Transversal Problems In Sparse Graphs

by

Hao Sun

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2022

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Adrian Vetta
Professor, Dept. of Mathematics and Statistics, McGill University

Supervisor: Jochen Koenemann
Professor, Dept. of Combinatorics & Optimization,
University of Waterloo

Internal Members: Joseph Cheriyan
Professor, Dept. of Combinatorics & Optimization,
University of Waterloo

Chaitanya Swamy
Professor, Dept. of Combinatorics & Optimization,
University of Waterloo

Internal-External Member: Therese Biedl
Professor, David R. Cheriton School of Computer Science,
University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Chapters 3 and 4 are based on the paper [30] that I coauthored with Alexander Göke, Jochen Koenemann, and Matthias Mnich. Chapter 5 is based on research I worked on with the same authors. Chapter 6 is based on my own work [56].

Abstract

Graph transversals are a classical branch of graph algorithms. In such a problem, one seeks a minimum-weight subset of nodes in a node-weighted graph G which intersects all copies of subgraphs F from a fixed family $\mathcal{F}$.

In the first portion of this thesis we show two results related to even cycle transversal. In Chapter 4, we present our 47/7-approximation for even cycle transversal. To do this, we first apply a graph "compression" method of Fiorini et al. which we describe in Chapter 2. For the analysis we repurpose the theory behind the 18/7-approximation for "uncrossable" feedback vertex set problems of Berman and Yaroslavtsev noting that we do not need the larger "witness" cycles to be a cycle that we need to hit. This we do in Chapter 3.

In Chapter 5 we present a simple proof of an Erdos Posa result, that for any natural number k a planar graph G either contains k vertex disjoint even cycles, or a set X of at most $9k$ such that $G\backslash X$ contains no even cycle.

In the rest of this thesis, we show a result for dominating set. A dominating set S in a graph is a set of vertices such that each node is in S or adjacent to S. In Chapter 6 we present a primal-dual $(a+1)$-approximation for minimum weight dominating set in graphs of arboricity a. At the end we propose five open problems for future research.

Acknowledgements

First and foremost, I would like to express my heartfelt thanks to my supervisor Professor Jochen Koenemann for his invaluable guidance throughout my PhD program. His energy, expertise and encouragement resulted in a very enlightening experience for me. He not only encouraged me to attend a variety of international conferences and workshops, but also brought me into research collaborations with other scholars such as Dr. Matthias Mnich. His guidance, through weekly meetings over the years and innumerable back-and-forth discussions of ideas and proofs, gave me a lot of research experience and gradually trained me to become a scholar. I especially want to thank Prof Koenemann who paved the way forward for me to pursue my academic research in the future.

I would also like to thank my master's thesis supervisors Professors William Cook and Henry Wolkowicz. Prof Wolkowicz has introduced me to the world of mathematical research and given me a lot guidance since my days as his Undergraduate Research Assistant. He inspired my interest and broadened my horizons in optimization and with him I had my first collaborated paper published on a mathematical journal. Prof Cook introduced me to and cultivated my interest in the interplay between combinatorics, formulations and optimization and has been a valuable guide in the world of academia. Also thanks to Prof Jim Geelen for his inspiring class and helping develop one of my inadvertent ideas to a published paper as a new proof of Rado's Theorem.

Throughout my student life at the Faculty of Mathematics, I would like to thank many professors and instructors who have taught me over the years, especially those in the C&O department their passion and enthusiasm seems to have rubbed off on me.

Special thanks to my thesis Examining Committee: to Prof. Adrian Vetta for his valuable feedback in my defence; to Prof. Therese Biedl for her careful checkup of many proofs, arguments, diagrams, and comments on writing styles; to Prof. Joseph Cheriyan and Prof. Chaitanya Swamy for providing their constructive suggestions from my research proposal to my thesis defence, all of which help improve the quality of my thesis.

Finally, I would like to thank the Faculty of Mathematics and the Department of Combinatorics and Optimization for providing an academic environment for many enlightening mathematical discussions and for providing my financial supports in a variety of scholarships throughout my graduate studies.

Table of Contents

List of Figures

Chapter 1

Introduction

Transversal problems in graphs have received a significant amount of attention from the perspective of algorithm design [6, 9, 20, 41]. Node (resp. edge) transversal problems take as input a node-weighted (resp. edge-weighted) graph G, and seek a minimum-weight subset S of nodes (resp. edges) which intersect all graphs F from a fixed graph family $\mathcal{F}$ that appears as subgraph in G. A set $X \subset V(G)$ is an *$\mathcal{F}$-transversal* if $G \backslash X$ contains no subgraph isomorphic to a graph of $\mathcal{F}$. A major part of this thesis is about the $\mathcal{F}$-transversal problem where $\mathcal{F}$ is defined to be the set of all even cycles of our graph.

Let us give an overview of the literature on $\mathcal{F}$-transversal problems. Many prominent $\mathcal{F}$-transversal problems are at the same time generalized *Feedback Vertex Set* (FVS) problems. In a generalized FVS problem (see [8]), we are given a (possibly directed) graph $G = (V, E)$, weights $w_v \ \forall v \in V$, a set of cycles $\mathcal{C} \subset \{0,1\}^E$, and wish to choose a set of vertices F such that each $C \in \mathcal{C}$ is hit by F, that is, F contains a vertex of C. The cases of *undirected FVS* and *directed FVS (DFVS)* are when $\mathcal{C}$ is the set of all cycles, or dicycles of a directed graph, respectively. The cases of *subset FVS* (SFVS), *odd cycle transversal* (OCT), and *even cycle transversal* (ECT) are given by setting $\mathcal{C}$ equal to the set of cycles going through a given set of "special" vertices, the set of cycles of odd length and the set of cycles of even length.

Many results are known for FVS and its variants (the ones for our main problem ECT will be reviewed in Section 1.1). FVS is one of Karp's 21 NP-complete problems [36]. FVS has many applications such as deadlock prevention, node weighted network design and minimum feasible subsystem problem (MinFs2) [8, 22, 42]. MinFs2 is the problem of removing a minimum weight set of equations from an infeasible system to obtain feasibility.

For $\alpha \in \mathbb{R}$, we say that an algorithm is an α-approximation algorithm, if it is guaranteed to return a solution at most α times the optimum. Undirected FVS admits a 2-approximation in polynomial-time [2, 6], which cannot be improved to a $(2-\varepsilon)$-approximation for any $\varepsilon > 0$ assuming the Unique Games Conjecture holds [38]. We say that a problem admits a polynomial-time approximation scheme (PTAS) if, for any fixed $\epsilon > 0$, there is a $(1+\epsilon)$-approximation algorithm running in polynomial-time. If such an algorithm runs in time $n^{O(1)} f(1/\epsilon)$ for some function f where n is the problem size, we say that the problem admits an efficient PTAS (EPTAS). Many unweighted graph transversal problems have been shown to admit PTASs for *planar* input graphs G, using a variety of techniques like the shifting technique (Baker, J. [3]), bidimensionality (Fomin et al. [26]), or connectivity domination (Cohen-Addad et al. [10]). Informally speaking, the techniques in [3, 26] employ a "divide and conquer" approach which first looks for a set of vertices called a "separator" whose size is small compared to the size of an optimal solution, that is $o(OPT)$ where OPT is the size of an optimal solution, and whose removal divides the graph into pieces for which the problem is easy. These techniques do not extend to weighted graphs as [3, 26] do not show that weighted graphs contain a separator of small weight. A key advantage of the technique of [10] is that it can be applied to weighted graphs.

A 2-approximation for (undirected) FVS is given in [2]. Undirected FVS with unit weights admits an efficient polynomial-time approximation scheme (EPTAS) in H-minor free graphs for any fixed graph H via the divide and conquer technique mentioned previously [26, 41]. Undirected (weighted) FVS also admits an EPTAS on graphs of bounded genus [10]. In fact it is shown in [41], that local search yields a PTAS in H-minor free graphs. To be precise, given any $\epsilon > 0$, there exists some positive integer c for which the following algorithm is a $(1 + \epsilon)$-approximation. Given a graph $G = (V, E)$, positive integer c, initialize $S := V$. For each $A \subset S$ of size at most c, $B \subset V$ of size at most $|A| - 1$, if $(S\backslash A) \cup B$ is a feasible FVS replace S with $(S\backslash A) \cup B$. If for each $A \subset S$ of size at most c, $B \subset V$ of size at most $|A| - 1$, we have that $(S\backslash A) \cup B$ is not a feasible FVS return S.

A tournament is a complete directed graph. A bipartite tournament is a directed complete bipartite graph. DFVS has a 2-approximation in tournaments [45] and bipartite tournaments [59], is polynomial-time solvable on graphs of bounded treewidth, and has an $O(\log n \log \log n)$-approximation in general graphs [20]. DFVS cannot be approximated within $2 - \epsilon$ for any $\epsilon > 0$ assuming the unique games conjecture holds [38]. SFVS has an 8-approximation [21] and also admits an 11-approximation based on Linear Program (LP) rounding [9]. A characterization of digraphs for which the natural linear program (see ($\mathrm{P}_{\mathrm{ECT}}$)) has integral primal and dual solutions in the unit weight case is given in [31]. A characterization of the feedback vertex set polytope in series-parallel graphs is given in [25].

We say that a vertex deletion problem Π has a kernel of size f (where $f : \mathbb{N} \to \mathbb{N}$), if the following holds. For any instance L of Π and $k \in \mathbb{N}$ we can in polynomial-time find a problem L' of Π of "size" at most $f(k)$ and an integer $k' \leq f(k)$ such that L has a solution which deletes k or fewer vertices if and only if L' has a solution which deletes k' or fewer vertices. ECT, OCT, FVS, DFVS and SFVS admit $O(k^2), O(k^{4.5}), O(k^2), O(k^4)$and $O(k^9)$, kernels [7, 33, 39, 49, 54, 57], respectively.

Feedback edge set (FES) problems, that is, the problem of given a graph $G = (V, E)$ weights w_e $\forall e \in E$, a set of cycles $\mathcal{C} \subset \{0, 1\}^E$, and finding a minimum weight set of edges that intersects all cycles, can be easier than their FVS counterparts. Undirected FES is equivalent to the maximum weight forest problem, which is polynomial-time solvable [40]. Thus, undirected FES is solvable in polynomial-time. It was shown in [48] that the natural LP for directed FES (DFES) is totally dual integral in planar graphs. Directed FES (DFES) also has a polynomial-time approximation scheme (PTAS) in tournaments [37].

1.1 Approximating Even Cycle Transversal

Recall that two variants of FVS are *Odd Cycle Transversal* (OCT) and *Even Cycle Transversal* (ECT), where one wishes to intersect the odd-length and even-length cycles of the input graph G, respectively. These fall in the category of graph problems under parity constraints, a topic that has been studied for graph transversal problems [46, 47, 49, 51].

Since FVS is NP-hard it is easy to see that ECT is NP-hard. OCT is known to be NP-hard [61]. The approximability of these problems is much less understood than that of FVS: for OCT, only an $\mathcal{O}(\sqrt{\log n})$-approximation is known [1], which requires unit weights, and for ECT, only a 10-approximation is known [49]. It is NP-hard to approximate OCT to $1.3606 - \epsilon$ for any $\epsilon > 0$ [16].

Planar graphs are a natural subclass of graphs in which to consider graph transversal problems. We provide a quick proof that ECT is hard in planar graphs for completeness. Lichtenstein [44] showed that vertex cover is NP-hard in planar graphs. Given an instance of vertex cover on a planar graph G, for each edge uv add another parallel edge between u and v. Call this new graph G'. Given $S \subset V(G)$, if S is a vertex cover of G, then given any cycle of G' let uv be an edge of G', then uv is an edge of G and S contains at least one vertex in u, v. If S is an ECT of G', then note that for each $uv \in G$, G' contains an even cycle consisting of two parallel edges between u and v and hence S contains one of u, v. So S is a vertex cover. This reduction thus shows that ECT is NP-hard in planar graphs.

For many NP-hard problems, there exist good approximations in planar graphs. One of the first results here was Baker's shifting technique [3], which yielded a PTAS for Vertex Cover in planar graphs (which is an $\mathcal{F}$-transversal problem where $\mathcal{F}$ is the single graph consisting of an edge). The technique was generalized by Demaine et al. [11] who gave EPTASs for graph transversal problems satisfying a certain bidimensionality criterion, including FVS in *unweighted* planar graphs. That result was later extended to yield an EPTAS for FVS in unweighted H-minor free graphs [27], for any fixed graph H. In the edge-weighted *Steiner Forest* problem, we are given a graph G and a list of pairs of nodes of G and wish to find a minimum weight set of edges that connect all pairs of nodes in our list. In *edge-weighted* planar graphs, PTAS are known for edge-weighted Steiner Forest and OCT [5, 18, 32]. One naturally hopes that better approximations exist for ECT in planar graphs than general graphs.

On *node-weighted* planar graphs, the situation appears to be more complex. First, the existence of a PTAS for FVS on node-weighted planar graphs was a long-standing open question which was resolved only recently in a paper of Cohen-Addad et al. [10]. The authors presented a PTAS for FVS in node-weighted planar graphs, crucially exploiting the fact that the treewidth of $G - S$ is bounded for feasible solutions S. The existence of an EPTAS for FVS in node-weighted planar graphs is still open.

Goemans and Williamson [29] first proposed a primal-dual based framework for FVS problems where the cycle family $\mathcal{F}$ satisfies a certain uncrossing property. In an embedded planar graph, two cycles C_1, C_2 *cross* if C_1 contains an edge intersecting the interior of the region bounded by C_2 and C_2 contains an edge intersecting the interior of the region bounded by C_1; see Figure 1.1. Informally, a set $\mathcal{S}$ of cycles is *uncrossable* if for any two crossing cycles C_1, C_2 of $\mathcal{S}$ that cross one can find two cycles $C_1', C_2' \in \mathcal{S}$ that do not cross and use the same set of vertices. The latter property can be seen to be satisfied by OCT, *Directed FVS* in directed planar graphs, and *Subset FVS*, which seeks a minimum-cost node set hitting all cycles containing a node from a given node set T. For those problems, the authors obtained 3-approximations[1]. The framework by Berman and Yaroslavtsev [29] also yields a 3-approximation for *S*teiner Forest in node-weighted planar graphs [15, 50]. Berman and Yaroslavtsev [8] later improved the approximation factor for the same class of uncrossable cycle transversal problems from 3 to 2.4.

The main question driving this work is whether the framework of [8, 29] can be extended to cycle transversal problems that do not satisfy uncrossability. We focus on ECT in node-weighted planar graphs as a natural such problem: even cycles are not uncrossable, and hence the frameworks of [8, 29] do not apply.

Figure 1.1 gives an example of a graph whose even cycles are not uncrossable: the two even cycles depicted in red and black edges cross, and they are the only two even cycles in the graph. One cannot find two distinct even cycles that do not cross in that graph, which shows that even cycles are not uncrossable. Furthermore, the framework of Cohen-Addad et al. [10] requires that contracting edges only reduces the solution value, which is not the case for even cycles either. For example, the graph G consisting of a single odd cycle has the empty set as an ECT, while G/e for any $e \in G$ is an even cycle and does not contain the empty set as an ECT. Our main result, which we prove in Chapter 4 is a 47/7-approximation algorithm for ECT in node-weighted planar graphs.

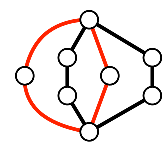


Figure 1.1: Here the red cycle crosses the black cycle.

Theorem 1. *[30] ECT admits an polynomial-time* $47/7 \approx 6.71$*-approximation algorithm on node-weighted planar graphs.*

This improves the previously best 10-approximation by Fiorini et al. [24] for planar graphs.

1.2 Even Cycles Satisfy Erdős-Pósa in Planar Graphs

For a graph G, a set $\{F_1, \ldots, F_\ell\}$ of subgraphs of G is called an $\mathcal{F}$*-packing* if the F_i are pairwise vertex-disjoint and isomorphic to graphs in $\mathcal{F}$. Clearly, the maximum size of an $\mathcal{H}$

[1] 18/7-approximations were claimed but later found to be incorrect [8].

packing is no greater than the minimum size of an $\mathcal{H}$-transversal. A question of fundamental interest is how large the gap between these two quantities can be. Erdős and Pósa proved that there is a function $f(k)$ such that for every graph G and every $k \in \mathbb{N}$, either G contains k vertex-disjoint cycles, or there is a set X of at most $f(k)$ vertices such that $G \backslash X$ is a forest.

Definition 1. *We say that the* Erdős-Pósa *property holds for a set of graphs $\mathcal{H}$ in a class of graphs $\mathcal{G}$ if there is a function $f(k)$ (called the bounding function) such that any graph of $\mathcal{G}$ either contains k vertex disjoint copies of graphs in $\mathcal{H}$ or an $\mathcal{H}$-transversal $X \subset V$, $|X| \leq f(k)$.*

Recently van Batenburg et al. [58] showed that for any planar graph H any graph G either contains k vertex disjoint subgraphs each of which contain H as a minor, or a set X of at most $f(k) \in O(k \log k)$ vertices such that $G \backslash X$ does not contain H as a minor. As a corollary they show that even cycles satisfy the Erdős-Pósa property with $f(k) \in O(k \log k)$ and $f(k) \in O_H(k)$ for H-minor free graphs. By $O_H(k)$ we mean $f \in O(k)$ for fixed H. Going through the details of their proof one finds their result shows that one can take $f(k) = 145k$.

Odd cycles do not satisfy the Erdős-Pósa property: Reed [55] showed that "Escher walls" are graphs which do not contain two node-disjoint odd cycles, but for which a minimum odd cycle transversal can be arbitrarily large. Given a graph G, T an even set of vertices of G, a set of edges J is a *T-join* if each vertex of T is incident to an odd number of edges in J and each vertex of G not in T is incident to an even number of edges of J. The minimum T-join problem is the problem of finding a minimum weight T-join of a graph with weights on its edges. Fiorini et al. [23] show that odd cycles satisfy the Erdős-Pósa property in planar graphs: any planar graph G either contains k node-disjoint odd cycles, or a set X of at most $10k$ vertices such that $G - X$ has no odd cycles. Informally speaking, they achieve this result by relating odd cycle transversals to the T-join problem in the dual graph of the planar graph G. To illustrate their ideas, consider the problem of removing a set of edges X from our planar graph such that $G \backslash X$ contains no odd cycle. Suppose that every face of G is a cycle. Let T be the set of odd faces of the dual graph G^*. For a cycle C of G, one can show that C is an odd cycle of G if and only if an odd number of faces in T lie in the region bounded by C. Cycles in planar graphs are cuts of the dual graph and one can see that C is a T-cut of G^*, that is, a cut for which the number of nodes in T on either side of the cut is odd. They then use the theory of T-cuts and T-joins to obtain their result.

In Chapter 5, we give a simple proof that the Erdős-Pósa property holds for the set of even cycles in the class of planar graphs with bounding function $f(k) = 9k$ via a primal-dual algorithm.

Theorem 2. *For $k \in \mathbb{N}$, a planar graph either has as set of at most $9k$ vertices that intersect every even cycle in G, or a set of k vertex disjoint even cycles.*

1.3 Approximation Algorithm for Dominating Set

In the Minimum Weighted Dominating Set (MWDS) problem, we are given a graph $G = (V, E)$ with weights $w_v, \forall v \in V$, and wish to find a minimum weight set D of vertices for which each vertex $v \in V$ is either in D, or has a neighbour in D. When all weights are 1 we call this the minimum dominating set (MDS) problem. One can see that MWDS is a special case of weighted set cover. Hence, by applying the greedy algorithm for weighted set cover, one can obtain a H_n approximation for MWDS, where $n := |V|$ and H_n is the n-th harmonic number. Bansal and Umboh [4] made the observation that it is NP-hard to approximate MDS to within $(1 - \epsilon) \ln n$ by reducing it to set cover and using a hardness result for set cover proven in [17].

Baker [3] showed that MDS in planar graphs admits a PTAS. Fomin et. al. [26] extended this to an EPTAS on H minor free graphs for any fixed graph H.

Given that planar graphs are sparse, it may seem natural to generalize the previous results on MDS to sparse graphs. Lenzen and Wattenhofer [43] observed that MDS remains hard on graphs of low average degree and unit weights by the following reduction. Given any graph on n nodes, add a star on $n^2 - n$ nodes. The resulting graph G' has average degree at most 2 and MDS one more than that of G. Hence, approximating the minimum dominating set in H is as hard as approximating the minimum dominating set in G.

Lenzen and Wattenhofer [43] proposed studying MDS on a class of graphs that informally speaking have a local sparsity property. A graph has *arboricity* a if a is the minimum number of edge-disjoint forests into which its edges can be partitioned. It is well known that a graph has arboricity a, if and only if each subgraph induced by a nonempty subset of vertices $S \subset V$ has at most $a(|S| - 1)$ edges [52]. In this sense, bounded arboricity is equivalent to local sparsity. We use a-MWDS and a-MDS to refer to MWDS and MDS in graphs of arboricity a.

Lenzen and Wattenhofer presented a distributed $\mathcal{O}(a^2)$ approximation algorithm for a-MDS. Bansal and Umboh [4] improved this by giving a $3a$-approximation for a-MDS by rounding the natural LP relaxation. They also show that it is NP-hard to approximate a-MDS to within $a - 1 - \epsilon$ for any $\epsilon > 0$. Dvořak [19] showed that the algorithm of Bansal and Umboh [4] actually gives a $(2a + 1)$-approximation for a-MDS. We present an $(a + 1)$-approximation algorithm for a-MWDS using the primal-dual method. We also show in Theorem 12 that the algorithm of Bansal and Umboh is no better than a $(2a - 1)$-approximation in the worst case. Our primal-dual algorithm, which has a combinatorial flavour in the sense that it produces a “fractional packing”, beats the direct LP rounding of Bansal and Umboh [4] in the worst case.

In Chapter 6, we show the following result for dominating set:

Theorem 3. *[56] There is a polynomial-time $(a+1)$-approximation algorithm for a-MWDS.*

Our analysis actually requires a slightly weaker condition than arboricity a, namely that for our graph G, $|E(G[S])| \leq a|S| \quad \forall S \subset V(G)$, in other words, our graph has *maximum* *average degree* at most a. Informally speaking, we show that the dual variables of nodes only "pay" towards themselves and their neighbours. We use the sparsity of our graph, that is, nodes of our graph have at most $2a$ neighbours on average to bound how much an average dual variable pays. We also use the concept of a "witness" [29], which is a node that only pays for one other node. Together with sparsity, this allows us to derive an $(a+1)$-approximation.

Chapter 2

Preliminaries

In the even cycle transversal (ECT) problem, we are given a graph $G = (V, E)$, costs c_v for all $v \in V$, and we wish to find a minimum cost set of nodes that intersects every even cycle of G. Throughout this thesis, V will be the set of vertices $V(G)$ of a graph G. Frequently G will be a *planar graph* i.e. it can be drawn without crossing.

Given a matrix A, denote by $A_{i,:}$ the ith row of A and $A_{:,j}$ the jth column. Given a graph $G = (V, E)$ and $v \in V$, denote by $N_G(v)$ the neighbours of v in G. If it's clear what graph we are talking about, we will use $N(v)$ instead of $N_G(v)$.

Let V be a set, $c \in \mathbb{R}^V$ costs on V, $\mathcal{C} \subset 2^V$ a set of subsets of V. The $\mathcal{C}$ *hitting set problem* is to find the minimum weight set $S \subset V$ that hits $\mathcal{C}$, that is, $C \cap S \neq \emptyset \ \ \forall C \in \mathcal{C}$. We call nodes of S *hit nodes*. Let $A \in \mathbb{R}^{\mathcal{C} \times V}$ where $A_{C,v} = 1$ if $v \in C$ and 0 otherwise. Consider a linear program ($\mathrm{P}_{\mathcal{C}}$) and its dual ($\mathrm{D}_{\mathcal{C}}$).

$$\begin{aligned} \min \quad & c^T x && (\mathrm{P}_{\mathcal{C}}) \\ \text{s.t.} \quad & Ax \geq \mathbb{1} && (2.1) \\ & x \geq \mathbb{0} \end{aligned}$$

$$\begin{aligned} \max \quad & \mathbb{1}^T y && (\mathrm{D}_{\mathcal{C}}) \\ \text{s.t.} \quad & A^T y \leq c \\ & y \geq \mathbb{0} \end{aligned}$$

Given feasible solutions x and y to ($\mathrm{P}_{\mathcal{C}}$) and ($\mathrm{D}_{\mathcal{C}}$) respectively, the *residual cost* of node $v \in V$ is $c_v - A_{:,v}^T y$. Our general approach is the primal-dual method which proceeds as follows: We start with a set $S = \emptyset$ and $y = \mathbb{0}$. Then, in each iteration, increase y_C for all C in some subset of $\mathcal{C}$, maintaining dual feasibility until some condition is achieved. Usually this condition is a node becoming tight, that is, its residual cost becoming 0. We call this an *iteration* of our primal-dual algorithm. When a node becomes tight, we add it to S. Once S is a feasible $\mathcal{C}$-hitting set, we perform a *reverse deletion* procedure defined as follows. Consider each node in the reverse order in which it was added to S, and if deleting the node from S maintains feasibility and another condition that we specify later holds, we delete it. Call the resulting set S'. Given an inequality $\sum_{v \in V} A_{C,v} x_v \geq 1$ of ($\mathrm{P}_{\mathcal{C}}$) and the corresponding dual variable y_C, we say that y_C "*pays* for" $\sum_{v \in S'} A_{C,v}$ hit nodes. We also call $\sum_{v \in S'} A_{C,v}$ the *primal increase rate*. For brevity we will sometimes say C "*pays* for"

$\sum_{v\in S} A_{C,v}$ hit nodes instead of the dual variable y_C "*pays* for" $\sum_{v\in S} A_{C,v}$ hit nodes. Let y^* be the dual solution output by the primal-dual algorithm, and $\mathbb{1}^{S'}$ the characteristic vector for S' i.e. $[\mathbb{1}^{S'}]_v = 1$ if $v \in S'$ and $[\mathbb{1}^{S'}]_v = 0$ otherwise. It is well known e.g. [28], that if during any iteration dual variables y_i in some set $\mathcal{C}'$ were incremented uniformly, and the dual variables $\{y_C : \ C \in \mathcal{C}'\}$ pay for α hit nodes (of S') on average, then $c^T\mathbb{1}^S \leq \alpha \mathbb{1}^T y^*$.

Lemma 1. *[28] : Suppose that S and y are solutions to the primal and dual LP ($\mathrm{P}_{\mathcal{C}}$), ($\mathrm{D}_{\mathcal{C}}$) output by our primal-dual algorithm such that the following holds.*

1. *y is obtained starting with the initial feasible solution $y := \mathbb{0}$ and incrementing some set of dual variables $\{y_C : C \in \mathcal{C}_t\}$ uniformly and maintaining feasibility of y for iterations $t = 1, 2, .., l$ for some $l \in \mathbb{N}$.*

2. *For each iteration $t \in \{1, 2, 3, .., l\}$, the set $\{y_C : C \in \mathcal{C}_t\}$ of incremented dual variables satisfies $\sum_{C\in\mathcal{C}_t} |S \cap C| \leq \alpha|\mathcal{C}_t|$ for some $\alpha > 0$. Intuitively, the number of nodes of S each dual variable y_C pays for, $\sum_{v\in S} A_{C,v}$ is at most α on average.*

3. *$\forall v \in S, \quad A^T_{:,v} y = c_v$.*

Then S is a α-approximation. In fact, the characteristic vector $\hat{x} := \mathbb{1}^S$ (that is, $\hat{x}_v = 1$ for $v \in S$ and $\hat{x}_v = 0$ for $v \notin S$) satisfies $\sum_{v\in V} c_v \hat{x}_v \leq \alpha \sum_{v\in V} y_v$.

For the even cycle transversal problem and the rest of this thesis, $\mathcal{C}$ will refer to the set of (vertices of) even cycles of our graph G. For a set $T \subset V(G)$, define $x(T) = \sum_{t\in T} x_t$. Let us define the even cycle LP and its dual.

$$\begin{aligned} \min\ & c^T x && (\mathrm{P_{ECT}}) \\ \text{s.t.}\ & x(C) \geq 1 \quad \forall\, C \in \mathcal{C} && (2.2) \\ & x \geq \mathbb{0} \end{aligned}$$

$$\begin{aligned} \max\ & \mathbb{1}^T y && (\mathrm{D_{ECT}}) \\ \text{s.t.}\ & \sum_{C\in\mathcal{C}, v\in C} y_C \leq c_v \quad \forall v \in V(G) && (2.3) \\ & y \geq \mathbb{0} \end{aligned}$$

A key part of the primal-dual method is the reverse deletion procedure, which guarantees every node we pick is informally speaking, needed. Let's make this formal. Let us call a hitting set S *minimal*, if S is an ECT, but no proper subset of S is an ECT. Note that if S is output by a primal-dual algorithm that applies a reverse deletion procedure that deletes a node from our hitting set, if feasibility is maintained, then the resulting set S is minimal.

Definition 2. *We will say that A is a set of* pseudo-witness *cycles for a set S of vertices if, for each node v of S, there is a cycle $C \in A$ with $C \cap S = \{v\}$. If C is even we will call C a* witness cycle *for v.*

Note that if S is a minimal hitting set then there is a set A of witness cycles for S.

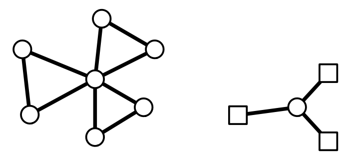


Figure 2.1: An example of a graph on the left and its block graph on the right.

2.1 Blended inequalities and compression

We begin by outlining the concept of "blended inequalities" for ECT and the necessary graph compression operations in order to define such inequalities. Blended inequalities were used by Fiorini et al. [24] in their work on diamond hitting sets, and our definitions follow theirs closely.

A *block* of G is an inclusion-maximal 2(-vertex)-connected subgraph of G. The *block graph* of G is the bipartite graph B_G with bipartition $V(B_G) = B_1 \cup B_2$, where B_1 are the blocks of G, B_2 are the cut nodes of G, and $(b_1, b_2) \in B_1 \times B_2$ is an edge if b_2 is a node of b_1. Here we slightly abuse notation and use the blocks of G as vertices, formally for each block B of our original graph, the block graph has a node v_B which we informally refer to as B. Let S be a partial solution to the given ECT instance at some point during the execution of our algorithm. Then let G^S be the corresponding residual graph that we obtain from $G - S$ by deleting all nodes that do not lie on even cycles. Our primal-dual algorithm now first looks for an even cycle C in G^S such that at most two nodes of C have neighbours outside C. If such a cycle C is found, we increment its dual variable y_C until a node becomes tight. The reason for doing this is that such a cycle will pay for at most two hit nodes, which we will show later.

If there is no even cycle C in G^S such that at most two nodes of C have neighbours outside C, we successively *compress* the residual graph G^S using two types of graph compression operations. To this end, first note that any minimal solution will only contain one node in the interior of any induced path in G^S. The interior of a path $v_1, v_2, .., v_l$ is the path $v_2 v_3, .., v_{l-1}$. Suppose that we contract some path P of G^S of length at least 2 down to an edge e, that is, we repeatedly *fold* degree-2 nodes v in P, as long as they exist, which means to delete v and adding the edge uw between its two neighbors u, w. Each cycle C of our new graph that uses the new edge uw "corresponds" to the cycle $(C \backslash \{uw\}) \cup P$ obtained by replacing the edge uw with the path P of our original graph. Each cycle C' of our original graph "corresponds" to the cycle $(C' \backslash P) \cup uw$ obtained by replacing the path P by uw of our new graph. Under this correspondence, removing a node p in the interior of P is now "equivalent" to removing the edge e, in the sense that a cycle of our original graph is removed by the deletion of p if and only if the corresponding cycle of our new graph is removed by the deletion of e. This is the motivation for the 1-*compression*, which we formally define later.

Recall that we assumed that there is no even cycle C in G^S such that at most two nodes of C have neighbours outside C. It follows that for $u, v \in V$ there are no induced $u - v$ paths of the same parity. Thus if we contract two u-v paths P_1, P_2 down to edges e_1, e_2 P_1 and P_2 must have different parity. Suppose that we contract two u-v paths P_1, P_2 with lengths of different parity down to edges e_1, e_2, respectively (see Figure 2.2). We will find it helpful to think of these edges as a single *twin* edge e between u and v whose parity is *flexible*. Formally, note that any cycle that uses both e_1 and e_2 consists of just those edges and hence corresponds to the odd cycle $P_1 \cup P_2$. If a cycle C in our contracted graph uses exactly one e_i, where $i \in \{1, 2\}$ then either C corresponds to an even cycle in our original graph or $(C \cup \{e_1, e_2\}) \backslash \{e_i\}$ corresponds to an even cycle in our original graph. As an example consider the blue and red edges of G in Figure 2.3. The blue and red paths get compressed into two uw edges in G_1 and replaced by a single twin edge in $\bar{G}_1$. The twin edge in $\bar{G}_1$ indicates that there are two u-w paths in G of different parity which were compressed to the single twin edge uw in $\bar{G}_1$. These are the blue u-w path of length 2 and the red u-w path of length 3.

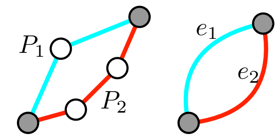


Figure 2.2: Two paths contracted to edges.

This is the motivation for the 2-*compression*.

Formally, we will successively compress G^S as follows:

- We obtain the 1-*compression* G_1^S (see Figure 2.3) of G^S by repeatedly folding degree-2 nodes v, as long as they exist.
- Note that no pair of nodes in G_1^S is connected by more than two edges. This is because each edge of G_1^S is the result of contracting a path. If there were 3 edges e_1, e_2, e_3 between a pair u, w of nodes in G_1^S, then there are 3 node disjoint u, w paths P_1, P_2, P_3, whose internal nodes have degree 2 in G^S which were contracted to e_1, e_2, e_3 respectively. Then one of $P_1 \cup P_2, P_2 \cup P_3, P_3 \cup P_1$ is an even cycle of G^S whose only nodes with outside neighbours are u and w, which contradicts our assumptions.

  We obtain $\bar{G}_1^S$ from G_1^S by replacing each pair of parallel edges by a single *twin* edge. In Figure 2.3 the two uw edges in G_1 are replaced with a single twin edge.

  In $\bar{G}_1^S$, we now once again fold degree-2 nodes as long as those exist. The resulting graph is the 2-*compression* G_2^S of G^S.

See Figure 2.3 for examples of 1- and 2-compression of a graph. In the following, we will sometimes call the subgraph Q of G^S whose compression yields a subgraph R of G_2^S the *preimage* of R. If R is an edge uv, call Q a *piece*, and say Q *corresponds* to R. Furthermore, call u, v *ends* of Q and other nodes of Q *internal nodes*. If the edge was twin, call the piece *twin*, otherwise, call the piece *single*. The blocks of a piece are cycles and paths, and the block graph of a piece is a path. Each cycle of a piece is called an *elementary cycle*. For an elementary cycle C, call its two nodes u_C and v_C with neighbours outside C *branch nodes*. Call the two $u_C - v_C$-paths P_1, P_2 in C the *handles* of C, which form the *handle pair* (P_1, P_2). For an illustration, see the red and light blue edges in Figure 2.3.

