Area-universal and constrained rectangular layouts

Citation for published version (APA):
Eppstein, D., Mumford, E., Speckmann, B., & Verbeek, K. A. B. (2012). Area-universal and constrained rectangular layouts. SIAM Journal on Computing, 41(3), 537-564. DOI: 10.1137/110834032

DOI:
10.1137/110834032

Document status and date:
Published: 01/01/2012

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

SIAM J. COMPUT.
Vol. 41, No. 3, pp. 537–564

AREA-UNIVERSAL AND CONSTRAINED RECTANGULAR LAYOUTS*

DAVID EPPSTEIN†, ELENA MUMFORD, BETTINA SPECKMANN‡, AND KEVIN VERBEEK‡

Abstract. A rectangular layout is a partition of a rectangle into a finite set of interior-disjoint rectangles. These layouts are used as rectangular cartograms in cartography, as floorplans in building architecture and VLSI design, and as graph drawings. Often areas are associated with the rectangles of a rectangular layout and it is desirable for one rectangular layout to represent several area assignments. A layout is *area-universal* if any assignment of areas to rectangles can be realized by a combinatorially equivalent rectangular layout. We identify a simple necessary and sufficient condition for a rectangular layout to be area-universal: a rectangular layout is area-universal if and only if it is *one-sided*. We also investigate similar questions for perimeter assignments. The adjacency requirements for the rectangles of a rectangular layout can be specified in various ways, most commonly via the dual graph of the layout. We show how to find an area-universal layout for a given set of adjacency requirements whenever such a layout exists. Furthermore we show how to impose restrictions on the orientations of edges and junctions of the rectangular layout. Such an orientation-constrained layout, if it exists, may be constructed in polynomial time, and all orientation-constrained layouts may be listed in polynomial time per layout.

Key words. computational geometry, rectangular layout, rectangular cartogram, area-universal, one-sided

AMS subject classification. 68U05

DOI. 10.1137/110834032

1. Introduction.

Raisz [21] introduced *rectangular cartograms* in 1934 as a way of visualizing spatial information, such as population or economic strength, of geographic regions. Rectangular cartograms represent regions by rectangles; the positioning and adjacencies of these rectangles are chosen to suggest their geographic locations, while their areas are chosen to represent the numeric values being communicated by the cartogram. The stylization inherent in replacing the complicated shapes of geographic regions by rectangles is a feature of such diagrams: as Raisz writes, "simple distortion of the map would be misleading," because it is important to emphasize that a cartogram is not a map.

Often more than one quantity should be displayed as a cartogram for the same set of geographic regions. The first three figures Raisz shows, for instance, are cartograms of land area, population, and wealth within the United States. To make the visual comparison of multiple related cartograms easier, it is desirable that the arrangement of rectangles be combinatorially equivalent in each cartogram, although the relative

*Received by the editors May 13, 2011; accepted for publication (in revised form) February 6, 2012; published electronically May 29, 2012. Preliminary versions of parts this paper appeared in Procceedings of the 25th ACM Symposium on Computational Geometry, 2009, pp. 267–276, and in Proceedings of the 11th Algorithms and Data Structures Symposium (WADS), Lecture Notes in Comput. Sci. 5664, Springer, 2009, pp. 266–277.

http://www.siam.org/journals/sicomp/41-3/83403.html

†Department of Computer Science, University of California, Irvine, CA 92697 (eppstein@ics.uci.edu). This author's work was supported in part by NSF grant 0830403 and by the Office of Naval Research under grant N00014-08-1-1015.

‡Department of Mathematics and Computer Science, TU Eindhoven, 5612 AZ Eindhoven, The Netherlands (b.speckmann@tue.nl, k.a.b.verbeek@tue.nl). The work of these authors was supported by the Netherlands' Organisation for Scientific Research (NWO) under project 639.022.707.

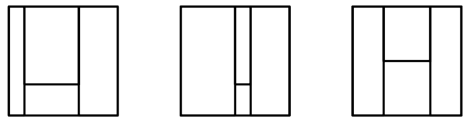

FIG. 1. *An area-universal layout.*

sizes of the rectangles will differ. This naturally raises the question: when is this possible?

Mathematically, a rectangular cartogram is a *rectangular layout*: a partition of a rectangle into finitely many interior-disjoint rectangles. We call a layout $\mathcal{L}$ *area-universal* if, no matter what areas we require each of its regions to have, some combinatorially equivalent layout $\mathcal{L}'$ has regions with the specified areas. For instance, the four-region rectangular layout shown in Figure 1 with three different area assignments is area-universal: any four numbers can be used as the areas of the rectangles in a combinatorially equivalent layout.

Area-universal rectangular layouts are useful not only for side-by-side display of cartograms with different data on the same regions, but also for dynamically morphing from one cartogram into another. Additionally, in other applications of rectangular layouts it may be advantageous to choose a layout first and then later assign varying areas while keeping the combinatorial type of the layout fixed. For instance, in circuit layout applications of rectangular layouts [27], each component of a circuit may have differing implementations with differing tradeoffs between area, energy use, and speed. In building design it is desirable to be able to determine the areas of different rooms according to their function [7]. And, in treemap visualizations, alternative area-universal layouts may be of use in controlling rectangle aspect ratios [4]. Thus, it is of interest to identify the properties that make a rectangular layout area-universal, and to find area-universal layouts when they exist.

For applications such as in cartography, where the spatial position of the rectangles is meaningful, we also want to impose certain restrictions on the orientations of the adjacencies of regions of a rectangular layout. For example, in a cartogram of the U.S., we might require that a rectangle representing Nevada be right of or above a rectangle representing California, as geographically Nevada is east and north of California.

1.1. Results. We identify a simple necessary and sufficient condition for a rectangular layout to be area-universal: it is area-universal if and only if it is *one-sided*. One-sided layouts are characterized via their *maximal line segments*. A *line segment* of a layout $\mathcal{L}$ is formed by a sequence of consecutive inner edges of $\mathcal{L}$. A segment of $\mathcal{L}$ that is not contained in any other segment is *maximal*. In a one-sided layout every maximal line segment s must be the side of at least one rectangle R; any vertices interior to s are T-junctions that all have the same orientation away from R (Figure 2).

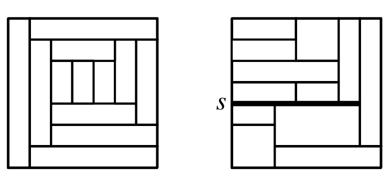

FIG. 2. *The left layout is one-sided, but the one on the right is not: the maximal segment s is not the side of any rectangle.*

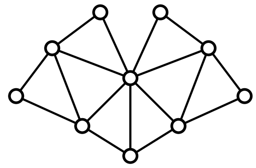

FIG. 3. *A graph that is not the dual of an area-universal layout: the rectangle dual to the bottom center vertex may not be arbitrarily large* [22].

Given an area-universal layout $\mathcal{L}$ and an assignment of areas for its regions, we describe a numerical algorithm that finds a combinatorially equivalent layout $\mathcal{L}'$ whose regions have a close approximation to the specified areas. These results can be found in section 4. In section 5 we investigate *perimeter cartograms* in which the perimeter of each rectangle is specified rather than its area. Any rectangular layout can have at most one combinatorially equivalent layout for a given perimeter assignment; it is possible in polynomial time to find this equivalent layout, if it exists.

The rectangles of a rectangular cartogram should have the same adjacencies as the regions of the underlying map. Hence, the dual graph of the cartogram (a graph with one node per region, with two nodes adjacent if their regions share a boundary segment) should be the same as the dual graph of the map. The dual of a rectangular layout is called a *proper graph*. Not every proper graph has an area-universal rectangular dual; Rinsma [22] described an outerplanar proper graph $\mathcal{G}$ and an assignment of weights to its vertices such that no rectangular dual of $\mathcal{G}$ has these weights as its regions' areas (Figure 3). Thus, it is of interest to determine which proper graphs have an area-universal rectangular dual. In section 6 we describe algorithms that, given a proper graph $\mathcal{G}$, find an area-universal rectangular dual of $\mathcal{G}$ if it exists. These algorithms are not fully polynomial, but are fixed-parameter tractable for a parameter related to the number of separating four-cycles in $\mathcal{G}$.

In section 7 we extend the approach of section 6 to construct a rectangular dual with certain orientation constraints, if it exists, in polynomial time. Further, we can list all layouts obeying the constraints in polynomial time per layout. Our algorithms can handle constraints limiting the allowed orientations of a shared edge between a pair of adjacent regions, as well as more general kinds of constraints restricting the possible orientations of the three rectangles meeting at any junction of the layout. We also discuss the problem of finding area-universal layouts in the presence of constraints of these types.

Motivated by architectural plans, where only a subset of the room adjacencies might be specified, Rinsma [23] considered the following related problem: given a tree $\mathcal{T}$, does there exist a rectangular layout $\mathcal{L}$ such that $\mathcal{T}$ is a spanning tree of the dual graph of $\mathcal{L}$? She showed that such a layout always exists, but the layouts constructed by her algorithm are not necessarily area-universal. In section 8 we modify her construction to yield area-universal layouts, proving that for every tree $\mathcal{T}$ there is an area-universal layout $\mathcal{L}$ such that $\mathcal{T}$ is a spanning tree of the dual graph of $\mathcal{L}$.

2. Preliminaries. As stated above, a rectangular layout (or sometimes simply *layout*) is a partition of a rectangle into a finite set of interior-disjoint rectangles. We assume that no four regions meet in a single point, as this is true (with a notable exception in the American Southwest) for most geographic partitions of interest. We denote the dual graph of a layout $\mathcal{L}$ by $\mathcal{G}(\mathcal{L})$; $\mathcal{G}(\mathcal{L})$ has a vertex for every rectangle in the layout, and an edge for every pair of rectangles that abut each other along a shared line segment. A layout $\mathcal{L}$ such that $\mathcal{G} = \mathcal{G}(\mathcal{L})$ is called a *rectangular dual* of graph $\mathcal{G}$. $\mathcal{G}(\mathcal{L})$ is a plane triangulated graph and is unique for any layout $\mathcal{L}$. Not every plane triangulated graph has a rectangular dual, and if it does, then the rectangular dual is not necessarily unique.

