technische universiteit eindhoven
Department of Mathematics and Computer Science

Master's Thesis

Characterizing graphs with a sliceable rectangular dual

by
Vincent Kusters

Supervisor
dr. Bettina Speckmann

Eindhoven, July 20, 2011

ABSTRACT

A *rectangular partition* of a rectangle R is a partition of R into a set $\mathcal{R}$ of non-overlapping rectangles such that no four rectangles in $\mathcal{R}$ meet at one common point. A *rectangular dual* of a planar graph $\mathcal{G}$ is a rectangular partition $\mathcal{R}$, such that *(i)* there is a one-to-one correspondence between the rectangles in $\mathcal{R}$ and the vertices of $\mathcal{G}$, and *(ii)* two rectangles in $\mathcal{R}$ share a common boundary if and only if the corresponding vertices of $\mathcal{G}$ are connected. A given graph can have many rectangular duals. A rectangular dual is *sliceable* if it can be recursively subdivided along horizontal or vertical lines. In this thesis, we aim to characterize the graphs which have a sliceable rectangular dual.

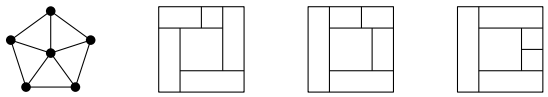
If a graph $\mathcal{G}$ has a rectangular dual and does not have a separating 4-cycle, then $\mathcal{G}$ has a sliceable rectangular dual [12]. No characterization of the sliceable subset of graphs with separating 4-cycles is known. We introduce a variation of series-parallel graphs and show that all such graphs are sliceable. The remainder of this thesis contains the majority of our results and focuses on graphs with exactly one separating 4-cycle. We describe a new class of graphs with exactly one separating 4-cycle which is adjacent to the outer cycle and show that all graphs in this class are sliceable. In addition, we present a generic method for proving the sliceability of such graphs. We introduce *rotating windmills* and prove that they are nonsliceable. Finally, we perform experiments which lead us to believe that rotating windmills are the only nonsliceable graphs which have a rectangular dual and exactly one separating 4-cycle.

CONTENTS

1

INTRODUCTION

A *rectangular partition* of a rectangle R is a partition of R into a set $\mathcal{R}$ of non-overlapping rectangles such that no four rectangles in $\mathcal{R}$ meet at one common point. A *rectangular dual* of a planar graph $\mathcal{G}$ is a rectangular partition $\mathcal{R}$, such that *(i)* there is a one-to-one correspondence between the rectangles in $\mathcal{R}$ and the vertices of $\mathcal{G}$, and *(ii)* two rectangles in $\mathcal{R}$ share a common boundary if and only if the corresponding vertices of $\mathcal{G}$ are connected. A given graph can have many rectangular duals, as shown in Figure 1. There are several known classes of rectangular duals. For example, a rectangular partition is called *area-universal* if any assignment of areas to rectangles can be realized by a combinatorially equivalent rectangular dual [4]. We focus on the class of *sliceable* rectangular partitions: rectangular partitions which can be recursively subdivided along horizontal or vertical lines. In this thesis, we aim to characterize the graphs which have a sliceable rectangular dual.



(a) (b) (c) (d)

Figure 1: A graph with three of its rectangular duals. Duals (c) and (d) are sliceable. Dual (d) is area-universal.

1.1 RECTANGULAR DUALS

Not all plane graphs admit a rectangular dual. An *extended graph* $E(\mathcal{G})$ of a plane graph $\mathcal{G}$ is an extension of $\mathcal{G}$ with four vertices in such a way that the four vertices form the outer face of $E(\mathcal{G})$. The four vertices are labeled $t(\mathcal{G})$, $r(\mathcal{G})$, $b(\mathcal{G})$ and $l(\mathcal{G})$ in clockwise order. These vertices are called the *poles* of $E(\mathcal{G})$, whereas the vertices from the original graph $\mathcal{G}$ are called the *interior* vertices. Since the choice of extended graph fixes the vertices that correspond to the four corners (and therefore the vertices along the four sides) of the rectangular dual, extended graphs are also called *corner assignments*. See Figure 2 for an example.

A *separating k-cycle* of an extended graph $E(\mathcal{G})$ is a k-cycle with vertices both inside and outside the cycle. A separating k-cycle is *maximal* if its interior is not contained in any other separating k-cycle of $E(\mathcal{G})$. A *triangle* is a 3-cycle. The *outer cycle* of a plane graph is the cycle which has

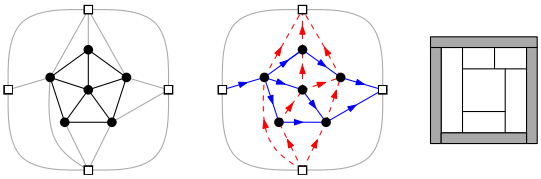


Figure 2: An extended graph $E(\mathcal{G})$, one of its regular edge labelings and the corresponding rectangular dual.

no vertices in its exterior, that is, the cycle formed by the edges incident to the unbounded face. An *irreducible triangulation* is a triangulated plane graph with four vertices on its outer cycle and no separating triangles.

Theorem 1 (Ungar [11], Bhasker and Sahni [1], Koźmiński and Kinnen [7]). *A graph $\mathcal{G}$ has a rectangular dual if and only if $\mathcal{G}$ has an extended graph which is an irreducible triangulation.*

Two rectangular duals $\mathcal{R}$ and $\mathcal{R}'$ of a graph $\mathcal{G}$ are *equivalent* if and only if for each edge (u, v) in $\mathcal{G}$ it holds that either *(i)* u and v share a horizontal line segment in both $\mathcal{R}$ and $\mathcal{R}'$ or *(ii)* u and v share a vertical line segment in both $\mathcal{R}$ and $\mathcal{R}'$. The equivalence classes of the rectangular duals of an irreducible triangulation $E(\mathcal{G})$ correspond one-to-one to the *regular edge labelings of* $E(\mathcal{G})$. A regular edge labeling of an extended graph $E(\mathcal{G})$ is a partition of the interior edges of $E(\mathcal{G})$ into two subsets of red and blue directed edges such that: *(i)* around each inner vertex in clockwise order we have four contiguous nonempty sets of incoming blue edges, outgoing red edges, outgoing blue edges, and incoming red edges and; *(ii)* $l(\mathcal{G})$ has only outgoing blue edges, $t(\mathcal{G})$ has only incoming red edges, $r(\mathcal{G})$ has only incoming blue edges and $b(\mathcal{G})$ has only outgoing red edges (see Figure 3; red edges are dashed). Kant and He [6] showed how to find a regular edge labeling and construct the corresponding rectangular dual in linear time. Fusy [5] also studied regular edge labelings (he calls them *transversal pairs of bipolar orientations*) and gave another linear time algorithm to compute a regular edge labeling. This algorithm can also be used to traverse all regular edge labelings of an extended graph.

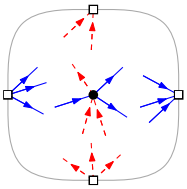


Figure 3: The local conditions on a regular edge labeling.

It has proven to be difficult to characterize specific classes of rectangular duals. There is currently no characterization of the graphs that have an area-universal rectangular dual. Eppstein et al. [4] gave a fixed-parameter tractable algorithm to find an area-universal rectangular dual of a graph if one exists.

1.2 SLICEABLE GRAPHS

An extended graph $E(\mathcal{G})$ is *sliceable* if and only if it has a sliceable rectangular dual. A graph $\mathcal{G}$ is sliceable if and only if it has a sliceable extended graph. Since we can determine if a graph is sliceable by testing each of its

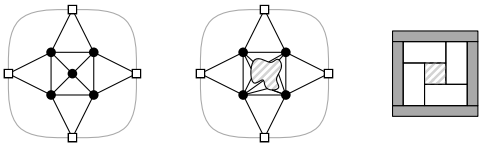


Figure 4: The windmill, the generalized windmill (the hatched shape is an arbitrary graph), and a rectangular dual for the generalized windmill.

polynomially many corner assignments, we will consider only extended graphs from now on. No characterization of the sliceable subset of extended graphs is known yet. The smallest nonsliceable extended graph is the *windmill* depicted in Figure 4. This extended graph can be generalized to a *generalized windmill* by replacing the center vertex with an arbitrary graph. All generalized windmills are nonsliceable.

Yeap and Sarrafzadeh [12] showed that all irreducible triangulations $E(\mathcal{G})$ without separating 4-cycles are sliceable. Dasgupta and Sur-Kolay [3] claim to have partially characterized the sliceable graphs with at least one separating 4-cycle, but their claim and proof are incorrect, as will be shown in Section 2.4.

1.3 RESULTS AND ORGANIZATION

In this thesis we attempt to characterize the sliceable subset of irreducible triangulations with at least one separating 4-cycle. We introduce a variation of series-parallel graphs and show that all such graphs are sliceable. The remainder of this thesis contains the majority of our results and focuses on graphs with exactly one separating 4-cycle. We describe a new class of sliceable graphs and a new class of nonsliceable graphs. Finally, we discuss our experimental analysis and the algorithms we implemented.

We first revisit regular edge labelings and discuss previous results on determining the sliceability of graphs with and without separating 4-cycles in Chapter 2. We discuss a sliceable variation of series-parallel graphs called triangulated series-parallel graphs in Chapter 3. We study a class of sliceable graphs with exactly one separating 4-cycle whose separating 4-cycle is adjacent to the outer cycle in Chapter 4.

Chapter 5 contains our most important result: the nonsliceability of rotating windmills. We have a strong indication that rotating windmills are the only nonsliceable irreducible triangulations with exactly one separating 4-cycle, but we do not have a conclusive proof of this conjecture yet. We briefly discuss two conjectures we considered but ultimately refuted during our research in Chapter 6.

Finally, in Chapter 7, we describe the experimental analysis that led to our conjecture that rotating windmills are precisely the nonsliceable graphs with exactly one separating 4-cycle.

2

PRELIMINARIES

In this chapter, we summarize some earlier findings on regular edge labelings and sliceable graphs.

2.1 CHARACTERIZATION USING FORBIDDEN MINORS

A graph $\mathcal{H}$ is a *minor* of a graph $\mathcal{G}$ if $\mathcal{H}$ is isomorphic to a graph that can be obtained by zero or more edge contractions on a subgraph of $\mathcal{G}$. Sometimes a family of graphs can be characterized using forbidden minors. It follows from the Robertson–Seymour theorem [10] that every family of graphs that is closed under minors can be defined by a finite set of forbidden minors.

Unfortunately, this theorem cannot be used to characterize the set of sliceable extended graphs. This is illustrated by Figure 5. The three extended graphs depicted in the figure are irreducible triangulations. The leftmost graph is sliceable by Theorem 4. Contracting the bold edge yields the windmill, which is not sliceable. Finally, contracting the bold edge in the windmill yields the graph on the right, which is sliceable by Theorem 4. We conclude that the class of sliceable graphs is not closed under minors.

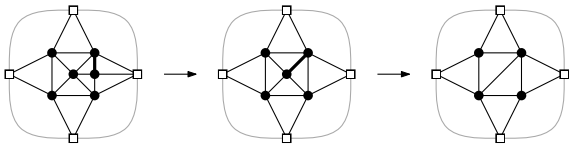


Figure 5: The class of sliceable graphs is not closed under the minor operation.

2.2 REGULAR EDGE LABELINGS

We use several definitions and properties of regular edge labelings that were introduced or proven by Fusy [5]. A *regular edge coloring* is a regular edge labeling, with the directions of the edges omitted.

Lemma 2 (Fusy [5], Proposition 2)**.** *A regular edge coloring uniquely determines a regular edge labeling.*

A *monochromatic triangle* is a triangle where all edges have the same color.

Lemma 3 (Fusy [5], Lemma 1)**.** *A regular edge labeling (of an irreducible triangulation) induces no monochromatic triangles.*

2.3 GRAPHS WITHOUT SEPARATING 4-CYCLES

Yeap and Sarrafzadeh [12] gave the following sufficient condition for sliceability:

Theorem 4 (Yeap and Sarrafzadeh [12])**.** *Let $E(\mathcal{G})$ be an irreducible triangulation. If $\mathcal{G}$ has no separating 4-cycle, then $E(\mathcal{G})$ is sliceable.*

A *cut* is a partition of the vertices of a graph in two disjoint subsets. The *cut-set* of the cut is the set of edges whose endpoints are in different subsets of the partition. A cut-set in $\mathcal{G}$ uniquely defines two *boundary paths* in $E(\mathcal{G})$: two paths with the same source and sink whose nonterminal vertices are exactly the endpoints of the edges in the cut-set. A cut is a *vertical slice* if its boundary paths start at $t(\mathcal{G})$ and end at $b(\mathcal{G})$. A cut is a *horizontal slice* if its boundary paths start at $l(\mathcal{G})$ and end at $r(\mathcal{G})$. A path $v_1, \cdots, v_k$ is *chordless* if and only if v_i and v_j are not connected for each $1 \leq i < j - 1 \leq k$. In other words, all edges between vertices on the path must be part of the path. A *proper slice* is a slice with chordless boundary paths. See Figure 6.