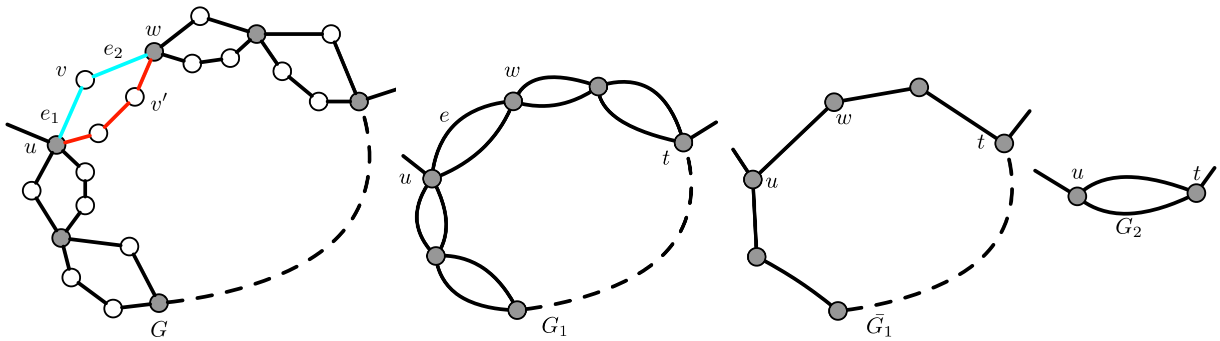


Figure 2.3: The figure shows the graph G and the 1 and 2-compression G_1 and G_2.

In the following, we will omit the superscript S from G_1^S, $\bar{G}_1^S$, and G_2^S if this is clear from the context. Let G_3 be obtained from G_2 by replacing every edge of G_2 with a path of length 2. If a twin edge was replaced, call the two edges of the path added *twin edges*. By an abuse of notation, we will say that a cycle of G_1, G_2 or G_3 is even if it contains a twin edge, or if its preimage in G is even. The reason for this is that we are interested in even cycles of G and are interested in the cycles of our compressed graphs that correspond to or contain even cycles of G. The reason for defining G_3 is that intuitively selecting a node inside a piece corresponds to selecting the edge corresponding to the piece in G_2. It will be simpler for us if our hitting set consists of only nodes, so we subdivide each edge of G_2. Suppose that S is the partial (and infeasible) hitting set for the cycles in $\mathcal{C}$ at some point during the algorithm. Further, assume that G^S has even cycles, but none with at most two outside neighbours. In this case, one can see that if an even cycle C' in G^S contains an internal node of some piece Q, then $C' \cap Q$ is a path between the ends of Q. We illustrate this in Figure 2.4. There, any even cycle that contains v,v' or w consists of a u-t path that goes through w and a u-t path that does not go through w. It follows that C' has the form $v_1 P_1 v_2 P_2 \dots v_k P_k v_1$, where for $i = 1, \dots, k$ the nodes $v_i, v_{i+1 \mod k}$ are ends of some piece Q_i, and P_i is a v_i-v_{i+1} path in Q_i. For $i = 1, \dots, k$, the pieces Q_i, Q_j for $i \neq j$ are disjoint except for their ends. We will say that C' in G^S *corresponds* to the cycle $C = (v_1, \dots, v_k)$ in G_2^S.

For a cycle C' in G^S corresponding to the cycle $C = (v_1, \dots, v_k)$ in G_2^S, we wish to define a valid inequality for ($\mathrm{P}_{\mathrm{ECT}}$). Note that the preimage of a non-twin piece is a path whose endpoints are the only nodes with outside neighbours. If C contains no twin piece, then its preimage is a cycle of G^S and hence there is only a single cycle C' in G^S which corresponds to C in G_2^S.

If C contains a twin piece, then there may be many cycles C' that correspond to C. Each such cycle uses exactly one handle of a pair and further, if $(P_{1,1}, P_{1,2}), \dots, (P_{i,1}, P_{i,2})$ are the set of handle pairs of C and $\hat{f} : \{1, 2, .., i-1\} \to \{1, 2\}$ then there is an even cycle C' which contains handles $P_{j,\hat{f}(j)}$ for $1 \leq j \leq i$ and one handle from the handle pair $(P_{i,1}, P_{i,2})$. The inequality $\sum_{v \in C'} x_v \geq 1$ is an inequality of ($\mathrm{P}_{\mathcal{C}}$). For $v \in V$, define $a_v^{C,\hat{f}} = 1$ if $v \in C' \cup (P_{i,1}, P_{i,2})$. Then for $x \in \mathbb{R}_+^V$, $\sum_{v \in V} a_v^{C,\hat{f}} x_v \geq \sum_{v \in C'} x_v \geq 1$ so $\sum_{v \in V} a_v^{C,\hat{f}} x_v \geq 1$ is

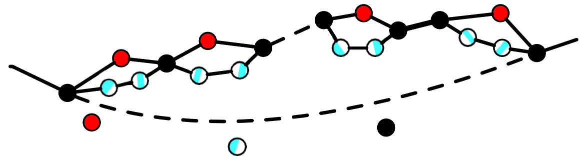


Figure 2.5: The bottom dashed black path has odd length. The number of length-5 faces at the top is assumed to be even.

a valid inequality for ($P_{\mathcal{C}}$).

Suppose that C' uses handle $P_{i,\hat{i}}$ $\hat{i} \in \{1,2\}$. In the primal-dual method, incrementing (the dual variable of) the the inequality $\sum_{v \in C'} x_v \geq 1$ decreases the residual cost of nodes on $P_{j,f(j)}$ for $1 \leq j \leq i-1$, but does not decrease the residual cost of nodes in the interiors of $P_{j,\{1,2\}\setminus f(j)}$ for $1 \leq j \leq i-1$, that is, the "other handle" of the handle pairs $(P_{1,1}, P_{1,2}), ..., (P_{i-1,1}, P_{i-1,2})$. Informally speaking, our algorithm wants some control on which nodes become tight and are picked. Consider the graph in Figure 2.5, assume that the nodes are unit weight. Removing a single black node would remove all even cycles. Removing a red node and a blue striped node would also remove all even cycles and has size twice the optimum. A really bad ECT is a set S consisting of one blue striped node on each cycle of length 5. S is minimal, in that any proper subset is not an ECT, but has arbitrarily large size. Informally speaking, we wish to increment our dual variables in a way that avoids this last case and we do this by choosing which handle of the handle pairs $(P_{1,1}, P_{1,2}), ..., (P_{i-1,1}, P_{i-1,2}))$ is used by C' carefully. For the last handle pair $(P_{i,1}, P_{i,2})$ we cannot decide which handle C' will use, however for reasons we explain later, we want the residual cost of nodes on both handles to decrease at the same rate. Thus we set $a_v^{C,\hat{f}} = 1$ for $v \in P_{i,1} \cup P_{i,2}$.

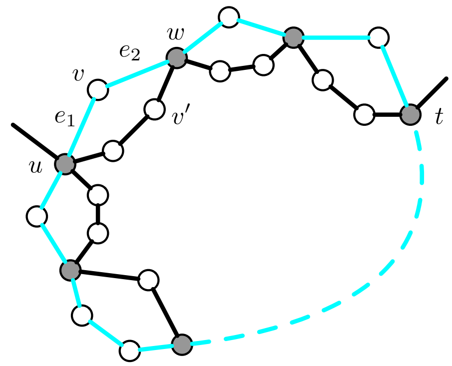


Figure 2.4: The light blue cycle in G has two u-t paths lying in different pieces of G; the dashed path has odd length.

To be more precise, for $\hat{f}, \bar{f} : \{1, 2, .., i-1\} \to \{1, 2\}$ define $a_v^{C,\hat{f},\bar{f}} = \frac{1}{2}(a_v^{C,\hat{f}} + a_v^{C,\bar{f}})$. We refer to the set of inequalities $\sum_{v \in V} a_v^{C,\hat{f},\bar{f}} x_v \geq 1$ over all functions $\hat{f}, \bar{f} : \{1, 2, .., i-1\} \to \{1, 2\}$ as a "family of blended inequalities". This will be explained precisely in Chapter 4.

Chapter 3

Generalizing a Feedback Vertex Set Result

In this chapter we outline and generalize the 2.4 approximation of Berman and Yaroslatvsev [8] for planar FVS problems satisfying a certain "uncrossability" criterion. This exposition follows Section 4 of [8].Looking into Berman and Yaroslatvsev's proof that their algorithm is a 2.4-approximation, we see they prove the following result. Given a minimal hitting set S (for an uncrossable FVS problem), a face of our graph is incident to at most 2.4 hit nodes on average. We generalize their result by showing that the minimality assumption can be replaced with each node of S has a "pseudo-witness cycle" satisfying certain properties we specify later. This allows us to generalize their result to Theorem 4, which will be crucial for the analysis of our even cycle transversal algorithm.

3.1 Pockets and their variants

Throughout this thesis, an embedding will refer to an embedding in the euclidean plane. The following definition of crossing cycles is elementary in the approach of Goemans and Williamson [29] for cycle transversal problems in planar graphs.

Definition 3. *Fix an embedding of a planar graph. Two cycles C_1, C_2* cross *if C_i contains an edge intersecting the interior of the region bounded by C_{3-i}, for $i = 1, 2$. That is, the plane curve corresponding to the embedding of the edge in the plane intersects the interior of the region of the plane bounded by C_{3-i} (see Figure 3.1 (ii), the two crossing cycles are depicted in red and black). A set of cycles $\mathcal{C}$ is* laminar *if no two elements of $\mathcal{C}$ cross (see Figure 3.1 (iii), the pair of laminar cycles is depicted in red and black).*

The improvement of Berman and Yaroslatvsev's work on FVS [8] over Goeman's and Williamson [29] involves looking at a type of subgraph called a *pocket*. We formally define pockets, and also introduce the new notion of "pseudo-pockets", the lack of which will help us "cover" our graph with even cycles.

Definition 4. *Let G be a graph and let $\mathcal{C}$ be a collection of cycles in G. A* pseudo-pocket *of $(G, \mathcal{C})$ is a connected subgraph G' of G which contains a cycle such that at most two nodes of G' have neighbours outside G'. A* pocket *of $(G, \mathcal{C})$ is a pseudo-pocket that contains a cycle of $\mathcal{C}$ (see Figure 3.1 (i) the graph formed by red nodes is a pocket.). A pocket is* minimal *if it contains no pocket as a proper induced subgraph.*

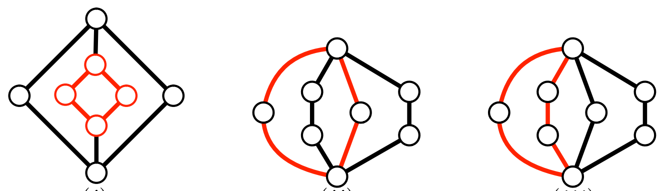


Figure 3.1: (i) pocket in red (ii) crossing cycles in red and black (iii) uncrossed cycles in red and black

One can generalize the 18/7-approximation for feedback vertex set problems satisfying a certain "uncrossability property" of [8] to the more general statement of Theorem 4. Informally speaking Theorem 4 says given a "minimal" hitting set in a planar graph and a set of faces in a minimal pocket, what the average degree of those faces is. That is our main contribution, which we state formally below.

Theorem 4. *Let G be a planar graph. Let H be an inclusion-wise minimal pocket of G. Let $S \subset V(G)$ be a set of nodes with some set $\mathcal{A}$ of laminar pseudo-witness cycles. Let $\mathcal{R}$ be a set of finite faces of H such that each cycle A of $\mathcal{A}$ contains a face of $\mathcal{R}$ in its interior (which may be A itself if $A \in \mathcal{R}$). Then*

$$\sum_{M \in \mathcal{R}} |M \cap S| \leq \frac{18}{7}|\mathcal{R}| \ .$$

Let $G, H, \mathcal{R}, S, \mathcal{A}$ be as in the statement of Theorem 4. To bound $\sum_{M \in \mathcal{R}} |M \cap S|$ we introduce the following definition.

Definition 5. *[29][8] Let G be a graph, let $\mathcal{R}$ be a set of cycles of G, and let $S \subset V(G)$. The* debit graph *for $\mathcal{R}$ and S is the bipartite graph $\mathcal{D}_G = (\mathcal{R} \cup S, E)$ with edges $E_{\mathcal{R}} = \{(C, s) \in \mathcal{R} \times S \mid s \in V(C)\}$. We will call $\mathcal{R}$ the* face nodes *of our debit graph.*

For us $\mathcal{R}$ will generally be a set of faces of some subgraph of G.

Given an embedding of G and a set $\mathcal{R}$ of faces of G, we can obtain an embedding of $\mathcal{D}_G$ by placing a node v_R inside the face R for each $R \in \mathcal{R}$ and drawing an edge from v_R to each node of S on R. This shows the following observation.

Observation 1. *[8, 29] If $\mathcal{R}$ is a set of faces of G, then the debit graph is planar and simple.*

Recall that for an even cycle C, we said that the dual variable y_C (or just C) corresponding to the inequality $\sum_{v \in C} x_v \geq 1$ in ($\mathrm{P}_{\mathcal{C}}$) pays for $|C \cap S|$ hit nodes. We say that $R \in \mathcal{R}$ *pays* for $|R \cap S|$ hit nodes. Note that $|R \cap S|$ is the degree of R in the debit graph $\mathcal{D}_G$. As a consequence, $\sum_{R \in \mathcal{R}} |R \cap S| = |E(\mathcal{D}_{\mathcal{G}})|$.

We introduce the notion of "balance", which for subsets $\mathcal{R}' \subseteq \mathcal{R}$ of cycles will be non-negative if cycles of R' are incident to at most 18/7 nodes of S on average and negative otherwise.

Definition 6. *For each subset $\mathcal{R}' \subseteq \mathcal{R}$, its* balance $\mathsf{bal}(\mathcal{R}')$ *is the quantity $|\mathcal{R}'| - \frac{7}{18}|E(\mathcal{D}_{\mathcal{G}}[S \cup \mathcal{R}'])|$.*

To build some intuition and to better understand the proof of Theorem 4, it may be helpful to the consider the following proof that $\sum_{M \in \mathcal{R}} |M \cap S| \leq 3|\mathcal{R}|$ given the assumptions of Theorem 4 (where $\mathcal{A}, \mathcal{R}$,.. are as in Theorem 4) in the special case that every cycle of $\mathcal{A}$ is a cycle of $\mathcal{R}$.

Lemma 2. *Let G be a planar graph. Let $S \subset V(G)$ be a set of nodes with some set $\mathcal{A}$ of laminar pseudo-witness cycles such that each element of $\mathcal{A}$ is a face. Let $\mathcal{R}$ be a nonempty set of finite faces of G such that $\mathcal{A} \subset \mathcal{R}$. Then*

$$\sum_{M \in \mathcal{R}} |M \cap S| \leq 3|\mathcal{R}| - 2 \ .$$

Proof. If $\mathcal{R} \backslash \mathcal{A} = \emptyset$, that is $\mathcal{A} = \mathcal{R}$, then $|E(\mathcal{D}_G[\mathcal{R} \backslash \mathcal{A}])| = \sum_{R \in \mathcal{R}} |R \cap S| = |\mathcal{A}| = |\mathcal{R}| \leq 3|\mathcal{R}| - 2$. If $\mathcal{A} = \emptyset$, then $S = \emptyset$ and $\sum_{R \in \mathcal{R}} |R \cap S| = 0 \leq \mathcal{R}| - 2$. Since the debit graph $\mathcal{D}_G[S \cup \mathcal{R} \backslash \mathcal{A}]$ is planar, simple and bipartite, and $\mathcal{A} \neq \emptyset$, $|\mathcal{R}| \geq 2$, Euler's formula for bipartite planar graphs with at least 2 vertices yields $|E(\mathcal{D}_G[S \cup \mathcal{R} \backslash \mathcal{A}])| \leq 2|V(\mathcal{D}_G[S \cup \mathcal{R} \backslash \mathcal{A}])| - 2$.

Since each node of S has a pseudo-witness $|\mathcal{A}| = |S|$. Substituting in, we obtain $|V(\mathcal{D}_G[\mathcal{R} \backslash \mathcal{A}])| = |\mathcal{R}| - |\mathcal{A}| + |S| = |\mathcal{R}|$ and $|E(\mathcal{D}_G[\mathcal{R} \backslash \mathcal{A}])| \leq 2|\mathcal{R}| - 2$. Recall that for $R \in \mathcal{R}$ $|R \cap S|$ is the degree of R in $\mathcal{D}_G$, so $\sum_{R \in \mathcal{R} \backslash \mathcal{A}} |R \cap S| = |E(\mathcal{D}[\mathcal{R} \backslash \mathcal{A}]) \leq 2|\mathcal{R}| - 2$.

From $|E(\mathcal{D}_G[S \cup \mathcal{R} \backslash \mathcal{A}])| \leq 2|V(\mathcal{D}_G[S \cup \mathcal{R} \backslash \mathcal{A}])| - 2$, it follows that $\sum_{R \in \mathcal{R}} |R \cap S| \leq 3|\mathcal{R}| - 2$. ☐

The following corollary follows from immediately from Lemma 2.

Corollary 1. *Suppose that G, S, $\mathcal{A}$ and $\mathcal{R}$ are as in Lemma 2. Let A be a cycle in G containing no node of S. Denote $\mathcal{R}_A$ the set of cycles $\mathcal{R}$ lying inside the region bounded by A. Then*

$$\sum_{R \in \mathcal{R}_A} |R \cap S| \leq 3|\mathcal{R}| - 2$$

Goemans and Williamson [29] essentially prove the following.

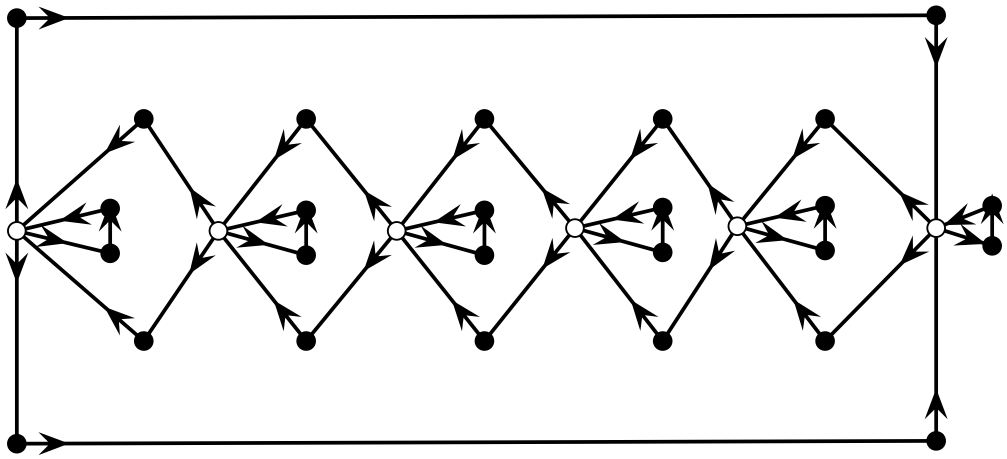


Figure 3.2: Tight example to algorithm of Goemans and Williamson.

Theorem 5. *Suppose that G, H S, $\mathcal{A}$ and $\mathcal{R}$ are as in Theorem 4. Then*

$$\sum_{R\in\mathcal{R}} |R \cap S| \leq 3|\mathcal{R}|$$

If $\mathcal{A} \subset \mathcal{R}$, then Theorem 5 follows from Lemma 2. Otherwise, for a witness cycle $A \in \mathcal{A}$ that is not a face, let $v_A \in S$ be the node that A is a witness of. Denote $\mathcal{R}_A$ the set of cycles $\mathcal{R}$ lying inside the region bounded by A. One can show using Corollary 1 that

$$\sum_{R\in\mathcal{R}_A} |R \cap (S\backslash\{v_A\}| \leq 3|\mathcal{R}| - 2.$$

Goemans and Williamson [29] manage to show that

$$\sum_{R\in\mathcal{R}_A} |R \cap (S\backslash| \leq 3|\mathcal{R}| - 2$$

. Informally speaking this means that whenever our instance contains a witness cycle A that is not a face, the set $\mathcal{R}_A$ of cycles $\mathcal{R}$ lying inside the region bounded by A pay for slightly less than 3 nodes of S on average.

Informally this is how Goemans and Williamson [29] (resp. Berman and Yaroslatvsev [8]) deal with witness cycles A that are not faces, namely by arguing that the total number of hit nodes that the cycles of $\mathcal{R}$ that are inside A pay for, is slightly less than 3 (resp. $\frac{18}{7}$ for minimal pocket and 2.4 for minimal pocket or 3-pocket) on average. Goemans and Williamson essentially show that for $H = G$ in Theorem 4, one can have $\sum_{R\in\mathcal{R}} |R \cap S| = 3|\mathcal{R}| - 6$ (see [29] Figure 2 also Figure 3.2 $\mathcal{R}$ is the set of finite faces that are dicycles $\mathcal{A}$ is the set triangles that are dicycles).

Recall that for our proof above that $\sum_{R\in\mathcal{R}} |R \cap S| \leq 3|\mathcal{R}|$, we used Euler's formula $|E(\mathcal{D}_G[S_A \cup \mathcal{R}\backslash\mathcal{A}])| \leq 2|V(\mathcal{D}_G[S_A\mathcal{R}\backslash\mathcal{A}])|$ on $\mathcal{D}_G[S_A \cup \mathcal{R}\backslash\mathcal{A}])$. If we could get a better bound of $|E(\mathcal{D}_G[S_A \cup \mathcal{R}\backslash\mathcal{A}])| \leq \alpha|V(\mathcal{D}_G[S_A\mathcal{R}\backslash\mathcal{A}])|$ for some $\alpha < 2$ one could get $\sum_{R\in\mathcal{R}} |R \cap S| \leq (\alpha + 1)|\mathcal{R}|$ which would be a better bound. Informally speaking, the only case this doesn't happen, that is, $\frac{\sum_{R\in\mathcal{R}} |R\cap S|}{|\mathcal{R}|} \approx 3$ our debit graph has to be as dense as a planar bipartite graph can be. Berman and Yaroslatvsev show that this cannot happen if our graph contains no pockets.

The previous paragraphs were for intuition only and are not necessary to understand the proof of Theorem 4. Our proof follows the same methodology as Berman and Yaroslavtsev [8]. Let A be a pseudo-witness cycle that is not a face and is minimally so, that is, any pseudo-witness cycle lying in the finite region bounded by A is a face. First, we show the instance formed by the set of faces of $\mathcal{R}$ lying inside or on such a pseudo-witness cycle A, has balance at least $1 - \frac{7}{18}$. Then our proof uses this to apply a reduction on G where a pseudo-witness cycle $A \in \mathcal{A}$ that is not a face and is minimally so, is selected, all nodes of G and all cycles of $\mathcal{R}$ lying in the interior of the region bounded by A are removed from G and $\mathcal{R}$ respectively, and A is added to $\mathcal{R}$. The key is that after sufficiently many reductions we obtain an instance of smaller balance and informally speaking almost no pseudo-witness cycles that are not faces. We then show that this reduced instance has positive balance, which will show that our original instance had positive balance. We will use the following result of Berman and Yaroslavtsev [8], which will be used both to show that reductions decrease balance and that the final reduced instance has positive balance. We include the proof for completeness.

Proposition 1. *[[8, Lemma 4.3]] Let W be a planar graph, $\hat{S}$ be a set of nodes on the outer face of W and $Q \subset \hat{S}$ be a set of nodes of W that we call* outer nodes*. Let $\mathcal{R}_W$ be a set of faces of W such that each non-outer node of $\hat{S} \cap W$ has a pseudo-witness cycle in $\mathcal{R}_W$. If W contains $a \leq 2$ outer nodes, then bal($\mathcal{R}_W$) $\geq 1 - \frac{7}{18}a$.*

Proof. We define the debit graph $\mathcal{D}_G$ and balance as before with $\mathcal{R}_W$ in place of $\mathcal{R}$ and again refer to the nodes $\mathcal{R}_W$ in $\mathcal{D}_G$ as face nodes. If $|\mathcal{R}_W| = 1$, let $\{M\} = \mathcal{R}_W$. The balance of $\mathcal{R}_W$ is at least $1 - \frac{7}{18}|M \cap \hat{S}|$. If M is a witness cycle, then $|M \cap \hat{S}| = 1$ and the balance of $\mathcal{R}_W$ is at least $\frac{11}{18}$. Otherwise, M is not in $\mathcal{A}$, so $\mathcal{A} = \emptyset$. So $\hat{S}$ consists only of outer nodes. Thus $|\hat{S}| \leq a$ and the balance of W, $1 - \frac{7}{18}|M \cap \hat{S}|$ is at most $1 - \frac{7}{18}a$.

Otherwise, $|\mathcal{R}_W| \geq 2$ and we apply the following "pruning procedure" for cycles of $\mathcal{R}$ of small degree. If a cycle M $\in \mathcal{R}_W\backslash\mathcal{A}_H$ is a node in at most two edges in $\mathcal{D}_G$, we remove M from $\mathcal{R}$. If $A_h \in \mathcal{R}_W$ for some $h \in \hat{S}$ and h is a node in at most two edges in E, including (A_h, h), we remove h from $\hat{S}$ and $\mathcal{A}_h$ from $\mathcal{R}$. To be clear, we don't prune W if $|\mathcal{R}_W| = 1$, but instead apply the analysis in the previous paragraph.

Assume that we apply pruning on A_h, h. Our new debit graph $\mathcal{D}'_G$ has $\mathcal{R}'_W = \mathcal{R}_W\backslash\{A_h\}$ as its face nodes. We claim bal$(\mathcal{E}_{\mathcal{R}'}) \leq$ bal$(\mathcal{E}_{\mathcal{R}})$. The new debit graph $\mathcal{D}'_G$ has at most 2 fewer edges as h is incident to at most 2 other nodes. $|\mathcal{R}'_W| = |\mathcal{R}_W| - 1$ and $|E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}'_W])| \geq |E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}])| - 2$. The balance of our new debit graph $\mathcal{D}'_G$ is $|\mathcal{R}'_W| - \frac{7}{18}|E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}'_W])| \leq |\mathcal{R}| - 1 - \frac{7}{18}(|E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}_W])| - 2)$ which is less than the balance of $\mathcal{R}_W$.

We will henceforth assume that $\mathcal{R}\backslash\mathcal{A}$ has no cycles of degree at most two.

Define $A = \mathcal{R}_W \cap \mathcal{A}_H$, $B = \mathcal{R}_W \backslash \mathcal{A}_H$ and Z as the set of faces in W, but not in $\mathcal{R}$. Let $\mathcal{G} = (A \cup Z, E)$ be a subgraph of the dual graph of W (we have an edge between two nodes of G if the corresponding faces share an edge). Let $C_1, C_2, \ldots C_q$ be the components of $\mathcal{G}$. Let E_i be the set of boundary edges of C_i that are adjacent to one face in C_i and one face in B. We call the E_i envelopes and the corresponding C_i as the inside of the envelope E_i. We also associate the outside face of W with being an envelope which we call the outer envelope; Figure 3.3 shows an example. Let us call the elements of A (resp. B), A- (resp. B-) faces.

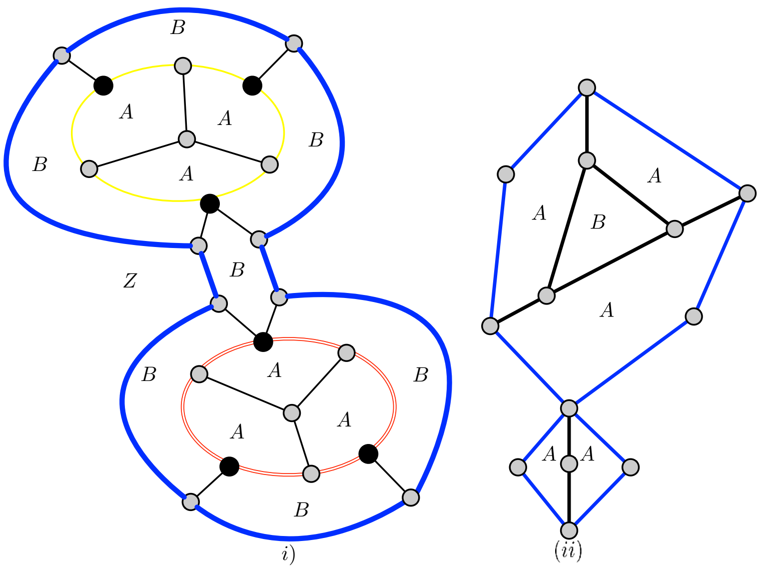


Figure 3.3: The left figure shows graph G with the 3 envelopes of G depicted by the yellow edges, double stroke red edges and thick blue edges, and hit nodes (large black dots). On the right shows a graph G with an envelope shown in blue, which is not a simple cycle.

Remark 1. *We assume that the envelopes here are simple cycles. If the outside face of W is a walk that intersects itself, then it is possible that the outer envelope is not a simple cycle. (However in this case let $Y_1, Y_2, ..., Y_t$ be the cycles forming the outside face of W. We can just apply our analysis to the subgraph of W lying in the interior of the region bounded by Y_i for each i.)*

It is clear that any non-outer nodes in $W \cap \hat{S}$ lie inside an envelope.

Definition 7. *For an envelope $\mathcal{S}$, its* principal neighbours *are the B-faces that have edges on $\mathcal{S}$.*

For example, in Figure 3.3, the black B-faces are principal, but the green B face is not. If a non-outer envelope had two principle neighbours B_1 and B_2, then the intersection of B_i with our envelope forms a path P_i for $i = 1, 2$. Further P_1, P_2 partition the edges of the outer face of our envelope and only the nodes of $V(P_1) \cap V(P_2)$ can have neighbours outside our envelope (see Figure 3.4, there u, v are the only two nodes in the envelope with outside

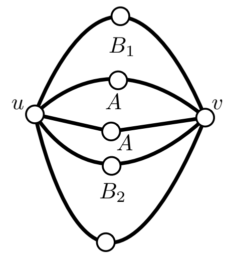


Figure 3.4: An envelope where only 2 nodes have outside neighbours.

neighbours). Likewise if an envelope has only one principle neighbour, then it has at most one node with an outside neighbour. Since our graph does not have pockets, any envelope that is not the outer envelope must have at least 3 principal neighbours.

Definition 8. *The intersection of a principal neighbour with an envelope is a path P, whose endpoints we call* contact nodes*.*

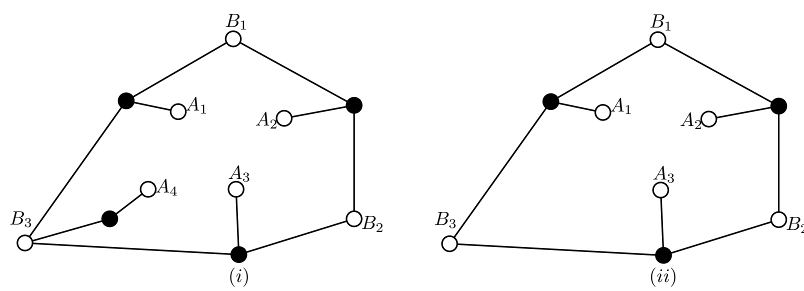


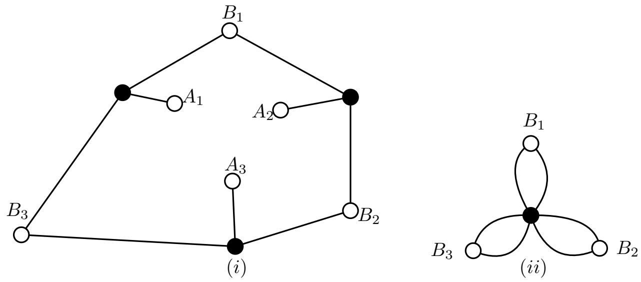
Figure 3.5: The figure shows the result of pruning a hit node of degree 2. On the left is a debit graph with face nodes $A_1, A_2, A_3, A_4, B_1, B_2$, where the nodes A_i correspond to pseudo-witness cycles. On the right is depicted the result of pruning A_4 the face node A_4 and the hit node that A_4 is a witness of is removed as well.

If a hit node h was not a contact node, then it lies in the interior of the shared path of a principal neighbour B_1 and the envelope it belongs to. This implies that h is incident to no other B-faces and is hence incident to at most 2 faces of $\mathcal{R}_W$. Thus we would have pruned h. Henceforth we assume that all hit nodes are contact nodes.

Let us construct a minor of our graph $\mathcal{D}_G$ as follows: We delete edges of the form (h, A_h) (and the resulting isolated nodes A_h) and contract the hitting set nodes of a single envelope to a single node $\mathbb{S}$. At this point $\mathcal{S}$ may be connected to its principal neighbours by 2 parallel edges and if this happens we delete one. Let us call this new graph $\mathcal{E}$. Further, since $\mathcal{D}_G$ has a natural embedding we get a natural embedding of $\mathcal{E}$.

Remark 2. *This embedding of $\mathcal{E}$ contains no parallel edges.*

Proof. By construction, $\mathbb{S}$ and B_i do not have parallel edges between them for principal neighbours B_i. Now let us consider a non-principal neighbour B of $\mathbb{S}$.



If B intersects two consecutive contact nodes h_i, h_{i+1} then we have a pocket.

If B does not, then for any two edges between B and S through the nodes h_i, h_j, we have $j \geq i + 2$.

Consider the portion of our envelope bounded by the two edges $\{h_i, B\}$ and $\{h_{i+1}, B\}$. Assume, without loss of generality that this portion is $h_i, h_{i+1}, \ldots h_j$; thus, B_i is bounded by these two edges by property of contraction $\{\mathbb{S}, B_i\}$ lies inside $\{B, h_i\}$ $\{B, h_{i+1}\}$ after contraction (and thus the two edges cannot be parallel). $\square$

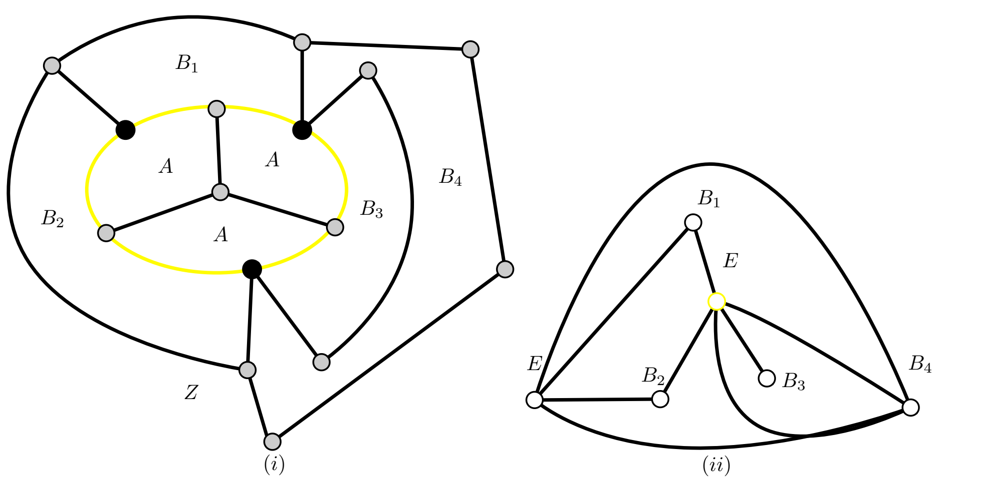


Figure 3.6: Non-principal face B_4 hits the yellow envelope twice, B_3 lies between the two edges between B_4 and E in $\mathcal{E}$ and thus they are not parallel edges.

