}]$, then for some $i' \ge i$ $\{S_{i-1}\}^{(m+1)} = \{S_{i'}\}^{(m+1)}$. This also implies (9) for $i-1$, a contradiction.

11 Maintaining the dragon-decomposition

Definition 11.1. A **dragon-decomposition** is given by a forest $T \subseteq E$ which has the following properties. Let $A^T := A \cup V(T)$.

1. For each $a \in A$ there is a component T_a of the (forest) subgraph (A^T, T) such that $A \cap V(T_a) = \{a\}$. Moreover, $\{T_a : \text{for each } a \in A\}$ are all the components of (A^T, T).

2. Let $\omega^T : A^T \to \Omega$ be the (uniquely defined) function s.t. each edge in T is ω^T-balanced. Let F be the set of ω^T-balanced edges. Let $\mathcal{K}$ is the family of components of $G - F$.

3. $T/\mathcal{K}$ is a tree.

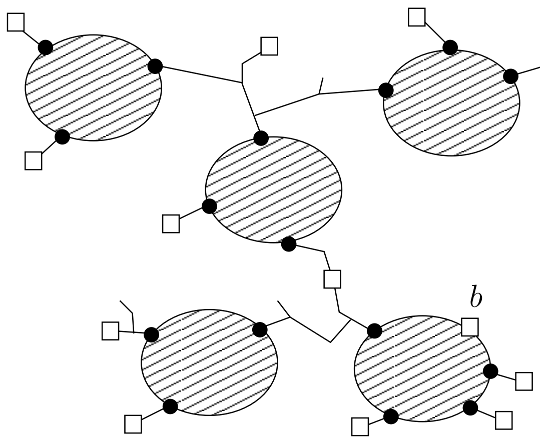

Figure 7: A dragon-decomposition.

4. $K, A^T \cap V(K), \omega^T, \pi$ is critical for every $K \in \mathcal{K}$.

Some remarks on the definition of dragon decompositions: The definition implies that $|A|$ is odd. In Figure 7, squares are the terminals, bold dots are so-called "borders" of components, the shaded areas are components. Edges in $F - T$ are not depicted. In trees T_a (joining components to each other and to the terminals) some nodes may collapse; however, to keep components disjoint, borders must have positive distance. For example in Figure 7, a collapsed with one of the borders in T_a, and so did b in T_b. Note that, T_a may be any kind of tree. However, by shrinking some of the edges in some T_a, in some of the proofs we may assume that T_a bounded by a and the borders in T_a, i.e. T_a has no pendant branches.

Firstly, let us show the main idea about dragon-decompositions, which is showing that this structure implies the dragon property. We will in fact be able to determine precisely which nodes are uniquely-/multiply reachable. These facts are shown in the following claim. It is easy to see that the assertions of Lemma 6.2 follow from this, if G is a dragon having a dragon-decomposition.

Claim 11.2. *A p-graph with a dragon-decomposition is a dragon. A^T is exactly the set of uniquely reachable nodes. A node $x \in A^T$ is $\omega^T(a)$-reachable by the unique half-A-path inside some T_a. $2\nu = |A| - 1 = |A^T| - c_{odd}(G - F, A^T)$.*

Proof. We begin with an easy but important observation, which is in fact the one and only tool to construct packings in a p-graph with a dragon-decomposition. Let $\mathcal{K} = \{K_1, K_2, \cdots, K_k\}$. Consider a component K_i, let us call nodes in $V(K_i) \cap V(T)$ "borders of K_i". Choose one of the borders, say $b \in V(K_i) \cap V(T) := \{b_1, \cdots, b_{2m+1}\}$. The following assertion is easy to see from Definition 11.1.

$$\text{There is a packing of } m \text{ non-returning paths inside } T \cup K_i \text{ disjoint from } b. \quad (15)$$

Notice that, a packing given by (15) only uses edges in K_i or edges in a component T_a of T s.t. $V(T_a)$ contains a border of K_i other than b. To prove Claim 11.2, we distinguish two cases:

Case A. $v \in V(T_a)$ for some $a \in A$. Let b_i denote the border of K_i which is "closest to v" (i.e. b_i is the end of the pre-image of the unique v–$\{K_i\}$ path in tree $T/\mathcal{K}$). Let $\mathcal{P}_i$ be a packing given by (15) applied for K_i, b_i. It is easy to see that $\mathcal{P} := \bigcup \mathcal{P}_i$ is a packing, and is disjoint from $V(T_a)$. Also notice that, $\mathcal{P}$ covers all terminals in $A - a$. So v is reachable, just use the unique a–v path in T_a.