The proof assumes that an irreducible triangulation without separating 4-cycles $E(\mathcal{G})$ is given. The authors show that a proper slice can always be found in this graph. The slice partitions $\mathcal{G}$ into $\mathcal{G}_l$ and $\mathcal{G}_r$ (without loss of generality, we will assume that the slice is vertical). The intention is to inductively apply the theorem to obtain sliceable rectangular duals of $\mathcal{G}_l$ and $\mathcal{G}_r$ and to combine these duals into a sliceable rectangular dual for $\mathcal{G}$. Since $\mathcal{G}$ has no separating 3-cycles or 4-cycles, its subgraphs $\mathcal{G}_l$ and $\mathcal{G}_r$ also have no separating 3-cycles or 4-cycles. It remains to show that there are extended graphs $E(\mathcal{G}_l)$ and $E(\mathcal{G}_r)$ such that *(i)* neither $E(\mathcal{G}_l)$ nor $E(\mathcal{G}_r)$ have separating triangles and *(ii)* the sliceable rectangular duals of $E(\mathcal{G}_l)$ and $E(\mathcal{G}_r)$ can be used to construct a sliceable rectangular dual of $\mathcal{G}$. For this to work, the corner assignment of $\mathcal{G}_l$ must be as follows:

- if a vertex v of $\mathcal{G}_l$ is adjacent to a pole p in $E(\mathcal{G})$, where p is one of $b(\mathcal{G})$, $t(\mathcal{G})$ and $l(\mathcal{G})$, then v is also adjacent to p in $E(\mathcal{G}_l)$, and
- if a vertex v is part of the left boundary path of the cut, then v is adjacent to $r(\mathcal{G}_l)$ in $E(\mathcal{G}_l)$.

The corner assignment of $\mathcal{G}_r$ is determined symmetrically. The crucial observation is that $E(\mathcal{G}_l)$ and $E(\mathcal{G}_r)$ contain no separating triangles: neither $\mathcal{G}_l$ nor $\mathcal{G}_r$ contain separating triangles and the addition of $r(\mathcal{G}_l)$ (or $l(\mathcal{G}_r)$) does not introduce any separating triangles, since the boundary paths of the cut are chordless. It follows that $E(\mathcal{G}_l)$ and $E(\mathcal{G}_r)$ satisfy the requirements of the theorem and hence have a sliceable layout. By choosing the corner assignments for $\mathcal{G}_l$ and $\mathcal{G}_r$ correctly, the authors make sure that the rectangular duals of the left and right graph can be combined into a rectangular dual for $\mathcal{G}$.

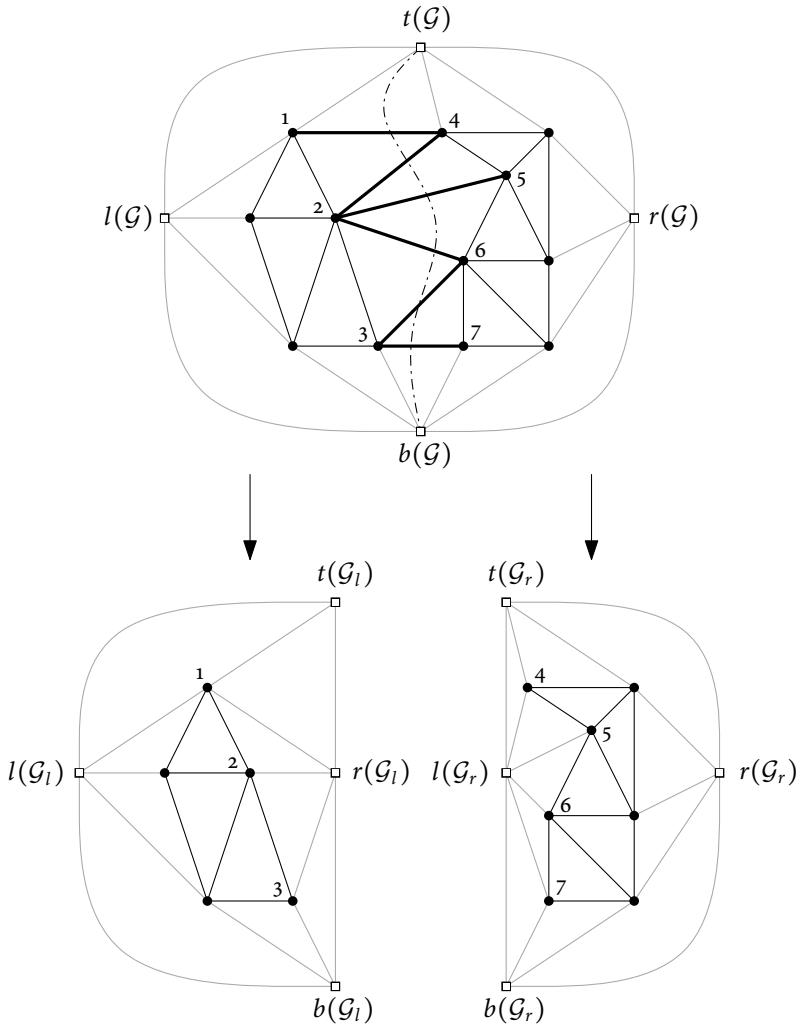


Figure 6: An extended graph with a proper slice indicated by a dash-dotted line. The edges of the cut-set are bold. The boundary paths are $t(\mathcal{G}), 1, 2, 3, b(\mathcal{G})$ and $t(\mathcal{G}), 4, 5, 6, 7, b(\mathcal{G})$. Both boundary paths are chordless. Original by Yeap and Sarrafzadeh [12].

2.4 GRAPHS WITH SEPARATING 4-CYCLES

It remains to characterize the sliceable subset of the class of irreducible triangulations $E(\mathcal{G})$ where $\mathcal{G}$ has at least one separating 4-cycle. There are examples of both sliceable and nonsliceable extended graphs in this class. The smallest sliceable one is shown in Figure 7. The smallest nonsliceable one is the windmill shown in Figure 4. A 4-cycle with exactly one vertex in its interior is called a *pyramid*. Mumford showed that it is sufficient to consider extended graphs where all separating 4-cycles are pyramids:

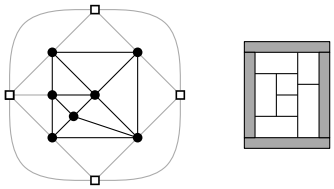


Figure 7: The smallest sliceable extended graph $E(\mathcal{G})$ where $\mathcal{G}$ has at least one separating 4-cycle.

Lemma 5 (Mumford [8], Theorem 2.16)**.** *Let $E(\mathcal{G})$ be an irreducible triangulation. Let $MQ(\mathcal{G})$ be the set of all maximal separating 4-cycles in $\mathcal{G}$. Given a cycle $C \in MQ(\mathcal{G})$, let $\mathcal{G}_C$ be the interior of C. Let $E(\mathcal{G}')$ be the extended graph obtained by replacing the subgraph $\mathcal{G}_C$ for every $C \in MQ(\mathcal{G})$ in $E(\mathcal{G})$ by a single vertex connected to all vertices of C. The extended graph $E(\mathcal{G})$ is sliceable if and only if $E(\mathcal{G}')$ is sliceable and for each $C \in MQ(\mathcal{G})$, the induced corner assignment of $\mathcal{G}_c$ is sliceable.*

The proof directly extends to maximal separating 4-cycles in $E(\mathcal{G})$ instead of $\mathcal{G}$. See Figure 8 for an example. A *sliceable regular edge labeling* is a regular edge labeling which corresponds to a sliceable rectangular partition.

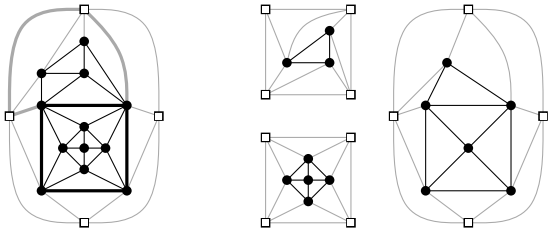


Figure 8: The extended graph on the left is sliceable if and only if the three extended graphs on the right are sliceable. The two maximal 4-cycles of the extended graph on the left are indicated with bold lines.

Lemma 6 (Mumford and Speckmann [9], without proof)**.** *Let $\mathcal{L}$ be a sliceable regular edge labeling of $E(\mathcal{G})$ and let P be a pyramid in $E(\mathcal{G})$. There must be at least one vertex on the outer cycle of P whose three incident edges in P have the same color.*

Proof. Let P be a pyramid in an extended graph $E(\mathcal{G})$. After removing duplicates with respect to rotation and symmetry, there are only four possible regular edge colorings for P. They are depicted in Figure 9. Note that (d) is the only nonsliceable dual. In all sliceable duals, the bottom left vertex has three incident edges in P, all of which have the same color. □

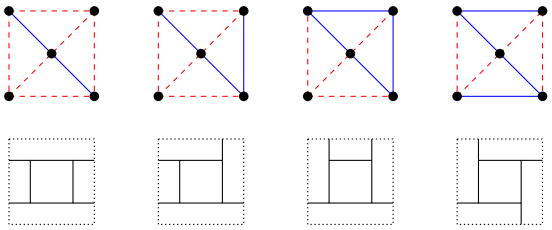


Figure 9: The four equivalence classes of the regular edge colorings of a pyramid.

Lemma 7 (Mumford and Speckmann [9], without proof)**.** *Let $E(\mathcal{G})$ be a sliceable extended graph. Every pyramid in $E(\mathcal{G})$ must have a vertex with degree 6.*

Proof. It follows from Lemma 6 that every pyramid P must have a vertex v whose three incident edges in P have the same color. Since v is an interior vertex of $E(\mathcal{G})$, it must have four nonempty contiguous sets of incoming blue edges, outgoing red edges, outgoing blue edges and incoming red edges. It follows that the degree of v is at least six. □

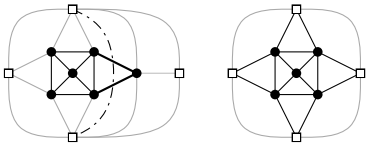
Dasgupta and Sur-Kolay [3] attempt to give a sufficient condition for sliceability of extended graphs with a separating 4-cycle. We first introduce the terminology used by Dasgupta and Sur-Kolay. A *rectangular graph* is a graph that admits a rectangular dual. A cycle is *complex* if it has at least one vertex in its interior. A vertex is a *corner* if it is adjacent to at least two poles. A corner is *nondistinct* if it is adjacent to at least three poles.

Claim 1 (Dasgupta and Sur-Kolay [3], invalid)**.** *A rectangular graph $\mathcal{G}$ with n vertices, $n > 4$, is sliceable if it satisfies either of the following two conditions:*

1. *its outermost cycle is the only complex 4-cycle in $\mathcal{G}$ and at least one of its four vertices is a nondistinct corner;*
2. *all the complex 4-cycles of $\mathcal{G}$ are maximal.*

Note that if $n > 4$, no corner is adjacent to all four poles, so in the context of the claim, nondistinct corners are adjacent to exactly three poles.

The formulation of the theorem is somewhat ambiguous. In the first paragraph of the proof, the authors write that they "[…] wish to show that if at least one of the above two conditions is true, then there always exists a proper slice E_s (vertical or horizontal) through $\mathcal{G}$". This would imply that the graph is sliceable if at least one of the two conditions holds. However, in the first paragraph of the Main Results section of the paper, the authors state that they "[…] prove a stronger condition for sliceability



(a) (b)

Figure 10: On the left: a graph $\mathcal{G}$ which is sliceable according to the construction in the proof, but is not sliceable for the given corner assignment. The proper slice found by the constructive proof is indicated with the dashed line. On the right: the graph $\mathcal{G}_l$ with corner assignment $E(\mathcal{G}_l)$ is a windmill. Original by Mumford [8].

[than Yeap and Sarrafzadeh do], namely, even if a rectangular graph contains complex 4-cycles, it is necessarily sliceable, provided all complex 4-cycles in the graph are maximal, and there does not exist any complex 4-cycle in the graph whose all four vertices are assigned to be corner blocks of the corresponding floorplan". This statement implies that both conditions must hold. In addition, the first condition appears to exclude all graphs that do not have a 4-cycle as their outermost cycle. This would, however, impose the clearly unintended restriction that $\mathcal{G}_l$ and $\mathcal{G}_r$ again have a 4-cycle as their outermost cycle. The illustrations they give suggest this is not their intention. We therefore interpret the first condition to mean that $\mathcal{G}$ is not a generalized windmill (see Figure 4).