Recall that the balance of $\mathcal{R}_W$ was defined as the quantity $|\mathcal{R}_W| - \frac{7}{18}|E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}_W])|$. We will refer to $|\mathcal{R}_W|$ as the "credit" and $|E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}_W])|$ as the "debit". In order to show that the balance is positive we "divide up" the credit and debit among the envelopes in such a way that every envelope has a non-negative balance. We break down the balance of $\mathcal{R}_W$ into envelopes, each of which has a non-negative balance and some of which have a strictly positive balance. That is, we partition $\mathcal{R}_W, E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}_W])$ into disjoint sets that we associate with the envelopes. Let $\mathbb{S}_1, \mathbb{S}_2, .., \mathbb{S}_\eta$ be the envelopes of our instance. We will define disjoint sets $\mathcal{R}_{W,1}, \mathcal{R}_{W,2}, .., \mathcal{R}_{W,\eta} \subset \mathcal{R}_W$ and partition $E(\mathcal{D}_\mathcal{G}[S \cup \mathcal{R}_W])$ into disjoint sets $E_1, E_2, .., E_\eta$ such that $|\mathcal{R}_{W,i}| - \frac{7}{18}|E_i| \geq 0 \;\; \forall i \in \{1, 2, .., \eta\}$. We will call $|\mathcal{R}_{W,i}| - \frac{7}{18}|E_i|$ the balance of envelope $\mathbb{S}_i$. We call $\mathcal{R}_{W,i}$ the faces assigned to envelope $\mathbb{S}_i$, and E_i the edges assigned to envelope $\mathbb{S}_i$. Each envelope $\mathbb{S}_i$ has the $\mathcal{A}$-faces that it contains assigned to it. Let $\hat{S}_i$ denote the set of hit nodes corresponding to the set of $\mathcal{A}$-faces assigned to $\mathbb{S}_i$. The set of edges of the debit graph assigned to $\mathbb{S}_i$ are all edges with one endpoint in $\hat{S}$. The balance of $\mathcal{R}_W$ is at least the sum of balances of all the $\mathbb{S}_i$. Thus this breakdown of the balance of $\mathcal{R}_W$ into envelopes will show that $\mathcal{R}_W$ has non-negative balance.

Let $n_\mathbb{S}$ be the number of contact nodes of $\mathbb{S}$, $d_\mathbb{S}$ be the degree of $\mathbb{S}$ in $\mathcal{E}$, $l_\mathbb{S}$ be the number of contact nodes that are not hit nodes, and $e_\mathbb{S}$ be the number of edges of $\mathcal{E}$ incident to node $\mathbb{S}$. Since our graph contains no pockets, $n_\mathbb{S} \geq 3$. Define bal := balance$(N) = |N| - \frac{7}{18}|E_N|$.

Let $\mathcal{E}$ have m nodes, d edges and f faces. Then $d = \Sigma_\mathbb{S} d_\mathbb{S}$ and $m = s + b$, where b is the number of faces in B and s is the number of envelopes. Since $\mathcal{E}$ is embedded without parallel edges and is bipartite, Euler's formula applies. Therefore, $m = d - f + 2 \geq d/2 + 2$. Thus, we can allocate $d_\mathbb{S}/2 - 1$ nodes for non-outer envelopes $\mathbb{S}$, and $d_\mathbb{S}/2 + 1$ nodes for the outer envelope with $a \leq 2$ outer nodes, and have enough to go around. For non-outer envelopes $\mathbb{S}$, we have $3(\frac{7}{18})n_\mathbb{S}$ units of debit from the principal neighbours and witness cycles, and $(d_\mathbb{S} - n_\mathbb{S})(\frac{7}{18})$ units of debit from the other B faces. Under this allocation, non-outer envelopes get $n_\mathbb{S} + d_\mathbb{S}/2 - 1$ units of credit. As $n_\mathbb{S} \geq 3$, it holds

$$\begin{aligned} bal_\mathbb{S} &= n_\mathbb{S} + d_\mathbb{S}/2 - 1 - (2n_\mathbb{S} + d_\mathbb{S})(\tfrac{7}{18}) \\ &= (3/2)n_\mathbb{S} - 1 + (d_\mathbb{S} - n_\mathbb{S})/2 - (3n_\mathbb{S} + (d_\mathbb{S} - n_\mathbb{S}))(\tfrac{7}{18}) \\ &\geq (3/2)n_\mathbb{S} - 1 - 3(\tfrac{7}{18})n_\mathbb{S} \\ &\geq 7/2 - 9(\tfrac{7}{18}) = 0 \end{aligned} \tag{3.1}$$

For the outer envelope $\mathbb{S}$ of W we do not get the credit from the outer face of W, that is the outer face of W is not in $\mathcal{R}_W$ and is thus not assigned to $\mathbb{S}$, but we allocate two extra B nodes. That is, we only get $n_\mathbb{S} - a$ units of credit from the witness cycles rather than the usual $n_\mathbb{S}$ units of credit, and we also have $\frac{7}{18}a$ less debit from the witness cycles.

Since we have $a \leq 2$ outer nodes, it holds

$$\begin{aligned} \mathsf{balance}_\mathbb{S} &\geq 3/2n_\mathbb{S} - 1 - 3n_\mathbb{S}(\frac{7}{18}) - a(1 - 1(\frac{7}{18})) + 2 \\ &\geq 3n_\mathbb{S}(\frac{1}{2} - 1(\frac{7}{18})) + 1 - a(1 - 1(\frac{7}{18})) \quad (n_\mathbb{S} \geq 2) \\ &\geq 4 - 6(\frac{7}{18}) - a(1 - 1(\frac{7}{18})) \\ &> (2 - a)(1 - \frac{7}{18}) \ . \end{aligned} \tag{3.1}$$

□

Definition 9. *If all nodes of a pseudo-witness cycle A are contained in H, call A a* hierarchical *pseudo-witness cycle. Otherwise, call A a* crossing *witness cycle. Denote the set of crossing pseudo-witness cycles by $\hat{\mathcal{A}}$.*

We are now ready to complete the proof of Theorem 4.

Proof. (of Theorem 4) We begin by reductions on our instance $(G, H, \mathcal{R}, \mathcal{A}, S)$ which simplify our instance and do not increase the balance. If after applying this reduction our instance has positive balance, then our instance had positive balance before the reduction. We define the reduction below.

Definition 10. *We define the following reduction on our instance* $(G, H, \mathcal{R}, \mathcal{A}, S)$*. If* H *contains a hierarchical pseudo-witness cycle* A *that is not a face of* $\mathcal{R}$*, delete all nodes, edges and faces of* $\mathcal{R}$ *inside* A *from* H *and add* A *to* $\mathcal{R}$*. If* H *does not contain a hierarchical witness cycle, we call the instance* $(G, H, \mathcal{R}, \mathcal{A}, S)$ reduced.

For a cycle C, let $\mathcal{R}_C$ be the faces in $\mathcal{R}$ contained in the region bounded by C. Let $H^1, \mathcal{R}^1$ be the result of applying the reduction in Definition 10 on $H, \mathcal{R}$. The balance of $H^1, \mathcal{R}^1$ is equal to

$$
\begin{aligned}
&|(\mathcal{R}\backslash\mathcal{R}_C) \cup \{C\}| - \sum_{M \in (\mathcal{R}\backslash\mathcal{R}_C)\cup\{C\}} |M \cap S| \\
=&|\mathcal{R}| - \sum_{M\in\mathcal{R}} |M \cap S| - (|\mathcal{R}_C| + 1 - (\sum_{M\in\mathcal{R}_C} |M \cap S|) + 1) \\
=&\mathsf{bal}(H) + 1 - \mathsf{bal}(\mathcal{R}_C) - \frac{7}{18} \ .
\end{aligned}
$$

That is to say, the reduction changes the balance by $1 - \mathsf{bal}(\mathcal{R}_C) - \frac{7}{18}$, which by Proposition 1 is non-positive.

Thus if after applying the reduction in Definition 10, our instance has positive balance then it initially had positive balance. We know apply the reduction in Definition 10 until our instance is reduced, for simplicity we will continue to call this graph H.

The crossing pseudo-witness cycles $\hat{\mathcal{A}}$ partition H into *regions*, see Figure 3.7. That is, consider the subgraph $K \subset H$ consisting of nodes and edges lying on a witness cycle of $\hat{\mathcal{A}}$ or on the outside face of H. The regions are defined as the portions of the plane bounded by the finite faces of K. Define a *subpocket* [8] as the subgraph of H consisting of the nodes and edges lying in or on the boundary of a region.

Proposition 2 ([8])**.** *The regions that the set of crossing cycles* $\hat{\mathcal{A}}$ *partition the plane into satisfy the following. For each region, there is a set* $\tilde{\mathcal{A}}$ *of at most 2 pseudo-witness cycles of* $\hat{\mathcal{A}}$ *such that each node bounding the region either does not lie on a pseudo-witness cycle in* $\hat{\mathcal{A}}$ *or lies on a cycle of* $\tilde{\mathcal{A}}$*.*

By the reduction described in Definition 10 each non-crossing cycle of $\mathcal{A}$ is a face. Since by Proposition 2, the outside face of each subpocket W contains nodes from at most two crossing pseudo-witness cycles, and contains all nodes that belong to pseudo-witness cycles lie on the outside face, there are at most two hit nodes of W whose pseudo-witness is not a face and they must lie on the outside face of W. Hence, each subpocket satisfies the

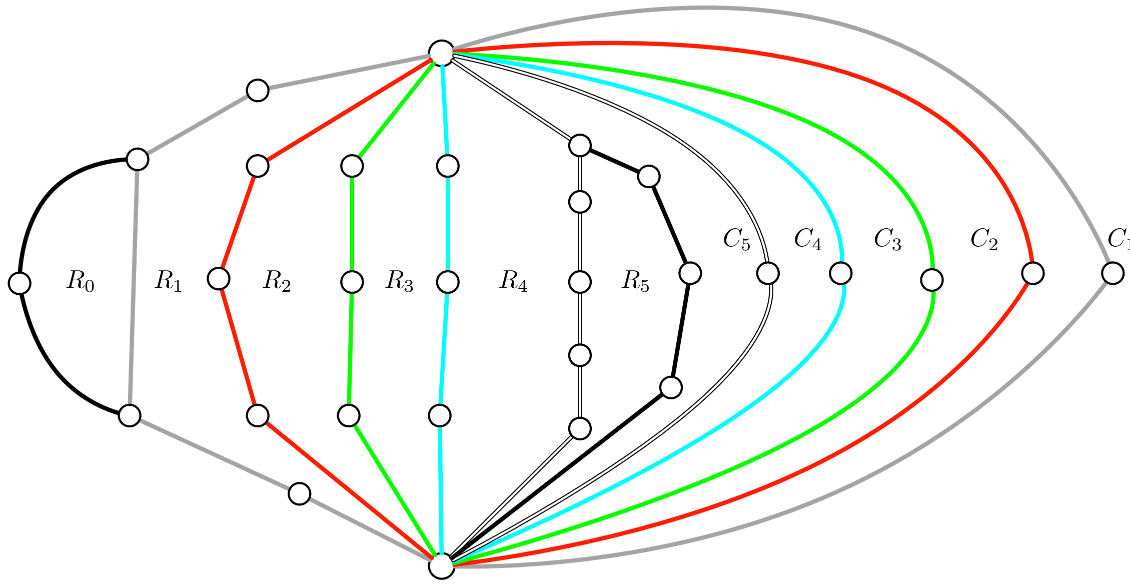


Figure 3.7: Pseudo-witness cycles $C_1, C_2, .., C_5$ divide H into regions $R_0, R_1, \ldots, R_5$.

conditions of Proposition 1 and hence has positive balance. Thus, H has positive balance, that is, $0 \leq |\mathcal{R}| - \frac{7}{18}|E_{\mathcal{R}}| = |R| - \sum_{M \in R} |M \cap S|$. Rearranging, $\sum_{M \in R} |M \cap S| \leq \frac{18}{7}|\mathcal{R}|$, which completes the proof of Theorem 4. $\square$

Chapter 4

Approximation Algorithm for Even Cycle Transversal

4.1 Introduction

In this chapter we show our 47/7-approximation algorithm for ECT in node-weighted planar graphs. This result was published first with Göke, Mnich and Koenemann in [30].

Theorem 1. *[30] ECT admits an polynomial-time* $47/7 \approx 6.71$*-approximation algorithm on node-weighted planar graphs.*

Our algorithm takes as input a node-weighted planar graph G with node weights $c_v \in \mathbb{N}$ for each $v \in V(G)$. We then employ a primal-dual algorithm that is based on the following natural covering LP for ECT and its dual, where $\mathcal{C}$ denotes the set of even cycles in G:

$$\begin{aligned} \min\ & c^T x && (\mathrm{P_{ECT}}) \\ \text{s.t.}\ & x(C) \geq 1 \quad \forall\, C \in \mathcal{C} && (4.1) \\ & x \geq \mathbb{0} \end{aligned}$$

$$\begin{aligned} \max\ & \mathbb{1}^T y && (\mathrm{D_{ECT}}) \\ \text{s.t.}\ & \sum_{C \in \mathcal{C}, v \in C} y_C \leq c_v \quad \forall v \in V(G) && (4.2) \\ & y \geq \mathbb{0} \end{aligned}$$

Fiorini et al. [24] proved that the integrality gap of this LP is $\Theta(\log n)$. Our main result is an improved integrality gap of this LP for ECT in planar graphs:

Theorem 6. *The integrality gap of the LP* ($\mathrm{P_{ECT}}$) *is at most* $47/7 \approx 6.71$ *in planar graphs.*

4.1.1 Our approach

Designing a primal-dual algorithm is far from trivial, as the imposed parity constraints rule out a direct application of the framework proposed by Goemans and Williamson [29]. Unlike in their work, the *face-minimal even cycles*—even cycles that contain a minimal set of faces in their interior—are not necessarily faces, and may thus overlap. Indeed, increasing the dual variables of face-minimal even cycles does not yield a constant-factor approximation in general.

Recall that in Section 2.1 we used an example of the unweighted graph in Figure 2.5 to illustrate the intuition in choosing which nodes in a piece end up in our ECT. Recall that in Section 2.1, we made the following observations about Figure 2.5. In Figure 2.5, a single black node, two nodes one on each handle of a handle pair and a set comprised of taking one blue node on each cycle of length 5 are all inclusion-wise minimal even cycle transversals. As the number of length 5 cycles becomes arbitrarily large, the set comprised of taking one blue node on each cycle of length 5 has arbitrarily larger size than taking a single black node. Informally speaking, we wish for our primal-dual algorithm to avoid selecting one blue node on each cycle of length 5 prior to the reverse deletion step as since this set of nodes is an inclusion-wise minimal ECT, this set would be returned by our algorithm. Let's see a weighted example of this phenomenon.

Consider Figure 4.1, and let F be the inner face that is only incident to blue striped and black nodes. For an even number of 5-cycles surrounding F, F is the only face-minimal even cycle in the graph. Using only F for the dual increase, even including a reverse-delete step, leaves one blue striped node of each 5-cycle. Yet, an optimal solution would take a single red and blue striped node from one 5-cycle.

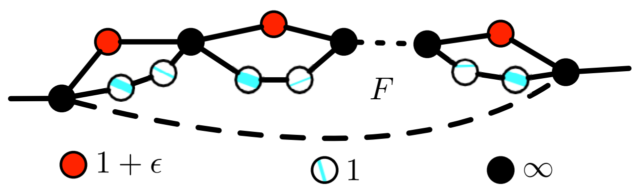


Figure 4.1: The bottom path has odd length, and the number of length-5 faces at the top is even.

To circumvent this impediment, we establish strong structural properties of planar graphs related to ECT. Those properties along with results from matching theory allow us to algorithmically find a large set of pairwise face-disjoint even cycles whose dual variables we can then increment. Even with this set of cycles found, it remains technically challenging to bound the integrality gap. For this purpose, we first use the structure of minimal hitting sets of our graph to associate each such set with a hitting set in a subdivision of the 2-compression of our graph. We then show that faces that are contained in even cycles we increment are incident to few nodes on average. Crucial in this step is a technical result that is implicit in the work of Berman and Yaroslavtsev [8]. Eventually, this approach leads to an integrality gap of 47/7, and an algorithm with the same approximation guarantee.

4.2 Primal-dual algorithm for ECT on node-weighted planar graphs

4.2.1 Blended inequalities formally defined

Recall that the 2-compression and related graph compressions were defined in Section 2.1. At the end of the section we briefly defined a family of "blended inequalities", which were dominated by a convex combination of inequalities $\sum_{v \in C} x_v \geq 1$ in ($\mathrm{P}_{\mathrm{ECT}}$). Let's be precise about which inequality we want.

For each cycle C, its *blended inequality* is

$$\sum_{v} a_v^C x_v \geq 1, \tag{$\circledast$}$$

where $a_v^C \in \{0, 1/2, 1\}$ for all nodes v, and where the support of a^C is contained in the node set of the preimage (defined in Section 2.1) of C. We next provide a precise definition of the coefficients of ($\circledast$). With those, one can show that ($\circledast$) is dominated by a convex combination of inequalities $x(C) \geq 1$ in ($\mathrm{P}_{\mathcal{C}}$).

Consider an elementary cycle of the preimage of C and let h_1, h_2 be its two handles. For each of these handles, we define its residual cost as the smallest residual cost of any of its internal nodes. Suppose that the residual cost of h_2 is at most that of h_1. We will also call h_1 the *dominant*, and h_2 the *non-dominant* handle of this cycle. As an invariant, our algorithm maintains that the designation of dominant and non-dominant handles of an elementary cycle does not change throughout the algorithm's execution.

Suppose first that the residual cost of h_1 is strictly larger than that of h_2. In this case, let $a_v^C = 1$ for all internal nodes of handle h_1, and let $a_v^C = 0$ of the internal nodes of h_2. If the residual cost of both handles is the same, we let $a_v^C = 1/2$ on internal nodes of both handles.

One can see that the current inequality is a convex combination of inequalities of the form $\sum_{v \in C} x_v \geq 1$ where C is a cycle. Namely consider the cycle C_1 formed by nodes that are on a dominant handle or not on a handle and the cycle C_2 formed by nodes that are on a non-dominant handle or not on a handle. One can see that the inequality $\sum_{v \in V} a_v^C x_v \geq 1$ is a convex combination of the inequalities $\sum_{v \in C_1} x_v \geq 1$ and $\sum_{v \in C_2} x_v \geq 1$. If C_1 and C_2 are both even cycles then $\sum_{v \in V} a_v^C x_v \geq 1$ is a convex combination of inequalities in ($\mathrm{P}_{\mathrm{ECT}}$).

Informally speaking, if C_1 and C_2 are not both even, we need to *correct the parity* of C_1 and C_2. In this case, we pick an arbitrary elementary cycle on C, and declare it *special*. For this special cycle, we then set $a_v^C = 1$ for the internal nodes on *both* handles. Following the same reasoning as Fiorini et al. [24] for DHS, we can show the following for ECT:

Lemma 3. *[24] Each feasible point of our LP* ($\mathrm{P}_{\mathrm{ECT}}$) *satisfies any blended inequality.*

Proof. We outlined this in Section 2.1, to recap, let $(P_{1,1}, P_{1,2}), ..., (P_{i,1}, P_{i,2})$ be the set of handle pairs of C with $P_{j,1}$ the handle of greater residual cost and $P_{i,1}, P_{i,2}$ then handles of the special cycle. For each choice of $f : \{1, 2, 3, .., i-1\} \to \{1, 2\}$, there is an even cycle C' which contains handles $P_{j,f(j)}$ for $1 \leq j \leq i-1$, one handle from the handle pair $(P_{i,1}, P_{i,2})$, all nodes not on a handle pair, and no other nodes, i.e. no node in the interior of $P_{j,\{1,2\}\setminus f(j)}$. For $v \in V$, define $a_v^{C,f} = 1$ if $v \in C' \cup (P_{i,1}, P_{i,2})$. Then for $x \in \mathbb{R}_+^V$, $\sum_{v \in V} a_v^{C,f} x_v \geq \sum_{v \in C'} x_v \geq 1$ so $\sum_{v \in V} a_v^{C,f} x_v \geq 1$ is a valid inequality for ($\mathrm{P}_{\mathcal{C}}$). Now define $\hat{f}(j) = 1 \;\; \forall j$ and $\bar{f}(j) = 1$ if $P_{j,1}$ has strictly greater residual cost than $P_{j,2}$ and $\bar{f}(j) = 2$ otherwise. Define $a_v^{C,\hat{f},\bar{f}} = \frac{1}{2}(a_v^{C,\hat{f}} + a_v^{C,\bar{f}})$. Then $a_v^{C,\hat{f},\bar{f}}$ are the coefficients of our blended inequality and $\sum_{v \in V} a_v^{C,\hat{f},\bar{f}} x_v \geq \frac{1}{2}(\sum_{v \in V} a_v^{C,\hat{f}} x_v + \sum_{v \in V} a_v^{C,\bar{f}} x_v) \geq 1$ so the blended inequality is valid. □

Just like in [24], in our algorithm, which we will define later, we assume that inequalities ($\circledast$) are part of ($\mathrm{P}_{\mathrm{ECT}}$). Throughout the algorithm, we increase dual variables $y_{\circledast}$ of such inequalities.

Recall that we say that variable $y_{\circledast}$ (or cycle C) *pays for* $\sum_{v \in S'} a_v^C$ hit nodes. Also recall from Lemma 1 that if during any iteration dual variables for a family of blended inequalities are incremented uniformly, and the dual variables pay for α hit nodes (of S') on average, then the final solution produced by the algorithm is α-approximate.

The motivation for blended inequalities is to pay for no more than one node in each piece. Consider the example in Figure 4.1. Here, the bottom black dashed path is odd, there are an even number of handle pairs in the top part, and ε is small. Suppose that we set $a_v^C = 1/2$ on internal nodes of each handle. If we were to increment the inequality ($\circledast$), all the blue nodes of weight 1 would become tight, and after reverse-delete, the algorithm would keep one blue node for each handle pair. This solution has cost equal to the number of length 5 faces in Figure 4.1. However, selecting a red node and a blue node would be a cheaper solution of cost $2 + \epsilon$. This could be achieved by setting $a_v^C = 1$ for red and black nodes, and $a_v^C = 0$ on blue nodes, until the residual costs of the red nodes become 1, and afterwards setting $a_v^C = 1/2$ on internal nodes of each handle.

Our algorithms will carefully in polynomial time choose a family of one or more even cycles $\mathcal{T}$ in G_2^S and increments the dual variables of certain blended inequalities for each $C \in \mathcal{T}$ until a node becomes tight, or the blended inequality changes; i.e. the residual costs of two handles of a handle pair, which were previously not equal, become equal.

The general framework of primal-dual for even cycles will be the following.

We start with the empty candidate $S := \emptyset$. In every iteration, the algorithm first looks for an even cycle C in the residual graph G^S such that at most two nodes of C have outside neighbours. If we find such a C, increment the variable y_C until a node becomes tight. If no such cycle exists, the algorithm increments the blended inequalities of some set $\mathcal{T}$ of even cycles. The algorithm then adds all nodes X that became tight to our candidate hitting set S.

During an iteration, for each handle pair (Q_1, Q_2) for which the set X of nodes that became tight contains a node in the interior of each handle, our algorithm will choose two nodes $a, b \in X$ with a in the interior of Q_1 and b in the interior of Q_2 and define (a, b) to be a *node pair*. For instance, in Figure 2.3 if v and v' are the only nodes added during some iteration then the algorithm would define (v, v') to be a node pair. For a set of nodes X added during the same iteration, nodes in a pair are considered to be added *before* any node not in a pair.

At the end of the algorithm, we perform a non-trivial reverse-delete procedure. Formally, let $w_1, \ldots, w_\ell$ be the nodes of S in the order they were added to S by the algorithm. For nodes w_i, w_j that were added during the same iteration we assume that the ordering satisfies the property that if w_i is in a pair and w_j is not, then $i < j$. That is, for reverse-delete purposes, nodes not in a pair are considered for deletion first. For $p = \ell, \ell-1, \ldots, 1$, if w_p is not in a node pair, then if $S \backslash \{w_p\}$ is a feasible hitting set, the algorithm deletes w_p from S; otherwise, it does not. If w_p is in a node pair (w_p, w'), then if $S \backslash \{w_p, w'\}$ is a feasible hitting set, then delete both w_p, w' from S; otherwise, keep both w_p, w'. The complete description is given in Algorithm 4.2.1.

To clarify, Algorithm 4.2.1 is the general framework that our 47/7-approximation algorithm will fit into. $\mathcal{T}$ is currently an arbitrary nonempty set of even cycles; we will clarify in Lemma 6 what its desired properties are and explain in Sections 4.2.2, 4.2.5 how to choose it.

The intuition behind the caveat in our reverse-delete step is that node pairs are often very useful to keep, because they disconnect a piece. Consider the example in Figure 4.2. There is a length-5 face with green square nodes of cost 2, and an odd number r of length-5

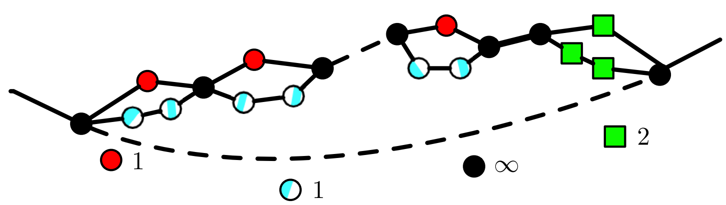


Figure 4.2: The red and blue striped nodes have weight 1, black nodes have infinite weight and green square nodes have cost 2. The bottom dashed black path has odd length. The number of length-5 faces at the top is assumed to be even.

faces with red and blue striped nodes of cost 1. The black nodes have cost infinity. The bottom dashed path has odd length. In the 2-compression, all length-5 faces in the figure belong to one piece. Suppose that for the blended inequality we choose the length-5 face with the green square nodes as the special cycle, and we increment this blended inequality. One sees that the red, blue striped and green square nodes become tight simultaneously.

To see that reverse delete order needs to be chosen carefully, consider the following adversarial ordering: in reverse delete, consider the two green square nodes other than v first, then consider the red nodes, and then consider one blue striped node on each handle.

Algorithm 4.2.1: EvenCycleTransversal(G, c)

Input : A graph G with node costs $c : V(G) \to \mathbb{N}$.
Output: An even cycle transversal S of G.

$S \leftarrow \emptyset$
while *Residual graph* G^S *contains an even cycle* **do**
  if G^S *contains a cycle* C *with at most* 2 *outside neighbours* **then**
    increase the dual variable y_C for C until a node v becomes tight.
  else
    compute the 2-compression G_2^S of G^S.
  $\mathcal{T} \leftarrow$ some (nonempty) set of cycles of G_2^S.
  Increment dual variables of blended inequalities of all $C \in \mathcal{T}$ until a node v becomes tight or the blended inequality changes.
  Add all nodes that became tight to S.
  Denote by X the set of nodes that became tight this iteration.
  for *each handle pair* (Q_1, Q_2) **do**
    if X *contains a node in the interior of each handle* **then**
      choose two nodes $a, b \in X$ with a in the interior of Q_1 and b in the interior of Q_2 and define (a, b) to be a node pair.
$w_1, \ldots, w_\ell \leftarrow$ nodes of S in the order they were added, where for nodes X added during the same iteration, any node of X in a node pair appears before others node of X not in a node pair.
for $i = \ell$ **downto** 1 **do**
  if w_i *is not part of a node pair* **then**
    if $S \backslash \{w_i\}$ *is feasible* **then**
      $S \leftarrow S \backslash \{w_i\}$.
  else
    Let (w_i, w_j) be the node pair containing w_i. **if** $S \backslash \{w_i, w_j\}$ *is feasible* **then**
      $S \leftarrow S \backslash \{w_i, w_j\}$.
return S

Finally, consider the remaining blue striped nodes. One can see that the algorithm would end up with v and one blue striped node per handle, which has cost $r + 2$ while the optimum (which selects the solution consisting of one red and one blue striped node on a handle pair) has cost 2.

Recall that a piece of our graph is the preimage of an edge of the 2 compression. Also recall that, informally speaking, we wish for a blended inequality to pay for at most one hit node inside a piece. This is made formal in the following theorem which is a modification of a result by Fiorini et al. [24, Theorem 5.7] for even cycles and tells us the structure of a minimal solution within a piece. (Fiorini et al. [24, Theorem 5.7] is for diamonds and has differences in its reverse deletion procedure.)

Lemma 4. *Let S' be the output of Algorithm 4.2.1 on input (G, c). Let $\bar{t}$ be a time during our algorithm, let $S_{\bar{t}}$ be the current hitting set at this time and let $G^{S_{\bar{t}}}$ be the residual graph. Consider an edge $uw \in E(G_2^{S_{\bar{t}}})$ on the even cycle whose dual variable we increase, and let Q be the piece corresponding to uw in G. Then exactly one of the following occurs:*

1. *S' contains no internal node of Q,*
2. *S' contains exactly one node of Q, and this node is a cut-node of Q,*
3. *S' contains exactly two nodes of Q, and they belong to opposite handles of a cycle of Q,*
4. *S' contains exactly one node per elementary cycle of Q, each belonging to the interior of some handle of the corresponding cycle.*

Proof. If S' contains two nodes a and b in the interiors of different handles of a handle pair, then since removing both a and b disconnects u from w in Q, our algorithm would delete all other nodes of $V(Q)\backslash\{u, w\}$ from S'. If u or w were in S', then our algorithm would delete both a and b. Thus $u, w \notin S'$, and case 3 holds.

Similarly, if S' contains a cut node z, then since removing z disconnects from u from v in Q, our algorithm would delete all other nodes of $V(Q)\backslash\{u, v\}$ from S'. If u or w were in S', then our algorithm would delete z. Thus $u, w \notin S'$, and case 2 holds.

If u or w is in S', then for any $r \in S' \cap (V(Q)\backslash\{u, w\})$ there cannot be an even cycle of G which intersects S' only at r as such a cycle would have to go through u or w, and thus S' contains no internal node of Q and case 1 holds.

By the above we may assume that S' does not contain a cut node of Q or two nodes in the interiors of different handles of a pair. Suppose that case 1 does not hold. Let (P_1, P_2) be a handle pair on Q such that P_1 contains a hit node t in its interior and P_2 does not. Suppose that Y_1, Y_2 was another handle pair with no hit node on either of Y_1 or Y_2. By our deletion procedure, there must be an even cycle C which intersects S' at t only. Such a cycle C uses the handle P_1 and one handle Y_i of the pair Y_1, Y_2. Let C' be the cycle obtained from C by replacing the paths P_1 and Y_i in C by the paths P_2 and Y_{3-i}. Since the lengths of different handles of a pair have different parity, C' is even. Since P_2, Y_1 and Y_2 contain no nodes of S', C' contains no nodes of S', which is a contradiction. Since a handle can only contain one hit node of S', this implies that case 4 holds. □

An immediate corollary is the following.

Lemma 5. *For a cycle C with k pieces, the set S of nodes output by our algorithm satisfies*

$$\sum_{v\in S} a_v^C \le k+1 \ . \tag{4.3}$$

If C contains no twin piece, then

$$\sum_{v\in S} a_v^C \le k, \tag{4.4}$$

Proof. Let $C = v_1v_2,..,v_l$ ($v_{l+1} := v_1$). Let $p(u,v)$ be the piece of G corresponding to uv in G_2. By Lemma 4 if the special cycle of the blended inequality for C does not lie on $p(v_i,v_{i+1})$, $\sum_{v\in p(v_i,v_{i+1})\setminus v_{i+1}} a_v^C \le 1$. If the special cycle of the blended inequality for C lies on $p(v_i,v_{i+1})$, $\sum_{v\in p(v_i,v_{i+1})\setminus v_{i+1}} a_v^C \le 2$. We sum up the inequalities $\sum_{v\in p(v_i,v_{i+1})\setminus v_{i+1}} a_v^C \le 1$ for pieces $p(v_i,v_{i+1})$ not containing the special cycle and the inequality $\sum_{v\in p(v_i,v_{i+1})\setminus v_{i+1}} a_v^C \le 2$ for the one piece $p(v_i,v_{i+1})$, if it exists, that contains the special cycle. This yields the result. □

Recall that we defined G_3 as being obtained from G_2 by replacing every edge of G_2 with a path of length 2. Given a hitting set S' output by Algorithm 4.2.1, we wish to construct a corresponding hitting set for $G_3^{S_{\bar{t}}}$ such that the primal increase rate of any particular blended inequality (with respect to S') is equals the number of nodes of S'_3 on the corresponding cycle of $G_3^{S_{\bar{t}}}$.

Definition 11. *Let S' be a hitting set output by Algorithm 4.2.1. The* corresponding hitting set for $G_3^{S_{\bar{t}}}$ *is the set $S'_3 \subset V(G_3^{S_{\bar{t}}})$ obtained by first taking the nodes of $S'\cap V(G_3^{S_{\bar{t}}})$. Now, consider an edge uv of $G_2^{S_{\bar{t}}}$ with corresponding piece P. Replace uv by the path uw_pv in $G_3^{S_{\bar{t}}}$, and add w_p to S'_3 if $P - S'$ has two components.*[1]

Claim 5.1. *Let C be the preimage of an even cycle in $G_2^{S_{\bar{t}}}$, and C_3 the corresponding cycle in $G_3^{S_{\bar{t}}}$. We claim $\sum_{v\in S'} a_v^C \le |C_3\cap S'_3|+1$. Further, if C does not contain a twin edge, then $\sum_{v\in S'} a_v^C \le |C_3\cap S'_3|$.*

Proof. Define b^C as follows: For a handle pair, while one handle has greater residual cost than the other set $b_v^C = 1$ for v on the handle of greater residual cost $b_v^C = 0$ on internal nodes of the other handle (change b^C whenever residual costs become equal). Otherwise, $b_v^C = 1/2$ on internal nodes of both handles. In short, b_v^C are the coefficients a_v^C if we had not redefined $a_v^C = 1$ for nodes on the special cycle.

Let $uw \in E(G_2^{S_{\bar{t}}})$, Q be the preimage of uw in $G^{S_{\bar{t}}}$ and uw_Qw be the subdivision of uw in $G_3^{S_{\bar{t}}}$. Let S'_3 be the corresponding hitting set of S' for $G_3^{S_{\bar{t}}}$. We claim $\sum_{v\in S'\cap(Q\setminus\{u,w\}} b_v^C = |S'_3\cap\{w_Q\}|$. We distinguish which case of Lemma 4 is satisfied by uw and S'.