Case B. Say $v \in V(K_1) - \bigcup V(T_a)$, so v is a non-border node of K_1. Suppose v is ω-reachable in K_1 (i.e. in the critical p-graph $K_1, A^T \cap V(K_1), \omega^T, \pi$). Next we show that v is also ω-reachable in the whole p-graph G. For this, we define $\mathcal{P}_i$ for $i \neq 1$ analogue to case A. It is easy to see that $\bigcup \mathcal{P}_i$ is a packing which is node-disjoint from K_1 and from the T_a's meeting K_1; moreover, $\bigcup \mathcal{P}_i$ covers all the terminals, except for those terminals a for which T_a meets K_1. We construct $\mathcal{P}$ by adding a packing "through K_1 and the trees T_a meeting K_1" s.t. there will also be a half-A-path ending in v with potential ω. This is easily constructed by using the unique paths in these T_a's linking K_1 to a.

So we have proved that G is a dragon, and that all the nodes in $V(K_1) - \bigcup V(T_a)$ are multiply reachable. To prove the remainder of of our claim we have to do the following considerations. Firstly, notice that F is A-balanced, and that the components K_i are exactly the odd components we count to determine c_{odd} in the min-max formula. Also notice that, $A^T = A^F$. So $|A^T| - c_{odd}(G-F, A^T)$ is a lower bound on 2ν (by "necessity" in the min-max formula). We show equality by using the construction given for case A, as follows. Let $\mathcal{P}$ be constructed for some $v \in V(T_a)$. For every K_i, $\mathcal{P}$ has the following property:

If K_i has $2r+1$ borders, then there are r paths in $\mathcal{P}$ which use two borders of K_i and each of these paths uses at most one border in any component other than K_i. (16)

So $|\mathcal{P}|$ must be equal to the number of all the borders minus k. Clearly, the set of borders is subset of A^T; and $k = c_{odd}(G - F, A^T)$.

We still need to show that the nodes in $\bigcup V(T_a)$ are uniquely reachable. This will be proved using equation $2|\mathcal{P}| = |A| - 1 = |A^T| - c_{odd}(G - F, A^T)$. Equality here implies "equality throughout", in other words, if $|\mathcal{P}| = (|A| - 1)/2$, then property (16) holds for every K_i. So a maximum packing $\mathcal{P}$ occupies at least all but one of the borders in any component. Thus a half-A-path in $G - V(\mathcal{P})$ either enters some $V(K_i)$ and never comes out of $V(K_i)$, or is disjoint from $\bigcup V(K_i)$. Hence, a node in $\bigcup V(T_a)$ is only reachable by half-A-paths on edges in F. As F is A-balanced, this implies that the nodes in $A^T = \bigcup V(T_a)$ are uniquely reachable. □

Notice that a dragon-decomposition – if there is one – must be more or less unique. F is unique, since it is determined as the set of edges xy s.t. x, y is uniquely reachable (say ω_x, ω_y-reachable) and $\pi(xy)(\omega_x) = \omega_y$. However, forest T is not defined uniquely, we only know that it is contained in F.

Lemma 11.3. *Dragons have a dragon decomposition.*

Our strategy to prove this lemma is the following. First we only prove that in our algorithm we are able to maintain a dragon-decomposition of dragons appearing in the sequence of contractions. In the end, Lemma 11.3 will follow by observing that a dragon has no other verifying pair than the trivial one. So we first prove the following weakening of Lemma 11.3.

