Towards Characterizing Graphs with a Sliceable Rectangular Dual

Vincent Kusters[1][*] and Bettina Speckmann[2]

[1] Department of Computer Science, ETH Zürich, Switzerland.
vincent.kusters@inf.ethz.ch
[2] Department of Computer Science, TU Eindhoven, The Netherlands.
b.speckmann@tue.nl

Abstract. Let $\mathcal{G}$ be a plane triangulated graph. A rectangular dual of $\mathcal{G}$ is a partition of a rectangle R into a set $\mathcal{R}$ of interior-disjoint rectangles, one for each vertex, such that two regions are adjacent if and only if the corresponding vertices are connected by an edge. A rectangular dual is sliceable if it can be recursively subdivided along horizontal or vertical lines. A graph is *rectangular* if it has a rectangular dual and *sliceable* if it has a sliceable rectangular dual. There is a clear characterization of rectangular graphs. However, a full characterization of sliceable graphs is still lacking. The currently best result (Yeap and Sarrafzadeh, 1995) proves that all rectangular graphs without a separating 4-cycle are sliceable. In this paper we introduce a recursively defined class of graphs and prove that these graphs are precisely the nonsliceable graphs with exactly one separating 4-cycle.

1 Introduction

Let $\mathcal{G}$ be a plane triangulated graph. A *rectangular dual* of $\mathcal{G}$ is a rectangular partition $\mathcal{R}$ such that *(i)* no four rectangles meet in the same point, *(ii)* there is a one-to-one correspondence between the rectangles in $\mathcal{R}$ and the vertices of $\mathcal{G}$, and *(iii)* two rectangles in $\mathcal{R}$ share a common boundary segment if and only if the corresponding vertices of $\mathcal{G}$ are connected. A graph can have exponentially many rectangular duals [6], but might not even have a single one. Rectangular duals have a variety of applications, for example, as rectangular cartograms in cartography or as floorplans in architecture and VLSI design.

There are several types of rectangular duals that are of particular interest. Often it is desirable to assign certain areas to each rectangle. A recent paper by Eppstein et al. [8] studies *area-universal* rectangular duals, which have the property that any assignment of areas to rectangles can be realized by a combinatorially equivalent rectangular dual. A rectangular dual is *sliceable* if it can be recursively subdivided along horizontal or vertical lines (such duals are also
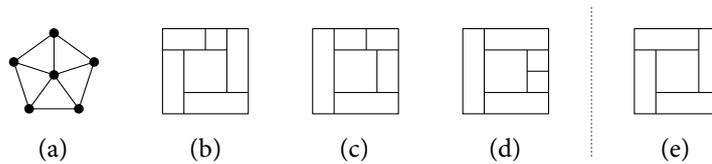
1

Fig. 1. A graph (a) with rectangular duals (b)-(d) and a rectangular dual of a different graph (e): (b) is not sliceable and not area-universal, (c) is sliceable and not area-universal, (d) is sliceable and area-universal, and (e) is area-universal and not sliceable.

called guillotine floorplans and can be constructed by glass cuts). While it is generally difficult to determine if an area assignment is feasible and to compute the corresponding layout of the rectangles, it is very easy to do so for sliceable duals. Furthermore, sliceable duals more easily facilitate certain layout steps in VLSI layout. Sliceability does not imply area-universality or vice versa (see Fig. 1).

A graph is *rectangular* if it has a rectangular dual and *sliceable* if it has a sliceable rectangular dual. Ungar [20], Bhasker and Sahni [4], and Koźmiński and Kinnen [12] independently gave equivalent characterizations of the rectangular graphs. Eppstein et al. [8] characterized the area-universal rectangular duals. However, despite an active interest in sliceable rectangular duals, a full characterization of sliceable graphs is still lacking. The currently best result by Yeap and Sarrafzadeh [22] from 1995 proves that all rectangular graphs without a separating 4-cycle are sliceable. Dasgupta and Sur-Kolay [7] modified the approach of Yeap and Sarrafzadeh and claimed two sufficient conditions for sliceability. However, Mumford [15] discovered a critical flaw that invalidates their results.[3]

Related work. Rectangular duals have been studied extensively by the VLSI community. Sliceable layouts more easily facilitate certain steps in the layout process [16]. For instance, the problem of minimizing the perimeter or area of modules in a rectangular layout according to a given measure can be solved in polynomial time for sliceable layouts, but is NP-complete in general [17]. Several papers focus on restricted classes of sliceable and nonsliceable graphs [5,18].