[1]Note that the minimality of S' implies that removing S' from P yields at most two connected components.

- If uw and S' satisfy (1), then $\sum_{v\in S'\cap(Q\setminus\{u,w\}} b_v^C = 0$. Since S' contains no internal node of Q, $Q\setminus S$ is connected and hence S'_3 does not contain w_Q. Hence $\sum_{v\in S'\cap(Q\setminus\{u,w\})} b_v^C = |S'_3 \cap \{w_Q\}|$.
- If uw and S' satisfy (2) or (3), then S' does not contain either end node of Q, and contains either a single cut node of Q, or exactly two nodes of Q in the interiors of two handles of a handle pair of Q. Thus, $S'\cap Q$ consists either of a single node v for which $b_v^C = 1$, or two nodes j,k for which $b_j^C = b_k^C = 1/2$, and so $\sum_{v\in S'\cap Q} b_v^C = 1$.

  In either case (2) or (3), $Q\setminus S'$ is disconnected so $|S'_3\cap\{w_Q\}| = 1$. Hence $\sum_{v\in S'\cap(Q\setminus\{u,w\})} b_v^C = |S'_3 \cap \{w_Q\}|$.
- Suppose that S' satisfies (4). Suppose for a contradiction that Algorithm 4.2.1 added a node pair (l,m) on some handle pair (P_1,P_2) of Q. It then follows from the reverse-delete step that the final solution S' contains both l and m, or none of them. Since we do not contain a node pair, the deletion procedure of Algorithm 4.2.1 implies the algorithm did not add a node pair with nodes in Q.

  Hence, throughout the algorithm, for each handle pair (P_1,P_2) of Q, the handle P_i, which contains a hit node in its interior must have strictly less residual cost than the other. Hence $b_v^C = 0$ on handle P_i. This implies

$$\sum_{v\in(V(Q)\setminus\{u,w\})} b_v^C = 0 \ . \tag{4.5}$$

Thus $\sum_{v\in S'\cap(Q\setminus\{u,w\}} b_v^C = |S'_3 \cap \{w_Q\}|$.

Let $C = v_1v_2..v_lv_1$. Let Q_i be the piece corresponding to $v_iv_{i+1 \mod l}$. Let q_i be the node resulting from subdividing $v_iv_{i+1 \mod l}$ in $G_2^{S_{\bar{t}}}$ to obtain $G_3^{S_{\bar{t}}}$. Let $C_3 := v_1q_1v_2, q_2, .., v_lq_l$ the cycle corresponding to C in $G_3^{S_{\bar{t}}}$. We showed

$$\sum_{v\in S'\cap(Q_i\setminus\{u,w\})} b_v^C = |S'_3 \cap \{q_i\}|. \tag{4.6}$$

Summing (4.6) for $i-1,..,l$ yields $\sum_{v\in S'\cap(\cup_{i=1}^l Q_i\setminus\{v_1,v_2,..,v_l\})} b_v^C = |\{q_1,q_2,..,q_l\}\cap C_3|$.

Noting $b_{v_i}^C = 1$ and $b_v^C = 0$ for $v \notin \cup_{i=1}^l Q_i$ for each i, yields

$$\sum_{v\in S'} b_v^C = |C_3 \cap S'_3| \ . \tag{4.7}$$

Let us now relate a_v^C to b_v^C. If C has no twin edge, then the blended inequality coefficients a_v^C are equal to b_v^C, therefore $\sum_{v\in S} a_v^C = |C_3 \cap S'_3|$.

In general, C may contain a twin edge. In this case, a_v^C differs from b_v^C only in the interior of the handles H_1,H_2 of the special cycle: then either $b_v^C = \frac{1}{2}$ in the interior of H_1 and H_2, or $b_v^C = 0$ in the interior of the dominant handle, and $b_v^C = a_v^C$ everywhere else.

If $b_v^C = \frac{1}{2}$ in the interior of H_1 and H_2, then note from Lemma 4 there are at most two nodes of S' on $H_1 \cup H_2$. Thus, $\sum_{v \in S'} a_v^C \leq \sum_{v \in S'} b_v^C + 1$.

Otherwise, $b_v^C = 0$ in the interior of the dominant handle, and $b_v^C = a_v^C$ everywhere else. Since S' contains at most one node from the dominant handle $\sum_{v \in S'} a_v^C \leq \sum_{v \in S'} b_v^C + 1$. Thus, $\sum_{v \in S'} a_v^C \leq |C_3 \cap S_3'| + 1$ completing the proof. □

Recall from Lemma 1 and the paragraph beforehand, that during any iteration of a primal-dual algorithm dual variables y_i in some set $\mathcal{C}'$ were incremented uniformly, and the dual variables $\{y_C : \ C \in \mathcal{C}'\}$ pay for α (for some $\alpha > 1$) hit nodes (of S') on average, then such an algorithm is an α-approximation. To show that Algorithm 4.2.1 is a α-approximation, we need to show the following.

Lemma 6. *During any iteration of Algorithm 4.2.1 the average number of hit nodes the incremented dual variables $\{y_C : \ C \in \mathcal{T}\}$ pay for, $\frac{1}{|\mathcal{T}|} \sum_{C \in \mathcal{T}} \sum_{v \in S'} a_v^C$ is at most $47/7$.*

The rest of this section will be the proof of Lemma 6. By Claim 5.1, if we can show that $\frac{1}{|\mathcal{T}|} \sum_{C \in \mathcal{T}} \sum_{v \in S'} |C_3 \cap S_3'| \leq \alpha - 1$, then $\frac{1}{|\mathcal{T}|} \sum_{C \in \mathcal{T}} \sum_{v \in S'} a_v^C \leq \alpha$. It would suffice to argue that $\frac{1}{|\mathcal{T}|} \sum_{C \in \mathcal{T}} \sum_{v \in S'} |C_3 \cap S_3'| \leq 40/7$. We will instead argue that $\frac{1}{|\mathcal{T}|} \sum_{C \in \mathcal{T}} \sum_{v \in S'} |C_3 \cap S_3'|$ is small while also taking into account of how many cycles of $\mathcal{T}$ contain a twin edge. To show that $|C_3 \cap S_3'| + 1$ is small on average we need the fact that S_3' is a minimal ECT, which is stated in the following remark.

Remark 3. *Let S' be the output of Algorithm 4.2.1 on input (G, c). Let S_3' be the corresponding hitting set for $G_3^{S_{\bar{t}}}$ in Definition 11. Then there is a witness cycle for each $v \in S_3'$.*

Proof. Recall that cycles of $G^{S_{\bar{t}}}$ take the form $v_1, R_1, v_2, R_2.., R_{n-1}, v_n, R_n, v_1$ where v_i are nodes of G_{D2} and R_i is a path between v_i and v_{i+1} in the piece P_i with ends v_i and v_{i+1} in G. Let $w_{v_i, v_{i+1}}$ ($v_{n+1} = v_1$) be the node resulting from the subdivision of $v_i v_{i+1}$ in G_2^S. Let $v \in S_3$. If $v \in V(G)$, denote $v_1 = v$ and let $v_1, R_1, v_2, R_2, \ldots, R_{n-1}, v_n, R_n, v_1$ be a witness cycle for v. Since there is a path R_i connecting the endpoints of P_i, $w_{v_i, v_{i+1}}$ is not in S_3. Since v_i for $i \neq 1$ are not in S, $v_1, w_{v_1, v_2}, v_2, w_{v_2, v_3}, \ldots, w_{v_{n-1}, v_n}, v_n, w_{v_n, v_1}, v_1$ is a witness cycle for v in $G_3^{S_{\bar{t}}}$.

If $v \notin V(G)$, then $v = w_{v_1 v_2}$ for some edge $v_1 v_2 \in G_2$. So S contains a hit node a of the piece P_1 (preimage of $v_1 v_2$). Either a has a witness cycle or a is part of a node pair (a, b) and b has a witness cycle. By replacing a with b if (a, b) is a pair we may assume that a has a witness cycle $v_1, R_1, v_2, R_2, \ldots, R_{n-1}, v_n, R_n, v_1$ where as before, R_i is a path between v_i and v_{i+1} in the piece P_i with ends v_i and v_{i+1} in G, and R_1 goes through a. For $i \neq 2$ since there is a path R_i connecting the endpoints of P_i, $w_{v_i, v_{i+1}}$ is not in S_3. Since v_i are not in S, $v_1, w_{v_1, v_2}, v_2, w_{v_2, v_3}, \ldots, w_{v_{n-1}, v_n}, v_n, w_{v_n, v_1}, v_1$ is a witness cycle for v in G_3. □

4.2.2 Identifying families of even cycles via tilings

It remains to explain how to choose the set $\mathcal{T}$ for Algorithm 4.2.1. We motivate our choice for this with the 12/5-approximation algorithm of Berman and Yaroslavtsev mentioned in Chapter 3. Recall the 12/5-approximation of Berman and Yaroslavtsev mentioned in Chapter 3. It starts with the empty hitting set $S = \emptyset$. As long as S is not a hitting set for the directed cycles of G, it first looks for a pocket H of the directed residual digraph $G^{S\rightarrow}$, that is the digraph obtained from $G - S$ by deleting all nodes not on a directed cycle. It then increments the dual variables for the set of face minimal directed cycles of H, which happen to be faces. It then adds any nodes that become tight to S. Once S is feasible, the algorithm performs a reverse deletion step.

Recall Definition 4: Given a graph G, a *pseudo-pocket* of G is a connected subgraph G' of G which contains a cycle such that at most two nodes of G' have neighbours outside G'. As pointed out, in our setting, face-minimal even cycles may not be faces, and may cross. Following Berman and Yaroslavtsev [8], we wish to "cover" our residual graph with face-minimal even cycles which do not cross, we call this a "tiling"; see Figure 4.3 (iii). As we will see, this tiling allows us to identify the dual variables to increase. Let us formalize the correspondence between edges of the dual between odd faces and even faces.

Definition 12. *Let H be a plane graph without pseudo-pockets. For each face f of H, let v_f be the corresponding node of the planar dual H^*.*

A tile *of H is an even cycle C of H bounding one or two faces. If C is a single face f, we say that C* corresponds *to the node v_f. If C bounds two faces f and g, we say that C corresponds to the edge $v_f v_g \in E(H^*)$. We say that nodes v_f, v_g and the faces f, g are* covered *by the tile.*

For a node v of H^*, let $f_v \subset E(H)$ be the edges on the boundary of the corresponding face of H. Denote by f_∞ the infinite face of H. Denote by v_∞ the node of H^* corresponding to the infinite face.

Given $v_f v_\infty \in E(H^*)$, a cycle $C \subset E(H)$ *corresponds* to $v_f v_\infty$ if C is a cycle of $f \Delta f_\infty$, or $C = C' \Delta f$ and C' is a cycle of $f \Delta f_\infty$. We also call such a cycle C a *tile* and say that C covers v_∞, v_f, and the corresponding faces. Figure 4.3 (*ii*) depicts a cycle C which covers vertices v_∞ and v_f of the dual. The cycle C is the disjoint union of the infinite face and f. Figure 4.3 (v) depicts a cycle C which corresponds to $v_f v_\infty$. Here $f_\infty \Delta f$ is the union of two cycles and C is one of them.

Given a matching $E' \subset E(H^*)$ and $V' \subset V(H^*)$, with $E' = \{e_1, \ldots, e_\ell\}$ and $V' = \{v_1, \ldots, v_t\}$, a set of tiles $\mathcal{T} = \{C_1, \ldots, C_{\ell+t}\}$ *corresponds* to $E' \cup V'$ if C_i corresponds to e_i for $i = 1, \ldots, \ell$ and $C_{j+\ell}$ corresponds to v_j for $j = 1, \ldots, t$.

In Figure 4.3 (i), cycle C bounds two faces f and g. In Figure 4.3 (ii), the cycle C corresponds to $v_f v_\infty$.

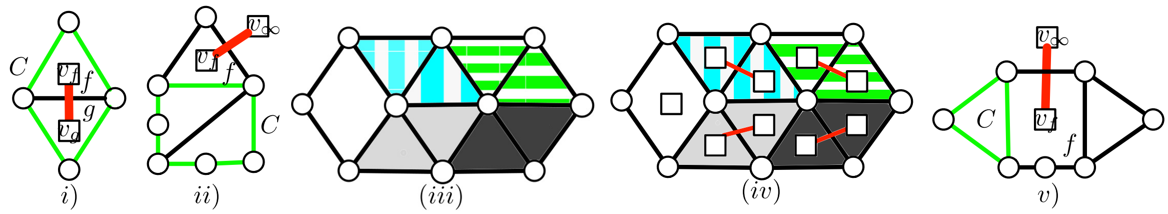


Figure 4.3: Diagrams(i) and(ii) show cycles in green and corresponding edges of the dual graph in red. (i) The red edge corresponds to the symmetric difference of two finite faces. (ii) The red edge corresponds to the symmetric difference of a finite and infinite face. Diagrams (iii) and (iv) show a tiling indicated by the boundaries of the various finite regions in white, light grey, etc and the corresponding matching.

Definition 13. *For a plane graph H, a set $\mathcal{T}$ of tiles is a* pseudo-tiling *if no face of H is covered by more than one tile. If the node v_{v_∞} corresponding to the infinite face of H is not covered by $\mathcal{T}$, we call $\mathcal{T}$ a* tiling.

Let C be an even cycle in G_2^S, and recall that we say that C pays for $\sum_{v\in S} a_v^C$ hit nodes. For an even cycle in a tiling covering two faces, we bound the number of hit nodes it pays for by the number of hit nodes each face pays for.

We will show that a finite face of our graph intersects at most $18/7$ hit nodes on average (over all finite faces). Ideally, we would want to cover all finite faces by a tiling $\mathcal{T}$. The average number of hit nodes that an even cycle of a tiling $\mathcal{T}$ is incident to is $\frac{1}{|\mathcal{T}|}\sum_{C\in\mathcal{T}}|C\cap S'|$. If all finite faces of our graph are covered by $\mathcal{T}$, then since each cycle of $\mathcal{T}$ covers at most 2 faces, $|\mathcal{T}| \geq |F|/2$, where F is set of faces of our graph. As all faces are covered by $\mathcal{T}$, $\frac{1}{|\mathcal{T}|}\sum_{C\in\mathcal{T}}|C\cap S'| = \frac{1}{|\mathcal{T}|}\sum_{C\in F}|C\cap S'| \leq \frac{2}{|F|}\sum_{C\in F}|C\cap S'|$. That is, an even cycle of our tiling is incident to at most twice the number of hit nodes as a finite face of of our graph is on average. So an even cycle of our tiling would be incident to at most $36/7$ hit nodes on average. Alas, tilings covering all faces need not always exist (see Figure 4.5 on page 41). Thus, we try to find a tiling that covers as many finite faces as possible.

Suppose that we find a tiling $\mathcal{T}$ that covers a set $\mathcal{T}_{\mathsf{Faces}}$ of finite faces consisting of α-fraction of the finite faces of our graph. It follows that a face of $\mathcal{T}_{\mathsf{Faces}}$ will be incident to at most $18/7\alpha$ hit nodes on average, and so an even cycle of the tiling $\mathcal{T}$ is incident to at most $36/7\alpha$ hit nodes on average. Intuitively, even faces pay for fewer hit nodes than even cycles containing two faces, so it is good if a tiling contains many even faces. The following definition is a combined measure of how well a tiling covers the faces of our graph and how many even faces are in the tiling. Note that for any tiling $\mathcal{T}$, we can add all even faces not in $\mathcal{T}$ to $\mathcal{T}$ and still remain a tiling. Since we want to cover as many faces as possible, we will focus on tilings containing all even faces.

Definition 14. *Let $\alpha \in (0,1)$. A tiling is* α-quasi-perfect *if it covers all even finite faces, a β-fraction of odd finite faces of G^S, and a ψ-fraction of the finite faces of G^S are even, where*

$$\beta(1-\psi) + 2\psi \geq \alpha \quad . \tag{4.8}$$

We prove the following key result in Subsection 4.2.5.

Theorem 7. *Let H be a 2-compression of some graph G such that H is planar, has an even cycle and contains no pockets. Then H has a 2/3-quasi-perfect tiling that can be found in polynomial-time.*

Note that the "bad" example in Figure 4.1 does not have a 2/3-quasi-perfect tiling. In some sense the purpose of the 2-compression is to have a 2/3-quasi-perfect tiling.

4.2.3 The algorithm in detail

We can now formally state our algorithm. It takes as input a planar graph G with cost function $c : V(G) \to \mathbb{N}$. Let $\mathcal{C}(G)$ denote the set of even cycles of G, and let $\mathsf{opt}(G, c)$ denote the minimum cost of an *even cycle transversal* of G, which is a set of nodes intersecting every cycle in $\mathcal{C}(G)$.

The algorithm maintains a candidate hitting set S at any time. Recall that we let G^S denote the residual graph consisting of the subgraph of G induced by those nodes of $G \backslash S$ that lie on an even cycle. The algorithm first looks for a minimal pocket H of the two compression of the residual graph. G_2^S. The algorithm looks for a 2/3-quasi-perfect tiling $\mathcal{T}_H$ of H and increments the dual variables of blended inequalities of all $C \in \mathcal{T}_H$ until a node v becomes tight or the blended inequality changes. We add all nodes that became tight to S. For each handle pair (Q_1, Q_2), if the set X of nodes that became tight contains nodes in the interior of each handle,choose two nodes $a, b \in X$ with a in the interior of Q_1 and b in the interior of Q_2 and define (a, b) to be a node pair. For a set of nodes X added during the same iteration, nodes in a pair are considered to be added before any node not in a pair. That is we let $w_1, \ldots, w_\ell$ be the nodes of S in the order they were added with the caveat that for nodes X added during the same iteration, any node of X in a pair appears before other nodes of X not in a pair. We then perform reverse delete on $S = \{w_1, \ldots, w_\ell\}$ with the caveat that when considering node w_i for deletion that is in a vertex pair (w_i, w_j), if deleting both w_i, w_j from S maintains feasibility, we do so, otherwise we keep both w_i and w_j.

As we will see, the algorithm returns an even cycle transversal S of G whose cost is at most $(47/7)\mathsf{opt}(G, c)$.

This completes the description of our approximation algorithm for ECT, whose complete pseudo-code is given as Algorithm 4.2.2. Since Algorithm 4.2.2 is a special case of Algorithm 4.2.1, all of the previous theorems we showed for Algorithm 4.2.1 also apply to Algorithm 4.2.2.

4.2.4 Analysis of the approximation ratio

We claim that Algorithm 4.2.2 is a 47/7-approximation for ECT on node-weighted planar graphs.

Algorithm 4.2.2: EvenCycleTransversal(G, c)

Input : A planar graph G with node costs $c : V(G) \to \mathbb{N}$.
Output: An even cycle transversal S of G of cost at most $\frac{47}{7}$ times the cost of an optimal ECT.

$S \leftarrow \emptyset$
while *Residual graph* G^S *contains an even cycle* **do**
  if G^S *contains a cycle* C *with at most* 2 *outside neighbours* **then**
    increase the dual variable y_C for C until a node v becomes tight.
  else
    compute the 2-compression G_2^S of G^S.
  $H \leftarrow$ minimal pocket of G_2^S.
  $\mathcal{T}_H \leftarrow$ a 2/3-quasi-perfect tiling of H.
  Increment dual variables of blended inequalities of all $C \in \mathcal{T}_H$ until a node v becomes tight or the blended inequality changes.
  Add all nodes that became tight to S.
  Denote by X the set of nodes that became tight this iteration.
  for *each handle pair* (Q_1, Q_2) **do**
    if X *contains a node in the interior of each handle* **then**
      choose two nodes $a, b \in X$ with a in the interior of Q_1 and b in the interior of Q_2 and define (a, b) to be a node pair.
$w_1, \ldots, w_\ell \leftarrow$ nodes of S in the order they were added, where for nodes X added during the same iteration, any node of X in a pair appears before others node of X not in pairs.
for $i = \ell$ **downto** 1 **do**
  if w_i *is not part of a pair* **then**
    if $S \backslash \{w_i\}$ *is feasible* **then**
      $S \leftarrow S \backslash \{w_i\}$.
  else
    Let (w_i, w_j) be the pair containing w_i. **if** $S \backslash \{w_i, w_j\}$ *is feasible* **then**
      $S \leftarrow S \backslash \{w_i, w_j\}$.
return S

Fix an input planar graph G with node costs $c_v \in \mathbb{N}$. Consider a set $S \subseteq V(G)$ of nodes and a node $v \in S$. Recall that a cycle C is a pseudo-witness cycle for v with respect to S if $C \cap S = \{v\}$. If C is additionally even, then C is a witness cycle for v. Note that if S is an inclusion-minimal ECT for G, then there is a set W_v of witness cycles for each node in $v \in S$. If the reverse-delete procedure does not delete any node of S, then each node not in a pair has a witness cycle and for each pair, at least one of the nodes in the pair has a witness cycle.

The analyses of the algorithms in [29] and [8] for Subset FVS on planar graphs rely crucially on the fact that, each node of an inclusion-wise minimal solution has a witness cycle. Goemans and Williamson [29] showed that one can find a laminar collection $\mathcal{A}$ of witness cycles. Laminar families $\mathcal{A}$ are well-known to have a natural tree representation [29] where we add an additional root node. Each cycle $A \in \mathcal{A}$ is the child of the smallest cycle $C \in \mathcal{A}$ such that A is contained in the closure of the region of the plane bounded by C. If no such cycle C exists then A is a child of the root. The key argument in [8, 29] is that for each *leaf* cycle C of the laminar family, one can increment the dual variable of at least one face contained in the region defined by C. Further, this dual variable pays only for the hit node that C is a witness of. Using the fact that leaf cycles form a large portion of any laminar set of cycles, this is used to argue that a large portion of the dual variables they incremented pay for a single hit node. An additional bound on how many nodes the other dual variables pay for is proven exploiting the fact that the "debit graph", which will be defined later is planar and hence sparse.

For the ECT problem, however, we do not have laminar witness cycles. Instead, we must extend the analysis of Berman and Yaroslavtsev [8] to find a set of laminar pseudo-witness cycles.

Consider some time $\bar{t}$ during the algorithm when applied to (G, c). Let $S_{\bar{t}}$ be the current hitting set and $G^{S_{\bar{t}}}$ the residual graph. Let $\{\sum_{v \in V(G)} a_v^C \geq 1\}_{C \in \mathcal{L}}$ be the set of inequalities of the dual variables increased during the next iteration. Here, $\mathcal{L}$ will be either a single cycle of $G^{S_{\bar{t}}}$, or a tiling of $G_2^{S_{\bar{t}}}$. We wish to show that the primal increase rate towards the final set S' at time $\bar{t}$, $\sum_{C \in \mathcal{L}} \sum_{v \in S'} a_v^C$ is at most $47/7$ times the dual increase rate $|\mathcal{L}|$.

If the algorithm incremented y_C, where C was a cycle of G for which at most two nodes have outside neighbours, then the inequality we increase is $\sum_{v \in C} x_v \geq 1$. As S' is minimal under reverse-delete, $|C \cap S'| \leq 2$, and hence the primal increase rate $\sum_{v \in S'} a_v^C = |C \cap S'|$ is at most twice the dual increase rate 1.

Otherwise, if the algorithm did not increment y_C, then there is no cycle C of $G^{S_{\bar{t}}}$ such that at most two nodes of C have neighbours outside C. Hence, the set of increased inequalities are the blended inequalities of a tiling $\mathcal{T}_H$ of an inclusion-minimal pocket H of $G_2^{S_{\bar{t}}}$. For a cycle C of $G_2^{S_{\bar{t}}}$, let $\sum_{v \in V(G^{S_{\bar{t}}})} a_v^C \geq 1$ be the blended inequality C (see Equation ⊛).

Recall the definition of the Subset FVS problem, which seeks a minimum-weight node set X which intersects all cycles from $\mathcal{C}_T$, the collection of cycles in G which contain some node from a given set $T \subseteq V(G)$. Observe that each node of S_3' has a witness cycle in $G_3^{S_{\bar{t}}}$;

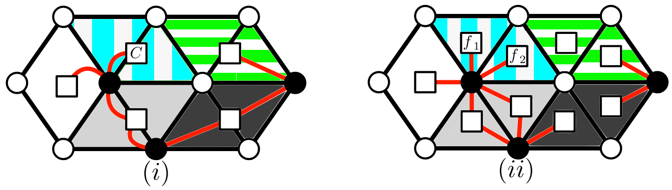


Figure 4.4: Left: A possible debit graph $\mathcal{D}$ with the cycles of the tiling in Figure 4.3. Right: the graph $\mathcal{D}'$ obtained by replacing each cycle with the faces that compose it.

therefore, it is an inclusion-minimal hitting set for the collection $\mathcal{C}_T$ with $T = S'_3$. Goemans and Williamson [29, Lemma 4.2] showed that any inclusion-minimal hitting set for $\mathcal{C}_T$ has a laminar set of witness cycles, which implies that there is a laminar set of pseudo-witness cycles $\mathcal{A}$ for hitting set S'_3.

Proposition 3 ([29, Lemma 4.2 specialized for Subset FVS])**.** *Let G' be a planar graph and let $T \subseteq V(G')$. Let $\mathcal{C}_T$ be the set of cycles of G' containing at least one node of T, and let X be an inclusion-minimal hitting set for $\mathcal{C}_T$. Then there is a laminar set of cycles $\mathcal{A} = \{A_x \mid x \in X\}$, satisfying $A_x \in \mathcal{C}_T$ and $A_x \cap X = \{x\}$.*

Applying Proposition 3 to $G' = G_3$ and $X = T = S'_3$ implies there is a laminar set $\mathcal{A} = \{A_x \mid x \in S'_3\}$ of cycles satisfying $A_x \cap S'_3 = \{x\}$. In other words, $\mathcal{A}$ is a laminar set of pseudo-witness cycles for S'_3. Note that cycles of $\mathcal{A}$ may not be even, hence they may be pseudo-witness cycles for S'_3, but not necessarily witness cycles for nodes of S'_3.

Recall that, during the current iteration, our algorithm incremented the blended inequalities of the cycles in a 2/3-quasi-perfect tiling $\mathcal{T}_H$ of H. Recall that H is an inclusion-minimal pocket of $G_2^{S_{\bar{t}}}$. By abuse of notation, let $\mathcal{T}_H$ denote the corresponding cycles of $G_3^{S_{\bar{t}}}$. Let $\mathcal{D}$ be the debit graph formed using $G_3^{S_{\bar{t}}}$, the cycle set $\mathcal{T}_H$ and hitting set S'_3. Obtain a graph $\mathcal{D}'$ from $\mathcal{D}$ by replacing each even cycle C containing two faces with the two faces that compose it.

For each even cycle C consisting of two faces f_1, f_2, and edge $(C, v) \in E(\mathcal{D})$, graph $\mathcal{D}'$ will have the edge (f_i, v) where v lies on the face f_i in G; see Figure 4.4. In Figure 4.4 (i) is depicted a possible debit graph $\mathcal{D}$, with face nodes depicted as squares, with the cycles of the tiling in Figure 4.3. In (ii) is shown the graph $\mathcal{D}'$ obtained by replacing each face node of our debit graph with the faces that compose it. The face node labelled C in (i) is replaced by two face nodes f_1 and f_2 in (ii). If f_i is not incident to any hit nodes v, we remove f_i from $\mathcal{D}'$. Let $\mathcal{T}_{\mathsf{Faces}(H)}$ be the "face nodes" of $\mathcal{D}'$. Let $\mathcal{F}_{\mathsf{all}(H)}$ denote the finite faces of H. Let $\mathcal{F}_H$ denote the set of finite faces of H that contain a hit node. Observe that $M \cap S'_3 = \emptyset$ for each $M \in \mathcal{F}_{\mathsf{all}(H)} \backslash \mathcal{F}_H$. Now

$$\sum_{M \in \mathcal{T}_H} |M \cap S'_3| \leq \sum_{M \in \mathcal{T}_{\mathsf{Faces}(H)}} |M \cap S'_3|$$
$$\leq \sum_{M \in \mathcal{F}_{\mathsf{all}(H)}} |M \cap S'_3| - |\mathcal{F}_H \backslash \mathcal{T}_{\mathsf{Faces}(H)}| = \sum_{M \in \mathcal{F}} |M \cap S'_3| - |\mathcal{F}_H \backslash \mathcal{T}_{\mathsf{Faces}(H)}| \ . \quad (4.9)$$

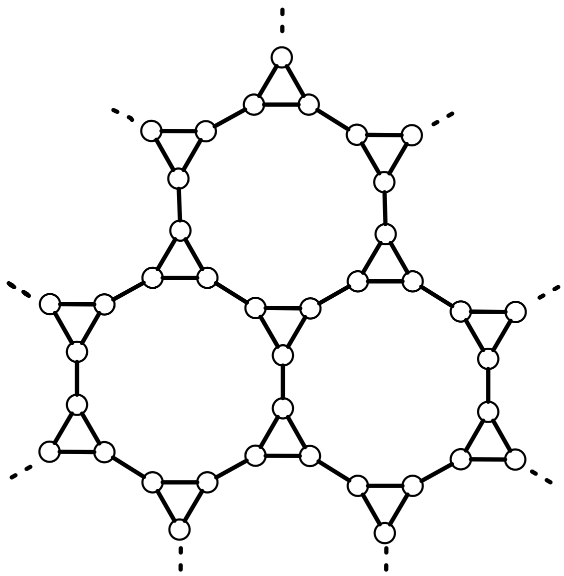


Figure 4.5: A graph consisting of a tessellation of the plane with twice as many triangles as dodecagons. None of the triangles are adjacent, so a maximum tiling covers only the even dodecagons.

The first inequality holds, because for each cycle C consisting of two faces f_1 and f_2 we have $|C \cap S'_3| \leq |f_1 \cap S'_3| + |f_2 \cap S'_3|$. The second inequality holds, because each face of $\mathcal{F}_H$ contains a hit node, and so $|C \cap S'_3| \geq 1$ for each $C \in \mathcal{F}_H$. The last inequality holds, because by definition $|M \cap S'_3| = 0$ for all $M \in \mathcal{F}_{\mathsf{all}(H)} \backslash \mathcal{F}_H$.

If our tiling covers 2/3 of all finite faces, then $|\mathcal{T}_{\mathsf{Faces}(H)}| \leq 2|\mathcal{T}_H|$ and $(2/3)|\mathcal{F}_H| \leq |\mathcal{T}_{\mathsf{Faces}(H)}|$, so $|\mathcal{F}_H| \leq 3|\mathcal{T}_H|$. Alas, one can show that a tiling that covers 2/3 of all finite faces does not always exist; see Figure 4.5. To overcome this impediment, we will show that $|\mathcal{F}_H| \leq 3|\mathcal{T}_H|$ holds for a 2/3-quasi-perfect tiling. Suppose that our 2/3-quasi-perfect tiling covers a b-fraction of the odd faces in $\mathcal{F}_H$, and a c-fraction of the faces in $\mathcal{F}_H$ which are even. Let $\mathcal{F}_{\mathsf{even}(H)}$ be the even finite faces of $\mathcal{F}_H$. Then, as $\mathcal{F}_H \backslash \mathcal{F}_{\mathsf{even}(H)}$ are the odd faces of $\mathcal{F}_H$, and $\mathcal{T}_{\mathsf{Faces}(H)} \backslash \mathcal{F}_{\mathsf{even}(H)}$ are the odd faces covered by our tiling, it holds that $b|\mathcal{F}_H \backslash \mathcal{F}_{\mathsf{even}(H)}| = |\mathcal{T}_{\mathsf{Faces}(H)} \backslash \mathcal{F}_{\mathsf{even}(H)}|$. Simplifying, we get

$$b|\mathcal{F}_H| + (1-b)|\mathcal{F}_{\mathsf{even}(H)}| \leq |\mathcal{T}_{\mathsf{Faces}(H)}| \leq 2|\mathcal{T}_H| - |\mathcal{F}_{\mathsf{even}(H)}| \ .$$

By rearranging, we get $b|\mathcal{F}_H \backslash \mathcal{F}_{\mathsf{even}(H)}| + 2|\mathcal{F}_{\mathsf{even}(H)}| \leq 2|\mathcal{T}_H|$. Noting that $b(1-c)+2c \geq 2/3$, and rearranging once more, yields

$$\frac{2}{3}|\mathcal{F}_H| \leq b|\mathcal{F}_H \backslash \mathcal{F}_{\mathsf{even}(H)}| + 2|\mathcal{F}_{\mathsf{even}(H)}| \leq |\mathcal{T}_{\mathsf{Faces}(H)}| \leq 2|\mathcal{T}_H| \ .$$

Noting that $|\mathcal{F}_{\mathsf{even}(H)}|/|\mathcal{F}_H| = c$ and $b(1-c)+2c \geq 2/3$, we get

$$3|\mathcal{T}_H| \geq \frac{3}{2}(b(1-c)+2c)|\mathcal{F}_H| \geq |\mathcal{F}_H| \ . \tag{4.10}$$

By (4.9), in order to bound $\sum_{M\in\mathcal{T}_H}|M\cap S_3'|$, it suffices to bound $\sum_{M\in\mathcal{F}}|M\cap S_3'|$. To do this, we need Theorem 4.

Let $\mathcal{A}$ be a set of laminar witness cycles for S_3'. If we were to set $\mathcal{R}=\mathcal{F}_H$ (the set of finite faces of H incident to a hit node), then each cycle $A\in\mathcal{A}$ contains a face of $\mathcal{R}$ in its interior, namely any face inside A that is incident to the hit node of S_3' on A. Thus, $S_3', \mathcal{A}$ and $\mathcal{R}$ meet the conditions of Theorem 4.

To recap, we wish to bound the primal increase rate $\sum_{M\in\mathcal{T}_H}\sum_{v\in S}a_v^M$, so we analyze the expression $\sum_{M\in\mathcal{T}_H}|M\cap S_3'|$. Recall that $\sum_{v\in S}a_v^M$ is at most one more than $|M\cap S_3'|$ and $\sum_{v\in S}a_v^M=|M\cap S_3'|$ if M contains no twin edge. We bound $\sum_{M\in\mathcal{T}_H}|M\cap S_3'|$ by looking at the quantity $\sum_{M\in\mathcal{F}_H}|M\cap S_3'|$, because $\mathcal{F}_H$ fits the conditions of Theorem 4. One could then use $|\mathcal{F}_H|\le 3|\mathcal{T}_H|$ (by (4.10)), to bound $\sum_{M\in\mathcal{T}_H}\sum_{v\in S}a_v^M$ in terms of the dual increase rate $|\mathcal{T}_H|$. We will use $3|\mathcal{T}_H|\ge\frac{3}{2}(b(1-c)+2c)|\mathcal{F}_H|$ to obtain a stronger bound.