Lemma 11.4. *Dragons in a sequence of contractions have a dragon decomposition.*

It is easy to see that Lemma 6.2 follows from Lemma 11.3 and Claim 11.2. In fact, to have our algorithm working, we in fact only need Lemma 11.4 and Claim 11.2. Lemma 11.3 will be proved in the end from Lemma 11.4 by showing that a dragon has no other decomposition than the trivial decomposition.

The proof of Lemma 11.4 goes by maintaining a dragon-decomposition through the following operations.

1. Given a dragon-decomposition of $G = (V, E)$, find the dragon-decomposition of $G' = G + ab = (V, E + ab)$ for some $a, b \in V$.

2. Given a dragon-decomposition of Z by a forest T. If G/Z is a rod, then find the decomposition of G.

3. Given a node-disjoint family $\mathcal{Z}$ of p-graphs which are subgraphs in a p-graph G. Suppose each member of $\mathcal{Z}$ is given together with a dragon-decomposition, and suppose $G/\mathcal{Z}$ is an odd cycle. Find a dragon-decomposition of G.

Operation 2 is easy to solve. Suppose G is built up by adding an edge ab to Z such that b is a new node. If T gives a dragon-decomposition of Z, then T or $T + ab$ also gives a dragon-decomposition of G, just check the definition.

Let us call a dragon decomposition given by a forest T a **pure decomposition** if $F = T$. So, for a pure decomposition property 3 means that $G/\mathcal{K}$ is a tree. A dragon does not necessarily have a pure decomposition! The proof of the following lemma will be given below.

Lemma 11.5. *Given a node-disjoint family $\mathcal{Z}$ of p-graphs which are subgraphs in a p-graph G. Suppose each member of $\mathcal{Z}$ is given together with a pure decomposition, and suppose $G/\mathcal{Z}$ is an odd cycle. Then one can construct a pure decomposition of G.*

Let us first see how to use this lemma to solve an operation of type 3 in general. We first omit some edges of G to get some family $\mathcal{Z}$ of purely decomposables. Lemma 11.5 produces a decomposition, then, in the end we put back the omitted edges. These are operations of type 1. Notice, however, that most operations of type 1 are in fact special cases of type 3, which will be solved by Lemma 11.5, too. So only those operations of type 1 remain which are not special cases of type 3. Up to this point we have a pure decomposition (by some forest T) of some spanning subgraph (V, E') for some $E' \subseteq E$. Now we have to put back a set $E - E'$ of omitted edges such that none of them maps to a non-returning loop (=odd cycle) in $G/(V, E')$. We claim that T also gives a decomposition of G, since we get $F = T \cup (E - E')$. This follows from the definition and Claim 11.2.

Proof of Lemma 11.5. G is built up by putting members of $\mathcal{Z}$ cyclically next to each other, and joining them by the pre-images of the odd cycle. So each $Z \in \mathcal{Z}$ will be incident with two such edges. Let us introduce some more notation for this. Let $\mathcal{Z} = \{Z_1, \cdots, Z_{2k+1}\}$, and let the pre-image of the odd cycle be $\{a_i b_{i+1} : i = 1, \cdots, 2k+1\}$ (addition mod $2k+1$), where $a_i, b_i \in Z_i$. Recall that, by definition of an odd cycle and of a contraction, for any i, a_i, b_{i+1} are reachable inside Z_i, resp. Z_{i+1} such that edge $a_i b_{i+1}$ produces a non-returning A-path.

To simplify the proof we will make one more assumption – besides assuming to have pure decompositions for the Z_i's, say given by forest T^i, with compoennts $\mathcal{K}^i$. Notice that one can build up a pure decomposition by growing a sub-forest of $T^i/\mathcal{K}^i$. So in the first place we assume that $T^i/\mathcal{K}^i$ is "path-like", and a_i, b_i are in the most distant parts of this path structure. Then, in the end, we can glue back the pendant branches – in the next paragraph we will be more precise in how to do this.