Independently, Koźmiński and Kinnen [15] and Ungar [25] proved that a plane triangulated graph $\mathcal{G}$ has a rectangular dual if and only if we can augment $\mathcal{G}$ with four external vertices in such a way that the *extended graph* $E(\mathcal{G})$ has the following two properties: (i) every interior face is a triangle and the exterior face is a quadrangle; (ii) $E(\mathcal{G})$ has no separating triangles—a *separating triangle* is a separating cycle (a simple cycle that has vertices both inside and outside) of length three. If a plane triangulated graph $\mathcal{G}$ allows such an augmentation, then we say that $\mathcal{G}$ is a *proper graph*. From an extended graph $E(\mathcal{G})$ of a proper graph $\mathcal{G}$ we can construct a rectangular dual for $\mathcal{G}$ in linear time [14] (Figure 4).

An extended graph $E(\mathcal{G})$ determines uniquely the vertices of its proper subgraph $\mathcal{G}$: they are the vertices that do not belong to the unique quadrilateral face of $E(\mathcal{G})$. However, for a given proper graph there might be several possible extended graphs and hence several possible *corner assignments*. In many cases we assume that a corner assignment, and hence an extended graph, has already been fixed, but if this is not the case then it is possible to test all corner assignments in polynomial time, as there can be only polynomially many of them.

A rectangular layout $\mathcal{L}$ naturally induces a labeling of its extended dual graph $E(\mathcal{G})$. If two rectangles of $\mathcal{L}$ share a vertical segment, then we color the corresponding edge in $E(\mathcal{G})$ blue (solid) and direct it from left to right. Correspondingly, if two rectangles of $\mathcal{L}$ share a horizontal segment, then we color the corresponding edge in $E(\mathcal{G})$ red (dashed) and direct it from bottom to top (Figure 5).

This labeling has the following properties: (i) around each inner vertex in clockwise order we have four nonempty contiguous sets of incoming blue edges, outgoing red edges, outgoing blue edges, and incoming red edges; (ii) the left exterior vertex has only blue outgoing edges, the top exterior vertex has only red incoming edges, the right exterior vertex has only blue incoming edges, and the bottom exterior vertex has only red outgoing edges.

Such a labeling is called a *regular edge labeling*. It was introduced by Kant and He [14] who showed that every regular edge labeling of an extended graph $E(\mathcal{G})$ uniquely defines an equivalence class of rectangular duals of a proper graph $\mathcal{G}$. Given any extended graph $E(\mathcal{G})$, a regular edge labeling for $E(\mathcal{G})$ can be found in linear time and the rectangular dual defined by it can also be constructed in linear time [14]. Regular edge labelings have also been studied by Fusy [12, 13], who refers to them

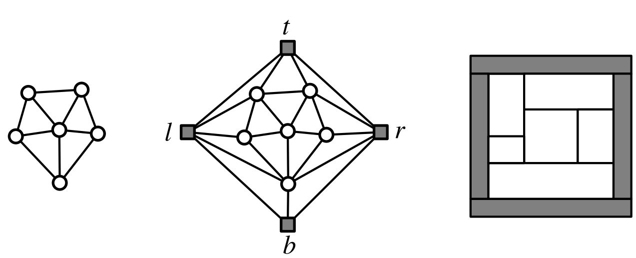

FIG. 4. *A proper graph $\mathcal{G}$, an extended graph $E(\mathcal{G})$, and a rectangular dual $\mathcal{L}$ of $E(\mathcal{G})$.*

as *transversal structures*. Regular edge labelings are closely related to several other edge coloring structures on planar graphs that can be used to describe straight line embeddings and orthogonal polyhedra [9, 10].

Two layouts $\mathcal{L}$ and $\mathcal{L}'$ are *equivalent*, denoted by $\mathcal{L} \sim \mathcal{L}'$, if they induce the same regular edge labeling of the same dual graph. We say that a rectangular layout $\mathcal{L}$ with n rectangles $R_1, \ldots, R_n$ realizes a weight function $w : R_1, \ldots, R_n \to \mathbb{R}, w(i) > 0$ as a *rectangular cartogram* if there exists a layout $\mathcal{L}' \sim \mathcal{L}$ such that for any $1 \leq i \leq n$ the area of rectangle R_i equals $w(R_i)$. Correspondingly, we say that a layout $\mathcal{L}$ realizes w as a *perimeter cartogram* if there exists a layout $\mathcal{L}' \sim \mathcal{L}$ such that the perimeter of each rectangle of $\mathcal{L}'$ equals the prescribed weight. A layout $\mathcal{L}$ is *area-universal* (*perimeter-universal*) if it realizes every possible weight function as a rectangular cartogram (perimeter cartogram).

It will be convenient to define a weaker equivalence relation on layouts than equivalence, which we call *order-equivalence*. For a layout $\mathcal{L}$, define a partial order on the vertical maximal segments, in which $s_1 \leq s_2$ if there exists an x-monotone curve that has its left endpoint on s_1, its right endpoint on s_2, and that does not cross any horizontal maximal segments. This partial order can be defined by a directed acyclic multigraph that has a vertex per maximal segment and an edge from the segment on the left boundary of each rectangle to the segment on the right boundary of the same rectangle; this graph is an st-planar graph, a planar directed acyclic graph in which the unique source and the unique sink are both on the outer face. The dual of this st-planar graph defines in a symmetric way a partial order on the horizontal maximal segments. We say that $\mathcal{L}$ and $\mathcal{L}'$ are order-equivalent if their rectangles and maximal segments correspond one-for-one in a way that preserves these partial orders (Figure 6).

Lemma 1. *A rectangular layout with n rectangular regions has $n - 1$ maximal (inner) segments.*

Proof. A rectangular layout can be seen as a plane graph $\mathcal{G}_{\mathcal{L}}$ with vertices located at the corners of the rectangles and edges formed by parts of the sides of the rectangles. Each maximal segment starts and ends with a vertex of such a graph. Each degree-three vertex is an endpoint of exactly one maximal segment. Thus the number of maximal segments is half of the number of the vertices of degree three in $\mathcal{G}_{\mathcal{L}}$. The number of degree-three vertices is $(4n-4)/2$ (four corners per rectangle; each degree-three vertex corresponds to a pair of corners, and the four outer corners of the layout are the only ones that do not contribute to degree-three vertices); hence the number of maximal segments in $\mathcal{L}$ is $n - 1$. ∎

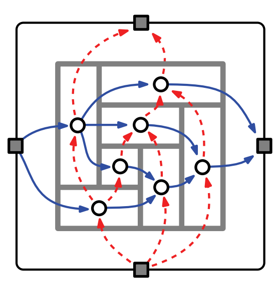

Fig. 5. *A rectangular layout and the regular edge labeling of its extended dual.*

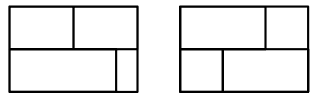

FIG. 6. *Two inequivalent but order-equivalent rectangular layouts.*

3. There can be only one. We first show that for any combination of layout and weight function there can be at most one rectangular cartogram, up to affine transformations. This result is also contained in [26]. We also show that there is at most one perimeter cartogram with a fixed bounding box. More generally, if two geometrically different but order-equivalent layouts share the same bounding box, there is a rectangle in one of the layouts that is larger in both of its dimensions than the corresponding rectangle in the other layout. The proof involves a graph-theoretic argument in an auxiliary graph constructed from the two layouts.

Thus, let $\mathcal{L}$ and $\mathcal{L}'$ be two geometrically different order-equivalent layouts with the same bounding box. The *push graph* $\mathcal{H}$ of $\mathcal{L}$ and $\mathcal{L}'$ is a directed graph that has a vertex for each rectangle in $\mathcal{L}$ and an edge from vertex R_i to vertex R_j if the rectangles R_i and R_j are adjacent and the maximal segment in $\mathcal{L}$ that separates R_i from R_j is shifted in $\mathcal{L}'$ towards R_j and away from R_i (Figure 7).

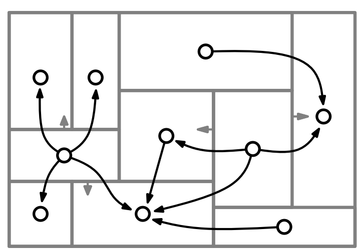

FIG. 7. *A push graph. The layout $\mathcal{L}$ is shown; the relative position of the maximal segments in the equivalent layout $\mathcal{L}'$ is indicated by the arrows attached to the maximal segments.*

LEMMA 2. *The push graph for $\mathcal{L}$ and $\mathcal{L}'$ contains a node with no incoming or no outgoing edges.*

Proof. Assume for contradiction that the push graph $\mathcal{H}$ has no source or sink. Then $\mathcal{H}$ must contain a cycle. Let C be a simple cycle in $\mathcal{H}$ that encloses as few vertices as possible, and assume without loss of generality that C is oriented clockwise. By construction, C cannot contain a rightward edge immediately followed by a leftward edge or an upward edge immediately followed by a downward edge. Hence it must contain a rightward edge e that is followed by a downward edge. We distinguish three cases depending on the relative positions of the bottom sides of the two rectangles L

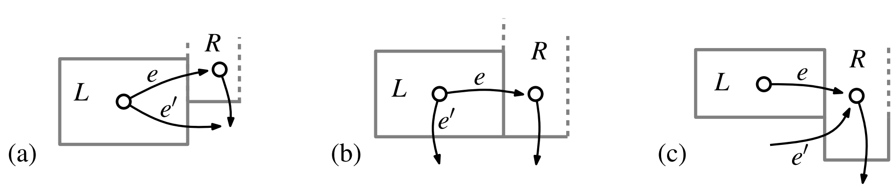

FIG. 8. *Cases for Lemma* 2.

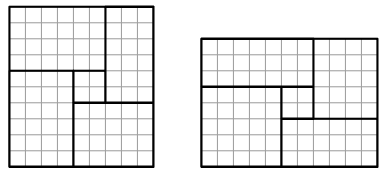

FIG. 9. *Two equivalent layouts in which corresponding rectangles have the same perimeter.*

and R that are connected by e (Figure 8):

(a) If the bottom edge of L lies below the bottom edge of R, then $\mathcal{H}$ must contain an edge e' that connects L to the rectangle below R. This edge e' shortcuts C, contradicting the minimality of C.