Rectangular duals are also studied in the context of rectangular cartograms, which represent geographic regions by rectangles. The positioning and adjacencies of these rectangles are chosen to suggest their geographic locations and their areas correspond to the numeric values that the cartogram communicates. Van Kreveld and Speckmann [13] gave the first algorithms to compute rectangular cartograms. Eppstein et al. [8] present a numerical algorithm for area-universal rectangular duals which computes a cartogram with approximately the correct areas. For sliceable rectangular duals one can easily compute a combinatorially equivalent rectangular dual with exactly the specified area assignment, if such a rectangular dual exists. Several papers consider *rectilinear duals*: a generalization of rectangular duals which uses simple (axis-aligned) rectilinear polygons instead of rectangles. Every triangulated graph has a rectilinear dual where ev-

---

[3] Confirmed by Dasgupta and Sur-Kolay, personal communication, 2011-2013.

ery polygon has eight sides, and eight sides are sometimes necessary [10,14,23]. A series of papers studies the question of how many sides are required to respect all adjacencies and area requirements in general. De Berg, Mumford and Speckmann [3] gave the first bound by showing that forty sides per polygon is always sufficient. After several intermediate results, Alam et al. [2] finally closed the gap by proving that eight sides per polygon is always sufficient.

Sliceable rectangular duals are also called *guillotine partitions* or *guillotine layouts*. In this context a different notion of equivalence is used, which is not based on a dual graph. Specifically, two guillotine partitions are equivalent if they have the same *structure tree* [19]. Yao et al. [21] show that the asymptotic number of guillotine partitions is the nth Schröder number. Ackerman et al. [1] derive the asymptotic number of guillotine partitions in higher dimensions.

Results and organization. It is comparatively easy to see that the class of sliceable graphs is not closed under minors. Hence we need to explore different approaches to characterize them. In Section 3 we introduce a recursively defined class of graphs, so-called *rotating pyramids*, which contain exactly one separating 4-cycle. We conjecture that configurations of rotating pyramids determine if a graph is sliceable. We verify our conjecture for the graphs that contain exactly one separating 4-cycle. The nonsliceable graphs in this class are exactly the graphs that reduce to *rotating windmills*: rotating pyramids with a specific corner assignment. In Section 4 we prove that rotating windmills are not sliceable and in Section 5 we argue that all other graphs with exactly one separating 4-cycle are sliceable.

2 Preliminaries

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 $E(\mathcal{G})$. These vertices are labeled $t(\mathcal{G})$, $r(\mathcal{G})$, $b(\mathcal{G})$ and $l(\mathcal{G})$ in clockwise order and are called the *poles* of $E(\mathcal{G})$. The vertices of the original graph $\mathcal{G}$ are called the *interior* vertices. Since choosing the extended graph fixes the vertices that correspond to the four corners (and hence the vertices along the four sides) of the rectangular dual, extended graphs are also called *corner assignments* (Fig. 2).

Fig. 2. An extended graph $E(\mathcal{G})$ and the corresponding rectangular dual.

A *separating k-cycle* of an extended graph $E(\mathcal{G})$ is a k-cycle with vertices both inside and outside the cycle. A *triangle* is a 3-cycle. The *outer cycle* of a plane graph is the cycle formed by the edges incident to the unbounded face. An *irreducible triangulation* is a plane graph without separating triangles and where all interior faces are triangles and
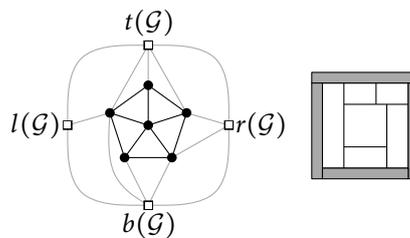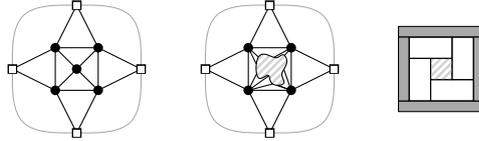
Fig. 3. The windmill, the generalized windmill (the hatched shape is an arbitrary graph), and a rectangular dual of the generalized windmill.

the outer face is a quadrangle. A graph $\mathcal{G}$ has a rectangular dual if and only if $\mathcal{G}$ has an extended graph which is an irreducible triangulation [4,12,20].