In both cases, the claim is incorrect. Although their proof is similar to the proof given by Yeap and Sarrafzadeh (construct a proper slice, recurse on $\mathcal{G}_l$ and $\mathcal{G}_r$), they fail to show that $\mathcal{G}_l$ and $\mathcal{G}_r$ again satisfy the constraints imposed by the theorem.

COUNTEREXAMPLES. Consider the graph $\mathcal{G}$ with nonsliceable corner assignment depicted in Figure 10. Suppose we obtain this graph at some point during the recursive application of Claim 1. Suppose furthermore that the depicted corner assignment is the only possible corner assignment, due to the restrictions imposed by previous slices on a larger graph that contains $\mathcal{G}$. The graph conforms to the second condition of the claim. If the first condition is not meant to exclude graphs with more than four vertices on the outer cycle, then it also conforms to the first condition. Hence, the claim incorrectly states that $\mathcal{G}$ is sliceable for this corner assignment.

There also exist graphs which are not sliceable for any corner assignment, yet contain only maximal separating 4-cycles [8]. One such graph is depicted in Figure 11. Note that there are five pyramids on the outer cycle of this graph. Suppose for the sake of deriving a contradiction that the graph is sliceable for some corner assignment $E(\mathcal{G})$. It follows from Lemma 7 that every pyramid in $E(\mathcal{G})$ must have a vertex with degree at

least six. Every pyramid P in $\mathcal{G}$ has two vertices on the outer cycle with degree 4, and two vertices not on the outer cycle with degree 5. The only way P can have a vertex with degree 6 in $E(\mathcal{G})$ is if a vertex of P on the outer cycle is connected to two poles. But since there are five pyramids and only four poles, this cannot be done. It follows that $\mathcal{G}$ is not sliceable.

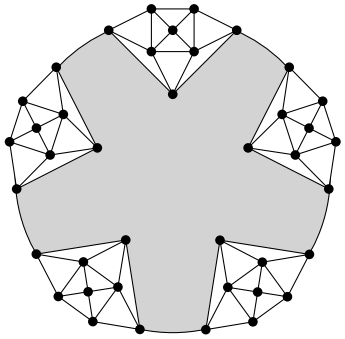


Figure 11: A graph which is not sliceable for any corner assignment. Original by Mumford [8].

2.5 SLICES IN REGULAR EDGE LABELINGS

Since we will frequently consider slices in regular edge labelings, it is important to discuss their properties. Recall that a cut is a partition of the vertices of a graph in two disjoint subsets. The cut-set of a cut is the set of edges whose endpoints are in different subsets of the partition. The boundary paths of a cut are the two paths with the same source and sink whose nonterminal vertices are exactly the endpoints of the edges in the cut-set. Let $\mathcal{R}$ be a rectangular dual of $E(\mathcal{G})$ and let $\mathcal{L}$ be the regular edge labeling that corresponds to $\mathcal{R}$. Any vertical slice in $\mathcal{R}$ has a blue cut-set and red boundary paths in $\mathcal{L}$. Any horizontal slice in $\mathcal{R}$ has a red cut-set and blue boundary paths.

Let us revisit the smallest sliceable extended graph $E(\mathcal{G})$ where $\mathcal{G}$ has at least one separating 4-cycle for an illustration. Figure 12 shows the slices in one of the sliceable regular edge labelings of this extended graph. Slices a and e are vertical. Slices b, c, d and f are horizontal. A slice is a *first slice* of $E(\mathcal{G})$ if it starts and ends at poles of $E(\mathcal{G})$. Slice a is the only first slice in Figure 12.

2.6 ONE-PYRAMID EXTENDED GRAPHS

Recall that Yeap and Sarrafzadeh proved that irreducible triangulations $E(\mathcal{G})$ for which $\mathcal{G}$ has no separating 4-cycle are sliceable (Theorem 4). In addition, Mumford showed that it is sufficient to consider only extended

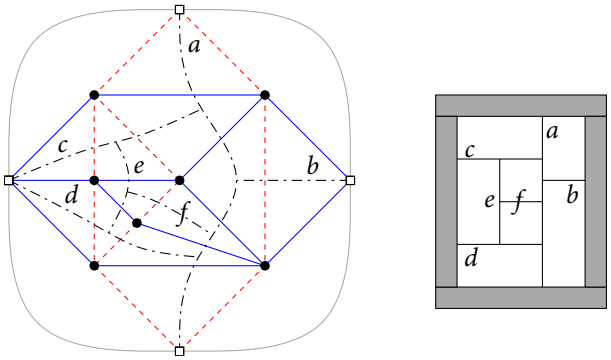


Figure 12: A regular edge labeling and corresponding rectangular dual of the smallest sliceable one-pyramid extended graph.

graphs where every separating 4-cycle is a pyramid (Lemma 5). Hence, the only class of extended graphs whose sliceable subset has not been fully characterized yet is the class of *many-pyramid extended graphs*: irreducible triangulations $E(\mathcal{G})$ such that $\mathcal{G}$ has at least one pyramid and all separating 4-cycles in $E(\mathcal{G})$ are pyramids. In this thesis, we focus on *one-pyramid extended graphs*: many-pyramid extended graphs $E(\mathcal{G})$ such that $\mathcal{G}$ has at exactly one pyramid P and P is the only pyramid in $E(\mathcal{G})$.

3

TRIANGULATED SERIES-PARALLEL GRAPHS

In this chapter we study a variation of series-parallel graphs: triangulated series parallel graphs.

3.1 DEFINITIONS

A *multigraph* is a graph which permits *multiple edges*: two edges between the same pair of vertices. A *series-parallel graph* is a multigraph with a source and a sink which can be defined by an *initial graph* and two operations. The initial series-parallel graph is the graph with source s and sink t and an edge from s to t. The *series* operation takes two series-parallel graphs $\mathcal{X}$ and $\mathcal{Y}$ and combines them by identifying the sink of $\mathcal{X}$ with the source of $\mathcal{Y}$. The *parallel* operation takes two series-parallel graphs $\mathcal{X}$ and $\mathcal{Y}$ and combines them by identifying the source of $\mathcal{X}$ with the source of $\mathcal{Y}$, and the sink of $\mathcal{X}$ with the sink of $\mathcal{Y}$. The multigraphs which result from the series and parallel operations are again series-parallel graphs. Figure 13 shows an example of the construction of a series-parallel graph.

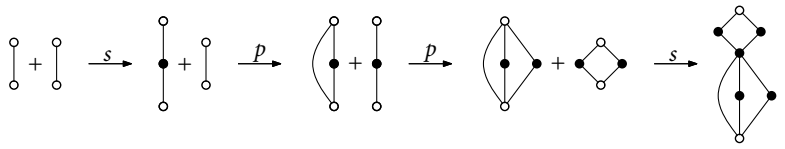


Figure 13: A construction sequence for a series-parallel graph. The sources and sinks are indicated with circles. The series operation is indicated with $\xrightarrow{s}$ and the parallel operation is indicated with $\xrightarrow{p}$.

We introduce *triangulated series-parallel graphs* which are closely related to series-parallel graphs, but do not allow multiple edges. Furthermore, all triangulated series-parallel graphs are embedded triangulations (except the outer face) without separating triangles. Triangulated series-parallel graphs can have an arbitrary number of separating 4-cycles. The *left boundary path* of a triangulated series-parallel graph $\mathcal{G}$ is the clockwise path along the outer cycle from the source of $\mathcal{G}$ to the sink of $\mathcal{G}$. Similarly, the *right boundary path* is the counterclockwise path along the outer cycle from the source of $\mathcal{G}$ to the sink of $\mathcal{G}$. Figure 14 shows an example of a triangulated series-parallel graph.

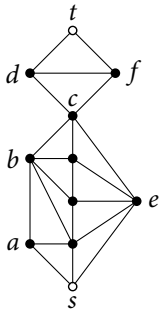


Figure 14: A triangulated series-parallel graph with left boundary path s, a, b, c, d, t and right boundary path s, e, c, f, t.

The series operation on triangulated series-parallel graphs is the same as the normal series operation. The parallel operation takes two triangulated series-parallel graphs $\mathcal{X}$ and $\mathcal{Y}$, neither of which is the initial graph, and

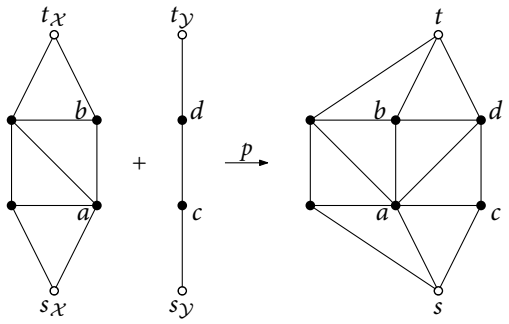


Figure 15: The parallel operation on two triangulated series-parallel graphs $\mathcal{X}$ and $\mathcal{Y}$. The paths along the outer cycles are $p_{\mathcal{X}} = s_{\mathcal{X}}, a, b, t_{\mathcal{X}}$ and $p_y = s_{\mathcal{Y}}, c, d, t_{\mathcal{Y}}$. The new face is triangulated by connecting vertices in $\{a, b\}$ to vertices in $\{c, d\}$. Note that the triangulation where ad is replaced by bc is also valid.

identifies the source of $\mathcal{X}$ with the source of $\mathcal{Y}$, and the sink of $\mathcal{X}$ with the sink of $\mathcal{Y}$. The parallel operation creates a new face f bounded by the right boundary $p_{\mathcal{X}}$ of $\mathcal{X}$ and the left boundary $p_{\mathcal{Y}}$ of $\mathcal{Y}$. This face is triangulated by connecting nonterminal vertices of $p_{\mathcal{X}}$ to nonterminal vertices of $p_{\mathcal{Y}}$ in any way that maintains planarity. The parallel operation is illustrated in Figure 15. Note that the operation is not defined if either graph is the initial graph, since the boundary paths of the initial graph do not have nonterminal vertices.

3.2 SLICEABILITY

We now prove that triangulated series-parallel graphs are sliceable.

Lemma 8. *Triangulated series-parallel graphs have no separating triangles.*

Proof. Let $\mathcal{G}$ be a triangulated series-parallel graph. We prove that $\mathcal{G}$ has no separating triangles by induction on the construction of triangulated series-parallel graphs. Since the initial graph has only two edges, it does not have a separating triangle.

Assume that all triangulated series-parallel graphs with fewer vertices than $\mathcal{G}$ do not have separating triangles. Suppose $\mathcal{G}$ was created from graphs $\mathcal{X}$ and $\mathcal{Y}$ with a series operation. Since neither $\mathcal{X}$ nor $\mathcal{Y}$ contain separating triangles by the induction hypothesis, and since the series operation does not introduce new triangles, graph $\mathcal{G}$ does not have separating triangles.

Suppose $\mathcal{G}$ was created from graphs $\mathcal{X}$ and $\mathcal{Y}$ with a parallel operation. Let $p_{\mathcal{X}}$ be the right boundary path of $\mathcal{X}$ and let $p_{\mathcal{Y}}$ be the left boundary path of $\mathcal{Y}$. Assume for the sake of deriving a contradiction that $\mathcal{G}$ has a separating triangle u, v, w. By the induction hypothesis, neither $\mathcal{X}$ nor $\mathcal{Y}$

contain separating triangles. Hence, $u \in p_{\mathcal{X}}$ and $v, w \in p_{\mathcal{Y}}$ or vice versa. Since u, v, w is a separating triangle, there must be at least one vertex between v and w on $p_{\mathcal{Y}}$. It follows that the left boundary path of $\mathcal{Y}$ has a chord. Note that the initial series-parallel graph does not have chords on its boundary paths, and chords cannot be introduced by the series and parallel operations. Hence, the left boundary path of $\mathcal{Y}$ does not have a chord. Contradiction. It follows that $\mathcal{G}$ does not have separating triangles. This concludes the proof. □

Let $\mathcal{G}$ be a triangulated series-parallel graph with source s and sink t. The *tight extended graph* $E_t(\mathcal{G})$ of $\mathcal{G}$ is the extended graph obtained by adding $l(\mathcal{G})$ and connecting it to the left boundary of $\mathcal{G}$, adding $r(\mathcal{G})$ and connecting it to the right boundary of $\mathcal{G}$, renaming s to $b(\mathcal{G})$ and renaming t to $t(\mathcal{G})$. Note that $E_t(\mathcal{G})$ does not have separating triangles, since the boundary paths of $\mathcal{G}$ are chordless. Figure 16 shows an example of a tight extended graph.