(b) If the bottom edges of L and R are aligned along a maximal segment, then $\mathcal{H}$ must contain an edge e' that points downward from L. By following a directed chain of edges starting with e' we either reach a repeated vertex within this chain of edges or a vertex that belongs to C. In either case we have found a cycle that encloses fewer vertices than C, contradicting the minimality of C.

(c) If the bottom edge of L lies above the bottom edge of R, then $\mathcal{H}$ must contain an edge e' that connects the rectangle below L to R. As in case (b), by following a chain of edges backward starting from e' we can find a cycle that encloses fewer vertices than C, contradicting the minimality of C. ◻

The following result was known already [26], and also follows immediately from Lemma 2.

LEMMA 3 (see [26, Theorem 3]). *For any layout $\mathcal{L}$ and weight function w, at most one order-equivalent layout $\mathcal{L}'$ (up to affine transformations) realizes w as a rectangular cartogram.*

Proof. Let $\mathcal{L}$ and $\mathcal{L}'$ be order-equivalent with the same area, but geometrically different; scale $\mathcal{L}'$ horizontally and vertically so that they have the same bounding box. By Lemma 2, one of the layouts contains a rectangle R that is strictly larger than the corresponding rectangle of the other. Thus, R cannot have the same area in both layouts and only one of the layouts can realize w. ◻

For perimeter, such strong uniqueness does not hold: there are equivalent layouts that are not affine transformations of each other in which the perimeters of corresponding rectangles are equal (Figure 9). However, if we fix the outer bounding box of the layout, the same proof method works.

THEOREM 1. *For any layout $\mathcal{L}$ and any weight function w there is at most one layout $\mathcal{L}'$ that is order-equivalent to $\mathcal{L}$ with the same bounding box and that realizes w as a perimeter cartogram.*

More generally the same result holds for any type of cartogram in which rectangle sizes are measured by any strictly monotonic function of the height and width of the rectangles.

4. Area-universality and one-sidedness.

As the next lemma states, all layouts are area-universal in a weak sense involving order-equivalence in place of equivalence. One possible proof uses Lemma 3 to invert the map from vectors of positions of segments in a layout to vectors of rectangle areas, along a line segment from the area vector of $\mathcal{L}$ to the desired area vector. However, we omit the proof, as the result was already known [26].

LEMMA 4 (see [26, Theorem 3]). *For any layout $\mathcal{L}$ and weight function w, there exists a layout $\mathcal{L}'$ that has a square outer rectangle, is order-equivalent to $\mathcal{L}$, and realizes w as a rectangular cartogram.*

One may find $\mathcal{L}'$ by hill-climbing to reduce the Euclidean distance between the current weight function and the desired weight function. No layout $\mathcal{L}$ can be locally but not globally optimal, because within any neighborhood of $\mathcal{L}$ the inverse image of the line segment connecting its weight vector to the desired weight vector contains layouts that are closer to w. Alternatively, one can find $\mathcal{L}'$ by a numerical procedure that follows this inverse image by inverting the Jacobean matrix of w at each step. We do not know whether it is always possible to find $\mathcal{L}'$ exactly by an efficient combinatorial algorithm (as may easily be done for the subclass of sliceable layouts), or whether the general solution involves roots of high-degree polynomials that can be found only numerically.

THEOREM 2. *The following three properties of a layout $\mathcal{L}$ are equivalent:*

1. *$\mathcal{L}$ is area-universal.*
2. *Every layout that is order-equivalent to $\mathcal{L}$ is equivalent to $\mathcal{L}$.*
3. *$\mathcal{L}$ is one-sided.*

Proof. We show that property 2 implies property 1, that the negation of property 2 implies the negation of property 1, that property 3 implies property 2, and that the negation of property 3 implies the negation of property 2.

$2 \Rightarrow 1$: Let $\mathcal{L}$ be a layout satisfying the property that every layout that is order-equivalent to $\mathcal{L}$ is equivalent to $\mathcal{L}$, and let w be an arbitrary weight function; we must show that $\mathcal{L}$ realizes w as a rectangular cartogram. By Lemma 4, there exists a layout $\mathcal{L}'$ that is order-equivalent to $\mathcal{L}$ and realizes w; by the assumption, $\mathcal{L}'$ is equivalent to $\mathcal{L}$, as desired.

$(\neg 2) \Rightarrow (\neg 1)$: Suppose that there exists a layout $\mathcal{L}'$ that is order-equivalent but inequivalent to $\mathcal{L}$. By scaling horizontally and vertically, we may assume that $\mathcal{L}$ and $\mathcal{L}'$ have the same bounding box. Let w be the weight function given by the areas of the rectangles in $\mathcal{L}'$. By Lemma 3, $\mathcal{L}'$ is the only layout that is order-equivalent to $\mathcal{L}$ and realizes w as a rectangular cartogram; therefore, there can be no layout that is equivalent to $\mathcal{L}$ and realizes w as a rectangular cartogram, showing that $\mathcal{L}$ is not area-universal.

$3 \Rightarrow 2$: Let $\mathcal{L}$ be a one-sided layout, and let $\mathcal{L}'$ be order-equivalent to $\mathcal{L}$. Then $\mathcal{L}'$ must be one-sided, because the property that each maximal segment is a side of a rectangle is preserved under order-equivalence. For every pair of adjacent rectangles R_1 and R_2 in $\mathcal{L}$ or in $\mathcal{L}'$, R_1 and R_2 are adjacent with a given orientation if and only if they are on opposite sides of a common maximal segment with the given orientation, and this property of being on opposite sides of a common maximal segment is also preserved by order-equivalence, so order-equivalence preserves the adjacencies of rectangles in $\mathcal{L}$ and $\mathcal{L}'$.

$(\neg 3) \Rightarrow (\neg 2)$: If $\mathcal{L}$ is not one-sided, then let s be a maximal segment of $\mathcal{L}$ that has more than one rectangle on both sides of s; without loss of generality assume that s is horizontal. We may form an order-equivalent but inequivalent layout $\mathcal{L}'$ by moving the vertical maximal segments that abut the top side of s rightward and the vertical maximal segments that abut the bottom side of s leftward until the order of their endpoints changes, as in Figure 6. $\square$

5. Finding perimeter cartograms.

Although our proof of uniqueness for rectangular cartograms generalizes to perimeter, our proof that any layout and weight function has a realization as an order-equivalent cartogram does not generalize: there exist one-sided layouts and weight functions that cannot be realized as a perimeter cartogram (Figure 10). Nevertheless, one can test in polynomial time whether a so-

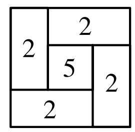

Fig. 10. *The outer rectangles each contribute at most one unit of shared boundary to the perimeter of the central rectangle, which is too large to be realized.*

lution exists for any layout and weight function. The technique involves describing the constraints on the perimeters of rectangles as linear equalities that reduce the dimension of the space of layouts to at most two, and forming a low-dimensional linear program from inequality constraints expressing the equivalence to $\mathcal{L}$ of the other layouts within this low-dimensional space.

Theorem 3. *For any layout $\mathcal{L}$ and any weight function w we can find a layout $\mathcal{L}'$ that is equivalent to $\mathcal{L}$ and that realizes w as a perimeter cartogram, if one exists.*

Proof. We can specify a layout by supplying one coordinate per maximal segment; together with the length and height of the bounding box this gives us a set of $n+1$ real values to be determined in a way consistent with the given weight function and layout. Each value of the weight function determines an equality constraint among these variables, stating that a certain linear combination of differences of segment positions equals the given perimeter. The constraints that the resulting layout be equivalent to $\mathcal{L}$ may be translated into linear inequality constraints, stating that the segment on the left side of each rectangle must have a smaller coordinate value than the segment on the right, the segment on the bottom side of each rectangle must have a smaller coordinate value than the segment on the top, and that the three-way junctions appearing along any maximal segment of the layout appear in the correct order.

The equality constraints determine a linear subspace S of $\mathbb{R}^{n+1}$ which we may find by Gaussian elimination. If there exists a layout $\mathcal{L}'$ realizing w, then, by Theorem 1, S contains only a single point with the same bounding box height and width as $\mathcal{L}'$, and hence has dimension at most two; conversely, if the dimension of this linear subspace is greater than two, then we may immediately infer from Theorem 1 that no solution exists.

If the dimension of the subspace is at most two, then, on the other hand, we may translate all the inequality constraints in $\mathbb{R}^{n+1}$ into linear inequality constraints in this two-dimensional subspace and solve the resulting two-dimensional linear program in linear time using standard algorithms (e.g., see [16]). ☐

The same algorithm can be used to find an order-equivalent layout rather than an equivalent layout by restricting the inequality constraints to the subset that determines order-equivalence.

6. Finding one-sided layouts. Recall that every proper triangulated plane graph has a rectangular dual, but not necessarily a one-sided rectangular dual. Since one-sided duals are area-universal, it is of interest to find a one-sided dual for a proper graph if one exists. Our overall approach is, first, to partition the graph on its separating four-cycles; second, to represent the family of all layouts for a proper graph as a distributive lattice, following Fusy [12, 13]; third, to represent elements of the distributive lattice as partitions of a partial order according to Birkhoff's theorem [2];

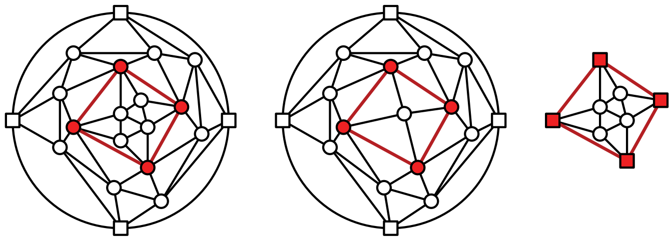

FIG. 11. *An extended graph with a nontrivial separating four-cycle (left), its outer separation component (center), and its inner separation component (right).*

fourth, to characterize the ordered partitions that correspond to one-sided layouts; and fifth, to search in the partial order for partitions of this type. Our algorithms are not fully polynomial, but they are polynomial whenever the number of separating four-cycles in the given proper graph is bounded by a fixed constant, or more generally when such a bound can be given separately within each of the pieces found in the partition we find in the first stage of our algorithms.