Suppose we have a pure decomposition of Z given by forest T, with components $\mathcal{K}$. Suppose some $K \in \mathcal{K}$ is a "leaf" – that is one of its borders x is a cutting node between K and $\mathcal{K} - K$. Say K is incident with trees $T_{a_1}, \cdots, T_{a_{2m+1}}$ with $x \in T_{a_{2m+1}}$. Then it is easy to see that $T' := T - \bigcup_{i=1}^{2m} T_{a_i}$ purely decomposes $Z' := Z - K - \bigcup_{i=1}^{2m} T_{a_i}$. Moreover, it is easy to see that Z/Z' is a dragon with a pure decomposition $\bigcup_{i=1}^{2m} T_{a_i}$. So a pure decomposition can be dismantled by cutting off one of its "leafs", or, by induction, cutting off some of its "branches". In the other direction, if some Z' has a pure decomposition, and Z/Z' has a pure decomposition, and the pre-image of $E(Z/Z')$ has exactly one (cutting) node in common with $V(Z')$, then the union of these two pure decompositions gives a pure decomposition of Z.

By the above reasoning, we may assume without loss of generality that each Z_i has the following "path-like" structure, with a_i, b_i being on the opposite ends of it. (We may assume this since "pendant braches" we cut off, and glue back in the end after a pure decomposition of G is found.) We may also assume – by shrinking some edges in the forest – that the forest only has path-components and possibly some "Y"-components, see Figure 8.

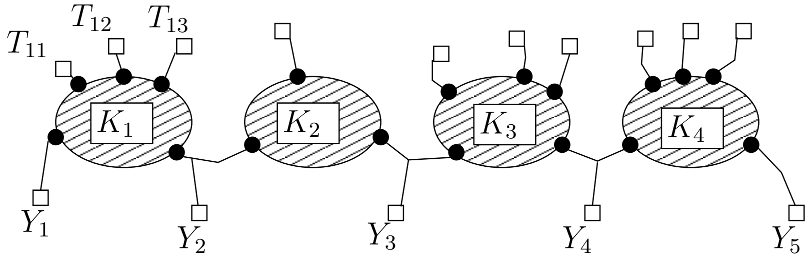

Figure 8: A pure decomposition with a "path-like structure".

In a path-like pure decomposition, let K_j $(j = 1, \cdots, l)$ be the compoents in the order of appearance, let Y_j $(j = 1, \cdots, l+1)$ be the tree before K_j or after K_{j-1}, and let T_{jq} be the paths incident with K_j (clearly there is an odd number of T_{jq}'s for a fixed j.) To interpret Figure 8, the shaded parts are the components K_j the inner structure of which is not in dicussion here – we only have property 3, i.e. the criticality with respect to some induced potential on the border. The bold dots denote the border of components in the figure, the squares denote the terminals. Our definition allows that some of the paths depicted in the figure vanish as zero-length paths. However the path on the tree Y_{j+1} joining K_j and K_{j+1} may not vanish – this would result in glueing K_j and K_{j+1} to form one bigger component. It is important that the bold dots (the borders) are all pairwise distinct nodes, and any component has an odd number of borders.

So, our assumption (besides having a pure decomposition) is that Z_i has a path-like decomposition with notation $\mathcal{K}^i, K^i_j, Y^i_j, T^i_{jq}$, moreover, $a_i \in K^i_1 \cup Y^i_1$ and $b_i \in K^i_l \cup Y^i_{l+1}$. So G has a cyclic structure, Figure 9 shows an example. This structure is built up by putting "path-like" decompositions cyclically next to each other, and joining their leafs $K^i_1 \cup Y^i_1$ and $K^{i+1}_l \cup Y^{i+1}_{l+1}$ by some edge $a_i b_{i+i}$. These edges $a_i b_{i+i}$ are drawn as dotted lines.