Sliceable graphs. A rectangular partition is *sliceable* if it can be recursively subdivided along horizontal or vertical lines. 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 a graph has only polynomially many corner assignments, we consider only extended graphs from now on. The smallest nonsliceable extended graph is the *windmill* depicted in Fig. 3. This extended graph can be generalized to a *generalized windmill* by replacing the center vertex with an arbitrary graph. All generalized windmills are nonsliceable.

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 of $\mathcal{G}$ with cut-set S is *vertical* if the edges dual to S form a path from an interior face incident to $t(\mathcal{G})$ to an interior face incident to $b(\mathcal{G})$. Order the edges in the cut-set $e_1, \ldots, e_m$, according to the order in which they are traversed by the dual path. The *left vertex* of e_i is the endpoint of e_i that is in the same component as $l(\mathcal{G})$ in the graph obtained by deleting $t(\mathcal{G})$, $b(\mathcal{G})$, and S from $E(\mathcal{G})$. The *right vertex* is defined analogously. Let the *left boundary walk* $W_\ell = t(\mathcal{G}), u_1, \ldots, u_\ell, b(\mathcal{G})$ be the sequence of left endpoints of $e_1, \ldots, e_m$ (removing consecutive duplicates), and let the *right boundary walk* $W_r = t(\mathcal{G}), v_1, \ldots, v_r, b(\mathcal{G})$ be the sequence of right endpoints of $e_1, \ldots, e_m$ (removing consecutive duplicates). A walk is a path if it visits every vertex at most once. A path $v_1, \ldots, v_k$ is *chordless* if and only if v_i and v_j are not adjacent for each $1 \leq i < j - 1 \leq k$. A vertical cut is a *vertical slice* if its boundary walks are chordless paths (Fig. 4). A vertical slice divides $\mathcal{G}$ into $\mathcal{G}_l$ and $\mathcal{G}_r$. Horizontal cuts, top and bottom boundary walks and horizontal slices are defined analogously.

Regular edge labelings. 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 (dashed) and blue (solid) 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
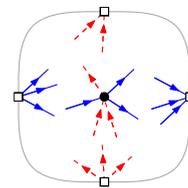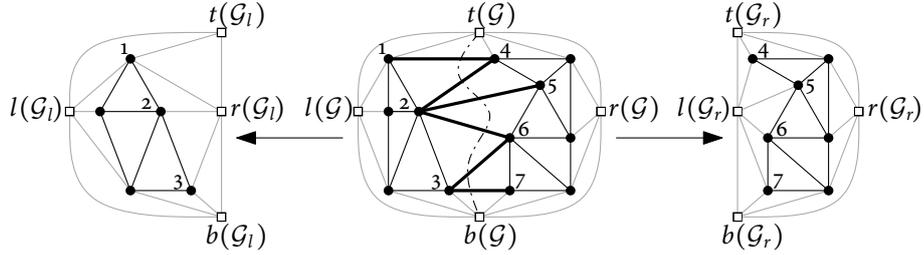
Fig. 4. An extended graph $E(\mathcal{G})$ with a vertical slice indicated by a dash-dotted line and the corresponding $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$. The edges of the cut-set are bold. The boundary paths are $t(\mathcal{G}), 1, 2, 3, b(\mathcal{G})$ and $t(\mathcal{G}), 4, 5, 6, 7, b(\mathcal{G})$. Both boundary paths are chordless. Figure based on [22].

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.

A regular edge labeling is *sliceable* if its corresponding rectangular dual is sliceable. One can find a regular edge labeling and construct the corresponding rectangular dual in linear time [11]. A *regular edge coloring* is a regular edge labeling, without the edge directions. A regular edge coloring uniquely determines a regular edge labeling [9, Proposition 2]. A *monochromatic triangle* is a triangle where all edges have the same color. A regular edge labeling (of an irreducible triangulation) induces no monochromatic triangles [9, Lemma 1].
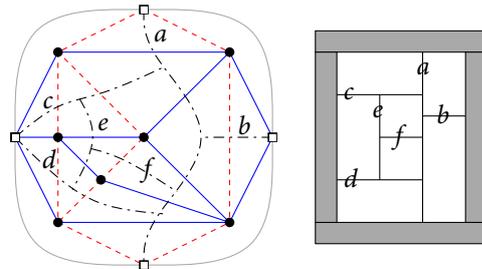
Fig. 5. A regular edge labeling and corresponding rectangular dual. Letters indicate the slices.