6.1. Eliminating separating four-cycles. Recall that a *separating four-cycle* in a plane graph $\mathcal{G}$ is a cycle of four vertices that has other vertices both inside and outside it. We say that a separating four-cycle is *nontrivial* if the number of vertices inside it is greater than one. Although a plane graph may have a quadratic number of separating four-cycles (for instance, this is true for the complete bipartite graph $K_{2,n-2}$) it is possible to represent all separating four-cycles in linear space by finding all maximal complete bipartite subgraphs $K_{2,i}$ of $\mathcal{G}$: a separating four-cycle is exactly a four-cycle in one of these graphs that is not a face of $\mathcal{G}$. Such a representation may be found in linear time [8]. In an extended graph $E(\mathcal{G})$, we allow the external vertices to be included as part of its separating four-cycles.

If $\mathcal{G}$ is a proper graph with a corner assignment $E(\mathcal{G})$, and C is a nontrivial separating four-cycle in $E(\mathcal{G})$, then we may form two minors of $\mathcal{G}$, the *separation components* of $\mathcal{G}$ with respect to C (see Figure 11). The *inner separation component* $\mathcal{G}_C$ is the subgraph induced by the vertices interior to the cycle, and its extended graph $E(\mathcal{G}_C)$ is the subgraph induced by the vertices on or interior to the cycle, interpreting the vertices of C as a corner assignment for its interior vertices. The *outer separation component* $E(\mathcal{G}) \setminus \mathcal{G}_C$ is formed by replacing the interior of C by a single vertex. We define a *minimal separation component* of $\mathcal{G}$ to be a minor of $\mathcal{G}$ formed by repeatedly splitting larger graphs into separation components until no nontrivial separating four-cycles remain. A partition of $E(\mathcal{G})$ into minimal separation components may be found in linear time by applying the algorithm for finding all maximal complete bipartite subgraphs $K_{2,i}$ as described above, and then for each such subgraph separating the exterior of the $K_{2,i}$ subgraph from each of the subgraphs within one of the inner faces of the $K_{2,i}$ subgraph.

LEMMA 5. *An extended graph $E(\mathcal{G})$ is dual to a one-sided layout if and only if both its inner and outer separation components are dual to one-sided layouts.*

Proof. In any layout dual to $E(\mathcal{G})$, the region enclosed by the four rectangles of the separating cycle C must be a four-sided polygon, that is, a rectangle. If we modify a one-sided layout of $E(\mathcal{G})$ by replacing the contents of this rectangle by a single rectangular area, or by removing the exterior of this rectangle, we obtain one-sided layouts of $E(\mathcal{G}) \setminus \mathcal{G}_C$ and $E(\mathcal{G}_C)$, respectively.

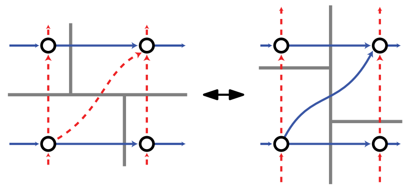

FIG. 12. *A move formed by recoloring the interior of an alternatingly colored four-cycle in a regular edge labeling, and its effect on the dual rectangular layout. In the case shown, the cycle is not separating: it contains a single edge of $\mathcal{G}$, but no vertices.*

Conversely, suppose that we have one-sided layouts of both $E(\mathcal{G}) \setminus \mathcal{G}_C$ and $E(\mathcal{G}_C)$. We may transform the layout of the inner separation component $E(\mathcal{G}_C)$ so that its bounding box matches the rectangle in the center of C in the layout for the outer separation component $E(\mathcal{G}) \setminus \mathcal{G}_C$, and combine these two layouts to obtain a layout of $E(\mathcal{G})$. The adjacencies between rectangles and maximal segments of the combined layout are unchanged except for the segments bounding the central rectangle. By the one-sidedness of the layout for the outer separation component, each such segment forms a side of one of the rectangles dual to the vertices of C (the inner rectangle on the other side of the segment has sides that are subsets of the sides of the rectangles dual to C), and this property remains true in the combined layout, which is therefore one-sided. ∎

COROLLARY 1. *An extended graph $E(\mathcal{G})$ is dual to a one-sided layout if and only if all of its minimal separation components are dual to one-sided layouts.*

Thus, if we seek to determine whether an extended graph $E(\mathcal{G})$ is dual to a one-sided layout, we may assume without loss of generality that $E(\mathcal{G})$ has no nontrivial separating four-cycles. The same idea of cutting the input on separating four-cycles has been previously applied to the problem of finding sliceable duals for a given proper graph [6, 17].

6.2. The lattice of regular edge labelings. Fusy [12, 13] (see also [24]) defines a family of moves by which one regular edge labeling can be changed to another. Let C be a four-cycle in $E(\mathcal{G})$ in which the colors alternate between red and blue around the cycle. Then a move consists of reversing the colors of the edges within C; when such a move is made, there can be only one way of setting the orientations of the recolored edges. In a graph with no nontrivial separating four-cycles, each move changes the edge labeling either of a single edge (as shown in Figure 12) or of all four edges surrounding a degree-four vertex. At each of the two or five vertices adjacent to the recolored edges, one of the boundaries between incoming red edges, incoming blue edges, outgoing red edges, and outgoing blue edges shifts by one position in the cyclic ordering of edges around the vertex. These shifts are the same direction for each affected vertex, and can also be interpreted as twisting the boundary between two rectangles in the dual layout by 90 degrees in the opposite direction. Consider the graph with one vertex per regular edge labeling of $E(\mathcal{G})$ and with an edge between every two labelings connected by one of these moves; direct each edge of this graph from the labeling in which the boundaries are more clockwise to the labeling in which the boundaries are more counterclockwise. Then this graph of labelings is acyclic and defines a partial ordering on the family of all regular edge labelings of $E(\mathcal{G})$. Figure 13 shows an example in which the edges in the graph of labelings are directed from the lower labelings to the higher ones.

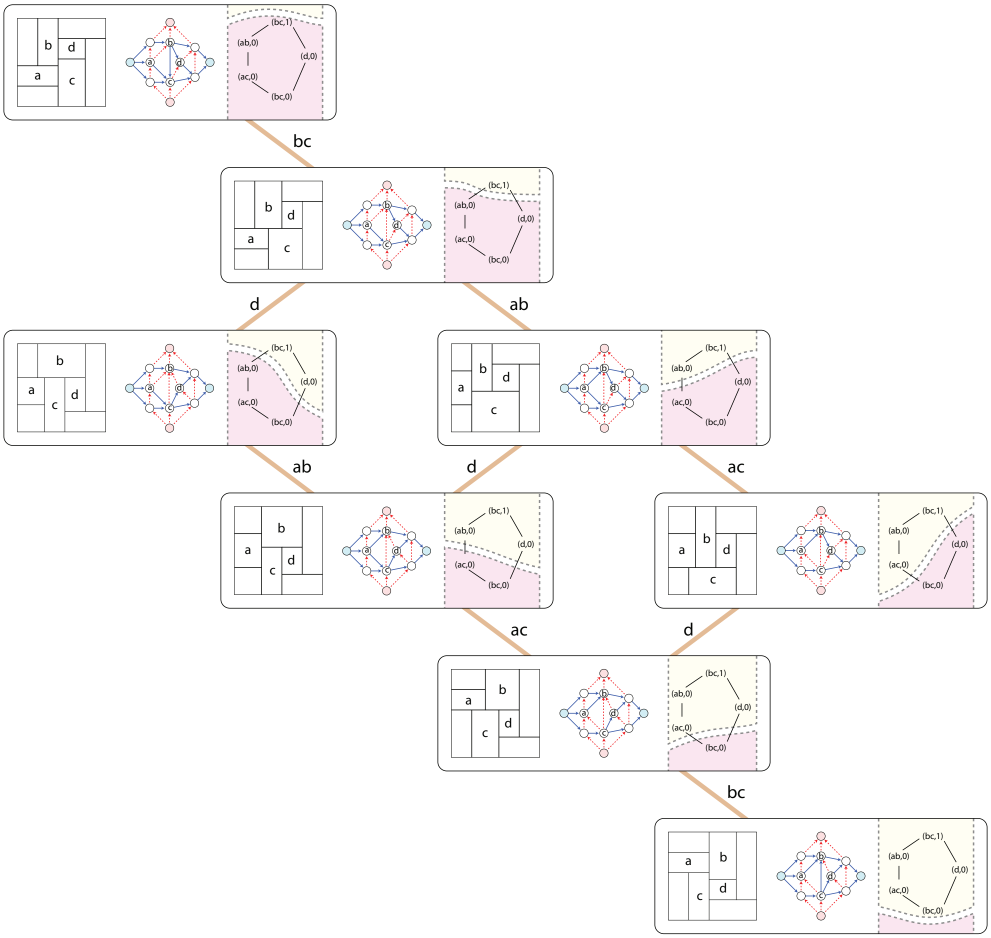

FIG. 13. *The family of rectangular layouts dual to a given extended graph $E(\mathcal{G})$, the corresponding regular edge labelings, and the corresponding partial order partitions. Two layouts are shown connected to each other by an edge if they differ by reversing the color within a single alternatingly colored four-cycle; these moves are labeled by the identity of the edge or vertex contained by the four-cycle.*

Moreover, as Fusy shows, the partial order defined in this way is a *distributive lattice*. A *lattice* is a partially ordered set in which each pair of elements (a, b) has a unique smallest upper bound (such an element is called the *join* of a and b and is denoted $a \vee b$) and a unique largest lower bound (such an element is called the *meet* of a and b and is denoted $a \wedge b$). A *distributive lattice* is a lattice in which the join and the meet operations are distributive over each other: $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$ and $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$. An element b of a lattice is said to *cover* an element a if $a < b$ and b are immediate neighbors in the lattice, that is, $a < b$ and there exists no element c such that $a < c < b$. In the distributive lattice defined in this way from the regular edge labelings of $E(\mathcal{G})$, the covering pairs are exactly the pairs of labelings connected by Fusy's moves. The minimal element of the lattice may be found from any lattice element by repeatedly performing clockwise moves until no more such moves are possible, and the maximal element may similarly be found by repeatedly performing counterclockwise moves. We say that a sequence of moves of the latter type, in which each move is counterclockwise, is *monotone*.