Lemma 9. *Tight extended graphs of triangulated series-parallel graphs are sliceable.*

Proof. Let $\mathcal{G}$ be a triangulated series-parallel graph and let $E_t(\mathcal{G})$ be the tight extended graph of $\mathcal{G}$. We will prove that $E_t(\mathcal{G})$ is sliceable by induction on the construction of triangulated series-parallel graphs. More specifically, we will show that the regular edge labeling of $E_t(\mathcal{G})$ obtained by the following procedure is sliceable: color all edges added by the triangulation step of the parallel operation blue and color all other edges red. Figure 16 shows a triangulated series-parallel graph with the described labeling. If $\mathcal{G}$ is the initial graph, then $E_t(\mathcal{G})$ is trivially sliceable, as shown in Figure 17.

Assume the tight extended graphs of all triangulated series-parallel graphs with fewer vertices than $\mathcal{G}$ are sliceable. If $\mathcal{G}$ was created from graphs $\mathcal{X}$ and $\mathcal{Y}$ with a series operation, then let v be the vertex that was formed by merging the sink of $\mathcal{X}$ and the source of $\mathcal{Y}$. Slice directly above and below v, as shown in Figure 17. The induced corner assignments of the remaining graphs are $E_t(\mathcal{X})$ and $E_t(\mathcal{Y})$, which are sliceable by the induction hypothesis.

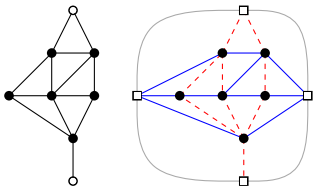


Figure 16: On the left: a triangulated series-parallel graph $\mathcal{G}$. On the right: the tight extended graph $E_t(\mathcal{G})$ of $\mathcal{G}$ with a sliceable regular edge labeling.

If $\mathcal{G}$ was created from graphs $\mathcal{X}$ and $\mathcal{Y}$ with a parallel operation, then let $p_{\mathcal{X}}$ be the right boundary of $\mathcal{X}$ and let $p_{\mathcal{Y}}$ be the left boundary of $\mathcal{Y}$. Let E_s be the edges internal to the cycle obtained by concatenating $p_{\mathcal{X}}$ and $p_{\mathcal{Y}}$. Slice vertically through E_s, as shown in Figure 17. The induced corner assignments of the remaining graphs are $E_t(\mathcal{X})$ and $E_t(\mathcal{Y})$, which are sliceable by the induction hypothesis. We conclude that $E_t(\mathcal{G})$ is sliceable. □

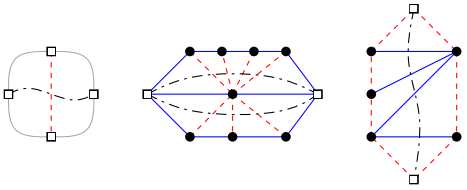


Figure 17: On the left: a sliceable labeling for the tight extended graph of the initial graph. In the middle: $\mathcal{G}$ was created using a series operation. On the right: $\mathcal{G}$ was created using a parallel operation.

We will use this lemma to prove the following theorem.

Theorem 10. *Triangulated series-parallel graphs have a sliceable corner assignment.*

Proof. Let $\mathcal{G}$ be a triangulated series-parallel graph with source s and sink t. We define $E(\mathcal{G})$ as the corner assignment where pole $t(\mathcal{G})$ is connected to t, pole $b(\mathcal{G})$ is connected to s, pole $l(\mathcal{G})$ is connected to every vertex in the left boundary of $\mathcal{G}$ and pole $r(\mathcal{G})$ is connected to every vertex in the right boundary of $\mathcal{G}$. This extended graph is shown in Figure 18. Let the first two slices in $E(\mathcal{G})$ be the slices that cut directly under t and directly above s. The induced corner assignment of the remaining graph is $E_t(\mathcal{G})$, which is sliceable by Lemma 9. □

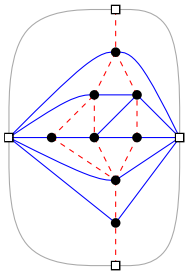


Figure 18: A corner assignment of the graph shown in Figure 16.

4

PYRAMIDS ADJACENT TO THE OUTER CYCLE

In this chapter we study a subset of the one-pyramid extended graphs where the pyramid is adjacent to the outer cycle and present a generic method for proving the sliceability of such extended graphs.

4.1 DEFINITIONS

A *vertical nucleus* $\mathcal{N}$ of $E(\mathcal{G})$ is a subgraph of $E(\mathcal{G})$ such that *(i)* $\mathcal{N}$ contains $b(\mathcal{G})$ and $t(\mathcal{G})$ and *(ii)* the outer cycle of $\mathcal{N}$ is chordless. The definition of *horizontal nucleus* is symmetrical. Although we only consider vertical nuclei in this chapter, all results can be trivially generalized to horizontal nuclei. We will henceforth refer to vertical nuclei as simply *nuclei*. The *left boundary path* of a nucleus $\mathcal{N}$ is the clockwise path along the outer cycle from $b(\mathcal{G})$ to $t(\mathcal{G})$. Similarly, the *right boundary path* is the counterclockwise path along the outer cycle from $b(\mathcal{G})$ to $t(\mathcal{G})$. The *tight extended graph* $E_t(\mathcal{N})$ of a nucleus $\mathcal{N}$ is the extended graph obtained by adding $l(\mathcal{N})$ and connecting it to the left boundary of $\mathcal{N}$, adding $r(\mathcal{N})$ and connecting it to the right boundary of $\mathcal{N}$, renaming $b(\mathcal{G})$ to $b(\mathcal{N})$ and renaming $t(\mathcal{G})$ to $t(\mathcal{N})$. A nucleus of $E(\mathcal{G})$ splits $\mathcal{G}$ into a left subgraph $\mathcal{G}_\ell$ and a right subgraph $\mathcal{G}_r$ with induced corner assignments $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$, in the same way that a vertical slice splits $\mathcal{G}$. Note that since the boundary paths of the nucleus are chordless, the extended graphs $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$ are irreducible triangulations.

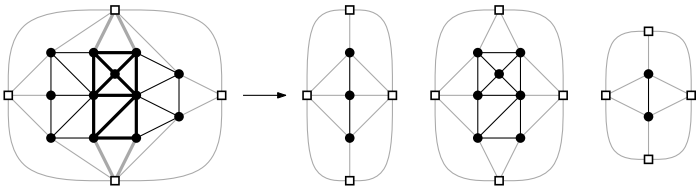


Figure 19: On the left: an extended graph $E(\mathcal{G})$ with a nucleus indicated with bold edges. On the right: the induced corner assignment of the left subgraph, the tight extended graph of the nucleus, and the induced corner assignment of the right graph.

4.2 SLICEABILITY

We now discuss the relation between the sliceability of an extended graph and the sliceability of a nucleus.

Lemma 11. *Let $E(\mathcal{G})$ be an extended graph with a nucleus $\mathcal{N}$. The nucleus splits $\mathcal{G}$ into a left and right subgraph with induced corner assignments $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$. If $E_t(\mathcal{N})$ and $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$ are sliceable, then $E(\mathcal{G})$ is also sliceable.*

Proof. Let $\mathcal{L}_\ell$ be a sliceable regular edge labeling of $E(\mathcal{G}_\ell)$, let $\mathcal{L}_\mathcal{N}$ be a sliceable regular edge labeling of $E_t(\mathcal{N})$ and let $\mathcal{L}_r$ be a sliceable regular edge labeling of $E(\mathcal{G}_r)$. Let $E_{s\ell}$ be the set of edges that connect $\mathcal{G}_\ell$ to $\mathcal{N}$ in $E(\mathcal{G})$ and let E_{sr} be the set of edges that connect $\mathcal{N}$ to $\mathcal{G}_r$ in $E(\mathcal{G})$.

Construct a regular edge labeling $\mathcal{L}$ of $E(\mathcal{G})$ as follows: *(i)* color the edges of $\mathcal{G}_\ell$ according to $\mathcal{L}_\ell$, *(ii)* color the edges of $\mathcal{N}$ according to $\mathcal{L}_\mathcal{N}$, *(iii)* color the edges of $\mathcal{G}_r$ according to $\mathcal{L}_r$, and *(iv)* color the edges of $E_{s\ell}$ and E_{sr} blue. Figure 20 shows an example of this construction. Note that the first two slices (with cut-sets $E_{s\ell}$ and E_{sr}) in $\mathcal{L}$ split $E(\mathcal{G})$ into three graphs with induced corner assignments $E(\mathcal{G}_\ell)$, $E_t(\mathcal{N})$ and $E(\mathcal{G}_r)$, all of which are sliceable. It follows that $E(\mathcal{G})$ is sliceable. □

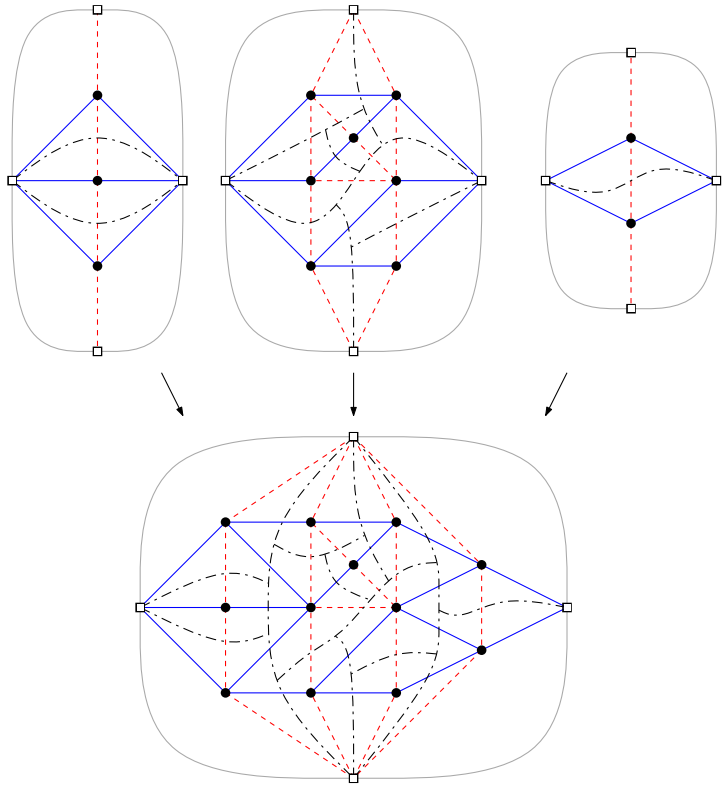


Figure 20: Constructing a sliceable regular edge labeling from sliceable regular edge labelings of a nucleus and the corresponding left and right subgraphs.

Theorem 12. *Let $E(\mathcal{G})$ be a one-pyramid extended graph with a nucleus $\mathcal{N}$ such that $\mathcal{N}$ contains the only pyramid of $E(\mathcal{G})$. If $E_t(\mathcal{N})$ is sliceable, then $E(\mathcal{G})$ is sliceable.*

Proof. The nucleus splits $\mathcal{G}$ into a left and right graph whose corner assignments are sliceable by Theorem 4. If the tight extended graph of the nucleus $\mathcal{N}$ is also sliceable, then $E(\mathcal{G})$ is sliceable by Lemma 11. □

Consider now the generic nucleus $\mathcal{N}_{x,y,z}$ depicted in Figure 21. The tight extended graph of $\mathcal{N}_{x,y,z}$ is sliceable for the labeling shown in the figure. The dotted edges represent paths $b, u_1, \cdots, u_x, t(\mathcal{G})$ and $b(\mathcal{G}), v_1, \cdots, v_y, c$ and $b(\mathcal{G}), w_1 \cdots, w_z, f$ for $x, y, z \geq 0$. All vertices u_i are connected with blue edges to a. Similarly, all vertices v_i and w_i are connected with blue edges to d.

Theorem 13. *Let $E(\mathcal{G})$ be a one-pyramid graph with subgraph $\mathcal{N}_{x,y,z}$ for some $x, y, z \geq 0$. If $\mathcal{G}$ does not contain the edge $(f, t(\mathcal{G}))$, then $E(\mathcal{G})$ is sliceable.*

Proof. We will show that $\mathcal{N}_{x,y,z}$ is a nucleus of $E(\mathcal{G})$. Since the tight extended graph of $\mathcal{N}_{x,y,z}$ is sliceable and contains a pyramid, it will follow from Lemma 12 that $E(\mathcal{G})$ is sliceable.