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 (see Fig. 5). 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 Fig. 5.

k-pyramid extended graphs. A *pyramid* is a 4-cycle with exactly one vertex in its interior. A *k-pyramid extended graph* is an irreducible triangulation $E(\mathcal{G})$ such that $\mathcal{G}$ has no cut-vertices, $\mathcal{G}$ has exactly k separating 4-cycles, and all separating 4-cycles in $E(\mathcal{G})$ are pyramids. We argue that it is sufficient for our investigation of sliceability to consider only k-pyramid extended graphs with $k \geq 1$. Firstly, we may assume $\mathcal{G}$ has no cut-vertex (all omitted proofs are in the full version of the paper):

Lemma 1. *Let $E(\mathcal{G})$ be an extended graph such that $\mathcal{G}$ has a cut-vertex v. Then v is adjacent to two opposite poles, say $t(\mathcal{G})$ and $b(\mathcal{G})$. Slice immediately left and immediately right of v. Then $E(\mathcal{G})$ is sliceable if and only if the three extended graphs that result from the two slices are sliceable.*

Secondly, Mumford [15] showed that it is sufficient to consider extended graphs $E(\mathcal{G})$ such that all separating 4-cycles in $\mathcal{G}$ are pyramids. Her proof directly extends to separating 4-cycles in $E(\mathcal{G})$ instead of $\mathcal{G}$, which immediately proves that generalized windmills (Fig. 3) are nonsliceable. Finally, 0-pyramid extended graphs are always sliceable [22].

Yeap and Sarrafzadeh's algorithm. In Section 5, we explicitly construct slices in a manner which is based on the algorithm by Yeap and Sarrafzadeh [22]. In Theorem 1 below we give a stronger version of their result and also add a missing case which was overlooked in their original analysis. A cycle C in $E(\mathcal{G})$ splits the plane into two parts: a bounded region and an unbounded region. We say that vertices in the bounded region including C are *enclosed* by C.

Theorem 1. *Let $E(\mathcal{G})$ be a k-pyramid extended graph ($k \geq 0$). Then there exists a vertical cut S such that (i) the left boundary walk P_ℓ of S is a chordless path that contains only vertices with distance 2 to $r(\mathcal{G})$ in $E(\mathcal{G}) \setminus \{t(\mathcal{G}), l(\mathcal{G}), b(\mathcal{G})\}$ and (ii) if the cycle $C_r := \langle r(\mathcal{G}), P_\ell, r(\mathcal{G}) \rangle$ does not enclose a pyramid, then S is a vertical slice. Analogous statements hold for $t(\mathcal{G})$, $l(\mathcal{G})$ and $b(\mathcal{G})$. Consequently, $E(\mathcal{G})$ is sliceable if $k = 0$.*

The following corollary of Lemma 1 gives a final simplification of our problem.

Lemma 2. *Let $E(\mathcal{G})$ be an extended graph with pole p such that p has only one neighbour v in $\mathcal{G}$. Let $E(\mathcal{G}')$ be the extended graph obtained by deleting v from $\mathcal{G}$ and connecting the neighbours of v in $\mathcal{G}$ to p. Then $E(\mathcal{G})$ is sliceable if and only if $E(\mathcal{G}')$ is sliceable.*

Exhaustively applying Lemma 2 to an extended graph $E(\mathcal{G})$ *reduces* $E(\mathcal{G})$ to an extended graph $E(\mathcal{G}')$. We say that $E(\mathcal{G}')$ is *reduced*. The extended graphs $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$ resulting from a slice in $E(\mathcal{G})$ might not be reduced even if $E(\mathcal{G})$ is. In this sense, Lemma 2 is different from Lemma 1 and Mumford's observation. In the following we focus on the 1-pyramid extended graphs, among which are both sliceable and nonsliceable extended graphs. The smallest nonsliceable one is the windmill in Fig. 3.

3 Rotating pyramids and windmills

The graph on the right is the *big pyramid* graph. *Rotating windmills* are recursively defined as follows. The windmill (see Fig. 3) is a rotating windmill. Furthermore, the extended graphs depicted in Fig. 6 are *base rotating windmills*: they are four corner assignments of the big pyramid graph. If $E(\mathcal{G})$ is a rotating windmill other than
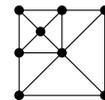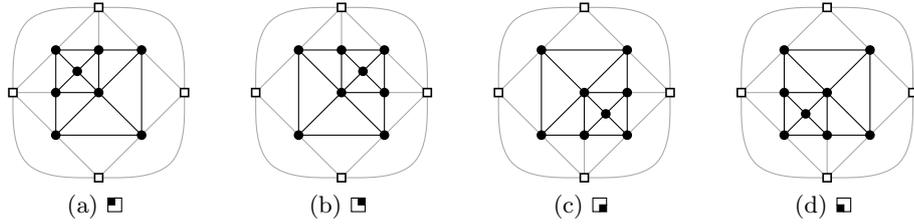
Fig. 6. The four base rotating windmills.

the windmill, then we can construct another rotating windmill by replacing the pyramid in $E(\mathcal{G})$ with a big pyramid using one of three construction steps, labeled $\uparrow$, $\nwarrow$ and $\nearrow$, each depicted in Fig. 7.