Birkhoff's representation theorem [2] states that the elements of any finite distributive lattice may be represented by sets in such a way that the join and meet operations may be represented by unions and intersections of sets. More precisely, let P be the partial order induced by the subset of the lattice consisting of elements that have exactly one predecessor in the covering relation. Then we may represent any lattice element x by a partition of the partial order into two sets $(L(x), U(x))$, where $L(x)$ consists of the members y of P with $y \leq x$ and $U(x)$ consists of the remaining members of P. Clearly, $L(x)$ is downward-closed (if $y \leq z$ in P and $z \in L(x)$, then $y \in L(x)$) and conversely $U(x)$ is upward-closed. If x and y are two members of the distributive lattice, then $x \leq y$ if and only if $L(x) \subset L(y)$ if and only if $U(x) \supset U(y)$, $x \wedge y$ is represented by the partition $(L(x) \cap L(y), U(x) \cup U(y))$, and $x \vee y$ is represented by the partition $(L(x) \cup L(y), U(x) \cap U(y))$. The lattice itself can be reconstructed as the set of all partitions of P into downward- and upward-closed subsets (L, U): each such partition corresponds in this way to a lattice element x.

6.3. The partial order of flippable items. We have seen that the layouts of an extended graph $E(\mathcal{G})$ may be described as partitions of a partial order P into downward-closed and upward-closed subsets; P is the order induced from the distributive lattice of layouts by the subset of layouts that have exactly one downward neighbor. Our goal in this section is to describe a partial order equivalent to P in a more concrete way, with elements that are not whole layouts themselves but rather correspond to individual features of rectangular layouts and their dual graphs in a way that helps us relate the distributive lattice operations more closely to their effect on a layout. Our more concrete partial order, and the partitions of it into subsets $L(\mathcal{L})$ and $U(\mathcal{L})$ that correspond to each rectangular layout $\mathcal{L}$, are depicted alongside the layouts in Figure 13.

Define a *flippable item* in the extended graph $E(\mathcal{G})$ to be either a degree-four vertex v or an edge e that is not adjacent to a degree-four vertex, with the additional property that there exists some regular edge labeling of $E(\mathcal{G})$ in which the four-cycle surrounding v or e is alternately colored and oriented. Thus, a flippable item is the edge that changes color, or the endpoint of a set of four edges that change color, in some move of $E(\mathcal{G})$. If x is a flippable item, and $\mathcal{L}$ is a rectangular layout represented by an element of the distributive lattice of labelings, define $f_x(\mathcal{L})$ as the number of moves involving x on any monotone sequence of moves from the minimal lattice element to $\mathcal{L}$.

The numbers $f_x(\mathcal{L})$ are in fact equivalent to the *potentials* defined by Felsner [11]. This equivalence can be shown using the theory developed by both Fusy [13] and Felsner [11]. The following lemma can also be found in [11] for potentials, but, for the sake of clarity, we include a proof using our notation.

LEMMA 6. *The number $f_x(\mathcal{L})$ is well defined and independent of the monotone path chosen to reach $\mathcal{L}$ from the minimal element.*

Proof. By Birkhoff's theorem, the length of any two upward paths between two elements of a distributive lattice is equal (it is equal to the size of the difference of the downward-closed subsets of the partial order representing those elements). By results of Birkhoff and Kiss [3], any three elements a, b, c of a distributive lattice have a unique median $m(a,b,c) = (a \vee b) \wedge (a \vee c) \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c) \vee (b \wedge c)$ belonging to shortest paths between any two of the three items (Figure 14). We prove by induction the following strengthening of the lemma: let $\mathcal{L} \leq \mathcal{L}'$ be two layouts. Then for any item x, and any two monotone paths from $\mathcal{L}$ to $\mathcal{L}'$, x is flipped the same

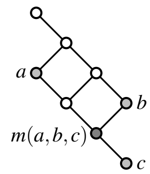

Fig. 14. *The median.*

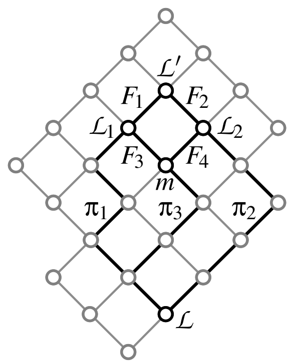

Fig. 15. *Notation for the proof of Lemma* 6.

number of times on both paths. Note that the number of times x is flipped, mod 4, must be the same on both paths, as the color and orientation of x may be determined from the number of flips mod 4.

As base cases for the strengthening, if the distance from $\mathcal{L}$ to $\mathcal{L}'$ is one, there can be only one monotone path, and if the distance is two, each path can flip x only once while the number of flips of x on both paths must be the same mod 4, so x must be flipped equally often. To finish the induction, suppose that we have two monotone paths π_1 from $\mathcal{L}$ to $\mathcal{L}_1$ and π_2 from $\mathcal{L}$ to $\mathcal{L}_2$ such that we can perform one more upward flip F_1 from $\mathcal{L}_1$ to $\mathcal{L}'$ and a flip F_2 from $\mathcal{L}_2$ to $\mathcal{L}'$. We must show that the number of flips of x on the two paths $\pi_1 F_1$ and $\pi_2 F_2$ are equal. Let $m = m(\mathcal{L}, \mathcal{L}_1, \mathcal{L}_2)$. Then there must exist a path π_3 from $\mathcal{L}$ to m, and flips F_3 and F_4 from m to $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively, such that $\pi_3 F_3$ and $\pi_3 F_4$ are monotone paths from $\mathcal{L}$ to $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively. By induction the number of flips of x on π_1 equals the number of flips of x on $\pi_3 F_3$, and the number of flips of x on π_2 equals the number of flips of x on $\pi_3 F_4$. Thus, the numbers of flips of x on $\mathcal{L}_1$ and $\mathcal{L}_2$ can differ only by one, and the numbers of flips of x on $\pi_1 F_1$ and $\pi_2 F_2$ can differ only by two. But again, these numbers of flips must be equal mod 4, so the result holds (see Figure 15). ∎

Lemma 7. *The number $f_x(\mathcal{L})$ is $O(n)$, where n is the number of rectangles in the layout.*

Proof. Define a flipping graph where the nodes are the degree-four vertices and nondegree-four edges of $\mathcal{G}$ and where two nodes are connected if they belong to the same triangle of $\mathcal{G}$. In any monotone sequence of moves, whenever a move on x increases $f_x(\mathcal{L})$, x cannot be flipped again until all its neighbors in the flipping graph have been flipped. Therefore, if x and y are adjacent in the flipping graph, then $f_x(\mathcal{L})$ and $f_y(\mathcal{L})$ are always within one of each other. But because the outer edges (or outer degree-four vertices) of the layout can never change color or orientation, the flippable items adjoining them can have f_x at most equal to one. Therefore, the maximum value of $f_x(\mathcal{L})$ for any x is at most the length of the shortest path in the flipping graph to one of the boundary nodes, and is $O(n)$. ☐

Let $\hat{\mathcal{L}}$ denote the maximal element in the distributive lattice of labelings. We define a partial order $P(\mathcal{G})$ that has as its elements the pairs (x, i), where x is a flippable element and i is an integer satisfying $0 \leq i < f_x(\hat{\mathcal{L}})$. Thus, if element x has k different states in different layouts, then it participates in $k-1$ pairs of P; the pairs correspond not to states but to transitions between states. In this partial order $P(\mathcal{G})$, we define $(x, i) \leq (y, j)$ when for all layouts $\mathcal{L}$ with $f_x(\mathcal{L}) \leq i$, it holds that $f_y(\mathcal{L}) \leq j$; that is, it is not possible to move f_y from j to $j+1$ prior to moving f_x from i to $i+1$. We may represent a layout $\mathcal{L}$ by the partition of $P(\mathcal{G})$ into two subsets $L(\mathcal{L})$ and $U(\mathcal{L})$, where $(x, i) \in L(\mathcal{L})$ when $i < f_x(\mathcal{L})$ and $(x, i) \in U(\mathcal{L})$ otherwise.

LEMMA 8. *We can construct $P(\mathcal{G})$ in polynomial time.*

Proof. We may compute $f_x(\hat{\mathcal{L}})$ for each x, determining the set of elements in $P(\mathcal{G})$, by repeatedly performing downward moves in the lattice of layouts until we reach the minimal layout, repeatedly performing upward moves from there until we reach the maximal layout, and counting the number of times a move involves each element x. The partial order of the pairs (x, i) may be determined from the neighboring objects of x in $E(\mathcal{G})$: we may make an upward move involving pair (x, i) in layout $\mathcal{L}$ if there is no pair (x, i') in $U(\mathcal{L})$ with $i' < i$ and when the regular edge labeling corresponding to $\mathcal{L}$ has the boundaries between incoming red edges, incoming blue edges, outgoing red edges, and outgoing blue edges in a position that would allow such a move at each of the vertices affected by a move at x. Each condition that one of these boundaries be in an appropriate position can be characterized by a pair (x', i') that must be moved prior to (x, i) in any monotone sequence of moves starting from the minimal layout, where x and x' are two features of $E(\mathcal{G})$ that belong to the same triangle. The minimal pair (x, i) in $U(\mathcal{L})$ can be characterized by a constraint that $(x, i') < (x, i)$ in the partial order for each $i' < i$. Thus, by such local considerations, we may find $O(n^2)$ order relations between pairs in $P(\mathcal{G})$ that include all covering relations in $P(\mathcal{G})$. These order relations define a directed acyclic graph from which the partial order $P(\mathcal{G})$ itself may be recovered as the transitive closure. ☐

In Figure 13, each layout is placed next to the corresponding partition of $P(\mathcal{G})$ into two subsets $L(\mathcal{L})$ and $U(\mathcal{L})$. Among the eight layouts in the figure, five of them have exactly one downward neighbor, and these five induce a partial order that is isomorphic to $P(\mathcal{G})$. This isomorphism is no coincidence.