It remains to show that the boundary paths of $\mathcal{N}_{x,y,z}$ are chordless in $E(\mathcal{G})$. We will prove this by showing that all possible chords induce a separating 3-cycle or separating 4-cycle. Since $E(\mathcal{G})$ is a one-pyramid extended graph, no separating 4-cycles other than the pyramid in $\mathcal{N}_{x,y,z}$ can exist. The paths $b(\mathcal{G}), w_1, \cdots, w_z, f$ and $b, u_1, \cdots, u_x, t(\mathcal{G})$ and $b(\mathcal{G}), v_1, \cdots, v_y, c$ are chordless since any chord would induce a separating 3-cycle.

Consider the left boundary path. Let $p \in \{b(\mathcal{G}), w_1, \cdots, w_z\}$. A chord (p, a) would induce a separating 3-cycle p, a, d. A chord $(w_i, t(\mathcal{G}))$ would induce a separating 4-cycle $w_i, t(\mathcal{G}), a, d$. Note that the chord

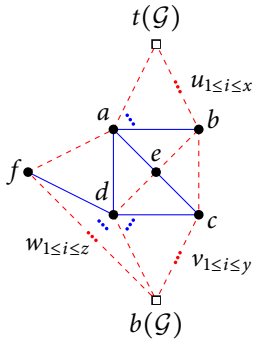


Figure 21: The generic nucleus $\mathcal{N}_{x,y,z}$, labeled such that $E_t(\mathcal{N}_{x,y,z})$ is sliceable.

$(f, t(\mathcal{G}))$ is not permitted by the theorem. We conclude that the left boundary path is chordless.

Consider the right boundary path. Let $p \in \{b(\mathcal{G}), v_1, \cdots, v_y, c\}$ and $q \in \{b, u_1, \cdots, u_x, t(\mathcal{G})\}$ such that $(p, q) \neq (c, b)$. A chord (p, q) would induce a separating 4-cycle p, q, a, d. We conclude that the right boundary path is chordless. It follows that $\mathcal{N}_{x,y,z}$ is a nucleus of $E(\mathcal{G})$. Hence, $E(\mathcal{G})$ is sliceable by Theorem 12. □

5 ROTATING WINDMILLS

In this chapter we study a subset of one-pyramid extended graphs: rotating windmills.

5.1 DEFINITIONS

We call the graph depicted in Figure 22 the *big pyramid* graph. The windmill depicted in Figure 24 is a *rotating windmill* and the four *base rotating windmills* depicted in Figure 23 are rotating windmills. The base rotating windmills are precisely the four corner assignments of the big pyramid graph. If $E(\mathcal{G})$ is a rotating windmill other than the windmill, then we can

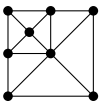


Figure 22: The big pyramid graph.

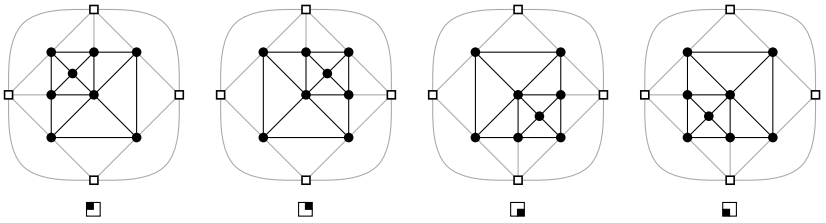


Figure 23: The base rotating windmills.

construct another rotating windmill by replacing the pyramid in $E(\mathcal{G})$ with a big pyramid using one of three construction steps, labeled $\uparrow$, $\curvearrowleft$ and $\curvearrowright$, each depicted in Figure 25. Intuitively, $\uparrow$ extends the rotating windmill in the same direction as the previous extension, $\curvearrowleft$ rotates the direction 90° counterclockwise and $\curvearrowright$ rotates the direction 90° clockwise. Note that the construction steps are not allowed to perform a rotation of 180°. We can uniquely identify a rotating windmill by its *construction sequence*. The

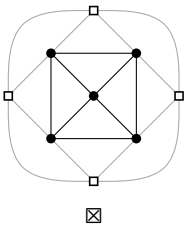


Figure 24: The windmill.

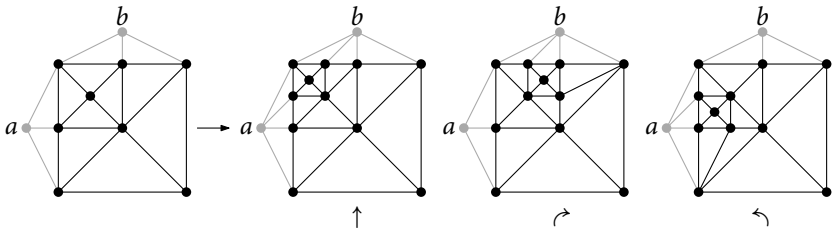


Figure 25: On the left: the big pyramid in the original rotating windmill, along with two of its neighbors in gray. On the right: the results of applying the three construction steps.

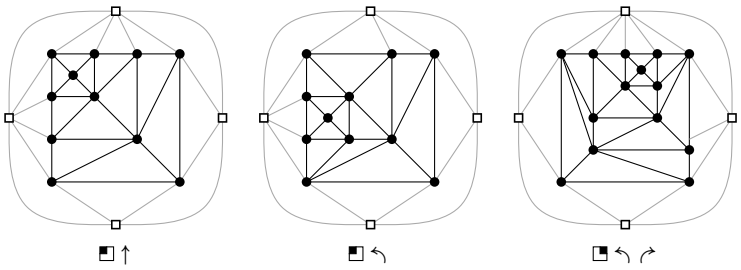


Figure 26: Three rotating windmills.

construction sequence of the windmill is ⊠. The construction sequences of the base rotating windmills depicted in Figure 23 are ◩, ◪, ◪ and ◩. If we apply the construction step $s_{k+1} \in \{\uparrow, \curvearrowleft, \curvearrowright\}$ to a rotating windmill $bs_1 \cdots s_k$ where $k \geq 0$ and $b \in \{◩, ◪, ◪, ◩\}$ and $s_1, \cdots, s_k \in \{\uparrow, \curvearrowleft, \curvearrowright\}$, then the resulting rotating windmill has construction sequence $bs_1 \cdots s_k s_{k+1}$.

Figure 26 shows three rotating windmills. If $E(\mathcal{G})$ is a rotating windmill, then we call $\mathcal{G}$ a *rotating pyramid*. The *inner graph* of a rotating pyramid $\mathcal{G}$ unequal to the pyramid is the largest strict subgraph $\mathcal{G}'$ of $\mathcal{G}$ such that $\mathcal{G}'$ is a big pyramid or a pyramid.

5.2 DRAWING CONVENTIONS

In all our figures of rotating pyramids $\mathcal{G}$, the edges of the outer cycle form a square. The *top side* of $\mathcal{G}$ in such a figure is the path from the topleft vertex of $\mathcal{G}$ up to and including the topright vertex. The definitions of *right side*, *bottom side* and *left side* are analogous. Note that every rotating windmill has two consecutive sides with exactly two vertices, and two consecutive sides with at least two vertices. For instance, every rotating windmill whose construction sequence starts with ◩ has exactly two vertices on its right and bottom side and at least two vertices on its left and top side.

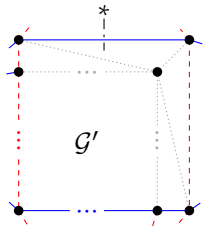


Figure 27: Graph $\mathcal{G}$.

Since we will frequently study subgraphs of rotating windmills, we will use a uniform way of depicting them. Consider the graph $\mathcal{G}$ shown in Figure 27. The small edges incident to the vertices on the outer cycle of $\mathcal{G}$ represent connections to vertices not shown in the figure. The figure shows only the outer cycle of the inner graph $\mathcal{G}'$ of $\mathcal{G}$. The lines along the top, right, bottom and left sides of $\mathcal{G}'$ contain the ⋯-symbol in their center to indicate that there may be zero or more extra vertices on the side. The edges which have not yet been colored in this graph are dotted. The start of a slice is denoted with $*$, and the end of a slice is denoted with $\times$ (not shown in Figure 27).

Every vertex on the top side of $\mathcal{G}'$ is connected to the topleft vertex in the figure, and every vertex on the right side of $\mathcal{G}'$ is connected to the bottomright vertex in the figure. Since $\mathcal{G}'$ is a rotating pyramid, a maximum of two sides of $\mathcal{G}'$ (and they must be consecutive) can have extra vertices.

5.3 NONSLICEABILITY

Mumford and Speckmann [9] stated (without proof) that the rotating windmills with a construction sequence of the form $b \uparrow\uparrow \cdots \uparrow$ where $b \in \{◩, ◪, ◪, ◩\}$ are not sliceable. We now prove a more general result.

Lemma 14. *Let $E(\mathcal{G})$ be an extended graph with a sliceable regular edge labeling $\mathcal{L}$. Let $\mathcal{G}'$ be a subgraph of $\mathcal{G}$ such that the outer cycle of $\mathcal{G}'$ under $\mathcal{L}$ has in clockwise order (i) a nonempty path of red edges followed by a nonempty path of blue edges oriented clockwise, and (ii) a nonempty path of red edges followed by a nonempty path of blue edges oriented counterclockwise. Let $E(\mathcal{G}')$ be the extended graph with labeling $\mathcal{L}'$ induced by coloring the edges of $\mathcal{G}'$ according to $\mathcal{L}$. The labeling $\mathcal{L}'$ is a sliceable labeling for $E(\mathcal{G}')$.*

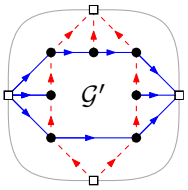


Figure 28: The labeled outer cycle of $\mathcal{G}'$ and the induced corner assignment $E(\mathcal{G}')$ with labeling $\mathcal{L}'$.

Proof. Figure 28 shows an example of the labeling of the outer cycle of $\mathcal{G}'$, the induced corner assignment $E(\mathcal{G}')$ and the labeling of $E(\mathcal{G}')$. Observe that the slices in $\mathcal{L}'$ are exactly the slices in $\mathcal{L}$ that cut through edges of $\mathcal{G}'$. Since $\mathcal{L}$ is a sliceable labeling of $E(\mathcal{G}')$, the labeling $\mathcal{L}'$ must also be sliceable. □

Theorem 15. *Rotating windmills are nonsliceable.*

Proof. We will prove the theorem by structural induction on rotating windmills. Our base case is the windmill, which is not sliceable.

Let $E(\mathcal{G})$ be a rotating windmill and assume that all rotating windmills with fewer vertices are nonsliceable. Assume without loss of generality that the construction sequence of $E(\mathcal{G})$ starts with ◩. For the sake of deriving a contradiction, suppose that $E(\mathcal{G})$ is sliceable. Without loss of generality, we can assume that the first slice in $E(\mathcal{G})$ will be a vertical slice from $t(\mathcal{G})$ to $b(\mathcal{G})$. We will show that any first slice either *(i)* cannot reach $b(\mathcal{G})$ or *(ii)* cuts $E(\mathcal{G})$ in such a way that a smaller graph is forced into a corner assignment that is a rotating windmill. Both cases result in a contradiction.

The graph $\mathcal{G}$ is depicted in Figure 29. The vertices along the outer cycle of $\mathcal{G}$ are connected to the poles in $E(\mathcal{G})$. Since $t(\mathcal{G})$ has only incoming red edges, the edges along the top side of $\mathcal{G}$ must be blue. A similar reasoning forces the coloring of all edges on the outer cycle of $\mathcal{G}$. Let $\mathcal{G}'$ be the inner graph of $\mathcal{G}$. We distinguish four cases.

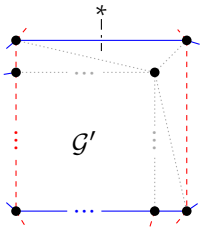


Figure 29: Graph $\mathcal{G}$.

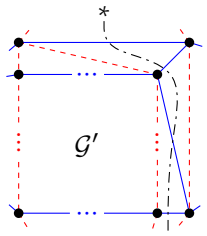
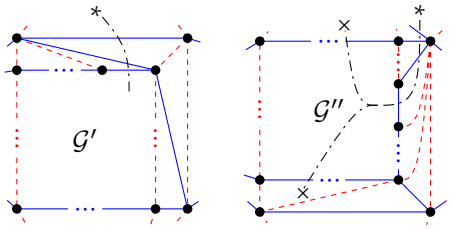


Figure 30: Case 1: graph $\mathcal{G}$.