Intuitively, $\uparrow$ extends the rotating windmill in the same direction as the previous extension, $\nwarrow$ rotates the direction 90° counterclockwise and $\nearrow$ 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 construction sequence of the windmill is ⊠. The construction sequences of the base rotating windmills are ◨, ◧, ◩ and ◪. If we apply a construction step $s_{k+1} \in \{\uparrow, \nwarrow, \nearrow\}$ to a rotating windmill $bs_1 \cdots s_k$ where $k \geq 0$, $b \in \{◨, ◧, ◩, ◪\}$, and $s_1, \ldots, s_k \in \{\uparrow, \nwarrow, \nearrow\}$, then the resulting rotating windmill has construction sequence $bs_1 \cdots s_k s_{k+1}$. Fig. 8 shows three examples. If $E(\mathcal{G})$ is a rotating windmill, then we call $\mathcal{G}$ a *rotating pyramid*. For a given rotating pyramid $\mathcal{G}$, which is not the pyramid, the *inner graph* $\mathcal{G}'$ is defined as the largest strict subgraph of $\mathcal{G}$ such that $\mathcal{G}'$ is a rotating pyramid.

Drawing conventions. We draw the edges of the outer cycle of a rotating pyramid $\mathcal{G}$ as a square. The *top side* of $\mathcal{G}$ is the path from the topleft vertex of $\mathcal{G}$ to the topright vertex (including both). The definitions of *right side*, *bottom side* and *left side* are analogous. Every rotating windmill has two consecutive sides with exactly two vertices, and two consecutive sides with at least two vertices.
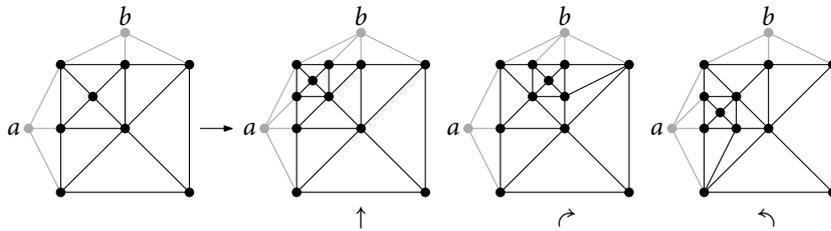

Fig. 7. On the left: the big pyramid in a rotating windmill, along with two of its neighbors in gray. On the right: the results of applying the three construction steps.
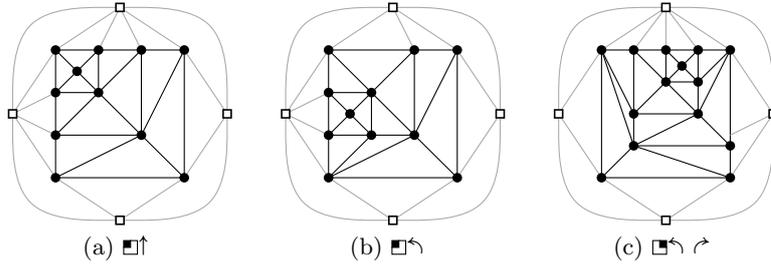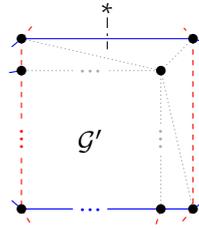
Fig. 8. Three rotating windmills.

Consider the graph $\mathcal{G}$ on the right. The partially drawn edges incident to the vertices on the outer cycle of $\mathcal{G}$ represent connections to vertices not shown in the figure. The inner graph $\mathcal{G}'$ of $\mathcal{G}$ is represented by only its outer cycle; its interior vertices (if any) are not shown. The lines along the top, right, bottom and left sides of $\mathcal{G}'$ contain the $\cdots\cdot$ symbol in their center to indicate that there may be zero or more extra vertices on the side. The edges whose color is not uniquely determined are gray (dotted). The start of a slice is denoted with $*$, and the end of a slice is denoted with $\times$ (not shown). 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.