LEMMA 9. *$P(\mathcal{G})$ is order-isomorphic to the partial order P defined in Birkhoff's representation theorem, and the representation of a layout as a partition of this partial order is the same as the representation in Birkhoff's representation theorem.*

Proof. We correspond elements of $P(\mathcal{G})$ one-for-one with elements of P: each element of P is a layout $\mathcal{L}$ with only one downward move, to a layout $\mathcal{L}'$. If this move is on item x, then we associate $\mathcal{L}$ with the pair (x, i), where $i = f_x(\mathcal{L}) - 1 = f_x(\mathcal{L}')$. This pair (x, i) is the single member of the singleton set $L(\mathcal{L}) \cap U(\mathcal{L}')$. Conversely, if (x, i) is any pair in $P(\mathcal{G})$, then we may associate with (x, i) a layout $\mathcal{L}$ that has only one downward move, as follows: starting from $\hat{\mathcal{L}}$, repeatedly perform downward moves that do not reduce $f_x(\mathcal{L})$ to i or below, until no more such moves exist; let $\mathcal{L}$ be the resulting layout. Each move between two layouts changes both the Birkhoff representation (L, U) and the representation $(L(\mathcal{L}), U(\mathcal{L}))$ in corresponding ways. Thus, the two representations are the same. Since $P(\mathcal{G})$ and P have a one-to-one correspondence between elements that causes the distributive lattices of their partitions into downward and upward components to have the same elements and the same covering relation, they must be order-isomorphic. ∎

Thus, we may search through the space of all possible layouts for a given extended graph by instead searching through partitions of $P(\mathcal{G})$ into a downward-closed and an upward-closed subset; the possible layouts correspond one-for-one with partitions of this type. The layout represented by a given partition (L, U) may be found by starting from the bottommost layout in the partial order, and repeatedly performing upward moves that do not increase $f_x(\mathcal{L})$ (where x is the flippable item involved in the move) to a value i such that $(x, i-1) \in U$, until no more such moves are possible. In the following sections we will simply denote $P(\mathcal{G})$ by P to simplify notation.

6.4. Order-theoretic characterization of one-sidedness. We say that a flippable item x is *free* in a layout $\mathcal{L}$ if there is a move on x available in $\mathcal{L}$, and *fixed* otherwise. Let $F(\mathcal{L})$ denote the set of free flippable items for $\mathcal{L}$. The following characterization of this set follows immediately from our representation of the distributive lattice of layouts in terms of the partial order P.

LEMMA 10. *$F(\mathcal{L})$ consists of the items x such that some pair (x, i) is a minimal element of $U(\mathcal{L})$ or a maximal element of $L(\mathcal{L})$.*

We may then characterize the one-sided layouts in terms of $F(\mathcal{L})$:

LEMMA 11. *Let layout $\mathcal{L}$ be dual to an extended graph $E(\mathcal{G})$. Then $\mathcal{L}$ is one-sided if and only if $F(\mathcal{L})$ contains no edges of $\mathcal{G}$.*

Proof. If $\mathcal{L}$ is not one-sided, let s be a maximal segment of $\mathcal{L}$ with multiple rectangles on both of its sides. Then some edge e of the layout from which s is formed must have as one of its endpoints a T-junction formed by the corners of two rectangles on one side of s, and must have on the other endpoint a T-junction formed by the corners of two rectangles on the other side of s, as shown in Figure 12. These four rectangles form an alternatingly colored cycle in the regular edge labeling dual to $\mathcal{L}$, containing a single edge dual to e; thus, one may perform a move on this cycle that recolors e, as shown in the figure, and $e \in F(\mathcal{L})$. Conversely, if an edge e belongs to $F(\mathcal{L})$, then the layout edge dual to e must be part of a segment that (because of the alternating coloring of the regular edge labeling cycle surrounding e) can be extended in both directions to a maximal segment of $\mathcal{L}$ that is not one-sided. Thus, in this case, $\mathcal{L}$ is itself not one-sided. ∎

Hence, the problem of finding a one-sided layout for $E(\mathcal{G})$ becomes equivalent to one of searching for a partition $(L(\mathcal{L}), U(\mathcal{L}))$ of the partial order P in which the free items consist only of degree-four vertices.

As special cases, it follows from Lemma 11 that for an extended graph with no flippable degree-four vertex, a one-sided layout exists if and only if there is exactly one possible layout, for only in that case can $F(\mathcal{L})$ be empty. Thus, we may find such a layout by constructing any layout and testing whether it is one-sided. In an extended graph with a single flippable degree-four vertex, a one-sided layout must be either the minimal or the maximal element of the distributive lattice of layouts, for only those two elements can correspond to partitions (L, U) in which L has no maximal elements or U has no minimal elements. Thus, in this case, we need merely construct both layouts and test them for one-sidedness.

LEMMA 12. *Let $E(\mathcal{G})$ be 4-connected. Then $E(\mathcal{G})$ has more than one regular edge labeling.*

Proof. Consider a regular edge labeling of $E(\mathcal{G})$. Let C be a cycle of $\mathcal{G}$ that is smallest, where the size of a cycle is the number of vertices on or inside C, with the property that it consists of four nonempty chains: a directed chain of red edges, followed by a directed chain of blue edges, followed by another directed chain of red edges, and another directed chain of blue edges. Note that each pair of chains with the same color must necessarily have opposite orientations. Since $E(\mathcal{G})$ is 4-connected, C must exist, because the outer edges of $\mathcal{G}$ have this property. We claim that each chain of C consists of at most one edge. This would imply that this regular edge labeling has a move and the stated result follows. Assume that C contains a chain with more than one edge. Assume without loss of generality that this chain consists of red edges. Choose an internal vertex of this chain and follow a directed sequence π of blue edges (either consistently following the direction of the edges or the opposite direction) from this vertex going into C until it hits a vertex on C again. The endpoints of π cannot be on the same chain, for then the order of the different types of edges around one of these vertices would be incorrect. One of the two cycles formed by C and π has the property described above and is smaller than C, contradicting the fact that C is smallest. ∎

Corollary 2. *Let $E(\mathcal{G})$ be 5-connected. Then every rectangular dual of $E(\mathcal{G})$ is not area-universal.*

6.5. Searching for extreme sets. We have seen in the previous section that one-sided layouts correspond to partitions (L, U) in which the maximal elements of L and the minimal elements of U correspond to degree-four vertices of $\mathcal{G}$. Each vertex v of $\mathcal{G}$ can take only one of these roles: it can be a maximal element of L or a minimal element of U, but not both, because only one move on v is possible in any layout. Thus, if $\mathcal{G}$ has k degree-four vertices, then either the maximal elements of L or the minimal elements of U consist of at most $k/2$ members of P. This motivates the following algorithm for finding one-sided layouts dual to a given graph $\mathcal{G}$:

For each possible extended graph $E(\mathcal{G})$ of the given graph $\mathcal{G}$, and each minimal component $\mathcal{G}'$ of the extended graph, test whether $\mathcal{G}'$ has a one-sided layout. If every minimal component has a one-sided layout, form a layout for $E(\mathcal{G})$ by gluing these component layouts together. If some minimal component does not have a one-sided layout, then neither does $E(\mathcal{G})$. To test whether $\mathcal{G}'$ has a one-sided layout, let k be the number of degree-four vertices in $\mathcal{G}'$, and loop through all sets S consisting of at most $k/2$ members of $P(\mathcal{G}')$ such that each member of S is a pair (x, i), where x is a degree-four vertex of $\mathcal{G}$ and all such degree-four vertices are distinct. For each set S of this type, form a partition (L_1, U_1) in which L_1 consists of all elements in the partial order that are less than or equal to an element in S; if U_1 has no minimal elements corresponding to single edges of $\mathcal{G}$, then return the one-sided layout corresponding to (L_1, U_1). Otherwise, form another partition (L_2, U_2) in which U_2 consists of all elements in the partial order that are greater than or equal to an element in S. If L_2 has no maximal elements corresponding to single edges of $\mathcal{G}$, return the one-sided layout corresponding to this partition. If neither partition formed in this way from each of the sets S gives rise to a one-sided layout, then $\mathcal{G}'$ has no one-sided layout.

Theorem 4. *Let K be the maximum number of flippable degree-four vertices in any minimal separation component of $\mathcal{G}$. Then the algorithm described above finds a one-sided layout dual to $\mathcal{G}$, if one exists, in time $O(n^{K/2+O(1)})$.*

Proof. The correctness of the algorithm follows from the sequence of lemmas above. The choice of the extended graph $E(\mathcal{G})$ multiplies the number of steps of the algorithm by a factor of $O(n^4)$. Within each minimal component $\mathcal{G}'$ we loop through $O(n^{K/2})$ sets S; there are $O(2^k)$ ways to choose a set of at most $k/2$ distinct degree-four vertices, and there are $O(n^{k/2})$ ways to choose numbers for each degree-four vertex (note that we hide a constant in the base). For each set we perform a polynomial amount of work. Thus, the total time is as stated. ∎

6.6. Fixed-parameter tractability. Although conceptually straightforward, the algorithm of Theorem 4 is unsatisfactory from the point of view of fixed-parameter tractability [18]: not just the constant factor in the O-notation, but also the exponent of n, grows with the parameter K. We address this shortcoming by describing an alternative fixed-parameter tractable algorithm for the same problem.