Let $\mathcal{T}$ be our 2/3-quasi-perfect tiling from Theorem 7. Recall from Definition 14 that the fraction β of odd finite faces that are covered by the tiling, and the fraction ψ of finite faces of H, that are even satisfy $\beta(1-\psi)+2\psi\ge\alpha$. Let $\mathcal{A}$ be a set of pseudo-witness cycles in H for S_3', the corresponding set for the hitting set S' returned by our algorithm. Define $\mathcal{R}=\mathcal{F}_H$. We have that every cycle of $\mathcal{A}$ contains a face of $\mathcal{R}$ in its interior. Thus, $\mathcal{R},\mathcal{A}$ and S_3' satisfy the conditions of Theorem 4. Therefore,

$$\sum_{M\in\mathcal{T}_H}|M\cap S_3'|\le\left(\sum_{M\in\mathcal{F}_H}|M\cap S_3'|\right)-|\mathcal{F}_H\backslash\mathcal{T}_{\mathsf{Faces}(H)}|\le\frac{18}{7}|\mathcal{F}_H|-|\mathcal{F}_H\backslash\mathcal{T}_{\mathsf{Faces}(H)}|\ . \tag{4.11}$$

Note that $\sum_{v\in S}a_v^M\le|M\cap S|$, unless M contains a twin edge. If $M\in\mathcal{T}$ is the disjoint union of two odd faces which share an edge, then M will not contain a twin edge. That is, M can only contain a twin edge if $M\in\mathcal{F}_{\mathsf{even}(H)}$, so M is an even face then. So

$$\sum_{M\in\mathcal{T}_H}\sum_{v\in S}a_v^M\le\sum_{M\in\mathcal{T}_H}|M\cap S|+|\mathcal{F}_{\mathsf{even}(H)}|\le\frac{18}{7}|\mathcal{F}_H|-|\mathcal{F}_H\backslash\mathcal{T}_{\mathsf{Faces}(H)}|+|\mathcal{F}_{\mathsf{even}(H)}|\ . \tag{4.12}$$

Recall that $c=|\mathcal{F}_{\mathsf{even}(H)}|/|\mathcal{F}_H|$ is the fraction of finite faces of $\mathcal{F}_H$ which are even, and that $b=|\mathcal{T}_{\mathsf{Faces}(H)}\backslash\mathcal{F}_{\mathsf{even}(H)}|/|\mathcal{F}_H\backslash\mathcal{F}_{\mathsf{even}(H)}|$ is the fraction of odd finite faces of $\mathcal{F}_H$ covered by our tiling. Note that

$$\begin{aligned}|\mathcal{F}\backslash\mathcal{T}_{\mathsf{Faces}(H)}|&=|\mathcal{F}_H\ \mathcal{F}_{\mathsf{even}(H)}|-|\mathcal{T}_{\mathsf{Faces}(H)}\backslash\mathcal{F}_{\mathsf{even}(H)}|\\&=|\mathcal{F}\backslash\mathcal{F}_{\mathsf{even}(H)}|-b|\mathcal{F}_H\backslash\mathcal{F}_{\mathsf{even}(H)}|\\&=(1-b)(1-c)|\mathcal{F}_H|\ .\end{aligned}$$

We now recall (4.10), by which $3|\mathcal{T}_H|\ge\frac{3}{2}(b(1-c)+2c)|\mathcal{F}_H|$.

Substituting these bounds for $|\mathcal{F}_H|$ and $|\mathcal{F}_H \backslash \mathcal{T}_{\mathsf{Faces}(H)}|$ into (4.12), we obtain

$$\sum_{M\in\mathcal{T}_H}\sum_{v\in S} a_v^M \le c|\mathcal{F}_H| + \frac{18}{7}\left(\frac{2}{b(1-c)+2c}|\mathcal{T}_H|\right) - (1-b)(1-c)|\mathcal{F}_H|$$
$$= \frac{2c}{b(1-c)+2c}|\mathcal{T}_H| + \frac{18}{7}\left(\frac{2}{b(1-c)+2c}|\mathcal{T}_H|\right) - \frac{2(1-b)(1-c)}{b(1-c)+2c}|\mathcal{T}_H| \ .$$

If we maximize the right-hand side factor $\frac{2c}{(b(1-c)+2c)} + \frac{36}{7(b(1-c)+2c)} - \frac{2(1-b)(1-c)}{(b(1-c)+2c)}$ subject to $b(1-c)+2c \ge 2/3$, we obtain that the right-hand side is bounded by $\frac{47}{7}|\mathcal{T}_H|$.

Assuming Theorem 7, each iteration of Algorithm 4.2.2 can be done in polynomial-time. Since each iteration adds a node to out hitting set, our algorithm has at most $|V|$ iterations and thus runs in polynomial-time.

This completes the proof of Theorem 1 modulo the proof of Theorem 7; i.e., the fact that large quasi-perfect tilings can be computed efficiently. The remaining part of this paper will provide details for this remaining task.

4.2.5 Obtaining a 2/3-quasi-perfect tiling

We now show how to find the 2/3-quasi perfect tiling in line 8 of Algorithm 4.2.2. The following result states that the minimal pockets picked by the algorithm have such tilings.

Theorem 7. *Let H be a 2-compression of some graph G such that H is planar, has an even cycle and contains no pockets. Then H has a 2/3-quasi-perfect tiling that can be found in polynomial-time.*

To prove this theorem we will use the following lemma.

Lemma 7. *For any set S, any pseudo-pocket contained in G_2^S contains an even cycle.*

Proof. Informally speaking, the proof will show that any pseudo-pocket without even cycles contains an odd cycle for which only two nodes have outside neighbours; this, however, cannot appear in the 2-compression, as we would have replaced this cycle by an edge in G_2^S. An example is given in Figure 4.6 on the left is depicted the two compression G_2^S of a graph G^S. The pseudo-pocket Q is formed by the black edges, C is a cycle for which only two nodes t, w have outside neighbours. On the right is shown the result of compressing cycle C into the green edge. This is a contradiction as the two compression G_2^S cannot be compressed further.

Suppose, for sake of contradiction, that G_2^S contained a pseudo-pocket Q without even cycles. Since each node of Q is in an even cycle of G_2 and Q contains no even cycle, Q contains exactly two nodes u and v with neighbours outside Q, and each node of Q lies on

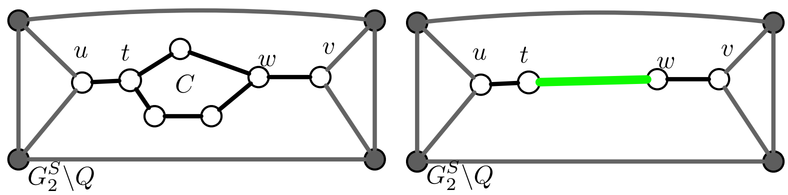


Figure 4.6: Cycle C compressed to edge.

a u-v path of Q. Let B_u and B_v be the blocks of Q containing u and v in the block graph $\mathcal{B}$ of Q, respectively.

If $\mathcal{B}$ is not a path, then there would be some block B_1 (see Figure 4.7) that does not lie on a B_u-B_v path in $\mathcal{B}$, and thus there would be a node of B_1 that would not lie on a u-v path in Q—a contradiction. Hence, $\mathcal{B}$ is a path. Figure 4.7 depicts an example of a graph with a pseudo-pocket Q depicted as the non-gray edges consisting of blocks labelled $B_1, B_2, B_3, B_4, B_u, B_v$, with blocks B_u and B_v containing u and v respectively. Block B_1 depicted in blue contains nodes not on any u-v path, which is a contradiction.

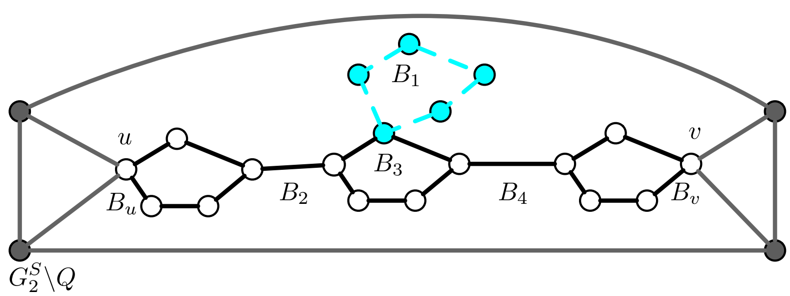


Figure 4.7: Graph Q consisting of blocks labelled $B_1, B_2, B_3, B_4, B_u, B_v$. Block B_1 depicted in blue/dashed contains nodes not on any u-v path, which is a contradiction.

Let B be a block of Q. Suppose for a contradiction that B contains a cycle C and a node v' of C with a neighbour $u' \in V(B)$ outside C. Since v' is not a cut node, there is a path P from u' to $C \backslash v'$. Construct the u'-v' path P' from P by traversing P from u' to the first node w' of $C \backslash v'$ and appending to that a w'-v' path in C. Since Q contains no even cycles, the cycles $P' \cup v'u'$ and C are odd. Then the cycle formed by the edges $E(C) \Delta E(P' \cup v'u')$, that is edges of C or $P' \cup v'u'$, but not both, has length $|E(C)| + |E(P' \cup v'u')| - 2|E(C) \cap E(P' \cup v'u')|$ which is even, and hence a contradiction. Thus if B contains a cycle then it does not contain nodes outside the cycle, or put simply B is a cycle. Since we assume that B contains no even cycles, B is an odd cycle. Thus, the blocks of Q are odd cycles or edges. Since Q contains at least one cycle, there is an odd cycle C'. Since $\mathcal{B}$ is a path, C' contains 2 nodes a and b with neighbours outside C'. However, G_2^S cannot contain such an odd cycle, as that we would have contracted the two a-b paths of C' to parallel edges and then replaced them by a twin edge; see Figure 4.8. This completes the proof. □

For any set S, if G_3^S contained a pseudo-pocket Q without even cycles, then Q was

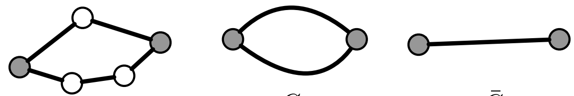


Figure 4.8: Cycle is replaced by an edge in 2-compression.

obtained from a subgraph Q' of G_2^S by subdividing edges. Then Q' would be a pseudo-pocket of G_2^S without even cycles. This contradicts Lemma 7. This shows the following corollary.

Corollary 2. *For any set S, any pseudo-pocket of G_3^S contains an even cycle.*

Recall from Definition 12 and the paragraph afterwards, that a pseudo-tiling of our graph corresponds to the union of a matching of the dual graph and a set of even faces. A tiling corresponds to the union of a matching of the dual graph not containing any edge incident to the infinite face and a set of even finite faces. Under this correspondence, the existence of large pseudo-tilings is a much more natural thing to prove. Let us first formally define a large pseudo-tiling.

Definition 15. *Let $\alpha \in (0,1)$. A pseudo-tiling $\mathcal{T}$ is* α-pseudo-perfect *if it covers all even faces (including the infinite face if it is even) and a β-fraction of the odd faces, and a ψ-fraction of the faces of H are even, where*

$$\beta(1-\psi) + 2\psi \geq \alpha \ . \tag{4.13}$$

We will first prove the existence of large pseudo-perfect pseudo-tilings. We fix an embedding of H. For any multigraph W, let $\mathsf{oc}(W)$ be the number of odd components of W. Since pseudo-tilings correspond to matchings, Tutte's Theorem stated below on the size of a maximum matching will be important to our proof that large pseudo-tilings exist.

Theorem 8 (Tutte's Theorem)**.** *For any graph G, the number of nodes of G which are not covered by a maximum size matching of G is at most*

$$\mathsf{oc}(G \backslash X) - |X| \ . \tag{4.14}$$

for some $X \subset V(G)$. Further, if some node $v \in V(G)$ is covered by every maximum matching of G, then (4.14) holds for some $X \subset V(G)$ containing v.

The main idea of why such large pseudo-perfect pseudo-tilings should exist is that by Tutte's Theorem, the absence of a large pseudo-tiling implies that for some set X of nodes of the dual graph H^*, the set of odd components of $H^* \backslash X$ is large relative to $|X|$.

Construct a new graph H^1 as follows. Start with the graph H^* and add as many edges as possible between nodes of X while preserving planarity and not creating any faces of length 2 (see Figure 4.9). We will show that each odd component of $H^1 \backslash X$ lies in a different face

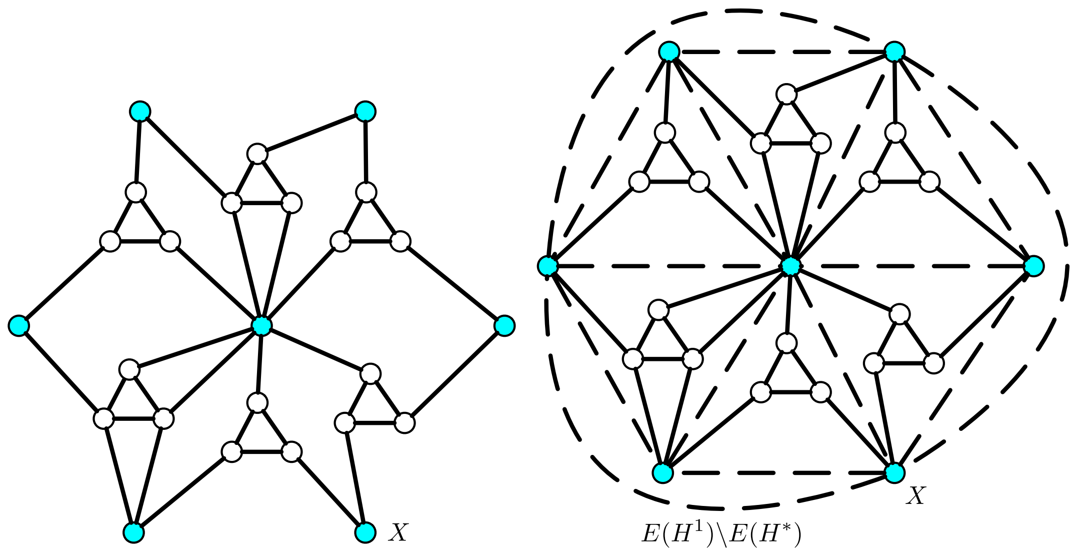


Figure 4.9: The graph H^* with set $X \subset V(H^*)$ (depicted in blue/shaded) on the left. On the right, the graph H^1 obtained from H^* by adding edges (dashed) between X.

of $H^1[X]$ and that H^1 contains at most two faces of length 2. Thus using Euler's formula, $|E(H^1[X])| \leq 3|V(H^1[X])|-4$, $H^1[X]$ does not have too many edges. The crucial observation is that since each odd component of $H^1\backslash X$ lies in a different face of $H^1[X]$, each node $x \in X$ is adjacent to more other nodes of X in H^1 than there are odd components of $H^1\backslash X$ which contain a neighbour of x. By facial region, we mean the region of the plane bounded by a face. We will also show there are at most 2 odd components J_1, J_2 for which at most 2 nodes of X have neighbours in J_i. (See Figure 4.10(ii). There for the odd component J_i there are 2 nodes $u, w \in X$ which have neighbours in J_i. Figure 4.10 (iii) shows the "corresponding dual graph" Q_i which contains only two nodes s and d with neighbours outside Q_i, which gives a contradiction.) We can then show that the number of odd components is at most 2/3 the number of edges of $H^1[X]$ plus $\frac{2}{3}$, which will contradict that the set of odd components is large.

Lemma 8. *Let H be as in Algorithm 4.2.2, that is, H is a minimal pocket of G_2^S. Then H has a 2/3-pseudo-perfect pseudo-tiling.*

Proof. Suppose, for sake of contradiction, that H does not have a 2/3-pseudo-perfect pseudo-tiling. Recall that each edge of the dual graph H^* of H between two nodes which correspond to odd faces in H corresponds to an even cycle of H. Thus, we may think of pseudo-tilings as the union of a set of even faces and a matching on the odd faces. Let Y be the set of even faces of H.

Consider a maximum matching of the odd faces of H, that is, a maximum matching Q of $H^*\backslash Y$. Assume that Q misses a $(1-b)$-fraction of the odd faces (of H), that is, $(1-b) = \frac{q'}{q}$, where q' is the number of odd faces not incident to an edge of the matching, and q is the total number of odd faces. By Theorem 8 applied to $G = H^*\backslash Y$ (by an abuse of notation we also use Y to denote the nodes of H^* which correspond to faces of Y), there is a set of nodes of $V(H^*)\backslash Y$ such that removing these nodes creates a relatively large number of odd components. More precisely, for some $X \subset V(H^*)\backslash Y$ we have

$$(1-b)|V(H^*\backslash Y)| \leq \mathsf{oc}(H^*\backslash(X \cup Y)) - |X| \ . \tag{4.15}$$

Tutte's Theorem also says that if v_∞ is covered by every maximum matching of $H^*\backslash Y$, then we may pick X containing v_∞. By rearranging (4.15), we obtain $|V(H^*)| - |Y| - b|V(H^*\backslash Y)| \leq \mathsf{oc}(H^*\backslash(X \cup Y)) - |X|$. Subtracting $|Y|$ from both sides, we get

$$|V(H^*)| - 2|Y| - b|V(H^*\backslash Y)| \leq \mathsf{oc}(H^*\backslash(X \cup Y)) - |X \cup Y| \ . \tag{4.16}$$

Note that a $|Y|/|V(H^*)|$-fraction of all the faces of H are even, and by definition, a b-fraction of all the odd faces are covered by Q. There is a pseudo-tiling $\mathcal{T}$ corresponding to $Y \cup Q$ under Definition 12 and the paragraph afterwards.

Let $J_1, \ldots, J_\ell$ be the odd components of $H^*\backslash(X \cup Y)$. Let $\hat{H}$ be the graph obtained from H^* by contracting each J_i deleting created parallel edges and loops. For $i = 1, \ldots, \ell$ let j_i be the node obtained by contracting J_i; let $J = \{j_1, \ldots, j_\ell\}$. Let H' be an edge maximal (multi) graph obtained from $\hat{H}$ by adding edges between nodes of $X \cup Y$ while preserving planarity and not creating any faces of length 2.

We will show the following 3 claims.

Claim 8.1. *The inequality $\sum_{i=1}^{\ell} |\delta_{\hat{H}}(j_i)| \geq 3\ell - 2$ holds.*

Claim 8.2. *It holds $|E(H'[X \cup Y])| \leq 3|X \cup Y| - 3$.*

Claim 8.3. *It holds $|E(H') \cap J \times \{x\}| \leq 2|E(H'(X \cup Y)|$.*

We defer the proofs for now and show how to finish the proof given these claims. From Claim 8.1, it follows that $3|J| - 2 \leq \sum_{i=1}^{\ell} |\delta_{H'}(j_i)| = |E(H') \cap J \times (X \cup Y)|$. Thus, by Claim 8.3 and Claim 8.2, it follows that

$$3|J| - 2 \leq 2|E(H'(X \cup Y)| \leq 6|X \cup Y| - 6 \ .$$

So $|X \cup Y| \geq 0.5|J| = \ell$.

Suppose for a contradiction that the pseudo-tiling $\mathcal{T}$ is not 2/3-pseudo-perfect, then (4.13) of Definition 15 is violated, that is,

$$b(1 - (|Y|/|V(H^*)|)) + 2|Y|/|V(H^*)| < 2/3 \ .$$

After simplifying, we obtain $2|Y| + b|V(H^*\backslash Y)| < \frac{2}{3}|V(H^*)|$. Therefore, it holds $\frac{1}{3}|V(H^*)| < |V(H^*)| - 2|Y| - b|V(H^*\backslash Y)|$. Substituting this into the left-hand side of (4.16), we obtain

$$\frac{1}{3}|V(H^*)| < |V(H^*)| - 2|Y| - b|V(H^*\backslash Y)| \leq \mathsf{oc}(H^*\backslash X \cup Y) - |X \cup Y|. \tag{4.17}$$

From $|J| + |X \cup Y| \leq |V(H^*)|$ and $|X \cup Y| \geq \frac{1}{2}|J|$, we get $\frac{2}{3}|V(H^*)| \geq |J|$. Consequently,

$$\frac{1}{3}|V(H^*)| \geq \frac{1}{2}|J| \geq |J| - |X \cup Y| = \mathsf{oc}(H^* \backslash (X \cup Y)) - |X \cup Y|,$$

which contradicts (4.17). Therefore, $\mathcal{T}$ is 2/3-pseudo-perfect. This completes the proof of the lemma. □

We use the notation in the proof of Lemma 8 throughout the rest of this section. Denote by Q_i the subgraph of H induced by the faces of H corresponding to J_i. Given a node $v \in V(H^*)$, denote by $v^* \subset H$ the face of H which v corresponds to. Let v^*_∞ denote the infinite face of H and v_∞ the node of the dual graph H^* corresponding to v^*_∞.

We need the following remark for the next claim.

Remark 4. *If $h\infty \notin J_i$, then the infinite face of Q_i is a cycle.*

Proof. Assume for a contradiction that the infinite face $f_{Q_i\infty}$ of Q_i was not a cycle. Then there is a cycle C of $f_{Q_i\infty}$ for which the region bounded by C contains at least one and not all finite faces of Q_i. Let F be the set of finite faces of Q_i bounded by C. Since C "separates" the faces of F from the other finite faces of Q_i, the vertices of J_i corresponding to faces of F are not reachable from the other vertices of J_i in $H^* \backslash v_\infty$. □

We argue that Q_i cannot be a pseudo-pocket. If Q_i is a pocket, then since Q_i is contained in H, this contradicts the fact that H is an inclusion-wise minimal pocket. Otherwise, Q_i is a pseudo-pocket with no even cycle, which by Lemma 7, cannot appear in the 2-compression of a graph. The following claim shows that a certain condition on j_i implies Q_i is a pseudo-pocket, which implies that such a condition cannot hold for j_i.

Claim 8.4. *Suppose that the degree $|\delta_{\hat{H}}(j_i)|$ of j_i in $\hat{H}$ is at most 2, $h\infty \notin J_i$ and no node of Q_i on the infinite face has a neighbour outside H (see node t in Figure 4.10 (vi)). Then Q_i is a pseudo-pocket.*

We illustrate the previous claim in Figure Figure 4.10 (*i*)-(*iii*). In (*i*), j_i has two neighbours u and w. In (*iii*), Q_i is bounded by the two faces u^* and w^* and only the nodes s and d in Q_i, the two nodes of Q_i which belong to both u^* and w^*, have neighbours outside Q_i.

Proof. Intuitively, the neighbours of J_i in $H^* \backslash J_i$ correspond to the faces of H bound J_i. Informally, if J_i has only 2 neighbours u, w in $H^* \backslash J_i$ and $u, w \neq v_\infty$, then the corresponding faces u^* and w^* bound Q_i, which implies Q_i is a pocket (see Figure 4.10 (*iii*)).

To be precise, suppose that j_i has degree 2 and u, w are the only nodes of $V(H^*) \backslash J_i$ with neighbours in J_i (see Figure 4.10 (*ii*)). Each edge e on the infinite face W_i of Q_i lies on a face a^* of H where a is a node of $H^* \backslash J_i$. The only nodes of $V(H^*)$ that have neighbours in J_i are u, w. Thus, a is either u or w. So the edge e lies on one of the faces u^* or w^*. We may assume that $u \neq h_\infty$ Recall that H contains no pseudo-pockets. Therefore, the intersection of any two finite faces of a subgraph of H with a common edge is a path. Let W_i denote the outside face of Q_i, which by Remark 4 is a cycle. It follows that $A_1 = W_i \cap u^*$ is a path. Let s and d denote the endpoints of A_1. Since each edge of A_1 lies on a face of Q_i and u^*, it does not lie on the face w^*. Thus $A_2 = W_i \cap w^*$ consists of the subgraph of W_i formed by the nodes not in the interior of A_1. Thus A_2 is a path with endpoints s and d. Thus, in the graph H, only nodes s and d of Q_i can have neighbours in $H \backslash Q_i$. Thus if no node of Q_i has a neighbour outside H, then Q_i is a pseudo-pocket of H.

Now suppose that j_i has a single neighbour u. Let W_i denote the outside face of Q_i, which is a cycle. If u^* is the infinite face, then $W_i \cap u^*$ is the infinite face of Q_i, which is a cycle. In this case $Q_i = H$. Suppose that $u \neq v_\infty$. Since each edge lies on two faces, each edge of W_i lies on u^*. Note that faces of graphs are enclosed by closed walks such that each cycle contains at most one node with a neighbour in the walk but outside this cycle. Thus there is exactly a single node $s \in W_i$ for which s contains a neighbour in $u^* \backslash W_i$. This node s is the only node of Q_i with a neighbour outside Q_i (see Figure 4.10 (vii)). Thus Q_i is a pseudo-pocket of H.

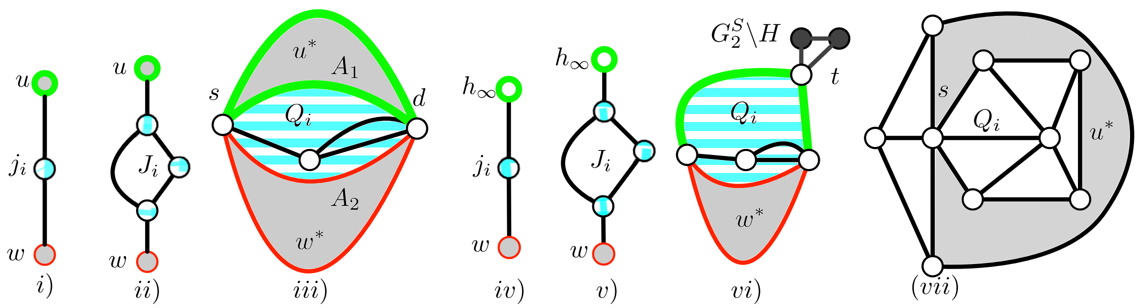


Figure 4.10: Figures $(i), (ii)$, and (iii) show how a degree 2 node in $\hat{H}$, not incident to v_∞, which is shown in (i), corresponds to a pseudo-pocket, which is shown in (iii). Figures $(iv), (v), (vi)$ show the exception when conditions of Claim 8.4 are not satisfied, that is, the node j_i is adjacent to v_∞, and a node t on the infinite face of Q_i has a neighbour outside H. In this case, j_i may not correspond to a pseudo-pocket of G_2^S. The shaded nodes in (vi) are part of $G_2^S \backslash H$. Figure (vii) shows Q_i bounded by a single face u^*. In this case Q_i is also a pseudo-pocket.

This completes the proof of Claim 8.4. □

The proof of Claim 8.1 will use the fact that $H^* \backslash v_\infty$ is connected, which we prove next.

Remark 5. *For the minimal pocket H found by Algorithm 4.2.2, $H^* \backslash v_\infty$ is connected.*

Proof. We show H is 2-connected. Note that if H has a cut node v, then some component of $H \backslash v$, say H^1, contains at most one node with a neighbour outside H. As a consequence, H^1 would be a smaller pocket, which would contradict the fact that H is a minimal pocket.

Thus, H is 2-connected. It is well known that if H is two connected, then the infinite face v_∞^* is a cycle. Thus each face of H lies in the finite region bounded by v_∞^* and thus $H^* \backslash v_\infty$ is connected. □

We now prove Claim 8.1, Claim 8.2 and Claim 8.3.

Proof of Claim 8.1. We distinguish two cases.

1. Some j_i contains only one neighbour in X.
2. Each j_i contains at least 2 neighbours in X.

In Case 1, we claim that for j_a such that $a \neq i$, $|\delta_{\hat{H}}(j_a)| \geq 3$. We consider 3 sub-cases.

Case 1a) $v_\infty \notin J_i$ and the one neighbour that J_i has in X is not v_∞. Then by Claim 8.4 the subgraph of H corresponding to the faces J_i is a pocket, which contradicts our assumption that H is a minimal pocket.

Case 1b) $v_\infty \notin J_i$ and the one neighbour that J_i has in X is v_∞. Then v_∞ separates J_i from the rest of $H^* \backslash h_\infty$. That is, J_i is a component of $H^* \backslash v_\infty$. By Remark 5, $H^* \backslash v_\infty$ is connected, so $J_i = H^* \backslash v_\infty$. Thus there do not exist J_a for $a \neq i$ and the condition is trivially true.

Case 1c) $v_\infty \in J_i$. Then J_i contains all nodes that have neighbours outside H and no other J_a contains a node with a neighbour outside H. Thus for each $a \neq i$, j_a satisfies $|\delta_{\hat{H}}(j_a)| \geq 3$.

In all three sub-cases, J_a does not contains a node with a neighbour outside H. Thus, $|\delta_{\hat{H}}(j_a)| \geq 3$ for all $a \in \{1, \ldots, \ell\} \backslash \{i\}$.

Therefore in Case 1, $\sum_{t=1}^{\ell} |\delta_{\hat{H}}(j_t)| \geq 3\ell - 2$. This completes the analysis of Case 1.

In the Case 2, each J_i contains at least two neighbours in X. If Q_i contains a node v_i with a neighbour outside H in the interior of the shared path between Q_i and the infinite face of H, then v_i has degree 2 in H. Thus, v_i is incident to only faces v_∞ and J_i. So v_i does not lie in any Q_t for $t \neq i$. Since at most two nodes of H have neighbours outside H, there are at most two Q_a that contain a node v_a with a neighbour outside H in the interior of the shared path between Q_a and the infinite face of H. For these Q_a, $|\delta_{\hat{H}}(j_a)| \geq 2$. For every other Q_r, $|\delta_{\hat{H}}(j_r)|$ is at least 3, and thus $\sum_{t=1}^{\ell} |\delta_{\hat{H}}(j_t)| \geq 3\ell - 2$.

In either case, we get $\sum_{t=1}^{\ell} |\delta_{\hat{H}}(j_t)| \geq 3\ell - 2$, as desired. This completes the proof of Claim 8.1. □

Proof of Claim 8.2. First note that if $H'[X \cup Y]$ contains parallel edges e_1, e_2 between two nodes $u, w \in X \cup Y$, then in the planar embedding of H', there are nodes of J that lie in the region bounded by e_1 and e_2. The faces corresponding to u and w in H then bound a pocket unless one of those faces is the infinite face, and the region bounded contains a node with a neighbour outside H. Hence, $H'[X \cup Y]$ contains at most two faces of length 2. Thus, if $|X \cup Y| \geq 2$, then $H'[X \cup Y]$ contains at most two more edges than a planar graph on at least two nodes, that is, at most $2 + 3|X \cup Y| - 5 = 3|X \cup Y| - 3$ edges. Otherwise, $|X \cup Y| \leq 1$, so $|E(H'(X \cup Y)| = 0$, which is at most $3|X \cup Y| - 3$. This completes the proof of Claim 8.2. □

Proof of Claim 8.3. We claim that in any embedding of H' each node $r \in X \cup Y$ does not have two consecutive neighbours in J in the clockwise orientation about r. Assume that some $r \in X \cup Y$ has two consecutive neighbours $j_a, j_b \in J$. Consider the face containing the nodes r, j_a, j_b. Let r' be a neighbour of j_b in this face. Then the edge rr' can be added to H' without creating a face of length 2, which contradicts the fact that H' is an edge maximal multigraph with respect to planarity and not having faces of length 2, that is, no edge can be added to H' while maintaining planarity and not creating any face of length 2.

This implies that, for each $x \in X \cup Y$, it holds

$$|E(H') \cap J \times \{x\}| \leq |E(H') \cap (X \cup Y) \times \{x\}| \ .$$

Summing up over all each $x \in X \cup Y$ we obtain

$$\begin{aligned} |E(H') \cap J \times (X \cup Y)| &= \sum_{x \in X \cup Y} |E(H') \cap J \times \{x\}| \\ &\leq \sum_{x \in X \cup Y} |E(H') \cap (X \cup Y) \times \{x\}| \\ &\leq 2|E(H'(X \cup Y)| \ . \end{aligned}$$

Thus, it holds $|E(H') \cap J \times (X \cup Y)| \leq 2|E(H'(X \cup Y)|$.

This completes the proof of Claim 8.3. □

So let $\mathcal{T}$ be a 2/3-pseudo-perfect pseudo-tiling of H. Let β' be the fraction of odd faces of H which are covered by $\mathcal{T}$, and let ψ' be the fraction of even faces of H. Next, we will show that if $\mathcal{T}$ covers more faces than a maximum tiling of H, then $\mathcal{T}$ satisfies a slightly stronger condition than 2/3-pseudo-perfect, namely, $\beta'(1 - \psi')|V(H^*)| + 2\psi'|V(H^*)| \geq \frac{2}{3}|V(H^*)| + \frac{4}{3}$. Formally, this means:

Lemma 9. *Let H be as in Algorithm 4.2.2, that is, H is a minimal pocket of G_2^S. Suppose that any maximum size pseudo-tiling of H covers the infinite face. Then H has a pseudo-tiling covering a β'-fraction of all odd faces such that*

$$\beta'(1 - \psi')|V(H^*)| + 2\psi'|V(H^*)| \geq \frac{2}{3}|V(H^*)| + \frac{4}{3} \ . \tag{4.18}$$

Proof. To show the statement of Lemma 9, we will need the following slight strengthening of Claim 8.2.

Claim 9.1. *Suppose that any maximum size pseudo-tiling of H covers the infinite face and H admits no $2/3$-quasi-perfect tiling. Then $|E(H'(X \cup Y)| \leq 3|X \cup Y| - 4$.*

Proof of Claim 9.1. Let $\mathcal{T}$, X, Y be as in the proof of Lemma 8. If the infinite face of H is odd, then by assumption, every maximum matching of $H^*\backslash Y$ covers v_∞. Recall that this meant we picked X to contain v_∞. Otherwise, $v_\infty \in Y$. So we may assume that $v_\infty \in X \cup Y$.

By Remark 5, if $X \cup Y = \{v_\infty\}$, then $\text{oc}(H^*\backslash(X \cup Y)) = 1$, which means that either $\beta' = 1$ or $X = \emptyset$.

Suppose that $X \cup Y = \{v_\infty\}$.

In case $\beta' = 1$, then a maximum pseudo-tiling covers all odd faces, and a maximum tiling covers all but at most one odd face.

In case $X = \emptyset$, we get that at most one odd face is not covered by a maximum pseudo-tiling. As $X \cup Y = \{v_\infty\}$, the infinite face is even. Thus, at most one odd face is missed by a maximum tiling.

In either case, a maximum tiling $\mathcal{T}$ misses at most one odd face.

Let β be the fraction of odd finite faces that are covered by $\mathcal{T}$, and ψ the fraction of finite faces of H that are even. As $\mathcal{T}$ misses at most one odd face, it holds $\beta(1-\psi)(|V(H^*)|-1) \geq (1-\psi)(|V(H^*)|-1) - 1$.

First, assume that H contains some even finite face. Then

$$\begin{aligned}\beta(1-\psi)(|V(H^*)|-1) + 2\psi(|V(H^*)|-1) &\geq (1-\psi)(|V(H^*)|-1) - 1 + 2\psi(|V(H^*)|-1) \\ &= (|V(H^*)|-1) - 1 + \psi(|V(H^*)|-1) \\ &\geq (|V(H^*)|-1)\ .\end{aligned}$$

So, $\mathcal{T}$ is $2/3$-quasi-perfect.