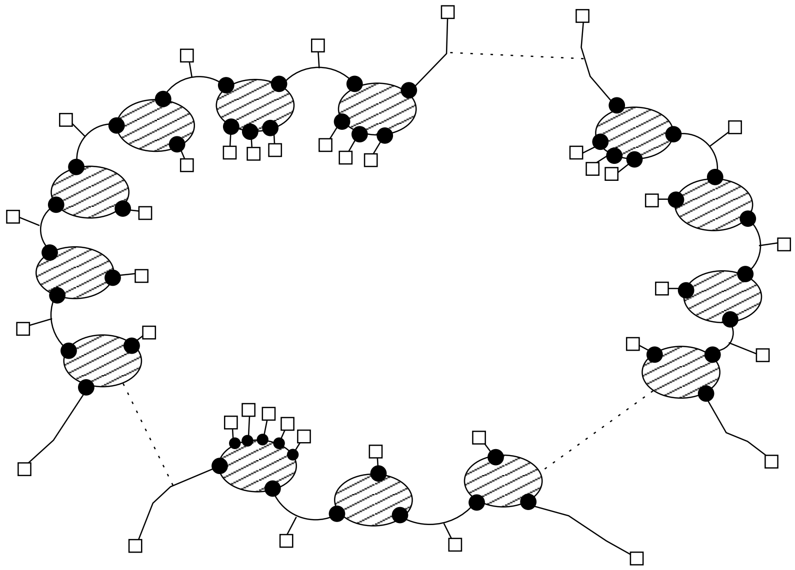

Figure 9: The cyclic structure.

It is easy to see that $G/\bigcup \mathcal{K}^i$ (shrinking the shaded parts) is a graph with a unique cycle Q with an odd number of pairwise node-disjoint pendant paths attached to Q. Let K denote the pre-image of Q. Clearly, K contains all the K^i_j's, and all the edges $a_i b_{i+1}$. Let $T := E - E(K)$, which is equal to the the union of the pre-image of Q's pendant paths.

Claim 11.6. *T determines a pure decomposition of G.*

Proof. Properties 1 and 3 are straightforeward, property 2 is just a definition. To prove property 4 we need to show that if $T = \emptyset$, then G is critical. (This case is exactly the case when pendant paths on Q have zero length.) Since Z_i are dragons by themselves it follows that all nodes in G are reachable. We need to show that nodes in $V - A$ are multiply reachable. Since the K^i_j's are critical by themselves – a pure decomposition is given inside the Z_i's – multiple reachability for the non-border nodes of K^i_j's also follows. We are only concerned with the non-terminal nodes in the paths in G joining two neighboring K^i_j's.

By shrinking, we may assume that the cyclic structure of K is built up as follows – demonstrated in Figure 11.

1. Put critical K^i_j's cyclically next to each other.
2. Join two neighboring K^i_j's by one of the ways in Figure 10 using altogether an odd number of dashed lines (the a_ib_{i+1}'s).

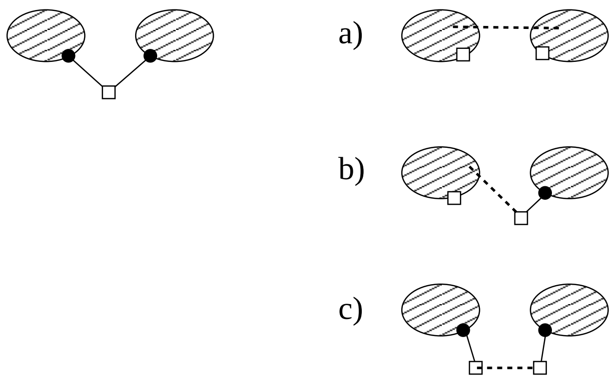

Figure 10: The possible paths connecting neighboring K^i_j's.

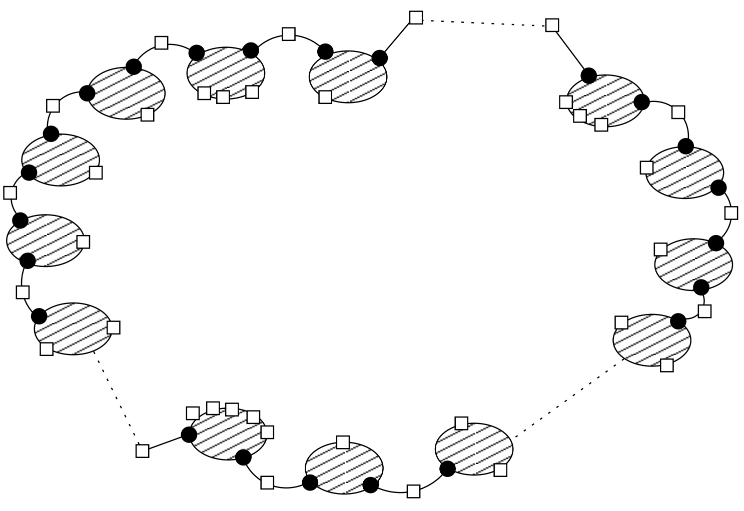

Figure 11: K

To prove that K is critical, first we prove the following claim.