4   Rotating windmills are not sliceable

Before we can prove the main result of this section, we need the following lemma:

Lemma 3. *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 $E(\mathcal{G}')$.*

Proof. The figure 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.   □
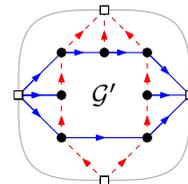
Theorem 2. *Extended graphs that reduce to rotating windmills are not sliceable.*

Proof. Since the reduction operation preserves sliceability, it is sufficient to consider rotating windmills. 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 and consider a sliceable regular edge labeling. We assume wlog that the first slice in $E(\mathcal{G})$ is a vertical slice from $t(\mathcal{G})$ to $b(\mathcal{G})$. We 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.

Fig. 9. Graph $\mathcal{G}$.

See Fig. 9. 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.

Case 1. The first slice does not cut through an edge in the top side of $\mathcal{G}'$, see Fig. 10. 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 Fig. 9. 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.

Fig. 10. Case 1.

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 Fig. 11(a). 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 Fig. 11(a) continues at the $*$ in Fig. 11(b). 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 top or bottom side of $\mathcal{G}''$, since the slice cannot continue to $b(\mathcal{G})$ from there. Since the first slice does not reach $b(\mathcal{G})$, it cannot be the first slice. Contradiction.

Case 3. The top side of $\mathcal{G}'$ has at least two edges and the first slice does not cut through the rightmost one, see Fig. 11(c). Hence, the construction sequence of $E(\mathcal{G})$ must start with ◪↷. The first slice continues at $*$ in Fig. 11(d). Let $\mathcal{G}''$ be

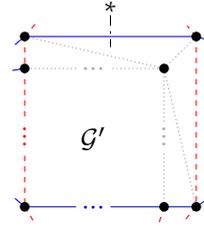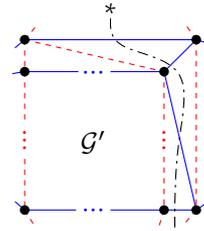(a) Case 2: $\mathcal{G}$. (b) Case 2: $\mathcal{G}'$. (c) Case 3: $\mathcal{G}$. (d) Case 3: $\mathcal{G}'$.
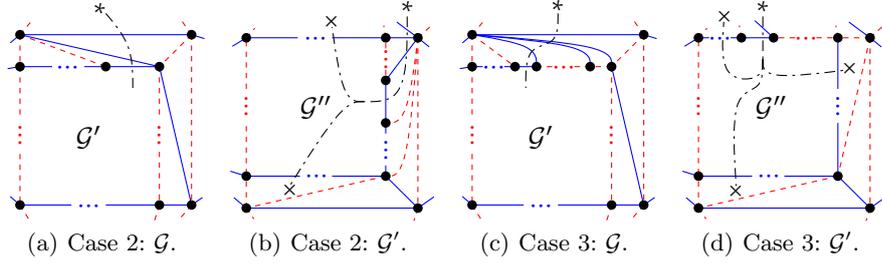
Fig. 11. (a-b) Graphs $\mathcal{G}$ and $\mathcal{G}'$ in Case 2. (c-d) Graphs $\mathcal{G}$ and $\mathcal{G}'$ in Case 3.

the inner graph of $\mathcal{G}'$. All edges in $\mathcal{G}'$ incident to the topright vertex in $\mathcal{G}'$ must be red. This forces the coloring of all remaining edges. So the first slice cannot continue to $b(\mathcal{G})$ after leaving $\mathcal{G}''$: hence it cannot be the first slice. Contradiction.

Case 4. The top side of $\mathcal{G}'$ has exactly one edge e and the first slice cuts through e, see Fig. 12(a). Since $\mathcal{G}'$ has only two vertices on its top side, the construction sequence of $E(\mathcal{G})$ must start with $\square\!\!\uparrow$ ($\mathcal{G}' = \mathcal{G}_1$) or $\square\!\!\nwarrow$ ($\mathcal{G}' = \mathcal{G}_2$). See Fig. 12(b) for $\mathcal{G}' = \mathcal{G}_1$ and Fig. 12(c) for $\mathcal{G}' = \mathcal{G}_2$. The only difference between $\mathcal{G}_1$ and $\mathcal{G}$ (Fig. 9) 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 3. But since $E(\mathcal{G}_1)$ is a smaller rotating windmill than $E(\mathcal{G})$, it is not sliceable by the induction hypothesis. Contradiction. □
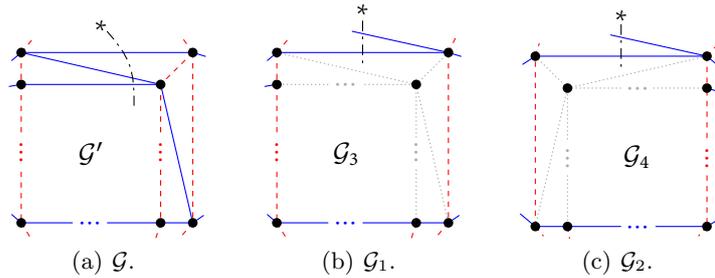