Second, suppose that H contains no even finite faces. If H contains a single odd finite face, then it contains no even cycle, which is a contradiction. If H contains exactly two odd finite faces, then since the maximum tiling misses at most one odd finite face, all odd finite faces of H are covered; so, a maximum tiling is 1-quasi-perfect.

If H contains three or more finite faces. Then noting that at most one face of H is not covered by $\mathcal{T}$, it follows that $\psi(|V(H^*)|-1) + \beta(1-\psi)(|V(H^*)|-1) \geq |V(H^*)|-2$. So the inequality $\beta(1-\psi)(|V(H^*)|-1) + 2\psi(|V(H^*)|-1) \geq \psi(|V(H^*)|-1) + \beta(1-\psi)(|V(H^*)|-1) \geq |V(H^*)|-2$ holds, which by algebra yields $\beta(1-\psi) + 2\psi \geq \frac{|V(H^*)|-1}{|V(H^*)|-2}$. As $|V(H^*)|-1 \geq 3$, $\frac{|V(H^*)|-1}{|V(H^*)|-2} \geq \frac{2}{3}$, so the tiling is $\frac{2}{3}$-quasi-perfect.

Henceforth, we assume that $|X \cup Y| \geq 2$.

Suppose first that $|X \cup Y| = 2$. If $H'(X \cup Y)$ contains three parallel edges e_1, e_2, e_3, then it contains three faces of length 2 each bounded by a pair of parallel edges. Since H' contains no parallel edges, the set R_i of nodes lying in the face bounded by the parallel edges $e_i e_{i+1}$ where

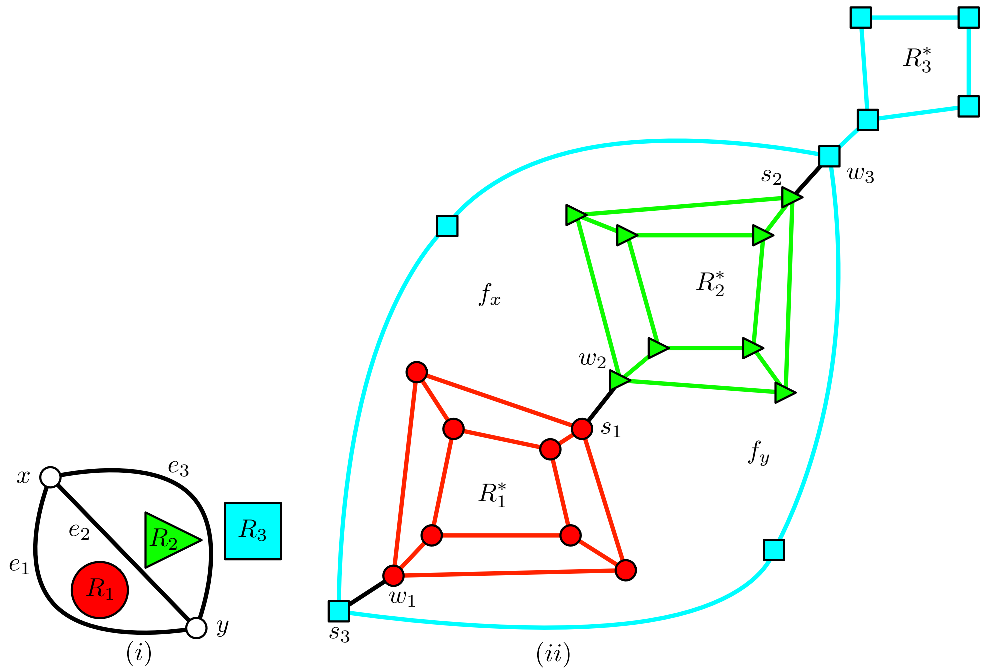


Figure 4.11: In (i) 3 parallel edges e_1, e_2, e_3 bounding node sets R_1, R_2, R_3 in H^* and in (ii) the duals R_1^*, R_2^*, R_3^* in H respectively.

$e_4 = e_1$ is nonempty for $i = 1, 2, 3$. For illustration, see Figure 4.11(i). Let R_i^* be the subgraph of H induced by the nodes that lie on a face which is the dual of a node of R_i. Note that R_i^* lies in a region T_i bounded by the faces f_x and f_y of H which are dual to x and y respectively, see Figure 4.11(ii). Denote by w_i and s_i the two nodes on the boundary of the region T_i belonging to both faces f_x and f_y. Then R_i^* is a pocket unless some node of $V(R_i^*)\backslash\{w_i, s_i\}$ has a neighbour outside H. Note that for $i \neq j$, $V(R_i^*)\backslash\{s_i, w_i\}) \cap (V(R_j^*)\backslash\{s_j, w_j\}) = \emptyset$. Since at most 2 nodes of H have neighbours outside H, at least one R_i has no node with a neighbour outside H and thus is a pocket, which is a contradiction. Hence $H'(X \cup Y)$ contains only two edges, and $|E(H'(X \cup Y))| \leq 2 = 3|X \cup Y| - 4$.

Second, suppose that $|X \cup Y| > 2$. By Euler's formula, any planar graph with nodes $X \cup Y$ without faces of length 2 has at most $3|X \cup Y| - 6$ edges. Suppose that F_i for $i = 1, \ldots, p$ are faces of length 2 in $H'(X \cup Y)$. Let q_i, r_i be the nodes, and e_i, d_i the edges of F_i. Since H' contains no parallel edges, the subgraph R_i of H' lying inside the region bounded by F_i, is nonempty. Let R_i^* denote the subgraph of H induced by the set of nodes that lie on a face which is the dual of a node of R_i. Then each R_i^* lies in a region T_i bounded by two faces f_{q_i} and f_{r_i} which are the dual of q_i and r_i. Let s_i, w_i be the nodes of H on the boundary of T_i that belong to both faces f_{q_i} and f_{r_i}. See Figure 4.12 for an illustration.

If no node of $V(R_i^*)\backslash\{s_i, w_i\}$ has a neighbour outside H, then R_i^* is a pseudo-pocket. Note that for $i \neq j$, the sets $V(R_i^*)\backslash\{s_i, w_i\}$ and $V(R_j^*)\backslash\{s_j, w_j\}$ are disjoint. Hence, if there were three length-2 faces F_1, F_2, F_3, then one of R_1, R_2, R_3 would be a pseudo-pocket, which is a contradiction. Thus, $H'(X \cup Y)$ contains at most two faces of length 2. Therefore, there are two edges $e'_1 e'_2$ that we can remove from $H'(X \cup Y)$ such that $H'(X \cup Y) \backslash \{e'_1, e'_2\}$ contains no face of length 2. Hence, $|E(H'(X \cup Y) \backslash \{e'_1, e'_2\}| \leq 3|X \cup Y| - 6$ and $|E(H'(X \cup Y))| \leq 3|X \cup Y| - 4$.

This completes the proof that $|E(H'(X \cup Y))| \leq 3|X \cup Y| - 4$. □

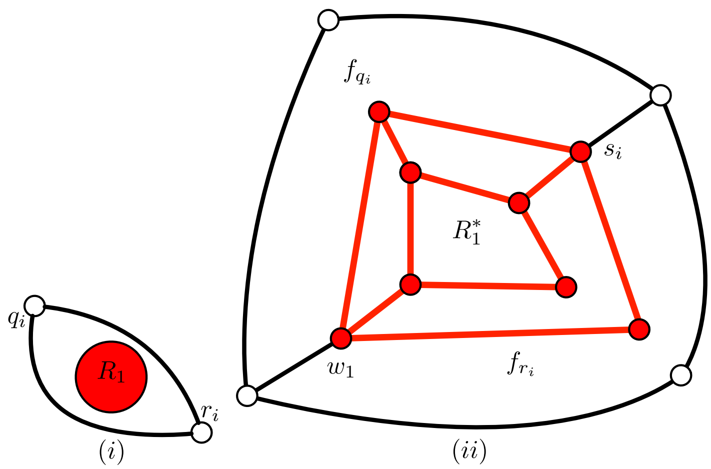


Figure 4.12: One the left, one parallel edge in H' bounding a region containing a set of nodes R_1. On the right is shown the dual graph, in which R_1^* is a pocket.

By assumption $\mathcal{T}$ covers more faces than a maximum tiling of H. Suppose for a contradiction that $\beta'(1-c)|V(H^*)| + 2c|V(H^*)| < \frac{2}{3}|V(H^*)| + 4/3$. Then, by Claim 9.1, it holds $|E(H'(X \cup Y)| \leq 3|X \cup Y| - 4$. Further, by Claim 8.3, it holds $|E(H') \cap J \times \{x\}| \leq 2|E(H'(X \cup Y)|$. Also, by Claim 8.1, we have $\sum_{i=1}^{\ell} |\delta_{\hat{H}}(j_i)| \geq 3\ell - 2$. So in summary, we obtain

$$3|J| - 2 \leq \sum_{i=1}^{\ell} |\delta_{H'}(j_i)| = |E(H') \cap J \times (X \cup Y)| \ .$$

Hence, $3|J| - 2 \leq 2|E(H'(X \cup Y)| \leq 6|X \cup Y| - 8$. Therefore,

$$|J| \leq 2|X \cup Y| - 2 \ . \tag{4.19}$$

Substituting $\psi' = \frac{|Y|}{|V(H^*)|}$ into $\beta'(1-\psi')|V(H^*)| + 2c|V(H^*)| < \frac{2}{3}|V(H^*)| + 4/3$, we obtain $\beta'(1-(|Y|/|V(H^*)|))|V(H^*)| + 2|Y| < \frac{2}{3}|V(H^*)| + \frac{4}{3}$. So $2|Y| + \beta'|V(H^* \backslash Y)| < \frac{2}{3}|V(H^*)| + \frac{4}{3}$, and thus $\frac{1}{3}|V(H^*)| - \frac{4}{3} < |V(H^*)| - 2|Y| - \beta'|V(H^* \backslash Y)|$. Substituting this into the left-hand side of (4.16), we obtain

$$\frac{1}{3}|V(H^*)| - \frac{4}{3} < |V(H^*)| - 2|Y| - \beta'|V(H^* \backslash Y)| \leq \mathsf{oc}(H^* \backslash X \cup Y) - |X \cup Y| \ . \tag{4.20}$$

Multiplying both sides by (-1) and adding to $\mathsf{oc}(H^* \backslash X \cup Y) + |X \cup Y| \leq |V(H^*)|$, we obtain $2|X \cup Y| < \frac{2}{3}|V(H^*)| + \frac{4}{3}$. Simplifying, we obtain $|X \cup Y| < \frac{1}{3}|V(H^*)| + \frac{2}{3}$. Thus,

$$\mathsf{oc}(H^* \backslash X \cup Y) > 2|X \cup Y| - 2 \ . \tag{4.21}$$

This, however, contradicts (4.19). Hence, $\beta'(1-\psi')|V(H^*)|+2\psi'|V(H^*)| \geq \frac{2}{3}|V(H^*)|+4/3$, which completes the proof. ☐

Theorem 9. *Let H be an inclusion-minimal pocket of G_2^S. Then we can obtain 2/3-quasi-perfect tiling of H in polynomial time.*

Proof. We first show that H admits a 2/3-quasi-perfect tiling. Let us show that if some tiling $\mathcal{T}$ is 2/3-pseudo-perfect, then it is 2/3-quasi-perfect. Let β' be the fraction of odd faces of H that are covered by $\mathcal{T}$ and ψ' the fraction of faces of H, that are even. As $\mathcal{T}$ is 2/3-pseudo-perfect, it covers all even faces. Since $\mathcal{T}$ is a tiling, the infinite face is odd. As the number of even finite faces is $\psi'|V(H^*)|$, so $\frac{\psi'|V(H^*)|}{|V(H^*)|-1}$ is the fraction of finite faces of H that are even. $(1-\psi')|V(H^*)|$ is the number of odd faces of H, so $\beta'(1-\psi')|V(H^*)|$ is the number of odd faces of H covered by $\mathcal{T}$. Since the infinite face is odd, $(1-\psi')|V(H^*)|-1$ is the number of odd finite faces. Thus $\frac{\beta'(1-\psi')|V(H^*)|}{(1-\psi')|V(H^*)|-1}$ is the fraction of odd finite faces of H covered by $\mathcal{T}$. Since

$$\begin{aligned}
&\frac{\beta'(1-\psi')|V(H^*)|}{(1-\psi')|V(H^*)|-1}\left(1-\frac{\psi'|V(H^*)|}{|V(H^*)|-1}\right)+\frac{2\psi'|V(H^*)|}{|V(H^*)|-1}\\
&=\frac{\beta'(1-\psi')|V(H^*)|}{(1-\psi')|V(H^*)|-1}(1-\psi')+2\psi'+\left(2-\frac{\beta'(1-\psi')|V(H^*)|}{(1-\psi')|V(H^*)|-1}\right)\left(\psi'-\frac{\psi'|V(H^*)|}{|V(H^*)|-1}\right)\\
&\leq\frac{\beta'(1-\psi')|V(H^*)|}{(1-\psi')|V(H^*)|-1}(1-\psi')+2\psi'\\
&\leq\frac{2}{3},
\end{aligned}$$

it holds that $\mathcal{T}$ is 2/3-quasi-perfect.

If there is a maximum size pseudo-tiling that is also a tiling, then it follows from Lemma 8 that such a tiling is 2/3-quasi-perfect.

Otherwise, if no pseudo-tiling exists, the largest pseudo-tiling is larger than the largest tiling. Let $\mathcal{T}$ be a maximum size pseudo-tiling.

If the infinite face of $\mathcal{T}$ is even, consider the tiling $\mathcal{T}'$ obtained by removing the infinite face from $\mathcal{T}$. Let $\psi^{(1)} := (\psi'|V(H^*)|-1)/(|V(H^*)|-1)$ be the fraction of finite faces of H which are even. As the infinite face is even, β' is the fraction of odd finite faces of H which are covered by $\mathcal{T}'$. It holds that

$$\begin{aligned}
\beta'(1-\psi^{(1)})(|V(H^*)|-1)+2\psi^{(1)}(|V(H^*)|-1) &= \beta'|V(H^*)|(1-\psi')+\psi'|V(H^*)|-1\\
&\geq\frac{2}{3}|V(H^*)|+\frac{4}{3}-1\\
&=\frac{2}{3}(|V(H^*)|-1)\ .
\end{aligned}$$

So $\mathcal{T}'$ is 2/3-quasi-perfect.

If the infinite face is odd, consider the tiling $\mathcal{T}'$ obtained by removing the even cycle covering the infinite face from $\mathcal{T}$. Let $\psi^{(2)} := \psi'|V(H^*)|/(|V(H^*)|-1)$ be the fraction of finite faces of H that are even. At least $\beta'|V(H^*)|-2$ of the finite faces of H are covered by $\mathcal{T}'$ so the fraction β''' of finite odd faces of H that are covered satisfies $b'' \geq (\beta'|V(H^*)|-1)/(1-\psi^{(2)})(|V(H^*)|-1)$. Therefore,

$$\begin{aligned} b''(1-\psi^{(2)})(|V(H^*)|-1)+2\psi^{(2)}(|V(H^*)|-1) &\geq (\beta'|V(H^*)|-1)+2c|V(H^*)| \\ &\geq \frac{2}{3}|V(H^*)|+\frac{4}{3}-1 \\ &= \frac{2}{3}(|V(H^*)|-1)\ . \end{aligned}$$

Hence also in this case, $\mathcal{T}'$ is 2/3-quasi-perfect.

Finally, since a tiling corresponds to the union of a matching and a set of even faces, finding a maximum tiling of H corresponds to finding a maximum matching of the odd finite faces of H. Computing such a maximum matching can be done in polynomial time. □

Chapter 5

Even Cycles in Planar Graphs Have The Erdős-Pósa Property

Recall the Erdős-Pósa property from Section 1.2. In this chapter, we will prove our main combinatorial result, namely, that even cycles in planar graphs satisfy the Erdős-Pósa Property with a linear function $f(k) = 9k$.

Theorem 2. *For $k \in \mathbb{N}$, a planar graph either has as set of at most $9k$ vertices that intersect every even cycle in G, or a set of k vertex disjoint even cycles.*

We will prove this by formulating a primal-dual 9-approximation algorithm for ECT that yields an integral dual solution for unit weights. Note that for $c = \mathbb{1}$, if y is an integral dual solution to ($\mathrm{P}_{\mathrm{ECT}}$), then the support of y, $Q := \{y_C : \quad y_C > 0\}$ is a set of vertex disjoint cycles. Further one can show that $|Q| = \mathbb{1}^t y$. Thus if our primal-dual solution outputs an ECT S of size at most $9(\mathbb{1}^t y)$, S is at most 9 times the size of Q, which will show the Erdős-Pósa property holds. The main idea of the proof will be to show that we can find short even cycles in the 2-compression (see Section 2.1).

This approach requires that the output dual solution y be integral, so the 47/7-approximation for ECT from Chapter 4 does not give an Erdős-Pósa result.

It can be shown that our 9-approximation algorithm runs in polynomial-time. Since the only purpose of our algorithm is to show an Erdős-Pósa result, we will not bother to show that our algorithm runs in polynomial-time.

A key first observation is captured in the following special case of *Kotzig's Theorem* on Light Planar Subgraphs; see the survey by Jendrol and Voss [34, Section 3]). Let us call a node *heavy* if it has degree 6 or more, and *light* otherwise.

Lemma 10. *(Kotzig's Theorem [34, Section 3]) Let G be a planar multigraph (with a fixed embedding) where every node has at least 3 distinct neighbours and that has no face of length 2. Then G contains* *(i) a node of degree at most 10 that is adjacent to a node of degree 3, or*

(ii) two adjacent nodes whose degrees sum to at most 11.

Further, this bound is tight, that is, there exist planar graphs where every node has at least 3 distinct neighbours and that has no face of length 2 such that any two adjacent nodes have degrees summing to 13. The proof of Lemma 10 will provide insights on when tightness can occur.

Proof. Suppose, for sake of contradiction that the statement is not true. Let G be an *edge-maximal counterexample*. In G, every node has degree at least three, and any two neighbouring nodes have degree sum exceeding 11. Edge-maximality implies that if u and v are nodes of combined degree less than 11, then $G + uv$ is not planar.

We assign a *charge* of $6 - d(v)$ to each node v. The sum of node charges is

$$\sum_v (6 - d(v)) = 6|V| - 2|E| \geq 12,$$

using Euler's formula. We now apply a discharging argument [34], and redistribute charges as follows. Each node v with positive charge, splits its charge evenly over its incident edges (and sends these portions to its neighbours).

Abusing notation, we refer to the resulting *redistributed* charge of a node simply as its *charge*. There clearly must be a node v with positive charge. Therefore, v must have a light neighbour. As we assumed G contained no two adjacent nodes whose degrees sum up to at most 11, v must itself be heavy.

Let $u_1, \ldots, u_\ell$ be the neighbours of v in clockwise order in our embedding, and let $u_{\ell+1} = u_1$. We first argue that u_i and u_{i+1} cannot both be light, for any $i \in \{1, \ldots, \ell\}$. For if they were, $u_i u_{i+1}$ is not an edge of G. Hence, the face containing u_i, u_{i+1} and v must contain another node, say w, and w must be heavy. But then adding vw to G preserves planarity; a contradiction to edge-maximality. Thus, v does not have two consecutive light neighbours, and so has at most as many light neighbours as heavy neighbours.

Let ℓ and h be the number of light and heavy neighbours of v. We consider three cases.

Case 1: v has a neighbour u of degree 3.
Node v starts with charge at most $6 - \ell - h$, and it receives charge at most 1 from each light neighbour and none from heavy neighbours. So the charge of v is at most $6 - \ell - h + \ell = 6 - h$. Thus, v has at most 5 heavy neighbours, and hence at most 10 neighbours. Therefore, v and u have combined degree no more than 13; a contradiction.

Case 2: v has no degree-3 neighbours, but at least one neighbour u of degree 4.
A node of degree 4 or more sends a charge of at most 0.5. Therefore, v has charge at most

$$6 - \ell - h + 0.5\ell = 6 - .5\ell + h \leq 6 - .75(h + \ell) \ .$$

The right-hand side is positive, by choice of v, showing that v has degree at most 7; a contradiction.

Case 3: all light neighbours of v have degree 5.
Then v has charge at most $6 - \ell - h + 0.2\ell$, which is at most $6 - 0.9(h + \ell)$. Because the charge is positive, v has degree at most 6. Let u be any light neighbour of v; then u and v are a pair of adjacent nodes whose degrees sum to at most 11. □

Applying Lemma 10 to the dual of a planar graph yields the following consequence.

Corollary 3. *Suppose that G_2 has no face of length 2. Then*

(a) it contains an even cycle that has no twin edges and that has length at most 11*, or*

(b) it contains a cycle C that contains a twin edge and that has length no more than 10*.*

Proof. Since G_2 contains no face of length 2, its dual G_2^* contains no node of degree 2. Applying Lemma 10 to G_2^*, we obtain that G^* contains two adjacent vertices v, w whose degrees sum to at most 11. In G_2, v^*, w^* are faces that share an edge whose lengths sum to at most 11. If either v^* or w^* contains a twin edge, then (b) holds. Otherwise, if either v^*, w^* is an even cycle, then it is an even cycle without twin edges of length at most 11. Otherwise, the cycle formed by their disjoint union $v^* \Delta w^*$ is an even cycle without twin edges of length at most 11. □

Lemma 10 is tight, that is, there exist planar graphs where every node has at least 3 distinct neighbours and that has no face of length 2 such that any two adjacent nodes have degrees summing to 13 (see the example in Figure 5.1), and its proof provides a set of useful features of tight instances. Let us make the following definition.

Definition 16. *For a graph G, a node v is* particular *if it has degree* 10*, its neighbours are alternating light and heavy, all of its light neighbours have degree 3, and all its incident faces have length* 3*.*

Let us classify what the tight instances look like.

Lemma 11. *Let G be a tight instance of Lemma 10. That is, G is a planar multigraph (with a fixed embedding) where every node has at least 3 distinct neighbours and that has no face of length 2 such that any two adjacent nodes have degrees summing to 13. Then G contains a particular node.*

Proof. Since no two adjacent nodes in G have combined degree at most 11, it follows from Lemma 10 that there must be a degree-10 node v neighbouring a degree-3 node u. From the proof of Lemma 10, we then learn that v's light and heavy neighbours alternate, and that v must have at least 4 neighbours of degree exactly 3. Furthermore, tightness implies that G is edge-maximal subject to satisfying the assumptions of Lemma 10, and this has consequences

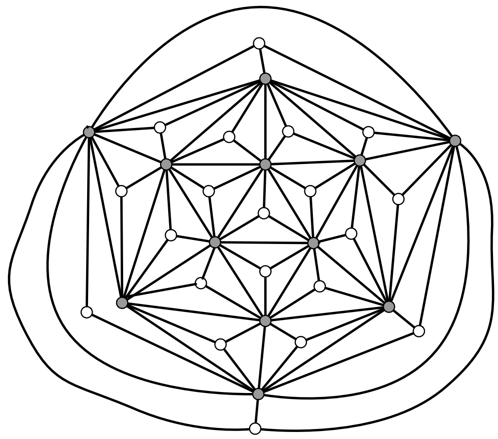


Figure 5.1: A tight example for Lemma 10. Each pair of adjacent nodes has combined degree at least 13.

for the structure of faces incident to v. Consider one such face f, and let v', u' be the two neighbours of v on f. As argued above, one of these two nodes, say v', is light, and the other is heavy. Then we see that v' and u' are neighbours. Otherwise, there exists a heavy neighbour w of v' on f such that $G + wv$ is planar; a contradiction to the edge-maximality of G. This shows that all faces incident to v have length 3. This shows that v is particular. □

Consider the following algorithm: If the residual graph G^S contains an even cycle C such that at most 2 nodes of C have outside neighbours, increase the dual variable y_C for C until a node v becomes tight ($\sum_{v \in C} y_C = w_v$). Otherwise, compute the 2-compression G_2^S of G^S. By Corollary 3 applied on G_2, there is an even cycle C of G_2 with at most 11 pieces and no twin piece, or at most 10 pieces. Increment the dual variable y_C for the blended inequality for C. At the end of the algorithm, perform the reverse-delete step of Algorithm 4.2.2. That is, for nodes added during the same iteration nodes in a pair are considered for deletion only after deleting nodes not in a pair added during that iteration. Since C either has at most 11 pieces and no twin piece, or at most 10 pieces, by Lemma 5 the following holds.

Corollary 4. *For C the cycle selected in Algorithm 4.2.2, the set S of nodes output by Algorithm 4.2.2 satisfies*

$$\sum_{v \in S} a_v^C \leq 11 \ . \tag{5.1}$$

By standard primal-dual arguments this shows our algorithm is an 11-approximation.

5.0.1 Improving to a 10-approximation

Let G^* be the dual of the graph G embedded in the plane. One can improve the 11-approximation by noting that tightness in Corollary 3 only happens in a very specific case.

In particular, it requires tightness of Lemma 10 for G_2^*. That is, suppose that nodes v and u of G_2^* have degrees summing up to 12 or less. Then either the face v, or face u, or the cycle C consisting of the disjoint union of the face v and the neighbouring face u is even. If u or v is an even face of G_2, then we get an even cycle with 9 or fewer pieces, otherwise C has at most 10 pieces, none of which is twin. Else, by Lemma 11, G_2^* has a particular node. Let us define a notion of what the corresponding face of a particular node of G_2^* looks like.

Definition 17. *A face f of G_2 is* particular *if f is adjacent to exactly 5 faces of degree 5 or less, and 5 faces of degree 6 or more, which alternate in clockwise order around f. We also require that at least 4 of the faces of length 5 or less have length 3, and all nodes on f have degree 3.*

We state our above observations in the following lemma:

Lemma 12. *If G_2 has no faces of length 2, one of the following holds:*

(a) G_2 contains an even cycle without twin edges of length at most 10.

(b) G_2 contains a cycle C that contains a twin edge of length no more than 9.

(c) G_2 contains an even cycle without twin edges of length 11 *that contains exactly two faces f_1 and f_2, and f_1 is particular.*

(d) G_2 contains a particular face f that contains a twin edge.

Our new algorithm is the same as the 9 approximation case except that we use Lemma 12 instead of Corollary 3. That is it starts with $S = \emptyset$. If the residual graph G^S contains an even cycle C such that at most 2 nodes of C have outside neighbours, increase the dual variable y_C for C until a node v becomes tight ($\sum_{v \in C} y_C = w_v$). Otherwise, compute the 2-compression G_2^S of G^S. The algorithm for this part looks for an even cycle with 10 or fewer edges in the 1-compression. If this cannot be found, it will look for an even cycle with 9 or fewer edges in G_2^S. If both previous steps fail to find an even cycle, we choose an even cycle of G_2 that is guaranteed by Lemma 12. Denote this cycle by C and construct its blended inequality as before. Increment the dual variable for this inequality as much as possible until a node becomes tight. Add newly tight nodes to S. The algorithm terminates when S is feasible, and runs a reverse-delete step to obtain $\bar{S}$.

Theorem 10. *The algorithm described above is a 10-approximation.*

Proof. To prove this, we will show that the set $\bar{S}$ returned by our algorithm satisfies $\sum_{v \in \bar{X}} a_v^C \leq 10$ whenever $y_C > 0$. If C is a cycle in the 1-compression with 10 or fewer pieces, then the claim is clear. If C satisfies case (a) or (b) of Lemma 12, the 10-approximation follows in the same manner as the previous 11-approximation. In case (c), denote the nodes of f_1 by $v_1, v_3, v_4, \ldots, v_{11}$, and the node of f_2 not on f_1 by v_2. Recall that a piece of G is the preimage of an edge of G_2. In the following remark, for neighbours u and v on cycle C, we let $p(u, v)$ be the corresponding piece in G.

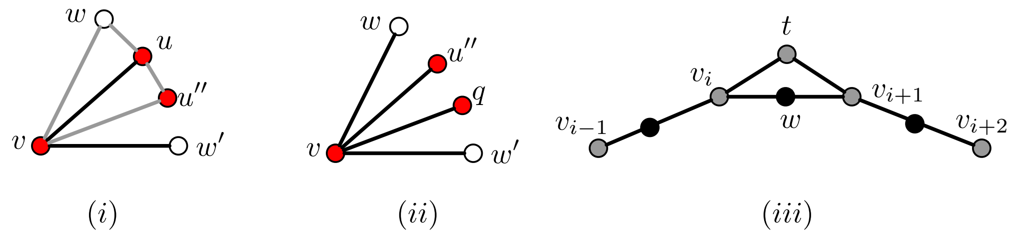


Figure 5.2: In (i), $w'' = w$. Since w, u are consecutive in G, u'' must lie before w'. In (ii), if there is a heavy node between w, w' then there is a non-d neighbour heavy node.

Remark 6. *Let tv_iv_{i+1} be a face of length 3 of G_2 without twin edges, whose projection in G is odd. Suppose that v_i and v_{i+1} have neighbours v_{i-1}, v_{i+1}, t, and v_i, v_{i+2}, t, respectively. Then $p(v_{i-1}, v_i) \cup p(v_i, v_{i+1}) \cup p(v_{i+1}, v_{i+2})$ contains at most two hit nodes.*

For an example, see Figure 5.2 (iii). In the figure the face tv_iv_{i+1} is odd and v_i, v_{i+1} have exactly 3 neighbours. Suppose that the hit node on v_iv_{i+1} had a witness cycle that does not contain any other black hit node. One can check that such a cycle must be $p(t, v_i), p(v_i, v_{i+1}), p(v_{i+1}, t)$.

Proof. Suppose, for sake of contradiction, that $p(v_{i-1}, v_i) \cup p(v_i, v_{i+1}) \cup p(v_{i+1}, v_{i+2})$ contains three hit nodes. Let w be the "middle" hit node. That is if $(p(v_i, v_{i+1}) \backslash v_i) \backslash v_{i+1}$ contains a hit node define w to be that hit node. Otherwise, one of $p(v_{i-1}, v_i)$, $p(v_{i+1}, v_{i+2})$ contains two hit nodes. If $p(v_{i-1}, v_i)$ contains 2 hit nodes, then v_i is a hit node and we define $w = v_i$, otherwise define $w = v_{i+1}$. Then each of $p(v_{i-1}, v_i)$ and $p(v_{i+1}, v_{i+2})$ contains a hit node besides w. Let A_w be a witness cycle for w, and consider the subpath Q_w of A_w containing w and lying in $p(v_{i-1}, v_i) \cup p(v_i, v_{i+1}) \cup p(v_{i+1}, v_{i+2}) \cup p(v_{i+1}, t) \cup p(v_i, t)$ such a path cannot use nodes of $p(v_{i-1}, v_i) \backslash v_i, p(v_{i+1}, v_{i+2}) \backslash v_{i+1}$ and hence both ends of the path are t and so tv_iv_{i+1} is the projection of A_w. But by assumption this cycle is odd in G, which is a contradiction. □

Let v_itv_{i+1} be a triangle face sharing an edge with f_1 that is not on f_2, and let v_{i-1}, and v_{i+2} be the neighbours of v_i and v_{i+1} on C. Remark 6 shows that the three pieces corresponding to the subpath v_{i-1}, v_i, v_{i+1} have at most two nodes from $\bar{X}$. The remaining 8 pieces have at most 8 hit nodes not counting v_{i-1}, v_{i+2}; hence, $\sum_{v \in X} a_v^{C_1', C_2'} x_v \leq 10$. □

5.0.2 Proof of Theorem 6

We now begin to describe our 9-approximation for ECT. For use in the following results, we define several quantities, for each node v of some graph G:

- a_1^v: number of neighbours of degree 3 for which the next neighbour of v in the clockwise order is heavy.

- a_2^v: number of other neighbours of degree 3.
- c_1^v:number of neighbours of degree 4 or 5 for which the next neighbour of v in the clockwise order is heavy.
- c_2^v: number of other neighbours of degree 4 or 5.
- b^v: number of faces of length 4 or more containing v and both a light and heavy neighbour of v in G.
- d: number of heavy neighbours w of v whose such that the next clockwise neighbour of v is also heavy.

See Figure 5.4 for an example. We may omit the superscript v from the above quantities if the node v in use is clear from context.

Lemma 13 ([34, Theorem 3.1])**.** *Any simple, plane multigraph G of minimum degree at least three contains an edge uv such that one of the following holds:*

(1) u has degree 3, and if v has degree more than 6 it has degree at most $2a_1+a_2+2c_1+c_2+d$ with $6-(a_1+a_2)-1.5(c_1+c_2)-d-b>0$.

(2) The sum of degrees of u and v is at most 11.

When G and $v \in V(G)$ are specified we will denote by $a'_1, a'_2, b', d', c'_1, c'_2$ *neighbours* as those neighbours of v counted by a_1, a_2, b, d, c_1, c_2 respectively and a $a'_1, a'_2, b', d', c'_1, c'_2$ *edge* as an edge between v and a $a'_1, a'_2, b', d', c'_1, c'_2$ neighbour respectively. We also say that a face of G^* is an $a'_1, a'_2, b', d', c'_1, c'_2$ face if the face corresponds to an $a'_1, a'_2, b', d', c'_1, c'_2$ node of G respectively.

Remark 7. *In case (1) of Lemma 13 we have that v has degree at most $10-a_2-c_2-2\lfloor 0.5(c_1+c_2)\rfloor-d-2b$*

Proof. Using the same variables as in Lemma 13, we get $6>(a_1+a_2)+1.5(c_1+c_2)+d+b$ so $5 \geq \lfloor (a_1+a_2)+1.5(c_1+c_2)+d+b \rfloor$ so $5-(a_1+a_2)-\lfloor 1.5(c_1+c_2)\rfloor-d-b \geq 0$ which implies $10 \geq 2(a_1+a_2)+2(c_1+c_2)+2\lfloor 0.5(c_1+c_2)\rfloor+2d+2b \geq \deg(v)+a_2+c_2+2\lfloor 0.5c\rfloor+d+2b$. □

Lemma 14. *We can find an even cycle C of G_2 such that either:*

(1) C is the union of 2 faces f_1 and f_2 of G_2 with, f_1 length 3 and at most max$(9, 11-a_2-c_2-2\lfloor 0.5(c_1+c_2)\rfloor-d-b)$ pieces.

(2) C has at most 9 pieces and is disjoint from the interior of any double piece.

(3) C has at most $10-a_2-c_2-2\lfloor 0.5(c_1+c_2)\rfloor-d-b$ pieces. One piece may be double.

(4) C has at most 8 pieces.

Above, we define a_1, a_2, c, d, b, c_2 as in Lemma 13 with $v = f_2$ $u = f_1$ in the dual of G_2.

Proof. We apply Lemma 13 to G_2^*.

In case (1), we find a node u of degree 3 adjacent to a node v of degree at most $10 - a_2 - c_2 - 2\lfloor 0.5(c_1 + c_2)\rfloor - d - b$ in G^* Let f_1, f_2 be the faces corresponding to v, u in G if f_1 or f_2 is even such an even cycle satisfies the conditions of the lemma. Otherwise, $f_1 \cup f_2$ is an even cycle in G with at most $11 - a_2 - 2\lfloor 0.5c\rfloor - d - b$ pieces.