In a layout $\mathcal{L}$ of an extended graph $E(\mathcal{G})$ with no nontrivial separating four-cycles, define an ordered pair (v, w) of degree-four vertices to be a *stretched pair* if there is no sequence of upward moves from $\mathcal{L}$ that moves v without moving w and no sequence of downward moves from $\mathcal{L}$ that moves w without moving v (see Figure 16). That is, if all relevant pairs belong to the partial order P, $(v, f_v(\mathcal{L})) > (w, f_w(\mathcal{L}))$ and $(v, f_v(\mathcal{L}) - 1) > (w, f_w(\mathcal{L}) - 1)$. We introduce a special symbol $\emptyset$, and we also define $(v, \emptyset)$ to be a stretched pair if v is in its maximal state ($f_v(\mathcal{L}) = f_v(\hat{\mathcal{L}})$), and we define $(\emptyset, w)$ to be a stretched pair if w is in its minimal state ($f_w(\mathcal{L}) = 0$). Thus, the stretched pairs form a directed graph on the vertex set V consisting of the degree-four vertices together with the special symbol $\emptyset$. We say that a stretched pair (v, w) *fixes* an edge e if $(v, f_v(\mathcal{L}) - 1) \geq (e, f_e(\mathcal{L}) - 1)$ (or $v = \emptyset$ and $f_e(\mathcal{L}) = 0$) and $(w, f_w(\mathcal{L})) \leq (e, f_e(\mathcal{L}))$ (or $w = \emptyset$ and $f_e(\mathcal{L}) = f_e(\hat{\mathcal{L}})$).

LEMMA 13. *If an edge e is fixed by a stretched pair, then e cannot belong to $F(\mathcal{L})$.*

Proof. Let the stretched pair be (v, w). Because $(v, f_v(\mathcal{L}) - 1) \geq (e, f_e(\mathcal{L}) - 1)$ (or $v = \emptyset$ and $f_e(\mathcal{L}) = 0$), $(e, f_e(\mathcal{L}) - 1)$ is not a maximal element of $L(\mathcal{L})$. Similarly, $(e, f_e(\mathcal{L}))$ is not a minimal element of $U(\mathcal{L})$, because $(w, f_w(\mathcal{L})) \leq (e, f_e(\mathcal{L}))$ (or $w = \emptyset$ and $f_e(\mathcal{L}) = f_e(\hat{\mathcal{L}})$). Therefore, e is fixed in $\mathcal{L}$. ☐

LEMMA 14. *Upward moves on flippable items that are part of the same triangle of $\mathcal{G}$ have a strict cyclical order.*

Proof. First assume that the triangle consists of three edges e_1, e_2, and e_3 not incident to degree-four vertices. In every valid regular edge labeling, a triangle (i) cannot be monocolored and (ii) the two edges with the same color must both be oriented toward or from the shared vertex. Let e_1 and e_2 have the same color in a layout $\mathcal{L}$. Any move on e_3 would violate property (i). Furthermore, it is easy to verify that we cannot do an upward move on both e_1 and e_2 (if this is allowed by the surrounding edges). Assume that we can do an upward move on e_1 resulting in

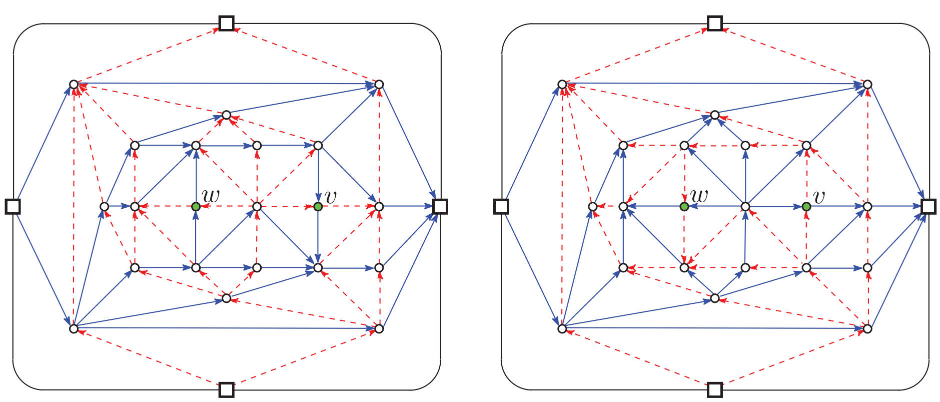

FIG. 16. *Two layouts with a stretched pair (v, w). From both layouts there is no sequence of upward moves that moves v without moving w and no sequence of downward moves that moves w without moving v.*

$\mathcal{L}'$. In $\mathcal{L}'$ we cannot do a move on e_2. Another upward move on e_1 can be performed only after performing upward moves on all surrounding edges, including e_2 and e_3. Hence we can do an upward move only on e_3. Continuing this argumentation, the sequence of upward moves on e_1, e_2, and e_3 from $\mathcal{L}$ must be $e_1, e_3, e_2, e_1, \ldots$. Hence the upward moves on e_1, e_2, and e_3 must follow a strict cyclical order. If a triangle contains a degree-four vertex, then only two flippable items v and e are part of this triangle. Using similar argumentation as above, upward moves on v and e have to alternate and hence these moves also must follow a strict cyclical order. ∎

LEMMA 15. *Let (y, j) cover (x, i) in the partial order P. Then x and y belong to the same triangle of $\mathcal{G}$.*

Proof. If (y, j) covers (x, i), then there must exist a monotone sequence of moves, starting from the minimal element of the distributive lattice of regular edge labelings, such that the penultimate move of the sequence changes $f_x(\mathcal{L})$ from i to $i + 1$ and the final move of the sequence changes $f_y(\mathcal{L})$ from j to $j + 1$. But if x and y did not belong to the same triangle of $\mathcal{G}$, then the four-edge cycle surrounding y would not have its colors or orientation changed by the move on x, and the final move on y could have been performed one step earlier, contradicting the assumption that (y, j) covers (x, i) in the partial order. ∎

LEMMA 16. *Suppose (x, i), $(x, i + 1)$, (y, j), and $(y, j + 1)$ all belong to P. Then $(x, i) \leq (y, j)$ if and only if $(x, i + 1) \leq (y, j + 1)$.*

Proof. By Lemma 15 it suffices to prove that, if (y, j) covers (x, i), then $(y, j+1) \geq (x, i+1)$. For, if we can prove this, then the opposite implication, that if $(y, j+1)$ covers $(x, i + 1)$, then $(y, j) \geq (x, i)$, will follow by clockwise-counterclockwise symmetry. And, if $(y, j) \geq (x, i)$ but (x, i) and (y, j) do not form a covering pair, then we can find a chain of covering pairs connecting them in the partial order, and this result will prove that there exists a corresponding chain of order-related pairs four steps higher, proving that $(y, j + 1) \geq (x, i + 1)$. By Lemmas 14 and 15, upward moves on x and y alternate if (y, j) covers (x, i). It easily follows that $(x, i) \leq (y, j)$ if and only if $(x, i + 1) \leq (y, j + 1)$. ∎

LEMMA 17. *Layout $\mathcal{L}$ is one-sided if and only if every flippable edge e is fixed by some stretched pair.*

Proof. If e is in its minimal state in $\mathcal{L}$, let $v = \emptyset$; otherwise, $(e, f_e(\mathcal{L}) - 1)$ belongs to $L(\mathcal{L})$ and there is a maximal element $(v, f_v(\mathcal{L}) - 1)$ of $L(\mathcal{L})$ above it in the partial order. If e is in its maximal state in $\mathcal{L}$, let $w = \emptyset$; otherwise, $(e, f_e(\mathcal{L}))$ belongs to $U(\mathcal{L})$ and there is a minimal element $(w, f_w(\mathcal{L}))$ of $U(\mathcal{L})$ below it in the partial order. We claim that (v, w) is a stretched pair. For, if all relevant pairs exist in P, then P contains a chain of inequality $(v, f_v(\mathcal{L})) \geq (e, f_e(\mathcal{L})) \geq (w, f_w(\mathcal{L}))$, where the first inequality arises by Lemma 16 and the second comes from the construction of w. Using Lemma 16, we also get $(v, f_v(\mathcal{L}) - 1) \geq (w, f_w(\mathcal{L}) - 1)$, so (v, w) must be stretched in $\mathcal{L}$. ∎

LEMMA 18. *If an edge e is fixed by a stretched pair (v, w) in layout $\mathcal{L}$, then e is fixed in any layout for which (v, w) are stretched.*

Proof. Assume that (v, w) are stretched in $\mathcal{L}'$, so that $(v, f_v(\mathcal{L}')) \geq (w, f_w(\mathcal{L}'))$. This means that $f_v(\mathcal{L}') - f_v(\mathcal{L}) = f_w(\mathcal{L}') - f_w(\mathcal{L})$, because if $f_v(\mathcal{L}') - f_v(\mathcal{L}) < f_w(\mathcal{L}') - f_w(\mathcal{L})$, then, by Lemma 16, $(v, f_v(\mathcal{L}) - 1) \geq (w, f_w(\mathcal{L}))$, which implies that $\mathcal{L}$ does not exist. Also, because of Lemma 16 and $(v, f_v(\mathcal{L})) \geq (w, f_w(\mathcal{L}))$, it must hold that $f_v(\mathcal{L}') - f_v(\mathcal{L}) \leq f_w(\mathcal{L}') - f_w(\mathcal{L})$. Now let $k = f_v(\mathcal{L}') - f_v(\mathcal{L}) = f_w(\mathcal{L}') - f_w(\mathcal{L})$. Because e is fixed in $\mathcal{L}$, we get that $(e, f_e(\mathcal{L}) - 1) \leq (v, f_v(\mathcal{L}) - 1)$ and $(w, f_w(\mathcal{L})) \leq (e, f_e(\mathcal{L}))$. By Lemma 16 we also get that $(e, f_e(\mathcal{L}) + k - 1) \leq (v, f_v(\mathcal{L}') - 1)$ and

$(w, f_w(\mathcal{L}')) \leq (e, f_e(\mathcal{L}) + k)$. This implies that $k = f_e(\mathcal{L}') - f_e(\mathcal{L})$, from which the lemma follows. ∎

LEMMA 19. *Let H consist of a set of pairs (v, w) that should be stretched. Then in polynomial time we may determine whether there exists a layout $\mathcal{L}$ of $E(\mathcal{G})$ in which all pairs in H are stretched.*

Proof. We perform a sequence of upwards moves, starting from the minimal layout, until either a layout satisfying the requirements of H is found or we reach the maximal layout $\hat{L}$. At each step, if the current layout $\mathcal{L}$ does not already meet the requirements, then it must contain a pair (v, w) that should be stretched but are not. If $v = \emptyset$, we terminate the search, as no sequence of upward moves can make w minimal if it is not already. Otherwise, we find a pair $(x, f_x(\mathcal{L}))$ that is minimal in $U(\mathcal{L})$ and below the pair $(v, f_v(\mathcal{L}))$ (possibly $v = x$), and move upward on x. Such a move must eventually be made to reach any layout that meets the requirements of $\mathcal{L}$ and is above $\mathcal{L}$ in the distributive lattice, so each move preserves the set of valid solutions and a solution will eventually be found if one exists. ∎

THEOREM 5. *Let K be the maximum number of degree-four vertices in any minimal separation component of $E(\mathcal{G})$, as before. Then it is possible to find a one-sided layout for $E(\mathcal{G})$, if one exists, in time $2^{O(K^2)}n^{O(1)}$.*

Proof. As above, we test each minimal separation component separately. Within each minimal separation component, we try all possible choices of the information H, consisting of a set of stretched pairs. For each value of H, we determine whether the stretched pairs in H fix all of the edges in $E(\mathcal{G})$. There are $2^{O(K^2)}$ choices, and each can be tested in polynomial time by Lemma 19. ∎

It may be possible to improve the $2^{O(K^2)}$ term in this time bound to $2^{O(K \log K)}$ by using the embedding structure of $E(\mathcal{G})$ to restrict the graph of stretched pairs to be a planar graph, but we have not worked out the details of such an improvement.