Claim 11.7. *Given an arbitrary terminal $a \notin \bigcup K^i_j$. Then there is a family of $\nu + 1$ non-returning paths such that they are pairwise node-disjoint, except for node a, where there start 2 paths.*

(Claim 11.7 is equivalent to: increasing the capacity of such a node a from 1 to 2 increases ν.)

Proof. Such a terminal a has degree 2 in G. Let us split a into 2 copies a' and a'' such that the two incident edges are split 1-1 among them. So G' has one more terminal, $G'/\bigcup K_j^i$ is an a'–a'' path on an even number of nodes, and the pre-images of nodes in $G'/\bigcup K_j^i$ are dragons. Consider the perfect matching M in $G'/\bigcup K_j^i$. By Claim 6.4, there is a $\mathcal{P}'$ packing in G' such that $\mathcal{P}'/\bigcup K_j^i = M$. The pre-image of $\mathcal{P}'$ is a family of non-returning paths in G as required. □

To prove that K is critical we need to prove that the bold dots are multiply reachable. Consider the bold dot z, i.e. a border which is not a terminal. We have already shown that z is reachable by the unique a–z path disjoint from $\bigcup K_j^i$. Then a is a terminal not in $\bigcup K_j^i$. Consider a family $\mathcal{P}$ as of Claim 11.7, say with paths $P_1, P_2 \in \mathcal{P}$ containing a. Then one of them, say P_1 contains the (unique) a–z path. Let P_1' denote the other section of P_1 ending in z. Then – since P_1 was non-returning – P_1' reaches z with another different potential. Also P_1' is disjoint from the maximum packing $\mathcal{P} - P_1$. Thus z is multiply reachable, hence K is critical. This proves Claim 11.6, and concludes the proof of Lemma 11.4. □

Proof of Lemma 11.3. Consider a dragon G, run the algorithm. We find a verifying pair X, F. In any maximum packing, nodes in X must be traversed by some path. So $X = \emptyset$.

References

[1] M. Chudnovsky, J.F. Geelen, B. Gerards, L. Goddyn, M. Lohman, P. Seymour, *Packing non-zero A-paths in group-labeled graphs*, submitted

[2] M. Chudnovsky, W.H. Cunningham, J.F. Geelen, *An algorithm for packing non-zero A-paths in group-labeled graphs*, manuscript

[3] J. Edmonds, *Path, trees, and flowers*, Canadian Journal of Mathematics 17 (1965) 449-467.

[4] T. Gallai, *Maximum-minimum Sätze und verallgemeinerte Faktoren von Graphen*, Acta Mathematica Academiae Scientarim Hungaricae 12 (1961) 131-137.

[5] L. Lovász, *Matroid matching and some applications*, Journal of Combinatorial Theory Ser. B 28 (1980) 208-236.

[6] W. Mader, *Über die Maximalzahl kreuzungsfreier H-Wege*, Archiv der Mathematik (Basel) 31 (1978) 387-402.

[7] G. Pap, *Packing non-returning A-paths*, EGRES TR-2005-15, submitted

[8] A. Schrijver, *A short proof of Mader's $\mathcal{S}$-paths theorem*, Journal of Combinatorial Theory Ser. B 85 (2001) 319-321.

[9] A. Sebő, L. Szegő, *The path-packing structure of graphs*, in: Integer Programming and Combinatorial Optimization (Proceedings of 10th IPCO Conference, New York, 2004; D. Bienstock, G. Nemhauser, eds.) Lecture Notes in Computer Science 3064, Springer-Verlag Berlin Heidelberg (2004) 256-270.