In case (2) if either f_1 or f_2 contains a twin edge, then note that as the lengths of f_1 and f_2 sum to 11 and G_2 contains no faces of length 2, both f_1 and f_2 have length at most 8 and the f_i where $i \in \{1, 2\}$ that contains the twin edge is an even cycle with at most 8 pieces. Otherwise, neither f_1 or f_2 contains a twin edge. If either f_i is even then it is an even cycle with at most 9 pieces and disjoint from the interior of any double piece. If not and both f_1 and f_2 are odd, then $f_1 \Delta f_2$ is an even cycle with at most 9 pieces and disjoint from the interior of any double piece. □

Our 9-approximation does the following: If the residual graph G^S contains an even cycle C such that at most 2 nodes of C have outside neighbours, increase the dual variable y_C for C until a node v becomes tight ($\sum_{v \in C} y_C = w_v$). Otherwise, compute the 2-compression G_2^S of G^S. Find a cycle C of G_2 as guaranteed by Lemma 14. Note from the proof of Lemma 14, it is clear that one can choose C so that C covers 2 faces. Hence such a cycle C can be found in polynomial-time by checking every pair of adjacent faces of G_2. Increment the dual variable y_C for the blended inequality for C. At the end of the algorithm perform a reverse deletion step where for nodes added during the same iteration nodes in a pair appear before nodes not in a pair. See Algorithm 5.0.1 for a complete description of the algorithm.

Theorem 11. *The algorithm in Algorithm 5.0.1 is a 9-approximation.*

Proof. We will consider each case of Lemma 14 separately. Define $c := c_1 + c_2$.

Case (1) If C has 9 or fewer pieces the proof follows from Lemma 5. Otherwise $b = 0$, $c \leq 1$, let $v_1, v_2, .., v_l$ be the cycle C in G_2, with v_1, v_2, v_3 nodes belonging to a triangle face of G_2. We denote:

$$p'(v_i, v_j) = \begin{cases} \cup_{u=i}^{j-1} p(v_u, v_{u+1}) & \text{if } \ i < j \\ \cup_{u=j}^{l} p(v_u, v_{u+1}) \cup (\cup_{u=1}^{i-1} p(v_u.v_{u+1})) & \text{otherwise} \end{cases} \tag{5.2}$$

where $v_{l+i} = v_i$ in the above equation. If $l = 11$, then $11 - a_2 - c_2 - 2\lfloor 0.5(c_1 + c_2)\rfloor - d - b = 11$. Since a_2, $a_2, c_2, d \geq 0$ this implies a_2, $a_2, c_2, d = 0$. So $v_1 v_3 ... v_l$ has length at most $2a_1 + 2c$ and so $a_1 \geq 4$, where a_1 is the number of triangle faces sharing an edge with $v_1 v_3, .., v_l$. In fact, just as in the proof of Theorem 10 $v_1 v_3 ... v_l$ must be particular. Let $v_j t_j v_{j+1}, v_k t_k v_{k+1}, v_r t_r v_{r+1}$ be 3 different a_1' faces of length 3 that have an edge on $v_1 v_3 ... v_l$

Algorithm 5.0.1: AlternativeEvenCycleTransversal(G, c)

Input : A planar graph G with non-negative node-costs c_v, for each $v \in V$.
Output: An even-cycle transversal S of G and solution $y \in \mathcal{C}$ to the dual LP ($\mathrm{D}_{\mathrm{ECT}}$) such that $\sum_{v\in S} c_v \leq 9 \sum_{C\in\mathcal{C}} y_C$.

$S = \emptyset$
while *Residual graph* G^S *contains an even cycle* **do**
  if G^S *contains a cycle* C *with at most* 2 *outside neighbours* **then**
    increase the dual variable y_C for C until a node v becomes tight.
  else
    compute the 2-compression G_2^S of G^S.
  Find an even cycle C satisfying Lemma 14.
  Increment the dual variable y_C for the blended inequality until a node v becomes tight or the blended inequality changes.
  Add all nodes that became tight to S.
Let $w_1, w_2, .., w_t$ be the nodes of S in the order they were added, where for nodes X added during the same iteration, any node of X in a pair appears before any other node of X not in a pair.
for $i = t$ **downto** 1 **do**
  if w_i *is not part of a pair* **then**
    if $S\backslash\{w_i\}$ *is feasible* **then**
      $S \leftarrow S\backslash\{w_i\}$.
  else
    Let (w_i, w_j) be the pair containing w_i.
    if $S\backslash\{w_i, w_j\}$ *is feasible* **then**
      $S \leftarrow S\backslash\{w_i, w_j\}$.
return S

that are not $v_1v_2v_3$. If any 2 distinct $q, r \in \{j, k, r\}$ satisfy $|q - r| \neq 3 \mod l$ then $p(v_{q-1}, v_q) \cup p(v_q, v_{q+1}) \cup p(v_{q+1}, v_{q+2}), p(v_{r-1}, v_r) \cup p(v_r, v_{r+1}) \cup p(v_{r+1}, v_{r+2})$ are disjoint and each contain at most 2 hit nodes. The remaining 5 pieces not containing nodes of $p(v_{q-1}, v_q) \cup p(v_q, v_{q+1}) \cup p(v_{q+1}, v_{q+2}), p(v_{r-1}, v_r) \cup p(v_r, v_{r+1}) \cup p(v_{r+1}, v_{r+2})$ contain at most 5 hit nodes. Thus C contains at most 9 hit nodes.

Finally suppose that $l = 10$, then $a_2 + c_2 + d \leq 1$. The length $l - 1$ of $v_1v_3, .., v_l$ is at most $2a_1 + a_2 + 2c + d$ and so $a_1 \geq 3$. At most one of the a_1' neighbours of $v_1v_3, .., v_l$ of $v_1v_3, .., v_l$ in G_2^* follows an a_2' or c_2' node in the clockwise order and at most one of them is $v_1v_2v_3$ in G_2. Let v_jtv_{j+1} be an a_1' face of $v_1v_3, .., v_l$ in G_2 that does not follow an a_2' or c_2' node, then by Remark 6, $p(v_{q-1}, v_q) \cup p(v_q, v_{q+1}) \cup p(v_{q+1}, v_{q+2})$ contains at most 2 hit nodes and the remaining 7 pieces of C not including nodes of $p(v_{q-1}, v_q) \cup p(v_q, v_{q+1}) \cup p(v_{q+1}, v_{q+2})$ contain at most 7 hit nodes. So C has at most 9 hit nodes.

Thus we can find an a_1 edge $v_r v_{r+1}$ distinct from $v_1 v_3$ that is not proceeded or followed by a c edge. Since one of v_1, v_3 is incident to more than 3 faces of G_2 either one of $v_l v_1, v_3 v_4$ is an a_2 or c edge or one of v_1, v_3 is a b face in G_2^* which implies that neither v_r or v_{r+1} can be a b-face in G_2^*, nor can $v_{r-1} v_r$ be an a_2 edge. Combined with $v_r v_{r+1}$ is an a_1 edge we get that v_r, v_{r+1} both have exactly 3 neighbours and by Remark 6 $p(v_{r-1}, v_{r+2})$ contains at most 2 hit nodes and hence C has at most 9 hit nodes.

Case (2). A 9-approximation follows from Lemma 5.

Case (3). Let G' be obtained from G by deleting the internal nodes of one handle of the special cycle of our blended inequality if our inequality has a special cycle, otherwise set $G' = G$. Denote the 2-compression of G' by G'_2.

Denote $m(u,v) = p(u,v) \cap G'$ and $m'(u,v) = p'(u,v) \cap G'$. Where the edges of C in G_2 are $v_1, v_2, .., v_l$ note that if v_i, v_{i+1}, t_i is a face of length 3 in G'_2 then $v_i v_{i+1}$ is a single piece in G' and remark 6 holds for $m'(v_{i-1}, v_{i+2})$. From $l \leq 2a_1 + a_2 + 2c + d$, one can check that either

(i) $a_1 \geq 4$ $c \leq 1$, $\quad b = d = 0$, or

(ii) $l \leq 9$, $a_2 + 2\lfloor 0.5c \rfloor + d + b \leq 1$.

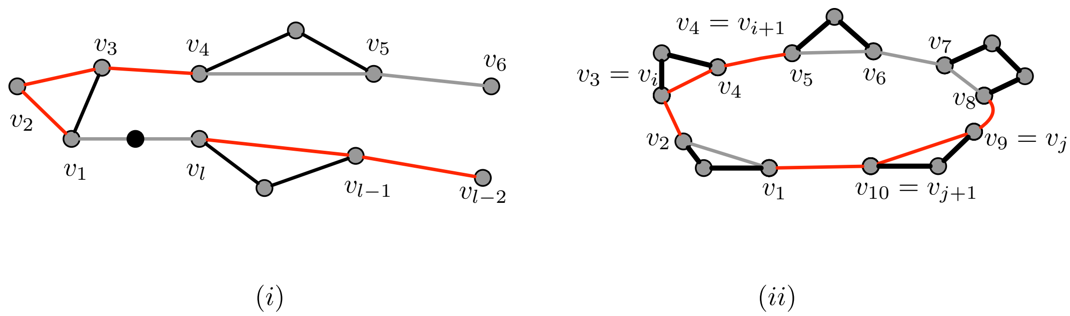


Figure 5.3: (i) shows case 1 and (ii) shows case 3a).

Case (3a). We have $b = \lfloor 0.5c \rfloor = d = a_2 = 0$. Then there are at least 4 a'_1 edges. Since $b = 0$ each endpoint of an a'_1 edge $v_i v_{i+1}$ has exactly 3 neighbours. Thus we can find 2 a'_1 edges $v_i v_{i+1}, v_j v_{j+1}$ with $i \neq j-2, j+2$. Hence $m'(v_{i-1}, v_{i+2})$, $m'(v_{i-1}, v_{i+2})$ each contain at most 2 hit nodes and hence C_1 contains at most 8 hit nodes.

Case (3b). We have $b + d + a_2 + 2\lfloor 0.5c \rfloor \ = 1$ and C_1 has 9 pieces. Here we wish to show there is an a_1 edge with both endpoints having exactly 3 neighbours. If $b = 1$, $a_2 = d = 0$ and from $l \leq 2a_1 + a_2 + c + d$ we get there are at least 3 a_1 edges and the endpoints and thus for at least 1 a_1 edge $v_i v_{i+1}$ v_i, v_{i+1} both (since they are not b nodes) have 3 neighbours. If $d = 1$ then $a_1 \geq 2$ and the endpoints of any a_1 edge have exactly 3 neighbours, since they are not b nodes.

Case (4). Our even cycle C has fewer than 8 pieces, then our inequality will count at most 9 hit nodes. □

We are now ready to complete the proof of Lemma 13.

Proof. If G contains 2 adjacent light neighbours, the proof is clear. Otherwise note that each heavy neighbour of v either follows an a_1 neighbour, a c_1 neighbour, or another heavy edge and are counted by a_1, c_1, d respectively. Thus v has degree at most $2a_1 + a_2 + 2c_1 + c_2 + d$. Let G' be a triangulation of G by drawing edges between heavy nodes. Note that this is possible, since any face of length four or more contains 2 heavy nodes, and we can add an edge between them. Just as Jendrol and Voss [34], by maximality of our triangulation v cannot have 2 consecutive light neighbours in G'.

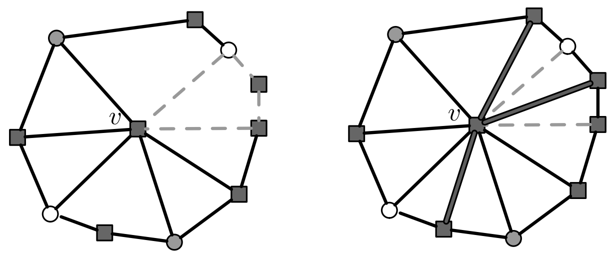


Figure 5.4: Example of a graph G and edges added in obtaining G'. The grey, white, and red nodes are a_2, other light neighbours of v, and heavy nodes, respectively. The grey face is a b-face.

In the proof of Lemma 10 it was shown there exists an edge uv, where v is a node with positive charge such that either

- u has degree 3 and $\deg_{G'}(v) \leq 10$, or
- $\deg_{G'}(u) + \deg_{G'}(v) \leq 11$.

We claim v has degree at least $2a_1 + 2a_2 + 2(c_1 + c_2) + d + b$ in G'. First, let $f_1, f_2, .., f_b$ be the b'-faces of G. Let E' be the set of edges of $\delta_{G'}(v)$ contained in the interior of some f_i and E'' be the d-edges. In any f_i one neighbour u of v is light and the other u' is heavy. Therefore uu' is not an edge of G'. So $\delta_{G'}(v)$ contains an edge in the interior of f_i. It thus suffices to prove that $|\delta_{G'}(v)\backslash E'\backslash E''| \geq 2a_1 + 2a_2 + 2c$. We claim that in $(G'\backslash E')\backslash E''$ there are no consecutive light neighbours of v. Assume for a contradiction that there were 2 such neighbours w, w' with w' the next neighbour after w in $(G'\backslash E')\backslash E''$ in the clockwise order about v. Suppose that there is a d neighbour u' between w and w' in G', let u'' be the last such d neighbour of v before w' in the clockwise order. Then there is a heavy non d -neighbour q of v in G which lies between u'' and w'. So henceforth we assume that no d-neighbours lie between w and w' in the clockwise order about v in G'.

Let u' be the previous neighbour before w' in G'. Assume that $vu \in E'$ so $u' \in f_j$ for some f_j. Note that f_j contains 2 consecutive neighbours of v in G and contains the edge vu in its interior, and u lies between w, w' in the clockwise order. So the neighbours of v in f_j

lie between w, w' in the clockwise order. So f_j must contain vw or vw'. Since one of them is light, that neighbour must be w or w'. Let w'' be the one of w, w' contained in f_j and let u'' be the other neighbour of v in G, then u'' lies between w, w' in the clockwise order, which is a contradiction. Hence there are no 2 consecutive light neighbours. Since each light edge is preceded and followed by a heavy edge hence $|\delta_{G'}(v) \backslash E' \backslash E''| \geq 2a_1 + 2a_2 + 2(c_1 + c_2)$. Calculating charge thus gives us $6 - (a_1 + a_2) - 1.5c - d - b > 0$. □

We are now ready to prove Theorem 2.

Proof. Consider the 9-approximation algorithm defined in Definition 5.0.1 on an unweighted graph. We prove by induction on $|V(G^S)|$ that at anytime during the algorithm, the dual solution is zero-one and the residual cost of any node of the residual graph G^S, where S is the current hitting set, is 1. Since our instance has unit weights, this holds at the beginning of the algorithm. Suppose that this holds after some iteration.

During the next iteration, if algorithm incremented the dual variable y_C of a cycle inequality $\sum_{v \in C} x_v \geq 1$, then since all nodes have residual cost 0 or 1, the nodes on C have residual cost 0, and the algorithm increments y_C by 1. Since prior to this step nodes on C had residual cost 0, y_C was 0 before this step. Hence y_C is 1 after this step.

If the algorithm incremented the dual variable y_C of a blended inequality $\sum_{v \in V} a_v^C x_v \geq 1$, then since all nodes have residual cost 0 or 1 and the maximum of a_v^C is 1, y_C is incremented by 1. Since prior to this step nodes on C had residual cost 0, y_C was 0 before this step. Hence y_C is 1 after this step.

Afterwards, the only nodes $v \in V(G^S)$ that can have residual cost not 1 or 0 are those with $a_v^C = 1/2$. Note that such nodes v are non-branch nodes of some piece P_v whose preimage is an edge of C. Thus $a_u^C = 1$ for branch nodes of P_v and thus nodes with residual cost $1/2$ are not present in the new residual graph. This is because for such nodes v of $V(G^S)$ with residual cost $1/2$ the new hitting set S' contains both endpoints of P_v, and hence there is no even cycle of $G_{S'}$ that uses v. This completes the induction.

Let $Q := \{C : \ y_C > 0\}$ be the cycles in the support of the dual solution y. Let D be the final hitting set output by the algorithm. Since the dual solution output by our algorithm is zero-one, $|Q| = \sum_{C \in \mathcal{C}} y_C \leq 9|D|$. Since $|Q| \leq 9|D|$, for any $k \in \mathbb{Z}$, either D is an even cycle transversal of size at most $9k$ or Q is a set of vertex disjoint even cycles of size at most k. This completes the proof of Theorem 2. □

Chapter 6

Minimum Dominating Set in Graphs of Bounded Arboricity

Recall that in the Minimum Weighted Dominating Set (MWDS) problem, we are given a graph $G = (V, E)$ with weights w_v for all $v \in V$, and wish to find a minimum weight set D of vertices for which each vertex $v \in V$ is either in D, or has a neighbour in D. A graph has arboricity a, if a is the smallest number of forests into which its edge set can be decomposed. It is well known that a graph has arboricity a if and only if each subgraph induced by a subset of vertices $S \subset V$ has at most $a(|S| - 1)$ edges. We use the term a-MWDS to denote MWDS in graphs of arboricity a.

We first introduce some preliminaries and show that the LP rounding algorithm of Bansal and Umboh [4] for the unweighted version of MWDS is no better than a $(2a - 1)$-approximation in the worst case. Afterwards, we present our main result, that MWDS admits an $(a + 1)$-approximation in graphs of arboricity a.

6.0.1 Preliminaries

Let $G = (V, E)$ be a graph, $v \in V$ and $S \subset V$. Denote by $N_H(v) := \{u \in V(H) : vu \in E(H)\}$ the *neighbourhood* of v in graph H, $N_H(S) := \cup_{v \in S} N_H(v)$ and $N(S) := N_G(S)$. Denote by $N[S] := S \cup N(S)$ and $N[v] := N(v) \cup \{v\}$ the *closed neighbourhood* of S and v respectively. Consider the following natural LP relaxation for MWDS and its dual. Given a graph G, for each $v \in V(G)$, the LP has a variable x_v indicating whether v is part of the minimum dominating set.

Algorithm 6.0.1: BansalUmbohMinWeightDominatingSet (G, c)

Input : A graph $G = (V, E)$ with non-negative node-costs w_v, for each $v \in V$.
Output: A dominating set S of G.

1 Solve LP (P_{MWDS}) get solution x^*.
2 $H = \{v \in V : x^*_v \geq \frac{1}{3a}\}$
3 $U := V \backslash N[H]$
4 **return** $H \cup U$

$$\begin{aligned} \min \quad & \sum_{v \in V} w_v x_v & (P_{\text{MWDS}}) \\ \text{s.t.} \quad & x_v + \sum_{u \in N(v)} x_u \geq 1 \quad \forall\, v \in V \\ & x \geq 0 \end{aligned}$$

$$\begin{aligned} \max \quad & \sum_{v \in V} y_v & (D_{\text{MWDS}}) \\ \text{s.t.} \quad & \sum_{u: v \in N(u) \cup \{u\}} y_u \leq w_v \quad \forall\, v \in V \\ & x \geq 0 \end{aligned}$$

The algorithm of Bansal and Umboh [4] first solves the LP (P_{MWDS}). Let x^* be this solution. Then they take the set H of vertices v whose LP value x^*_v is at least $\frac{1}{3a}$ and then taking the remaining set $U := V \backslash N[H]$ of undominated vertices. H has cost at most $3a$ times $\sum_{v \in H} c_v x_v$ where x_v is the LP value for node v in the a-MWDS LP. To bound the cost of U, they then use the fact the graph has arboricity a to obtain an orientation of the graph so that each node has at most a out neighbours. They analyze the size of U relative to $\sum_{v \in V \backslash H} x_v$ (their analysis requires unit weights) by effectively having each node of $V \backslash H$ "pay" its cost to each out neighbour. The final step of their proof is to observe that each node $u \in U$ satisfies $\sum_{v \in \delta_{in}(u)} x_v \geq \frac{1}{3}$, where $\delta_{in}(u)$ are the in neighbours of u. Combining this observation with the fact that each node of $V \backslash H$ pays towards at most a other nodes yields a $3a$-approximation.

Theorem 12. *For each $a > 1$ there exists a graph of arboricity at most a for which the algorithm of Bansal and Umboh in [4] (Algorithm 6.0.1) returns a solution that has size $2a - 1 - o(1)$ times the optimum.*

Proof. Let $a, k \in \mathbb{N}$ and $n = k(a-1)+1$. Construct graph G with vertex set $\{v_0, v_1, .., v_{k(a-1)}\}$ and edge set $\{v_i v_j : 0 \leq i - j \leq a - 1, \ i \neq j\} \cup \{v_i v_j : 0 \leq i + k(a-1) + 1 - j \leq a - 1\}$. See Figure 6.1 for an example of G for $a = 3, k = 4$, the red, thick green and double-stroke black edges form a partition of the edges of G into 3 forests. One can see that $|E(G(U))| \leq a(|U|-1)$ for any $U \subset V$ and hence G has arboricity at most a. For $i > k(a-1)$ and $i < 0$, define $v_i = v_{i \pmod{k(a-1)+1}}$.

Consider the Bansal and Umboh algorithm on G with unit weights. We claim $x = \frac{1}{2a-1}\mathbb{1} = [\frac{1}{2a-1}, \frac{1}{2a-1}, \ldots, \frac{1}{2a-1}]^T$ is the optimal LP solution for the a-MWDS LP on G with unit weights. Note that the LP cost $\sum_{v \in V(G)} x_v$ of x is $\frac{k(a-1)+1}{2a-1}$. Adding up $\sum_{i=-a+1}^{a-1} x_{i+j} \geq 1$

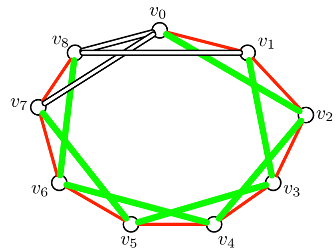


Figure 6.1: Graph G in proof of Theorem 12 for $a = 3, k = 4$.

for $j = 0, 1, .., k(a-1)$ we get

$$(2a-1) \sum_{v \in V(G)} x_v \geq k(a-1) + 1$$

so $\sum_{v \in V(G)} x_v \geq \frac{k(a-1)+1}{2a-1}$ and that equality can only hold if $\sum_{i=-a+1}^{a-1} v_{i+j} = 1$ for all i. It thus follows the unique optimal solution satisfies $x_{v_i} = x_{v_j} \quad \forall i, j \in \{0, 1, 2, .., k(a-1)\}$ and is thus $\frac{1}{2a-1}\mathbb{1}$.

The algorithm of Bansal and Umboh would thus take every vertex of G, which has a cost of $k(a-1)+1$. However, the vertices $\{v_{(2a-1)i} : 0 \leq i \leq \lceil \frac{k(a-1)+1}{2a-1} \rceil - 1\}$ are a solution of cost $\lceil \frac{k(a-1)+1}{2a-1} \rceil \approx \frac{k(a-1)+1}{2a-1}$ for large k. Hence the solution returned by the algorithm is $2a - 1 - o(1)$ times the optimum. □

6.1 Approximation Algorithm

In this section we obtain an approximation algorithm for minimum dominating set in graphs of arboricity a using the primal-dual method. As mentioned in Section 1.3, our analysis actually requires a slightly weaker condition than arboricity a, namely that for our graph G, $|E(G[S])| \leq a|S|$ for any subset $S \subset V(G)$, in other words, our graph has *maximum average degree* at most a.

This result was first published at WAOA 2021 [56].

Theorem 3. *[56] There is a polynomial-time $(a+1)$-approximation algorithm for a-MWDS.*

We use an integer programming formulation and apply the primal-dual method [28].

In each iteration, let S denote the current hitting set. The set of incremented dual variables are all y_v for which v is not in or adjacent to a vertex of S. As standard in the primal-dual method, we apply the standard reverse deletion procedure at the end of our algorithm. The complete description of our algorithm is given in Algorithm 6.1.1.

One can see that each iteration of Algorithm 6.1.1 runs in polynomial time. Since in each iteration the algorithm adds a node to S, there are at most $|V|$ iterations and so the overall run-time is polynomial as well.

Algorithm 6.1.1: MinWeightDominatingSet (G, c)

Input : A graph $G = (V, E)$ with non-negative node-costs w_v, for each $v \in V$.
Output: A dominating set S of G.

$S = \emptyset$
while *S is not a dominating set of G* **do**
    Increment all dual variables y_v for $v \in V \backslash N[S]$ uniformly until a node becomes tight. Add all nodes that became tight to S.
Reverse-Deletion:
    Let $s_1, s_2, .., s_l$ be nodes of S in the order they were added.
    for $t = l$ **downto** 1 **do**
        if $S \backslash \{s_t\}$ *is feasible* **then**
            $S \leftarrow S \backslash \{s_t\}$
return S

6.1.1 Analysis of our algorithm

We now show Algorithm 6.1.1 is an $(a+1)$-approximation. To do so we to show that during each iteration the number of hit nodes our dual variables pay for is at most $a+1$ on average [28].

Theorem 13. *Algorithm 6.1.1 is a polynomial-time $(a+1)$-approximation algorithm on graphs of arboricity a.*

Proof. Let $S^{\mathcal{A}}$ be the set of nodes returned by our primal-dual algorithm. We follow the standard method of analyzing the primal and dual increase rates.

Using Lemma 1 adapted for MWDS, the amount that a dual variable y_v pays for is $|S \cap N[v]|$. It suffices to prove that during any iteration t, the set W_t of nodes whose dual variables are incremented satisfies

$$\sum_{u \in W_t} |S^{\mathcal{A}} \cap N[u]| \leq (a+1)|W_t|. \tag{6.1}$$

We illustrate the intuition of our proof as follows. Graphs G of arboricity a have at most $a|V(G)|$ edges, so the average degree of G is at most $2a$. Suppose that each node of G had degree $2a$. Then for all $u \in V(G)$, $|S^{\mathcal{A}} \cap N[u]| \leq 1 + 2a$ which shows that $S^{\mathcal{A}}$ is a $2a+1$-approximation. This analysis does not work in general because it may be the case that incremented dual variables correspond exclusively to high degree nodes. We will show that because $S^{\mathcal{A}}$ is "minimal" in the sense that Algorithm 6.1.1 performed a reverse deletion step, it follows that although the nodes corresponding to incremented dual variables may have high average degree, they are not adjacent to too many nodes of $S^{\mathcal{A}}$ on average. Minimality of $S^{\mathcal{A}}$ also means that for each node u of $S^{\mathcal{A}}$ there is another node $v \in V(G)$ called a witness, for which $N[v] \cap S^{\mathcal{A}} = \{u\}$. Informally speaking, we show y_v for witnesses v will pay for only one solution node. Intuitively, these y_v which pay for a single solution node help to bring the average number of solution nodes a dual variable pays for down to $a+1$.

Now we make these concepts formal.

Definition 18. *[60] For $v \in S^{\mathcal{A}}$ we call a node $u \in V$ such that $N[v] \cap S^{\mathcal{A}} = \{v\}$ a* witness node *for v.*

Consider some iteration t during the algorithm. Let S_t be the solution set during iteration t, so $\{y_v : v \in V \backslash N[S]\}$ is the set of incremented dual variables. Thus W_t, the nodes corresponding to the dual variables incremented during this iteration, is equal to $V \backslash (S \cup N(S))$. To simplify notation, we will refer to W_t and S_t simply by W and S. Define $S_W^{\mathcal{A}} := W \cap S^{\mathcal{A}}$, the set of solution nodes in W and $S_{N(W)}^{\mathcal{A}} := (S^{\mathcal{A}} \cap N(W)) \backslash W$, the set of solution nodes outside W adjacent to nodes of W. Using the convention introduced in Theorem 1, $S_W^{\mathcal{A}} \cup S_{N(W)}^{\mathcal{A}}$ are the nodes "paid for" during this iteration.

Proposition 1. *At our fixed time t, each node $u \in S_W^{\mathcal{A}} \cup S_{N(W)}^{\mathcal{A}}$ has a witness node in W.*

Proof. Let t' be the time in our reverse deletion procedure when node u was considered for deletion. Let S' be the solution set the algorithm keeps at time t'. Since $u \in W \cup N(W)$, we have $u \notin S$. Since $u \in S^{\mathcal{A}}$, it follows that node u was added to our solution after all nodes of S. Therefore u gets considered in the reverse deletion procedure before the vertices of S, hence $S \subset S'$. Since we did not remove u from our solution, there is a node $w_u \in V$ such that $N[w_u] \cap S' = \{u\}$. Because $u \notin S \subset S'$, therefore $N[w_u] \cap S = \emptyset$. By definition of W, it follows that $w_u \in W$. $\square$

Now we actually argue Inequality (6.1) holds for any time t. For $s \in S_W^{\mathcal{A}} \cup S_{N(W)}^{\mathcal{A}}$ let $p(s)$ be a witness node for s in W. Denote $\hat{S}_W := \{s \in S_W^{\mathcal{A}} : \ p(s) \neq s\}$, the set of nodes in $S_W^{\mathcal{A}}$ that are not their own witness and suppose that $\hat{S}_W = \{s_1, s_2, .., s_l\}$. For brevity, we let $p_t := p(s_t)$.

Let $A := \{p_1, p_2, .., p_l\}$ be the set of witnesses for nodes of $\hat{S}_W$, $\tilde{A} := \{p(s) : \ s \in S_{N(W)}^{\mathcal{A}}\}$ the set of witnesses for $S_{N(W)}^{\mathcal{A}}$ and $B = W \backslash (A \cup S_W^{\mathcal{A}} \cup \tilde{A})$ be the nodes of W that are neither solution nodes nor witnesses (see Figure 6.2). Note that for $v \in A \cup (S_W^{\mathcal{A}} \backslash \hat{S}_W)$, the equality $|S^{\mathcal{A}} \cap N[v])| = 1$ holds as such a node is a witness.

Intuitively, $|N(v) \cap X|$ "counts" the number of edges with one endpoint v and the other in X. In this sense, $\sum_{v \in B} |N(v) \cap S_W^{\mathcal{A}}|$ counts edges between B and $S_W^{\mathcal{A}}$ once, while $\sum_{q \in S_W^{\mathcal{A}}} |(N(q)) \cap S_W^{\mathcal{A}}|$ counts edges within $S_W^{\mathcal{A}}$ twice. In the same line, $2\sum_{v \in B} |N(v) \cap S_W^{\mathcal{A}}| + \sum_{q \in S_W^{\mathcal{A}}} |(N(q)) \cap S_W^{\mathcal{A}}|$ counts each edge of $E(G[B \cup S_W^{\mathcal{A}}]) \backslash E \cap (B \times B)$ twice. Thus

$$\begin{aligned} & 2\textstyle\sum_{v \in B} |N(v) \cap S_W^{\mathcal{A}}| + \sum_{q \in S_W^{\mathcal{A}}} (|(N(q)) \cap S_W^{\mathcal{A}}| + 1) \\ = \ & 2|E(G[B \cup S_W^{\mathcal{A}}])| + |S_W^{\mathcal{A}}| - 2|E \cap B \times B| \\ \leq \ & 2|E(G[B \cup S_W^{\mathcal{A}}])| + |S_W^{\mathcal{A}}|. \end{aligned} \tag{6.2}$$

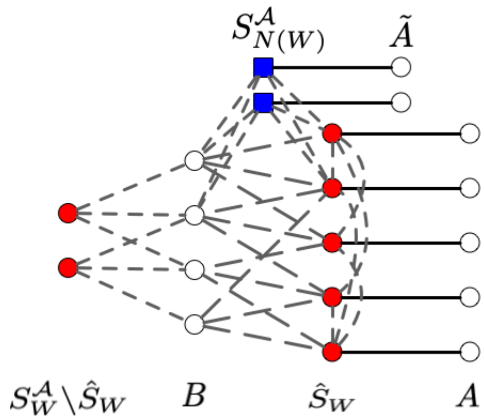


Figure 6.2: Partition of W into $S^{\mathcal{A}}_W \backslash \hat{S}_W, B, \hat{S}_W, \tilde{A}$ and A. Nodes of $S^{\mathcal{A}}_W$ are colored in red. The square blue nodes forming $S^{\mathcal{A}}_{N(W)}$, are solution nodes outside W but adjacent to some node inside W.

Note that each node of $S^{\mathcal{A}}_W \backslash \hat{S}_W$ is its own witness, and hence is not adjacent to other solution nodes. Thus, $\sum_{q \in (S^{\mathcal{A}}_W \backslash \hat{S}_W)} (|N(q) \cap S^{\mathcal{A}}_W| + 1) = |S^{\mathcal{A}}_W \backslash \hat{S}_W|$. This also implies $\sum_{q \in \hat{S}_W} (|N(q) \cap S^{\mathcal{A}}_W| + 1) = \sum_{q \in \hat{S}_W} (|N(q) \cap \hat{S}_W| + 1)$.

Note that $\sum_{q \in (S^{\mathcal{A}}_W \backslash \hat{S}_W)} |N(q) \cap \hat{S}_W|$ counts each edge of $G[\hat{S}_W]$ twice, so $\sum_{q \in \hat{S}_W} (|N(q) \cap \hat{S}_W| + 1) = 2|E(G[\hat{S}_W])| + |\hat{S}_W|$.

Thus,

$$\begin{aligned} & \textstyle\sum_{q \in S^{\mathcal{A}}_W} (|N(q) \cap S^{\mathcal{A}}_W| + 1) \\ = \; & \textstyle\sum_{q \in \hat{S}_W} (|N(q) \cap \hat{S}_W| + 1) + \sum_{q \in (S^{\mathcal{A}}_W \backslash \hat{S}_W)} (|N(q) \cap S^{\mathcal{A}}_W| + 1) \\ = \; & 2|E(G[\hat{S}_W])| + |\hat{S}_W| + |S^{\mathcal{A}}_W \backslash \hat{S}_W|. \end{aligned} \tag{6.3}$$

Adding (6.2) and (6.3) and dividing by 2 we get

$$\begin{aligned} & \textstyle\sum_{v \in B} |N(v) \cap \hat{S}_W| + \sum_{q \in S^{\mathcal{A}}_W} (|N(q) \cap S^{\mathcal{A}}_W| + 1) \\ \leq \; & \tfrac{1}{2}(2|E(G[B \cup S^{\mathcal{A}}_W])| + 2|E(G[\hat{S}_W])| + 2|\hat{S}_W|) + |S^{\mathcal{A}}_W \backslash \hat{S}_W|. \\ = \; & |E(G[B \cup S^{\mathcal{A}}_W])| + |E(G[\hat{S}_W])| + |S^{\mathcal{A}}_W|. \end{aligned} \tag{6.4}$$

To recap, we wish to bound $\sum_{v \in W} |N[v] \cap S^{\mathcal{A}}|$. Note that for $v \in W$, if $v \in B$, then $|S^{\mathcal{A}} \cap N[v]| = |N(v) \cap \hat{S}_W|$. For $v \in \hat{S}_W$, $|S^{\mathcal{A}} \cap N[v]| = |N(v) \cap \hat{S}_W| + 1$.