CASE 1. The first slice does not cut through an edge in the top side of $\mathcal{G}'$, as depicted in Figure 30. As noted previously, the colors of the edges along the outer cycle of $\mathcal{G}$ are forced by the corner assignment. The choice of the slice forces the colors of all dotted edges in Figure 29. The induced corner assignment of $\mathcal{G}'$ is a rotating windmill $E(\mathcal{G}')$ which is smaller than $E(\mathcal{G})$. By the induction hypothesis, $E(\mathcal{G}')$ is not sliceable. Hence, $E(\mathcal{G})$ is also not sliceable. Contradiction.

CASE 2. The top side of $\mathcal{G}'$ has at least two edges and the first slice cuts through the rightmost one, as depicted in Figure 31a. The induced corner assignment of $\mathcal{G}'$ is not a rotating windmill, so we cannot immediately conclude that $E(\mathcal{G})$ is not sliceable. Let us consider the structure of $\mathcal{G}'$. Note that the top side of $\mathcal{G}'$ has more than two vertices. This means that the construction sequence of $E(\mathcal{G})$ must start with ◧ ↷.

The slice that enters $\mathcal{G}'$ in Figure 31a continues at the $*$ in Figure 31b. Let $\mathcal{G}''$ be the inner graph of $\mathcal{G}'$. Note that the slice must enter $\mathcal{G}''$: if it did not, we would be in Case 1 again. It follows that the slice must enter $\mathcal{G}''$ through some edge on the right side of $\mathcal{G}''$. This forces the colors of all dotted edges in the figure. The slice cannot leave $\mathcal{G}''$ through an edge on the bottom side of $\mathcal{G}''$, since the slice cannot continue to $b(\mathcal{G})$ from there. Similarly, it cannot leave through an edge on the top side. Since the first slice does not reach $b(\mathcal{G})$, it cannot be the first slice. Contradiction.

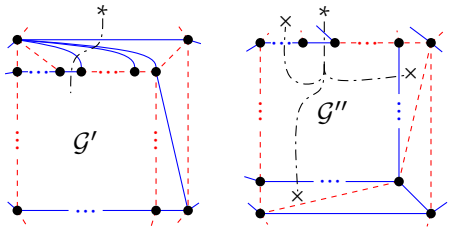


(a) $\mathcal{G}$ (b) $\mathcal{G}'$

Figure 31: Case 2: graphs $\mathcal{G}$ and $\mathcal{G}'$.

CASE 3. The top side of $\mathcal{G}'$ has at least two edges and the first slice does not cut through the rightmost one, as depicted in Figure 32a. Following the same reasoning as in Case 2, we can see that the construction sequence of $E(\mathcal{G})$ must start with ◧ ↷.

The first slice continues at $*$ in Figure 32b. Let $\mathcal{G}''$ be the inner graph of $\mathcal{G}'$. Note that all edges in $\mathcal{G}'$ incident to the topright vertex in $\mathcal{G}'$ must be red. This forces the coloring of all remaining edges. Similarly to Case 2, the first slice cannot continue to $b(\mathcal{G})$ after leaving $\mathcal{G}''$. Since the first slice does not reach $b(\mathcal{G})$, it cannot be the first slice. Contradiction.

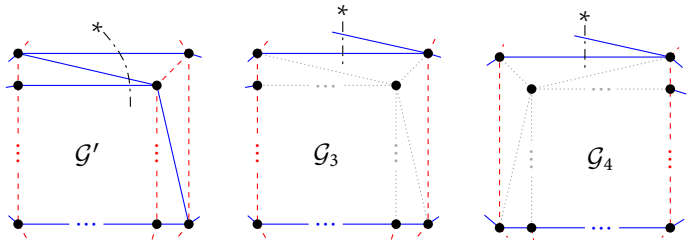


(a) $\mathcal{G}$ (b) $\mathcal{G}'$

Figure 32: Case 3: graphs $\mathcal{G}$ and $\mathcal{G}'$.

CASE 4. The top side of $\mathcal{G}'$ has exactly one edge e and the first slice cuts through e, as depicted in Figure 33a. Since $\mathcal{G}'$ has only two vertices on its top side, the construction sequence of $E(\mathcal{G})$ must start with ◧ ↑ ($\mathcal{G}' = \mathcal{G}_1$) or ◧ ↰ ($\mathcal{G}' = \mathcal{G}_2$). The case $\mathcal{G}' = \mathcal{G}_1$ is shown in Figure 33b and the case $\mathcal{G}' = \mathcal{G}_2$ is shown in Figure 33c. The only difference between $\mathcal{G}_1$ and $\mathcal{G}$ (Figure 29) is that the topright vertex of $\mathcal{G}_1$ has an extra blue edge.

Suppose that $E(\mathcal{G})$ is sliceable for $\mathcal{G}' = \mathcal{G}_1$ (the case $\mathcal{G}' = \mathcal{G}_2$ is similar). Let $\mathcal{L}_\mathcal{G}$ be a sliceable regular edge labeling of $E(\mathcal{G})$ and let $\mathcal{L}_\mathcal{G}[\mathcal{G}_1]$ be the restriction of $\mathcal{L}_\mathcal{G}$ to $\mathcal{G}_1$. All edges along the top side and bottom side of $\mathcal{G}_1$ in $\mathcal{L}_\mathcal{G}[\mathcal{G}_1]$ are blue and all the edges along the left side and right side are red. Let $E(\mathcal{G}_1)$ be the corner assignment of $\mathcal{G}_1$ such that $E(\mathcal{G}_1)$ is a rotating windmill. Coloring the edges of $\mathcal{G}_1$ inside $E(\mathcal{G}_1)$ according to $\mathcal{L}_\mathcal{G}[\mathcal{G}_1]$ yields a sliceable regular edge labeling for $E(\mathcal{G}_1)$ by Lemma 14. But since $E(\mathcal{G}_1)$ is a smaller rotating windmill than $E(\mathcal{G})$, it is not sliceable by the induction hypothesis. Contradiction.



(a) $\mathcal{G}$ (b) $\mathcal{G}_1$ (c) $\mathcal{G}_2$

Figure 33: Case 4: graph $\mathcal{G}$ and two cases for $\mathcal{G}'$: graphs $\mathcal{G}_1$ and $\mathcal{G}_2$.

We conclude that all rotating windmills are nonsliceable. □

The results from the experimental analysis detailed in Chapter 7 lead us to believe that the following conjecture holds:

Conjecture 1. *All nonsliceable one-pyramid extended graphs are rotating windmills.*

If this conjecture holds, then the rotating windmills are exactly the nonsliceable one-pyramid extended graphs.

6 CONJECTURES AND COUNTEREXAMPLES

In this chapter we discuss several conjectures we considered, but ultimately refuted.

Conjecture 2. *If the pyramid in a one-pyramid extended graph $E(\mathcal{G})$ is not connected to any of the four poles, then $E(\mathcal{G})$ is sliceable.*

The smallest counterexample to this conjecture has 18 vertices and is the rotating windmill ◧ ↷ ↷ shown in Figure 34. The nonsliceability of this graph follows directly from Theorem 15.

Conjecture 3. *If a one-pyramid extended graph $E(\mathcal{G})$ is sliceable, then $E(\mathcal{G})$ has a sliceable regular edge labeling $\mathcal{L}$ such that a first slice of $\mathcal{L}$ cuts through the pyramid.*

A smallest counterexample to this conjecture has 13 vertices and its only sliceable regular edge labeling is shown in Figure 34. The (only) first slice is indicated with a dash-dotted line and does not cut through the pyramid.

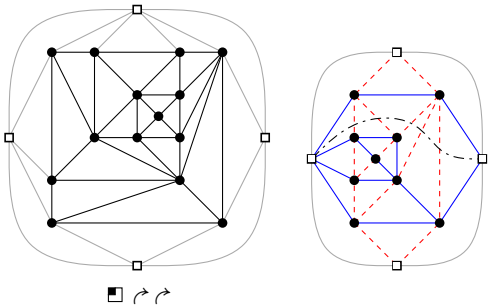


Figure 34: On the left: the smallest counterexample for Conjecture 2. On the right: a smallest counterexample for Conjecture 3.

7 EXPERIMENTAL ANALYSIS

We have created the RELENUM tool to determine the sliceability of all one-pyramid graphs with a given number of vertices n. The main purpose of the tool is to generate conjectures and to find counterexamples to our conjectures. In this chapter, we give a high-level overview of the components of the tool and their interactions, a detailed description of some of the components and finally an overview of our findings.

7.1 OVERVIEW

On a high level, our tool works as follows:

Algorithm SLICEABLEGRAPHS(n)
Input. A number of vertices n.
Output. The sliceable one-pyramid extended graphs with n vertices.

```
1.  for each irreducible triangulation E(G) with n vertices do
2.      if E(G) is a one-pyramid extended graph then
3.          for each regular edge labeling L of E(G) do
4.              if L is sliceable then
5.                  report that E(G) is sliceable
```

We enumerate the irreducible triangulations using the excellent PLANTRI tool by Brinkmann and McKay [2]. Given a natural number n, this tool can (among other things) output all (isomorphism classes of) triangulated graphs of size n with four vertices on the outer cycle. The inner workings of PLANTRI are outside the scope of this thesis. The output for PLANTRI is redirected to RELENUM, which filters out extended graphs with separating 3-cycles, more than one separating 4-cycle, or separating 4-cycles which are not pyramids. We enumerate the regular edge labelings of each extended graph using Fusy's algorithm [5]. Finally, we determine if an extended graph is sliceable by a simple algorithm detailed later in this chapter.

7.2 FUSY'S ALGORITHM

Fusy [5] presented an iterative algorithm to compute the regular edge labelings of a graph. The algorithm sweeps a cycle $\mathcal{C}$ over $E(\mathcal{G})$ while maintaining the following invariants:

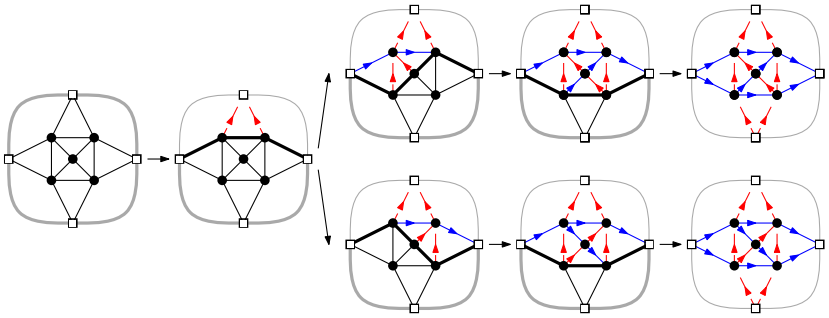


Figure 35: The execution of Fusy's algorithm on the windmill. The cycle $\mathcal{C}$ is indicated with bold edges.

1. The cycle $\mathcal{C}$ contains the two edges $(b(\mathcal{G}), l(\mathcal{G}))$ and $(b(\mathcal{G}), r(\mathcal{G}))$.
2. No edge interior to $\mathcal{C}$ connects two vertices of $\mathcal{C} \setminus \{b(\mathcal{G})\}$.
3. Around each inner vertex outside $\mathcal{C}$ in clockwise order we have four contiguous nonempty sets of incoming blue edges, outgoing red edges, outgoing blue edges and incoming red edges.
4. Around each inner vertex on $\mathcal{C}$ in clockwise order we have three contiguous sets of incoming blue edges, outgoing red edges and outgoing blue edges. The sets of blue edges may be empty, but the set of outgoing red edges must contain at least one edge.

The cycle $\mathcal{C}$ is initially composed of $b(\mathcal{G})$ and the neighbors of $t(\mathcal{G})$. The edges incident to $t(\mathcal{G})$, except the edges that connect $l(\mathcal{G})$ and $r(\mathcal{G})$ to $t(\mathcal{G})$, are colored red and oriented toward $t(\mathcal{G})$. Upon termination, the cycle is $b(\mathcal{G})$, $l(\mathcal{G})$, $b(\mathcal{G})$, $r(\mathcal{G})$. The second invariant implies that all interior vertices satisfy the local conditions for regular edge labelings when the algorithm ends.

The first and last vertices of a path are called *terminal vertices*. An *internal path* of $\mathcal{C}$ is a path of edges inside $\mathcal{C}$ that connects two vertices of $\mathcal{C}$. Let $\mathcal{C}[u, v]$ be the unique path from u to v on $\mathcal{C}$ that does not contain $b(\mathcal{G})$. The cycle formed by the concatenation of an internal path $p = u, \cdots, v$ and $\mathcal{C}[u, v]$ is called $\mathcal{C}_p$. An internal path $p = u, \cdots, v$ is *eligible* if the following conditions are satisfied:

1. The paths p and $\mathcal{C}[u, v]$ both have at least two edges.
2. Each edge interior to $\mathcal{C}_p$ connects a nonterminal vertex of p to a nonterminal vertex of $\mathcal{C}[u, v]$. In particular, the interior of $\mathcal{C}_p$ contains no vertex.
3. Let $\mathcal{C}'$ be the cycle obtained by replacing $\mathcal{C}[u, v]$ by p in $\mathcal{C}$. No interior edge of $\mathcal{C}'$ connects two vertices of $\mathcal{C}' \setminus \{b(\mathcal{G})\}$.