(a) $\mathcal{G}$. (b) $\mathcal{G}_1$. (c) $\mathcal{G}_2$.

Fig. 12. Case 4: graph $\mathcal{G}$ and two cases for $\mathcal{G}'$: graphs $\mathcal{G}_1$ and $\mathcal{G}_2$.

5 Sliceability of 1-pyramid extended graphs

In this section we prove that all reduced 1-pyramid extended graphs other than rotating windmills are sliceable. Given a 1-pyramid extended graph $E(\mathcal{G})$, let C_p be the cycle defined in Theorem 1 for each pole $p \in \{l(\mathcal{G}), b(\mathcal{G}), r(\mathcal{G}), t(\mathcal{G})\}$.

Lemma 4. *Let $E(\mathcal{G})$ be a reduced 1-pyramid extended graph. Suppose that there exists a slice S that splits $E(\mathcal{G})$ into $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$, such that $E(\mathcal{G}_\ell)$ (or $E(\mathcal{G}_r)$) can be reduced to a rotating windmill. Then we can construct a reduced 1-pyramid extended graph $E(\mathcal{G}')$ such that $E(\mathcal{G}')$ is not a rotating windmill, $\mathcal{G}'$ is a strict subgraph of $\mathcal{G}$ and $E(\mathcal{G})$ is sliceable if $E(\mathcal{G}')$ is sliceable.*

Proof (sketch). One can argue that that $E(\mathcal{G}_\ell)$ (or $E(\mathcal{G}_r)$) is already be a rotating windmill and then locally change S to a slice that does not induce a rotating windmill in the left or right graph. □

Lemma 5. *Let $E(\mathcal{G})$ be a reduced 1-pyramid extended graph. If C_p encloses the pyramid of $\mathcal{G}$ for all poles p, then $E(\mathcal{G})$ is the windmill.*

Proof (sketch). First, the proof argues that since C_ℓ and C_r both enclose the pyramid, there is a cycle C formed by vertices L from P_ℓ and R from P_r that encloses the pyramid. Since $l(\mathcal{G})$ ($r(\mathcal{G})$) has a path of length two to every vertex on P_ℓ (P_r), one can show that every vertex in $L \setminus R$ must have an edge to a vertex in R. It follows that C is a 4-cycle and since it encloses the pyramid in the 1-pyramid extended graph $E(\mathcal{G})$, the pyramid must be equal to C. Hence, P_ℓ and P_r contain an edge of the outer cycle of the pyramid. By a symmetric argument, P_t and P_b contain an edge of the outer cycle of the pyramid. Next, one can show that every edge of the outer cycle of the pyramid is on a different boundary path. Finally, we can use this property to show that every vertex on the outer cycle of the pyramid is connected to two adjacent poles. It follows that $E(\mathcal{G})$ contains the edges of the windmill. Since $E(\mathcal{G})$ is an irreducible triangulation, no other vertices can be present, which concludes the proof. □

The following algorithm computes a sliceable labeling of a reduced 1-pyramid extended graph that is not a rotating windmill.

1. If $\mathcal{G}$ is a single vertex, we are done.
2. Since $E(\mathcal{G})$ is not a rotating windmill, by Lemma 5, there is a pole p for which C_p does not enclose the pyramid. Use Theorem 1 to compute a slice from p. This slice splits $E(\mathcal{G})$ into $E(\mathcal{G}_\ell)$ and $E(\mathcal{G}_r)$. One of these, say $\mathcal{G}_\ell$, contains the pyramid of $\mathcal{G}$. By Theorem 1, $E(\mathcal{G}_r)$ is sliceable. If $E(\mathcal{G}_\ell)$ can be reduced to a rotating windmill, then proceed to Step 1 with the reduced extended graph $E(\mathcal{G}')$ guaranteed by Lemma 4. Otherwise, reduce $E(\mathcal{G}_\ell)$ using Lemma 2 and go to Step 1 with $E(\mathcal{G}_\ell)$.