We will next bound $\sum_{v \in B \cup S^{\mathcal{A}}_W} (|(N[v]) \cap (S^{\mathcal{A}}_{N(W)} \cup S^{\mathcal{A}}_W)| - |(N[v] \cap S^{\mathcal{A}}_W|)$, which will get us a bound on $\sum_{v \in B \cup S^{\mathcal{A}}_W} |N[v] \cap S^{\mathcal{A}}|$, the number of nodes dual variables indexed by $B \cup S^{\mathcal{A}}_W$ pay for. Using bounded arboricity we will bound the number of edges in subgraphs of G. Afterwards we will bound $\sum_{v \in A \cup \tilde{A}} |N[v] \cap S^{\mathcal{A}}|$, which will bound $\sum_{v \in W} |N[v] \cap S^{\mathcal{A}}|$ completing the analysis.

Recall that $S^{\mathcal{A}}_{N(W)} := (S^{\mathcal{A}} \cap N(W)) \backslash W$. Let us account for the amount nodes of W pay towards $S^{\mathcal{A}}_{N(W)}$, their neighbours outside W which are solution nodes. Each node of $\tilde{A}$ only pays towards the node of $S^{\mathcal{A}}_{N(W)}$ it is a witness of.

Note that $\sum_{v \in B \cup S^{\mathcal{A}}_W} (|(N[v] \cap (S^{\mathcal{A}}_{N(W)} \cup S^{\mathcal{A}}_W)| - |(N[v] \cap S^{\mathcal{A}}_W|)$ counts every edge of $G[B \cup S^{\mathcal{A}}_W \cup S^{\mathcal{A}}_{N(W)}]$ not in $G[B \cup S^{\mathcal{A}}_W]$, thus the following equality holds.

$$\begin{aligned} &\textstyle\sum_{v \in B \cup S^{\mathcal{A}}_W} (|(N[v] \cap (S^{\mathcal{A}}_{N(W)} \cup S^{\mathcal{A}}_W)| - |(N[v] \cap S^{\mathcal{A}}_W|) \\ &= |E(G[B \cup S^{\mathcal{A}}_W \cup S^{\mathcal{A}}_{N(W)}])| - |E(G[B \cup S^{\mathcal{A}}_W])|. \end{aligned}$$

Adding this to (6.4) we obtain the following.

$$\begin{aligned} & \textstyle\sum_{v \in B \cup S^{\mathcal{A}}_W} |N[v] \cap S^{\mathcal{A}}_W| + \\ & \textstyle\sum_{v \in B \cup S^{\mathcal{A}}_W} (|N[v] \cap (S^{\mathcal{A}}_{N(W)} \cup S^{\mathcal{A}}_W)| - |(N[v] \cap S^{\mathcal{A}}_W|) \\ \leq\ & |E(G[B \cup S^{\mathcal{A}}_W])| + |E(G[\hat{S}_W])| + |S^{\mathcal{A}}_W| + |E(G[B \cup S^{\mathcal{A}}_W \cup S^{\mathcal{A}}_{N(W)}])| - |E(G[B \cup S^{\mathcal{A}}_W])|. \end{aligned}$$

This simplifies to $|E(G[B \cup S^{\mathcal{A}}_W \cup S^{\mathcal{A}}_{N(W)}])| + |E(G[\hat{S}_W])| + |S^{\mathcal{A}}_W|$. So the following inequality holds.

$$\sum_{v \in B \cup S^{\mathcal{A}}_W} |(N[v] \cap (S^{\mathcal{A}}_{N(W)} \cup S^{\mathcal{A}}_W)| \leq |E(G[B \cup S^{\mathcal{A}}_W \cup S^{\mathcal{A}}_{N(W)}])| + |E(G[\hat{S}_W])| + |S^{\mathcal{A}}_W|. \quad (6.5)$$

We now have a bound on the number of nodes that dual variables for $B \cup S^{\mathcal{A}}_W$ pay for. Let us bound the right hand side of the above equation. From the fact that G has arboricity a, $|E(G[X])| \leq a|X|$ for any $X \subset V$ (in fact this weaker condition is sufficient for our analysis. The condition $|E(G[X])| \leq a|X|$ which we require, is also equivalent to G has maximum average degree a. Maximum average degree is defined as $\max_{H \text{ subgraph of } G} \frac{|E(H)|}{|V(H)|}$.). Thus, the right hand side of the above inequality is at most

$$\begin{aligned} & a(|B| + |S^{\mathcal{A}}_W| + |S^{\mathcal{A}}_{N(W)}|) + a|\hat{S}_W| + |S^{\mathcal{A}}_W| \\ \leq\ & a|B| + (2a+1)|\hat{S}_W| + (a+1)|S^{\mathcal{A}}_W \backslash \hat{S}_W| + a|S^{\mathcal{A}}_{N(W)}|. \end{aligned} \quad (6.6)$$

So combining inequalities (6.5) and (6.6), we obtain the following.

$$\sum_{v \in B \cup S^{\mathcal{A}}_W} |(N[v] \cap (S^{\mathcal{A}}_{N(W)} \cup S^{\mathcal{A}}_W)| \leq a|B| + (2a+1)|\hat{S}_W| + (a+1)|S^{\mathcal{A}}_W \backslash \hat{S}_W| + a|S^{\mathcal{A}}_{N(W)}|. \quad (6.7)$$

We also know that each node $v \in A \cup \tilde{A}$, has exactly one hit node in its closed neighbourhood, that is, $|S^{\mathcal{A}} \cap N[v]| = 1$.

Also note that $|A| = |\hat{S}_W|$, $|\tilde{A}| = |S^{\mathcal{A}}_{N(W)}|$, for each $v \in W$ we have $S^{\mathcal{A}} \cap N[v] = (S^{\mathcal{A}}_W \cup S^{\mathcal{A}}_{N(W)}) \cap N[v]$ and W is the disjoint union of $A, \tilde{A}, B$ and $S^{\mathcal{A}}_W$.

Putting these observations together, we get the total amount the dual variables "pay" for is $\sum_{v \in W} |S^{\mathcal{A}} \cap N[v]|$, which satisfies the following chain of inequalities.

$$\begin{array}{rl} & \sum_{v \in W} |S^{\mathcal{A}} \cap N[v]| \\ = & \sum_{v \in B} |N(v) \cap S^{\mathcal{A}}| + \sum_{q \in S_W^{\mathcal{A}}} (|N(q) \cap S^{\mathcal{A}}| + 1) + \sum_{q \in A} (|N(q) \cap S^{\mathcal{A}}| + 1) \\ & + \sum_{q \in \tilde{A}} (|N(q) \cap S^{\mathcal{A}}| + 1) \\ = & \sum_{v \in B} |N(v) \cap S^{\mathcal{A}}| + \sum_{q \in S_W^{\mathcal{A}}} (|N(q) \cap S^{\mathcal{A}}| + 1) + |A| + |\tilde{A}|. \end{array}$$

From (6.7), this is at most

$$a|B| + (2a+1)|\hat{S}_W| + (a+1)|S_W^{\mathcal{A}} \backslash \hat{S}_W| + a|S_{N(W)}^{\mathcal{A}}| + |A| + |\tilde{A}|.$$

Noting that $|A| = |\hat{S}_W|$ and $|\tilde{A}| = |S_{N(W)}^{\mathcal{A}}|$, the previous line is at most the following.

$$\begin{array}{rl} & a|B| + (2a+2)(|\hat{S}_W|) + (a+1)|S_W^{\mathcal{A}} \backslash \hat{S}_W| + (a+1)|\tilde{A}| \\ \leq & a|B| + (a+1)(|\hat{S}_W| + |A|) + (a+1)|S_W^{\mathcal{A}} \backslash \hat{S}_W| + (a+1)|\tilde{A}| \\ \leq & (a+1)|W|. \end{array}$$

Since the set of incremented dual variables is $\{y_v : v \in W\}$, by Theorem 1, this shows our algorithm is an $(a+1)$-approximation.

$\square$

Chapter 7

Open Problems

In this chapter we identify a number of questions related to ECT and more generally graph transversals that we could not resolve in this thesis. These questions provide potential research topics for future research in graph transversals.

Question 1. *Can we improve the approximation ratio of 47/7 for ECT in planar graphs?*

One immediate idea is to use the concept of 3 pockets in [8].

Suppose that we replaced the step of finding a minimal pocket H of G_2^S with finding a minimal pocket or 3-pocket of G_2^S and we were always able to find a $\frac{2}{3}$-quasi-perfect tiling of H. Then one could use a strengthening of Theorem 4 with a bound of $\sum_{M \in \mathcal{R}} |M \cap S| \leq 2.4|\mathcal{R}|$ to get Equation (7.1).

$$\sum_{M \in \mathcal{M}} |M \cap S_3| \leq \Big(\sum_{M \in \mathcal{F}_{\mathsf{all}}} |M \cap S_3| \Big) \leq 2.4|\mathcal{F}_{\mathsf{all}}| \leq 7.2|\mathcal{M}| \tag{7.1}$$

A slightly tighter analysis would yield a 31/5-approximation.

However this immediate idea does not work because we show in Figure 7.1 that 3-pockets do not always have $\frac{2}{3}$-quasi-perfect tilings.

Another method is to look at where the analysis is not quite tight. For instance, when we estimate $\sum_{M \in \mathcal{M}} |M \cap S_3|$ using $\sum_{M \in \mathcal{M}_{\mathsf{Faces}}} |M \cap S_3|$ in Equation (4.9) we get equality only if for each even cycle of $\mathcal{M}$, its faces do not share any hit nodes. It may be the case that we cannot obtain any α-quasi-perfect tiling for any $\alpha > 2/3$ (see Figure 7.1). However, we may need our pocket H to have a certain structure for equality to hold. We do not know if Theorem 4 is tight for our choice of $\mathcal{A}, S_3$, and $\mathcal{R}$ on any instance where we cannot obtain an α-quasi-perfect tiling for any $\alpha > 2/3$. We also estimated $\sum_{v \in S} a_v^C$ as $|C \cap S_3| + 1$ for all even faces C. However, we have $\sum_{v \in S} a_v^C \leq |C \cap S_3|$ unless C contains a twin edge. Thus we can obtain a better approximation if most of our even faces do not contain twin edges.

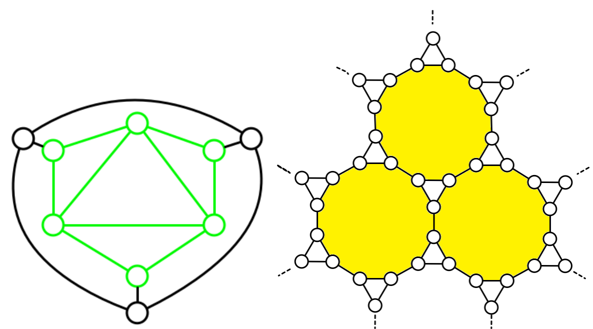


Figure 7.1: On the left, a 3-pocket is shown in green. One can check that the 3-pocket has no $\frac{2}{3}$-quasi-perfect tiling. On the right is shown a graph consisting of a tessellation of the plane with twice as many triangles as dodecagons. None of the triangles are adjacent, so a maximum tiling covers only the even dodecagons.

Our 47/7-approximation can only be tight if all previously mentioned parts of the approximation can be tight simultaneously. Thus looking at whether they can all be tight simultaneously could lead to improvements.

A different approach is to use the so-called "extended sparsity inequalities" of Fiorini et al. [24]. A *diamond* is a subdivision of the graph formed by 3 parallel edges. Fiorini et al. [24] showed that the "natural LP" for diamond transversal has an integrality gap of $\theta(\log n)$. They obtain a 9-approximation algorithm for diamond transversal by adding extended sparsity inequalities [24]. Note that if a set of vertices is an even cycle transversal, then it is a diamond transversal. Thus for any even cycle transversal S, the characteristic vector $\mathbb{1}^S$ also satisfy the extended sparsity inequalities. One naturally wonders how well can we approximate ECT in planar graphs using the LP of Fiorini et al. [24] with their extended sparsity inequalities?

Question 2. *How well can other graph transversal problems be approximated?*

We give the following example. For $c \in \mathbb{Z}$, a *c-pumpkin model* is a graph containing 2 vertices with c parallel edges as a minor [35]. A natural idea is to find a set of pumpkins to increment that are "small" similarly to how we found an even cycle with few pieces in Lemma 3. We believe this idea may be possible for certain small c.

One naturally considers graph transversal problems in directed graphs.

Question 3. *How well can we approximate directed even cycles in planar graphs?*

Question 4. *Can we improve the 2.4-approximation of [8] for DFVS in planar graphs?*

To explain our ideas on Question 4, we explain first how the approach in [8] works, and then why a natural generalization doesn't work.

The work in [8] extends the work of [29] which gives a simple primal-dual 3-approximation by iteratively incrementing the face-minimal cycles. The tight case only happens when the graph contains a proper subgraph U such that U contains a cycle and U has at most 2 nodes that have outside neighbours. That is, U is a pocket. They propose a new algorithm that increases all face-minimal cycles in a pocket. This algorithm achieves an approximation factor of 18/7. Reference [8] improves this algorithm by noting that the pocket oracle of [29] is only tight when there is a proper subgraph containing at least 3 cycles for which at most 3 vertices have outside neighbours. They call this subgraph a 3-pocket.

We generalize this idea with the following definition.

Definition 19. *For $p \geq 3$ and $r \geq 0$, a (p, r)-pocket for a planar graph G(V; E) and a cycle collection $\mathcal{C}$ is a set $U \subset V$ such that:*
1. The set U contains at most p nodes with neighbours outside U. (We call these contact nodes.)
2. The induced subgraph $G^S[U]$ has at least r faces in $\mathcal{C}$.

Given graph G, a collection of cycles $\mathcal{C}$ of G and $S \subset V(G)$, define the *residual graph* G^S as the subgraph of $G \backslash S$ induced by those vertices which belong to a cycle of $\mathcal{C}$ not containing a node of S. It can be shown that if at any time during the algorithm of [8], the residual graph does not contain a (q, r) pocket for any $q, r \geq 0$ with $q \leq p$ and $\frac{6r}{2q+r-2} \geq \frac{6p+6}{3p+1}$, then the algorithm of [8] is a $\frac{6p+6}{3p+1}$-approximation. The proof proceeds similarly to the proof of Proposition 1 and makes stronger assumptions on n_S and d_S in Equation 3.1. Berman and Yarolesev [8] deals with 3-pockets by incrementing all the face-minimal cycles within a 3-pocket. This strategy does not work for 4-pockets.

Consider the instance in Figure 7.2. Here each black vertex has cost equal to the number of A and B faces incident to the vertex, each green vertex has cost equal to the number of A and B faces incident to the vertex plus ϵ and each yellow vertex has cost ϵ. The set of cycles to hit is all the A, B and F faces. Here a 4-pocket algorithm would start by incrementing all the A, B faces, select the non-gray parts and return the black vertices of cost 17. This is worse than 2.4 times the green and yellow vertices, which form a solution of cost $7 + 11\epsilon$.

Recall that in the introduction we mentioned how bidimensionality and divide and conquer techniques were used to give approximation algorithms for graph transversal problems [12–14, 26, 27]. Another approach for improving the 2.4-approximation algorithm of [8] for DFVS in planar graphs would be to combine bidimensionality/kernels/divide and conquer techniques [12–14, 26, 27] with the primal-dual algorithm of [8].

We will in the following give some partial ideas about using a variation of local search [41, 53] that achieves a constant approximation for DFVS on planar graphs. Recall in the introduction we defined local search as the following. We are given a graph $G = (V, E)$ and positive integer c. Initialize $S := V$. For each $A \subset S$ of size at most c, set $B \subset V$ of size at most $|A| - 1$, if $(S \backslash A) \cup B$ is a feasible FVS replace S with $(S \backslash A) \cup B$. We call this a *c-opt move*. If not c-opt moves are possible, return S.

Reference [41] shows that the local search algorithm achieves a PTAS for undirected FVS in planar and H-minor free graphs for any fixed graph H. We construct the following example

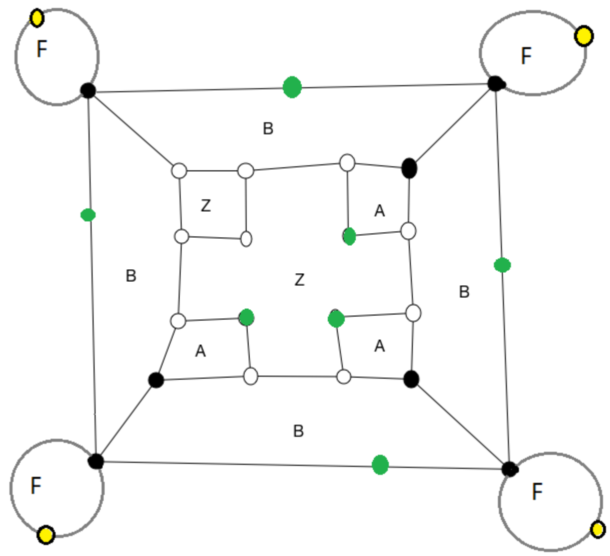


Figure 7.2: Here the set of cycles $\mathcal{C}$ that we need to hit are the A, B and F faces. The black vertices have cost equal to the number of A and B faces incident to the vertex, the green vertices have cost equal to the number of A and B faces incident to the vertex plus ϵ and the yellow vertices on the F faces have cost ϵ. Here a 4-pocket algorithm would select the non-gray parts and return the black vertices of cost 17, which is worse than 2.4 times the cost of the green and yellow vertices which form a solution of cost $7 + 11\epsilon$ (ϵ is small).

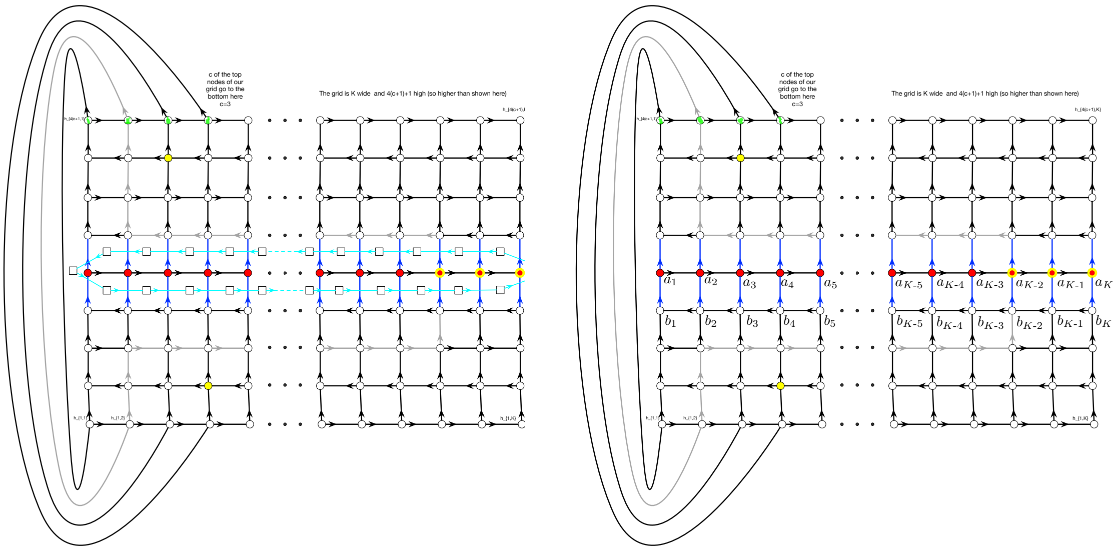
showing the local search algorithm does not achieve any constant factor approximation for DFVS on planar graphs:

Define our graph G to be a $(4(c+1)+1) \times K$ grid. $V(G) := \{h_{i,j} : \quad i \in [4(c+1)+1] \quad j \in [K]\}$, $E(G) = \{(h_{i,j}, h_{i+1,j}) : \quad i \in [4(c+1)+1] \quad j \in [K]\} \cup \{(h_{i,j}, h_{i,j+1}) : \quad i \text{ odd}) \quad j \leq K-1\} \cup \{(h_{i,j}, h_{i,j-1}) : \quad i \text{ even} \quad j \geq 2)\} \cup \{(h_{4(c+1),j}, h_{1,j}) : \quad j \in [c+1]\}$ (see Figure 7.3i)).

Now let the local solution B be $\{h_{2(c+1)+1,j} : \quad j \in [K]\}$. One can see that an optimal solution is $\{h_{4(c+1)+1,j} : \quad j \in [c+1]\}$. It can be shown that no local search on c vertices can decrease the local solution. So the local solution B of size K cannot be improved even though the optimal solution has size c. Since K can be an arbitrary positive integers this shows local search does not achieve any constant approximation.

Can we alter the local search method to achieve any constant approximation? For instance, we could define a c-flow move to select up to c vertices $u_1, u_2, .., u_l$ and up to c vertices $v_1, v_2, .., v_t$ such that no u_i-v_j paths exist in $G \backslash B$ where B is our local solution. Let B' be the nodes of B that lie on some $u_i - v_j$ path. We replace B' by a minimum set of vertices B'' disconnecting all u_i from v_j and add up to c vertices to B'' to fix the solution if it is not feasible.

Suppose that we had a local solution L that could not be improved by c-opt moves. Let $D_1, D_2, \ldots D_3$ be the disjoint directed acyclic graphs (DAGs) of $G \backslash S$. Consider $\delta(D_i)$



(i) (ii)

Figure 7.3: A counterexample to local search for DFVS.

in the dual of G which is a cycle C_i. Let us direct C_i arbitrarily, and for consecutive edges e_1, e_2 in C_i, let $e_1 = \{l_1, t_1\}$ and $e_2 = \{l_2, t_2\}$ in G with $l_i \in L$. We will call l_1, l_2 *consecutive* with respect to D_i. Let us define a *band* with respect to D_i as a series of vertices $\{a_1, a_2, \ldots a_k\}$ such that for $1 \leq j \leq k-1$, a_j and a_{j+1} are consecutive vertices in D_i. For $X \subset V$ define the *in neighbours* as $N_{in}(X) = \{y \in V : yx \in G\}$ and the *out neighbours* as $N_{out}(X) = \{y \in V : xy \in G\}$. We now define a c-band opt move as follows. Take a set $S = \cup_{j=1}^{c} S_j \subset L$ where each S_j is a band and compute directed min vertex cuts Y from S to $N_{in}(S) \backslash L$ and Z from $N_{out}(S) \backslash L$ to S. That is, Y (resp. Z) is a min cost set of nodes such that there is no dipath from S to $N_{in}(S) \backslash L$ (resp. $N_{out}(S) \backslash L$ to S). If either $|Z|$ or $|Y|$ is smaller than $|S|$, replace L with $L' := (L \backslash S) \cup Z$.

Does a local search algorithm with the addition of the above methods achieve any constant approximation? Consider the example in Figure 7.3(*ii*). The graph $D_1 = G \backslash L$, where L is the local solution indicated by the red nodes, is a DAG. The edges $\delta(D_1)$ are displayed in dark blue in Figure 7.3(*i*). $\delta(D_1)$ forms a cycle depicted in light blue in the dual graph. So in Figure 7.3(*ii*), the edges $b_1a_1, b_2a_2, ..., b_Ka_K$ are consecutive. So $L = \{a_1, a_2, .., a_K\}$ form a band. $N_{in}(L) \backslash L = \{b_1, b_2, .., b_K\}$ and a minimum vertex cut Y from L to $\{b_1, b_2, .., b_K\}$ is the set of green striped nodes. $|Y|$ has smaller size than L so the c-band opt move would replace L with Y. This example gives us hope that a modification of local search may be useful for DFVS.

For our final question, we need some definitions first. A 0-torus is a sphere (in 3 dimensional space). For $g \geq 1$, a g-torus is obtained from a $(g-1)$-torus Q and a torus U by removing a ball from both Q and U and "gluing" Q and U on this ball. A graph G has genus

g if g is the smallest integer for which G has an embedding on the g-torus. By bounded genus graphs we mean fix some $g \geq 0$ and consider the graphs of genus at most g. DVFS and ECT have constant factor approximation algorithms in planar graphs [8, 30]. One naturally asks whether this can be generalized to a larger class of graphs.

Question 5. *How well can DFVS or ECT be approximated (in polynomial-time) in bounded genus graphs?*

References

[1] Amit Agarwal, Moses Charikar, Konstantin Makarychev, and Yury Makarychev. $O(\sqrt{\log n})$ approximation algorithms for min uncut, min 2CNF deletion, and directed cut problems. In *Proc. STOC 2005*, pages 573–581, 2005.

[2] Vineet Bafna, Piotr Berman, and Toshihiro Fujito. A 2-approximation algorithm for the undirected feedback vertex set problem. *SIAM J. Discrete Math.*, 12:289–297, 1999.

[3] Brenda S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. ACM*, 41(1):153–180, January 1994.

[4] Nikhil Bansal and Seeun William Umboh. Tight approximation bounds for dominating set on graphs of bounded arboricity. *Information Processing Letters*, 122:21 – 24, 2017.

[5] Mohammadhossein Bateni, MohammadTaghi Hajiaghayi, and Dániel Marx. Approximation schemes for Steiner forest on planar graphs and graphs of bounded treewidth. *J. ACM*, 58(5), 2011.

[6] Ann Becker and Dan Geiger. Optimization of Pearl's method of conditioning and greedy-like approximation algorithms for the vertex feedback set problem. *Artif. Intelligence*, 83(1):167–188, 1996.

[7] Benjamin Bergougnoux, Eduard Eiben, Robert Ganian, Sebastian Ordyniak, and M. S. Ramanujan. Towards a polynomial kernel for directed feedback vertex set. *Algorithmica*, 83(5):1201–1221, 2021.

[8] Piotr Berman and Grigory Yaroslavtsev. Primal-dual approximation algorithms for node-weighted network design in planar graphs. In *Proc. APPROX 2012*, volume 7408 of *Lecture Notes Comput. Sci.*, pages 50–60, 2012.

[9] Chandra Chekuri and Vivek Madan. Constant factor approximation for subset feedback set problems via a new LP relaxation. In *Proc. SODA 2016*, pages 808–820, 2016.

[10] Vincent Cohen-Addad, Éric Colin de Verdière, Philip N. Klein, Claire Mathieu, and David Meierfrankenfeld. Approximating connectivity domination in weighted bounded-genus graphs. STOC '16, page 584–597, New York, NY, USA, 2016. Association for Computing Machinery.

[11] Erik D. Demaine, Fedor V. Fomin, Mohammad Taghi Hajiaghayi, and Dimitrios M. Thilikos. Bidimensional parameters and local treewidth. *SIAM J. Discret. Math.*, 18(3):501–511, 2004.

[12] Erik D. Demaine and MohammadTaghi Hajiaghayi. Bidimensionality: New connections between FPT algorithms and PTASs. SODA '05, page 590–601, USA, 2005. Society for Industrial and Applied Mathematics.

[13] Erik D. Demaine and MohammadTaghi Hajiaghayi. The bidimensionality theory and its algorithmic applications. *The Computer Journal*, 51(3):292–302, 2008.

[14] Erik D. Demaine and MohammadTaghi Hajiaghayi. Linearity of grid minors in treewidth with applications through bidimensionality. *Comb.*, 28(1):19–36, 2008.

[15] Erik D. Demaine, MohammadTaghi Hajiaghayi, and Philip N. Klein. Node-weighted Steiner tree and group Steiner tree in planar graphs. *ACM Trans. Algorithms*, 10(3), 2014.

[16] Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162, 07 2004.

[17] Irit Dinur and David Steurer. Analytical approach to parallel repetition. STOC '14, page 624–633, New York, NY, USA, 2014. Association for Computing Machinery.

[18] Y. Dorfman and G. Orlova. Finding the maximal cut in a graph. *Engineering Cybernetics*, 10(3), 1972.

[19] Zdenek Dvorak. On distance r-dominating and 2r-independent sets in sparse graphs. *Journal of Graph Theory*, 91:162–173, 10 2017.

[20] G. Even, J. Naor, B. Schieber, and M. Sudan. Approximating minimum feedback sets and multicuts in directed graphs. *Algorithmica*, 20(2):151–174, 1998.

[21] Guy Even. An 8-approximation algorithm for the subset feedback vertex set problem. *SIAM J. Comput.*, 30(4):1231–1252, oct 2000.

[22] Paola Festa, Panos M. Pardalos, and Mauricio G. C. Resende. *Feedback Set Problems*, pages 209–258. Springer US, Boston, MA, 1999.

[23] Samuel Fiorini, Nadia Hardy, Bruce Reed, and Adrian Vetta. Approximate min–max relations for odd cycles in planar graphs. *Math. Program.*, 110:71–91, 2007.

[24] Samuel Fiorini, Gwenaël Joret, and Ugo Pietropaoli. Hitting diamonds and growing cacti. In *Proc. IPCO 2010*, volume 6080 of *Lecture Notes Comput. Sci.*, pages 191–204, 2010.

[25] Samuel Fiorini and Odile Marcotte. On the feedback vertex set polytope of a series-parallel graph. *Discret. Optim.*, 6(3):271–287, August 2009.

[26] Fedor V. Fomin, Daniel Lokshtanov, Venkatesh Raman, and Saket Saurabh. Bidimensionality and EPTAS. SODA '11, pages 748–759, Philadelphia, PA, USA, 2011. Society for Industrial and Applied Mathematics.

[27] Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Dimitrios M. Thilikos. Bidimensionality and kernels. *SIAM J. Comput.*, 49(6):1397–1422, 2020.

[28] Michel X. Goemans and David P. Williamson. *The Primal-Dual Method for Approximation Algorithms and Its Application to Network Design Problems*, page 144–191. PWS Publishing Co., USA, 1996.

[29] Michel X. Goemans and David P. Williamson. Primal-dual approximation algorithms for feedback problems in planar graphs. *Combinatorica*, 18(1):37–59, 1998.

[30] Alexander Göke, Jochen Könemann, Matthias Mnich, and Hao Sun. Hitting weighted even cycles in planar graphs. In *APPROX/RANDOM 2021*, volume 207 of *LIPIcs*, pages 25:1–25:23. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021.

[31] Bertrand Guenin and Robin Thomas. Packing directed circuits exactly. *Combinatorica*, 31(4):397–421, Nov 2011.

[32] F. Hadlock. Finding a maximum cut of a planar graph in polynomial time. *SIAM J. Comput.*, 4(3), 1975.

[33] Eva-Maria C. Hols and Stefan Kratsch. A randomized polynomial kernel for subset feedback vertex set. *Theory Comput. Syst.*, 62(1):63–92, 2018.

[34] Stanislav Jendrol and Heinz-Jürgen Voß. Light subgraphs of graphs embedded in the plane—a survey. *Discrete Math.*, 313(4):406–421, 2013.

[35] Gwenaël Joret, Christophe Paul, Ignasi Sau, Saket Saurabh, and Stéphan Thomassé. Hitting and harvesting pumpkins. *SIAM J. Discret. Math.*, 28(3):1363–1390, 2014.

[36] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103, 1972.

[37] Claire Kenyon-Mathieu and Warren Schudy. How to rank with few errors: A PTAS for weighted feedback arc set on tournaments. *Electronic Colloquium on Computational Complexity (ECCC)*, 13, 01 2006.

[38] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within $2-\varepsilon$. *J. Comput. Syst. Sci.*, 74(3):335–349, 2008.

[39] Stefan Kratsch and Magnus Wahlström. Compression via matroids: A randomized polynomial kernel for odd cycle transversal. *ACM Trans. Algorithms*, 10(4):20:1–20:15, 2014.

[40] Joseph B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1):48–50, 1956.

[41] Hung Le and Baigong Zheng. Local search is a PTAS for feedback vertex set in minor-free graphs. *Theoretical Computer Science*, 838:17–24, 2020.

[42] Neele Leithäuser, Sven Oliver Krumke, and Maximilian Merkert. Approximating infeasible 2VPI-systems. volume 7551 of *Lecture Notes in Computer Science*, pages 225–236. Springer, 2012.

[43] Christoph Lenzen and Roger Wattenhofer. Minimum dominating set approximation in graphs of bounded arboricity. In *Distributed Computing*, pages 510–524, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[44] David Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982.

[45] Daniel Lokshtanov, Pranabendu Misra, Joydeep Mukherjee, Fahad Panolan, Geevarghese Philip, and Saket Saurabh. 2-approximating feedback vertex set in tournaments. *ACM Trans. Algorithms*, 17(2), apr 2021.

[46] Daniel Lokshtanov and MS Ramanujan. Parameterized tractability of multiway cut with parity constraints. In *Proc. ICALP 2012*, volume 7391 of *Lecture Notes Comput. Sci.*, pages 750–761, 2012.

[47] Daniel Lokshtanov, MS Ramanujan, Saket Saurab, and Meirav Zehavi. Parameterized complexity and approximability of directed odd cycle transversal. In *Proc. SODA 2020*, pages 2181–2200, 2020.

[48] C. L. Lucchesi and D. H. Younger. A minimax theorem for directed graphs. *J. LONDON MATH. SOC*, 17(2):369–374, 1978.

[49] Pranabendu Misra, Venkatesh Raman, M. S. Ramanujan, and Saket Saurabh. Parameterized algorithms for even cycle transversal. volume 7551 of *Lecture Notes in Computer Science*, pages 172–183. Springer, 2012.

[50] Carsten Moldenhauer. Primal-dual approximation algorithms for node-weighted steiner forest on planar graphs. *Inf. Comput.*, 222:293–306, 2013.

[51] Martin Nägele and Rico Zenklusen. A new contraction technique with applications to congruency-constrained cuts. *Math. Prog.*, 183:455–481, 2020.

[52] C. St.J. A. Nash-Williams. Decomposition of finite graphs into forests. *Journal of the London Mathematical Society*, s1-39(1):12–12, 1964.

[53] James Orlin, Abraham Punnen, and Andreas Schulz. Local search in combinatorial optimization. *SIAM J. Comput.*, 33:587–596, 01 2004.

[54] Igor Razgon and Barry O'Sullivan. Directed feedback vertex set is fixed-parameter tractable. *CoRR*, abs/0707.0282, 2007.

[55] Bruce A. Reed. Mangoes and blueberries. *Comb.*, 19(2):267–296, 1999.

[56] Hao Sun. An improved approximation bound for minimum weight dominating set on graphs of bounded arboricity. In *WAOA 2021*, volume 12982 of *Lecture Notes in Computer Science*, pages 39–47. Springer, 2021.

[57] Stéphan Thomassé. A $4k^2$ kernel for feedback vertex set. *ACM Trans. Algorithms*, 6(2):32:1–32:8, 2010.

[58] Wouter Cames van Batenburg, Tony Huyn, Gwenaël Joret, and Jean-Florent Raymond. A tight erdős-pósa function for planar minors. *Adv. Combinatorics*, 2019.

[59] Anke van Zuylen. Linear programming based approximation algorithms for feedback set problems in bipartite tournaments. *Theoretical Computer Science*, 412(23):2556–2561, 2011.

[60] Vijay V. Vazirani. *Approximation Algorithms*. Springer-Verlag, Berlin, Heidelberg, 2001.

[61] Mihalis Yannakakis. Node-and edge-deletion NP-complete problems. STOC '78, page 253–264, New York, NY, USA, 1978. Association for Computing Machinery.