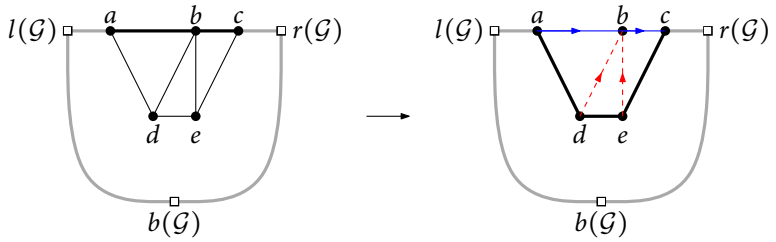


Figure 36: The cycle $\mathcal{C} = b(\mathcal{G}), l(\mathcal{G}), a, b, c, r(\mathcal{G})$ is updated with an eligible path $p = a, d, e, c$ to obtain a new cycle $\mathcal{C}' = b(\mathcal{G}), l(\mathcal{G}), a, d, e, c, r(\mathcal{G})$.

To update the cycle $\mathcal{C}$ with an eligible internal path $p = u, \cdots, v$ of $\mathcal{C}$, we *(i)* color each internal edge of $\mathcal{C}_p$ red and orient it toward $\mathcal{C}[u, v]$, *(ii)* color each edge of $\mathcal{C}[u, v]$ blue and orient it from u to v, and finally *(iii)* update $\mathcal{C}$ by replacing the path $\mathcal{C}[u, v]$ by p. An example is shown in Figure 36.

Fusy [5] orders the eligible paths for a given cycle from left to right. He computes the *minimum regular edge labeling* by choosing the rightmost eligible path in each iteration. Since we want to find *all* regular edge labelings of an extended graph, we compute all eligible paths and recursively compute a regular edge labeling for each of them. This process is shown in Figure 35. Since there are two eligible paths in the second iteration of the algorithm in Figure 35, we compute the new cycle for both and recurse. The following theorem proves that this procedure computes all regular edge labelings of an extended graph.

Theorem 16. *Fusy's algorithm can compute any regular edge labeling of an extended graph.*

Proof. An *intermediate regular edge labeling* $\mathcal{L}_f$ with cycle $\mathcal{C}$ of an extended graph $E(\mathcal{G})$ is a partial coloring and orientation of the edges of $E(\mathcal{G})$ outside $\mathcal{C}$ that can be computed using Fusy's algorithm. Since the edges on $\mathcal{C}[l(\mathcal{G}), r(\mathcal{G})]$ will inevitably be colored blue and oriented toward $r(\mathcal{G})$, we will assume for the remainder of this proof that these edges have already been colored blue and oriented toward $r(\mathcal{G})$.

An intermediate regular edge labeling $\mathcal{L}_f$ with cycle $\mathcal{C}$ *conforms* to a regular edge labeling $\mathcal{L}$ if and only if *(i)* all edges outside $\mathcal{C}$ have the same color and orientation in both $\mathcal{L}_f$ and $\mathcal{L}$, and *(ii)* all edges on $\mathcal{C}[l(\mathcal{G}), r(\mathcal{G})]$ are colored blue and oriented toward $r(\mathcal{G})$ in $\mathcal{L}$.

Let $\mathcal{L}$ be a regular edge labeling of an extended graph $E(\mathcal{G})$. Suppose for the sake of deriving a contradiction that $\mathcal{L}$ cannot be computed using Fusy's algorithm. There must be an intermediate regular edge labeling $\mathcal{L}_f$ with cycle $\mathcal{C}$ such that *(i)* $\mathcal{C}$ contains at least one vertex, *(ii)* $\mathcal{L}_f$ conforms to $\mathcal{L}$, and *(iii)* $\mathcal{C}$ cannot be shrunk to obtain an intermediate regular edge labeling which conforms to $\mathcal{L}$.

Let $p = v_1, \cdots, v_k$ be an internal path of $\mathcal{C}$ such that *(i)* $\mathcal{C}_p$ contains no internal blue edges and *(ii)* all edges of p are blue and oriented toward

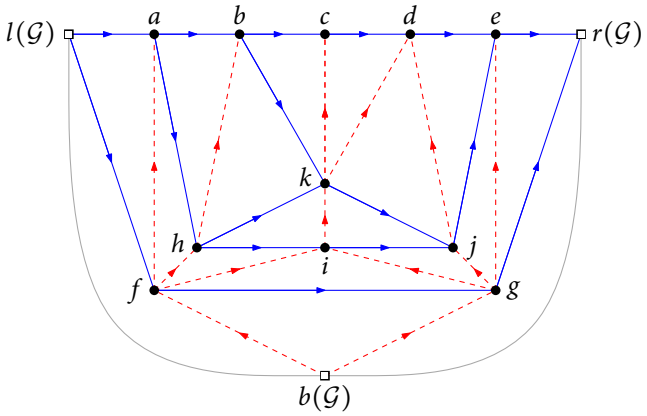


Figure 37: A labeling $\mathcal{L}$ with cycle $\mathcal{C} = b(\mathcal{G}), l(\mathcal{G}), a, b, c, d, e, r(\mathcal{G})$. The cycle $\mathcal{C}_p$ for the initial path $p = l(\mathcal{G}), f, g, r(\mathcal{G})$ contains a blue edge. The path is updated to $p = a, h, j, e$, then to $p = a, h, k, j, e$ and finally to $p = b, k, j, e$.

v_k in $\mathcal{L}$. We now show that such a path always exists. The path p formed by the neighbors of $b(\mathcal{G})$ satisfies *(ii)*. If it does not satisfy *(i)*, then there must be a directed path p' of blue edges in $\mathcal{C}_p$ that connects two vertices of $\mathcal{C}_p$. If p' connects two vertices of $\mathcal{C}$, then replace p by p' and repeat. If p' connects two vertices u and v of p, then replace the subpath from u to v in p by p' and repeat. If p' connects a vertex u of $\mathcal{C}$ to a vertex v of p, then replace the subpath of p before v by p' and repeat. Finally, if p' connects a vertex u of p to a vertex v of $\mathcal{C}$, then replace the subpath of p after u by p' and repeat. This procedure is shown in Figure 37.

By construction of p, all edges internal to $\mathcal{C}_p$ are red. Note that all edges inside $\mathcal{C}_p$ must be oriented toward $\mathcal{C}$ and connect nonterminal edges of p to nonterminal edges of $\mathcal{C}[v_1, v_k]$ to satisfy the regular edge labeling conditions. We will derive a contradiction by showing that p satisfies the three conditions for eligible paths. Since all internal edges in $\mathcal{C}_p$ connect nonterminal vertices of p to nonterminal vertices of $\mathcal{C}[v_1, v_k]$, both p and $\mathcal{C}[v_1, v_k]$ must have at least two edges, satisfying Condition 1. Condition 2 is satisfied by construction of p. Let $\mathcal{C}'$ be the cycle obtained by replacing $\mathcal{C}[v_1, v_k]$ by p in $\mathcal{C}$. Suppose that there exists an interior edge of $\mathcal{C}'$ that connects two vertices of $\mathcal{C}' \setminus \{b(\mathcal{G})\}$. Since $E(\mathcal{G})$ is triangulated, there must be a triangle u, v, w in $E(\mathcal{G})$ such that $u, v, w \in \mathcal{C}' \setminus \{b(\mathcal{G})\}$ and (u, v) and (v, w) are blue and oriented toward w. But then v cannot have an incoming red edge, as illustrated in Figure 38. We conclude that Condition 3 holds.

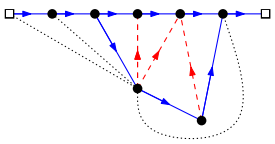


Figure 38: If any of the dotted edges exist, then some vertex will not have an incoming red edge.

Since p satisfies the three conditions for eligible paths, p is an eligible path such that updating $\mathcal{C}$ with p produces an intermediate regular edge labeling that conforms to $\mathcal{L}$. Contradiction. This concludes the proof. □

7.3 SLICEABILITY ALGORITHM

In this section we present a simple algorithm to determine if an extended graph is sliceable. We denote the face left of a directed edge e by $left(e)$ and the face right of e by $right(e)$. Given an extended graph $E(\mathcal{G})$ with a regular edge labeling $\mathcal{L}$, we construct the *labeling dual* $\mathcal{D}$ of $\mathcal{L}$ as follows:

- add a vertex to $\mathcal{D}$ for every face of $E(\mathcal{G})$, except the outer face;
- for each directed blue edge e in $\mathcal{L}$, add a directed red edge from $right(e)$ to $left(e)$ in $\mathcal{D}$; and
- for each directed red edge e in $\mathcal{L}$, add a directed blue edge from $left(e)$ to $right(e)$ in $\mathcal{D}$.

Figure 39 shows the labeling dual of the labeling of the smallest sliceable one-pyramid extended graph. Kant and He [6] use an equivalent dual graph to construct a rectangular dual for an extended graph with a regular edge labeling. Note that labeling duals are the graph equivalent of rectangular duals. The *top border* of a labeling dual is composed of the faces in $E(\mathcal{G})$ which are incident to $t(\mathcal{G})$. The *right border*, *bottom border* and *left border* are defined similarly. The nonterminal vertices along the left border are called *horizontal sources* and the nonterminal vertices along the right border are called *horizontal sinks*. The nonterminal vertices along the bottom border are called *vertical sources* and the nonterminal vertices along the top border are called *vertical sinks*. A *horizontal path* in $\mathcal{D}$ is a path from a horizontal source to a horizontal sink. A *vertical path* in $\mathcal{D}$ is a path from a vertical source to a vertical sink.

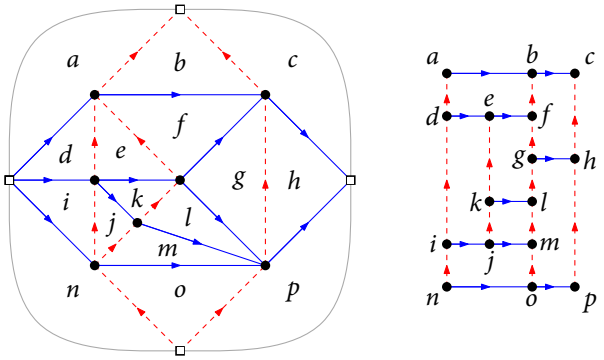


Figure 39: A regular edge labeling and corresponding labeling dual of the smallest sliceable one-pyramid extended graph. The top border of the labeling dual is a, b, c; the right border is c, h, p; the bottom border is p, o, n and the left border is n, i, d, a. The horizontal sources are d and i and the only horizontal sink is h. The only vertical source is o and the only vertical sink is b.

Lemma 17. *A horizontal slice in $\mathcal{L}$ corresponds to a blue horizontal path in $\mathcal{D}$. A vertical slice in $\mathcal{L}$ corresponds to a red vertical path in $\mathcal{D}$.*

Proof. We will prove the lemma for vertical slices; the proof for horizontal slices is symmetrical. Let p be a red vertical path in $\mathcal{D}$. We must show that *(i)* p corresponds to a cut in $\mathcal{G}$ that starts at $b(\mathcal{G})$ and ends at $t(\mathcal{G})$, *(ii)* the cut-set of the cut contains only blue edges, and *(iii)* the boundary paths of the cut are red.

Since p represents a list of consecutive faces that starts at a face incident to $b(\mathcal{G})$ and ends at a face incident to $t(\mathcal{G})$, condition *(i)* is satisfied. Each red edge in $\mathcal{D}$ connects two faces in $\mathcal{L}$ which are separated by a blue edge, satisfying condition *(ii)*. The first and last edge of each boundary path are red because they are incident to $b(\mathcal{G})$ or $t(\mathcal{G})$. Observe that the nonterminal vertices of p represent faces f through which a vertical slice enters and leaves. This means f has at least two blue edges in $\mathcal{L}$. Since a regular edge labeling has no monochromatic triangles, the remaining edge (the edge that contributes to the boundary path) must be red. It follows that condition *(iii)* holds, concluding the proof. □

Let $\mathcal{L}$ be a regular edge labeling for an extended graph $E(\mathcal{G})$ with a first (vertical) slice which cuts $E(\mathcal{G})$ into graphs $\mathcal{G}_\ell$ and G_r with induced corner assignments $E(\mathcal{G}_\ell)$ and $E(G_r)$. Let p be the red vertical path in $\mathcal{D}$ that corresponds to this slice. We wish to obtain labeling duals for the induced regular edge labelings of $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$ directly from $\mathcal{D}$.