The algorithm maintains the invariant that $E(\mathcal{G})$ is a reduced 1-pyramid extended graph that is not a rotating windmill at line 1. Combined with Theorem 2, this concludes the proof of our main result:

Theorem 3. *A 1-pyramid extended graph is sliceable if and only if it cannot be reduced to a rotating windmill.*

References

1. Ackerman, E., Barequet, G., Pinter, R.Y., Romik, D.: The number of guillotine partitions in d dimensions. Information processing letters 98(4), 162–167 (2006)
2. Alam, M.J., Biedl, T., Felsner, S., Kaufmann, M., Kobourov, S.G., Ueckerdt, T.: Computing cartograms with optimal complexity. In: SOCG'12. pp. 21–30 (2012)
3. de Berg, M., Mumford, E., Speckmann, B.: On rectilinear duals for vertex-weighted plane graphs. Disc. Math. 309(7), 1794–1812 (2009)
4. Bhasker, J., Sahni, S.: A linear time algorithm to check for the existence of a rectangular dual of a planar triangulated graph. Networks 17(3), 307–317 (1987)
5. Bhattacharya, B., Sur-Kolay, S.: On the family of inherently nonslicible floorplans in VLSI layout design. In: ISCAS'91. pp. 2850–2853. IEEE (1991)
6. Buchin, K., Speckmann, B., Verdonschot, S.: Optimizing regular edge labelings. In: Brandes, U., Cornelsen, S. (eds.) GD'11, LNCS, vol. 6502, pp. 117–128. Springer Berlin Heidelberg (2011)
7. Dasgupta, P., Sur-Kolay, S.: Slicible rectangular graphs and their optimal floorplans. ACM Trans. Design Automation of Electronic Systems 6(4), 447–470 (2001)
8. Eppstein, D., Mumford, E., Speckmann, B., Verbeek, K.: Area-universal rectangular layouts. In: SOCG'09. pp. 267–276 (2009)
9. Fusy, É.: Transversal structures on triangulations: A combinatorial study and straight-line drawings. Disc. Math. 309(7), 1870–1894 (2009)
10. He, X.: On floor-plan of plane graphs. SIAM J. on Comp. 28(6), 2150–2167 (1999)
11. Kant, G., He, X.: Two algorithms for finding rectangular duals of planar graphs. In: van Leeuwen, J. (ed.) WG'93, LNCS, vol. 790, pp. 396–410. Springer Berlin Heidelberg (1994)
12. Koźmiński, K., Kinnen, E.: Rectangular duals of planar graphs. Networks 15(2), 145–157 (1985)
13. van Kreveld, M., Speckmann, B.: On rectangular cartograms. Comp. Geom. 37(3), 175–187 (2007)
14. Liao, C.C., Lu, H.I., Yen, H.C.: Compact floor-planning via orderly spanning trees. J. Algorithms 48(2), 441–451 (2003)
15. Mumford, E.: Drawing Graphs for Cartographic Applications. Ph.D. thesis, TU Eindhoven (2008), `http://repository.tue.nl/636963`
16. Otten, R.: Efficient floorplan optimization. In: ICCAD'83. vol. 83, pp. 499–502 (1983)
17. Stockmeyer, L.: Optimal orientations of cells in slicing floorplan designs. Information and Control 57(2), 91–101 (1983)
18. Sur-Kolay, S., Bhattacharya, B.: Inherent nonslicibility of rectangular duals in VLSI floorplanning. In: FSTTCS'88. pp. 88–107. Springer (1988)
19. Szepieniec, A.A., Otten, R.H.: The genealogical approach to the layout problem. In: Proc. 17th Conf. on Design Automation. pp. 535–542. IEEE (1980)
20. Ungar, P.: On diagrams representing maps. J. L. Math. Soc. 1(3), 336–342 (1953)
21. Yao, B., Chen, H., Cheng, C.K., Graham, R.: Floorplan representations: Complexity and connections. ACM Trans. Design Auto. of Elec. Sys. 8(1), 55–80 (2003)
22. Yeap, G., Sarrafzadeh, M.: Sliceable floorplanning by graph dualization. SIAM J. Disc. Math. 8(2), 258–280 (1995)
23. Yeap, K.H., Sarrafzadeh, M.: Floor-planning by graph dualization: 2-concave rectilinear modules. SIAM J. on Comp. 22(3), 500–526 (1993)