Let $t_1, \cdots, t_h$ be the vertices along the top border of $\mathcal{D}$ in clockwise (left-to-right) order. Let $b_1, \cdots, b_k$ be the vertices along the bottom border of $\mathcal{D}$ in clockwise (right-to-left) order. Then p can be expressed as $b_i, \cdots, t_j$ for some $1 \le i \le h$ and $1 \le j \le k$. The left labeling dual $\mathcal{D}_\ell$ is the subgraph of $\mathcal{D}$ with the left border of $\mathcal{D}$, top border $t_1, \cdots, t_j$, right border p and bottom border $b_i, \cdots, b_1$. The right labeling dual $\mathcal{D}_r$ is defined similarly. Figure 40 shows how the labeling dual in Figure 39 is split into a left and a right labeling dual.

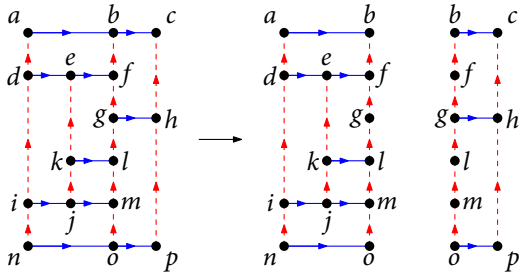


Figure 40: A labeling dual is split along a path $p = o, m, l, g, f, b$ into a left and a right labeling dual.

Combining the discussion above with the observation that a face in a labeling dual corresponds to a vertex in the extended graph, we obtain the following algorithm:

Algorithm IsSliceable($\mathcal{D}$)
Input. The labeling dual of a regular edge labeling $\mathcal{L}$ of an extended graph $E(\mathcal{G})$.
Output. A Boolean indicating if $\mathcal{L}$ is a sliceable labeling of $E(\mathcal{G})$.

```
1.  if the borders of D enclose a single face then
2.      return true
3.  else if there is a red vertical path p in D
4.      Split D along p into D_ℓ and D_r.
5.      return IsSliceable(D_ℓ) and IsSliceable(D_r)
6.  else if there is a blue horizontal path p in D
7.      ▷ symmetrical to the previous case
8.  else
9.      return false
```

Let $\mathcal{D}$ be a sliceable labeling dual with n vertices. Since there are $n-1$ slices in $\mathcal{D}$, the IsSliceable algorithm will be called $n-1$ times. In each call, it takes $O(n)$ time in the worst case to determine if the borders of $\mathcal{D}$ enclose a single face. Finding the next red vertical or blue horizontal path in $\mathcal{D}$ takes $O(n)$ time using a depth first search. Splitting $\mathcal{D}$ into $\mathcal{D}_\ell$ and $\mathcal{D}_r$ takes $O(n)$ time, which brings the total running time to $O(n^2)$.

Figure 41 shows the rectangular dual that corresponds to a worst-case example for our algorithm. Note that $m = \Theta(n)$. The first slice splits the rectangular dual into a left dual $\mathcal{D}_\ell$ and a right dual $\mathcal{D}_r$. The rectangle marked with R has $O(m)$ vertices on its left side in the labeling dual. The algorithm performs $O(m)$ recursive calls to verify the sliceability of $\mathcal{D}_r$. In each of these $O(m)$ calls, the algorithm checks if the borders of the current labeling dual enclose a single face. If the algorithm determines this by starting at the topleft vertex and walking down along the left side of R, then this takes $O(m)$ time. There are symmetric worst case examples for other execution orders. We conclude that the algorithm needs $\Omega(n^2)$ time to determine if the example is sliceable.

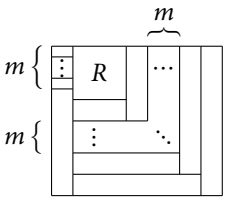


Figure 41: A worst-case example for IsSliceable.

Theorem 18. *The IsSliceable algorithm determines if a labeling dual is sliceable in $\Theta(n^2)$ worst case time.*

7.4 RESULTS

We used the RELENUM tool to determine the sliceability of all many-pyramid extended graphs with at most 19 vertices. The experiments were performed on the TIGER server of the Department of Mathematics

n	extended graphs	many (nonsliceable)		one (nonsliceable)	
5	1				
6	2				
7	8				
8	38				
9	219	1	(1)	1	(1)
10	1,404	1			
11	9,714	7		1	
12	70,454	35	(2)	3	(1)
13	527,235	180	(4)	11	
14	4,037,671	922	(4)	34	
15	31,477,887	4,726	(36)	128	(2)
16	249,026,400	24,218	(92)	449	
17	1,994,599,707	124,382	(274)	1,650	
18	16,147,744,792	640,747	(1,166)	6,054	(5)
19	131,959,532,817	3,313,477	(4,519)	22,451	

Table 1: The number of extended graphs, the number of many-pyramid extended graphs and the number of one-pyramid extended graphs (each with the number of nonsliceable ones) with at most 19 vertices.

and Computer Science of the Eindhoven University of Technology. The machine has an Intel Xeon Q6600 CPU with four cores, each at a clock speed of 2.4GHz and a total of 4GB internal memory. The tool (using only one CPU) processed all extended many-pyramid extended graphs with at most 14 vertices in under a minute. It took 4 minutes to process the extended graphs with 15 vertices and 36 minutes to process the ones with 16 vertices. Computing the results for 17, 18 and 19 vertices took 5 hours, 20 hours and 14 days, respectively.

The results are listed in Table 1. The table shows, for instance, that there are 219 extended graphs (more precisely, isomorphism classes) with 9 vertices. Only one of those graphs is a one-pyramid graph, and it is not sliceable. All of the nonsliceable one-pyramid extended graphs have been confirmed to be rotating windmills. This discovery led to the conjecture that all rotating windmills are nonsliceable, which we proved in Theorem 15. We believe that rotating windmills are in fact the only nonsliceable one-pyramid extended graphs. We have not been able to prove this conjecture yet.

Figure 42 shows all nonsliceable many-pyramid graphs with at most 14 vertices and nine such graphs with 15 vertices. All pyramids have been marked with a bold outline and a gray background. In graphs (k), (l) and (m), a big pyramid has been marked with a hatched gray background. It appears that a many-pyramid extended graph is nonsliceable if its rotating pyramids share vertices and cover most of the graph.

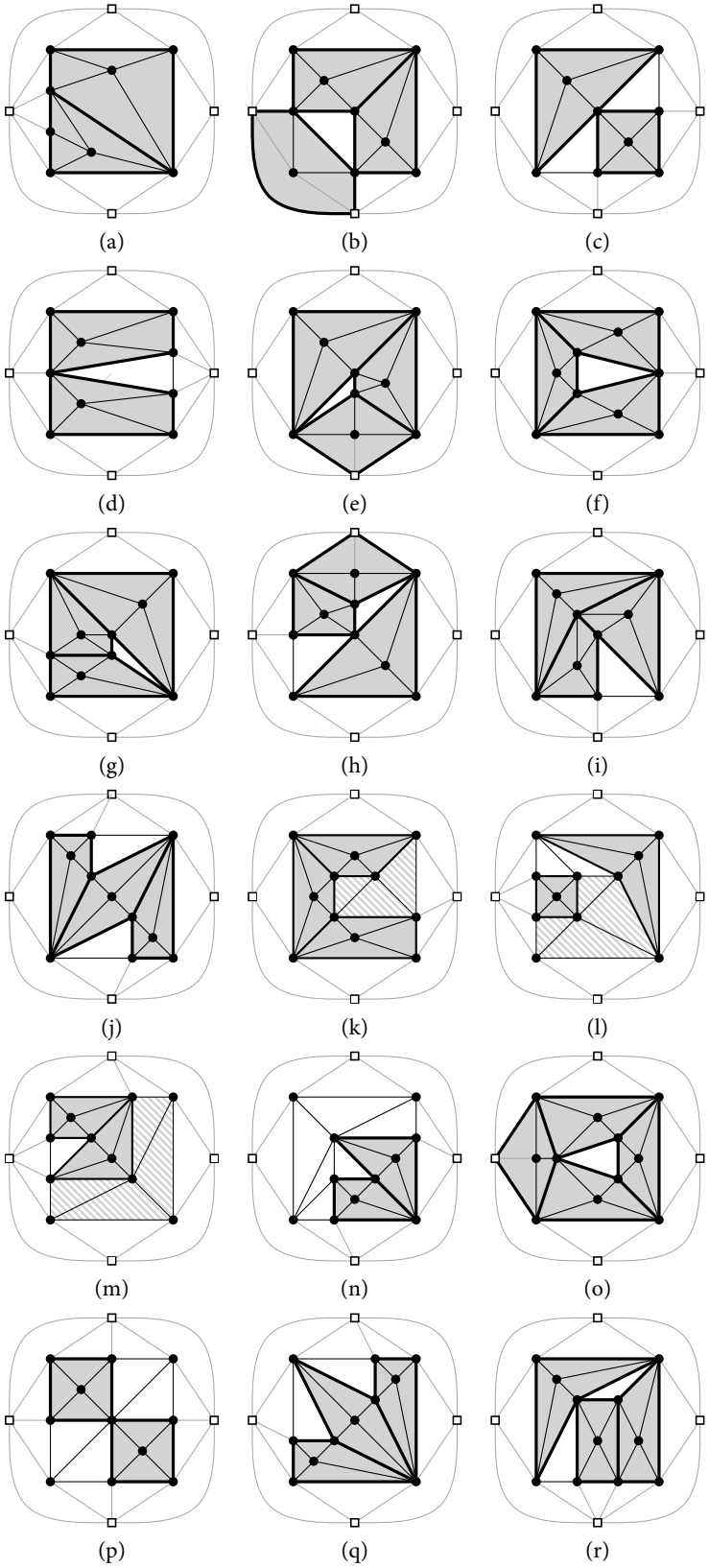


Figure 42: All nonsliceable many-pyramid graphs with at most 14 vertices and several such graphs with 15 vertices.

8

CONCLUSIONS AND FUTURE WORK

In this thesis, we showed that the triangulated series-parallel graphs and a subset of the one-pyramid graphs whose pyramid is adjacent to the outer cycle are sliceable. Moreover, we showed that rotating windmills are not sliceable, and we have a strong indication that all nonsliceable one-pyramid extended graphs are rotating windmills.

An important open problem is to prove that all nonsliceable one-pyramid extended graphs are rotating windmills. In addition, the problem of characterizing the sliceable subset of many-pyramid graphs remains open. Though our RELENUM tool has generated all nonsliceable many-pyramid extended graphs with at most 19 vertices, we have not analyzed their structure yet.

[1] J. Bhasker and S. Sahni. A linear time algorithm to check for the existence of a rectangular dual of a planar triangulated graph. *Networks*, 17(3):307–317, 1987.

[2] G. Brinkmann and B. McKay. Fast generation of planar graphs. *MATCH Communications in Mathematical and in Computer Chemistry*, 58(2):323–357, 2007.

[3] P. Dasgupta and S. Sur-Kolay. Slicible rectangular graphs and their optimal floorplans. *ACM Transactions on Design Automation of Electronic Systems*, 6(4):447–470, 2001.

[4] D. Eppstein, E. Mumford, B. Speckmann, and K. Verbeek. Area-universal rectangular layouts. In *Proc. 25th ACM Symposium on Computational Geometry*, pages 267–276, 2009.

[5] É. Fusy. Transversal structures on triangulations: A combinatorial study and straight-line drawings. *Discrete Mathematics*, 309(7):1870–1894, 2009.

[6] G. Kant and X. He. Two algorithms for finding rectangular duals of planar graphs. In *Graph-Theoretic Concepts in Computer Science*, pages 396–410. Springer, 1994.

[7] K. Koźmiński and E. Kinnen. Rectangular duals of planar graphs. *Networks*, 15(2):145–157, 1985.

[8] E. Mumford. Drawing Graphs for Cartographic Applications. PhD thesis. TU Eindhoven, 2008.

[9] E. Mumford and B. Speckmann. When a graph is sliceable. Unpublished manuscript. 2007.

[10] N. Robertson and P. Seymour. Graph minors–a survey. In *Surveys in combinatorics 1985*, pages 153–171. Cambridge Univ. Press, 1985.

[11] P. Ungar. On diagrams representing maps. *Journal of the London Mathematical Society*, 1(3):336–342, 1953.

[12] G. Yeap and M. Sarrafzadeh. Sliceable floorplanning by graph dualization. *SIAM Journal on Discrete Mathematics*, 8(2):258–280, 1995.