7. Orientation-constrained layouts. The approach of section 6, representing layouts as partitions of a partial order into downward- and upward-closed subsets using the lattice structure of regular edge labelings and Birkhoff's representation theorem, can be used to impose further restrictions on the layouts. In this section we show how to deal with constraints on the orientations of the adjacencies of the regions of a layout. We initially assume that $E(\mathcal{G})$ has only trivial separating four-cycles. Afterward we show how to deal with nontrivial separating four-cycles of $E(\mathcal{G})$. Finally we combine orientation constraints with one-sidedness.

7.1. Sublattices from quotient quasi orders. We first consider a more general order-theoretic problem. Let P be a partial order and let C be a (disconnected) undirected *constraint graph* having the elements of P as its vertices. We say that a partition of P into a lower set L and an upper set U *respects* C if there does not exist an edge of C that has one endpoint in L and the other endpoint in U. As we now show, the partitions that respect C may be described as a sublattice of the distributive lattice $J(P)$ defined via Birkhoff's representation theorem from P.

We define a quasi order (that is, reflexive and transitive binary relation) Q on the same elements as P, by adding pairs to the relation that cause certain elements of P to become equivalent to each other. More precisely, form a directed graph that has the elements of P as its vertices, and that has a directed edge from x to y whenever either $x \leq y$ in P or xy is an edge in C, and define Q to be the transitive closure of this directed graph: that is, (x, y) is a relation in Q whenever there is a path from x to y in the directed graph. A subset S of Q is downward-closed (respectively, upward-closed) if there is no pair (x, y) related in Q for which $S \cap \{x, y\} = \{y\}$ (respectively, $S \cap \{x, y\} = \{x\}$).

Denote by $J(Q)$ the set of partitions of Q into a downward-closed and an upward-closed set. Each strongly connected component of the directed graph derived from P and C corresponds to a set of elements of Q that are all related bidirectionally to each other, and Q induces a partial order on these strongly connected components. Therefore, by Birkhoff's representation theorem, $J(Q)$ forms a distributive lattice under set unions and intersections.

LEMMA 20. *The family of partitions in $J(Q)$ is the family of partitions of P into lower and upper sets that respect C.*

Proof. We show the lemma by demonstrating that every partition in $J(Q)$ corresponds to a partition of $J(P)$ that respects C, and the other way around.

In one direction, let (L, U) be a partition in $J(Q)$. Then, since $Q \supset P$, it follows that (L, U) is also a partition of P into a downward-closed and an upward-closed subset. Additionally, (L, U) respects C, for if there were an edge xy of C with one endpoint in L and the other endpoint in U, then one of the two pairs (x, y) or (y, x) would contradict the definition of being downward-closed for L.

In the other direction, let (L', U') be a partition of P into upper and lower sets that respect C, let (x, y) be any pair in Q, and suppose for a contradiction that $x \in U'$ and $y \in L'$. Then there exists a directed path from x to y in which each edge consists either of an ordered pair in P or an edge in C. Since $x \in U'$ and $y \in L'$, this path must have an edge in which the first endpoint is in U' and the second endpoint is in L'. But if this edge comes from an ordered pair in P, then (L', U') is not a partition of P into upper and lower sets, while if this edge comes from C, then (L', U') does not respect C. This contradiction establishes that there can be no such pair (x, y), so (L', U') is a partition of Q into upper and lower sets, as we needed to establish. ∎

If P and C are given as input, we may construct Q in polynomial time: by finding strongly connected components of Q we may reduce it to a partial order, after which it is straightforward to list the partitions in $J(Q)$ in polynomial time per partition.

7.2. Edge orientation constraints. Consider a proper graph $\mathcal{G}$ with corner assignment $E(\mathcal{G})$ and assume that each edge e is given with a set of *forbidden labels*, where a label is a color-orientation combination for an edge, and let $P = P(\mathcal{G})$ be the partial order whose associated distributive lattice $J(P)$ has its elements in one-to-one correspondence with the layouts of $E(\mathcal{G})$. Let x be the flippable item corresponding to e—that is, either the edge itself or the degree-four vertex e is adjacent to. Then in any layout $\mathcal{L}$, corresponding to a partition $(L, U) \in J(P)$, the orientation of e in $\mathcal{L}$ may be determined from $i \bmod 4$, where i is the largest value such that $(x, i) \in L$. Thus if we would like to exclude a certain color-orientation combination for x, we have to find the corresponding value $k \in \mathbb{Z}_4$ and exclude the layouts $\mathcal{L}$ such that $f_x(\mathcal{L}) = k \bmod 4$ from consideration. Thus the set of flipping values for x can be partitioned into *forbidden* and *legal* values for x; instead of considering color-orientation combinations of the flippable items we may consider their flipping values. We formalize this reasoning in the following lemma.

LEMMA 21. *Let $E(\mathcal{G})$ be a corner assignment of a proper graph $\mathcal{G}$. Let x be a flippable item in $E(\mathcal{G})$, let $\mathcal{L}$ be an element of the lattice of regular edge labelings of $E(\mathcal{G})$, and let (L, U) be the corresponding partition of P.*

Then $\mathcal{L}$ satisfies the constraints described by the forbidden labels if and only if for every flippable item x one of the following is true:

- *the highest pair involving x in L is (x, i), where $i+1$ is not a forbidden value for x, or*
- *$(x, 0)$ is in the upper set and 0 is not a forbidden value for x.*

Lemma 20 may be used to show that the set of all constrained layouts is a distributive lattice, and that all constrained layouts may be listed in polynomial time per layout. For technical reasons we augment P to a new partial order $A(P) = P \cup \{-\infty, +\infty\}$, where the new element $-\infty$ lies below all other elements and the new element $+\infty$ lies above all other elements. Each layout of $E(\mathcal{G})$ corresponds to a partition of P into lower and upper sets, which can be mapped into a partition of $A(P)$ by adding $-\infty$ to the lower set and $+\infty$ to the upper set. The distributive lattice $J(A(P))$ thus has two additional elements that do not correspond to layouts of $E(\mathcal{G})$: one in which the lower set is empty and one in which the upper set is empty. We define a constraint graph C having as its vertices the elements of $A(P)$, with edges defined as follows:

- If (x, i) and $(x, i+1)$ are both elements of $A(P)$ and $i+1$ is a forbidden value for x, then we add an edge from (x, i) to $(x, i+1)$ in C.
- If (x, i) is an element of $A(P)$ but $(x, i+1)$ is not, and $i+1$ is a forbidden value for x, then we add an edge from (x, i) to $+\infty$ in C.
- If 0 is a forbidden value for x, then we add an edge from $-\infty$ to $(x, 0)$ in C.

All together, this brings us to the following result.

LEMMA 22. *Let $E(\mathcal{G})$ be an extended graph without nontrivial separating four-cycles and with a given set of forbidden orientations, and let Q be the quasi order formed from the transitive closure of $A(P) \cup C$ as described in Lemma 20. Then the elements of $J(Q)$ corresponding to partitions of Q into two nonempty subsets correspond exactly to the layouts that satisfy the forbidden orientation constraints.*

Proof. By Lemma 21 and the definition of C, a partition in $J(P)$ corresponds to a constrained layout if and only if it respects each of the edges in C. By Lemma 20, the elements of $J(Q)$ correspond to partitions of $A(P)$ that respect C. And a partition of $A(P)$ corresponds to an element of $J(P)$ if and only if its lower set does not contain $+\infty$ and its upper set does not contain $-\infty$. ◻

COROLLARY 3. *Let $E(\mathcal{G})$ be an extended graph without nontrivial separating four-cycles and with a given set of forbidden orientations. There exists a constrained layout for $E(\mathcal{G})$ if and only if there exist more than one strongly connected component in Q.*

COROLLARY 4. *The existence of a constrained layout for a given extended graph $E(\mathcal{G})$ without nontrivial separating four-cycles can be proved or disproved in polynomial time.*

COROLLARY 5. *All constrained layouts for a given extended graph $E(\mathcal{G})$ without nontrivial separating four-cycles can be listed in polynomial time per layout.*

Figure 17 depicts the sublattice resulting from these constructions for the example from Figure 13, with constraints on the orientations of two of the layout edges.

7.3. Junction orientation constraints. So far we have considered forbidding only certain edge labels. However the method above can easily be extended to different types of constraints. For example, consider two elements of P (x, i) and (y, j) that are a covering pair in P; this implies that x and y are two of the three flippable items surrounding a unique T-junction of the layouts dual to $E(\mathcal{G})$. Forcing (x, i) and (y, j) to be equivalent by adding an edge from (x, i) to (y, j) in the constraint graph C can be used for more general constraints: rather than disallowing one or more of the four orientations for any single flippable item, we can disallow one or more of the twelve orientations of any T-junction. For instance, by adding equivalences of this

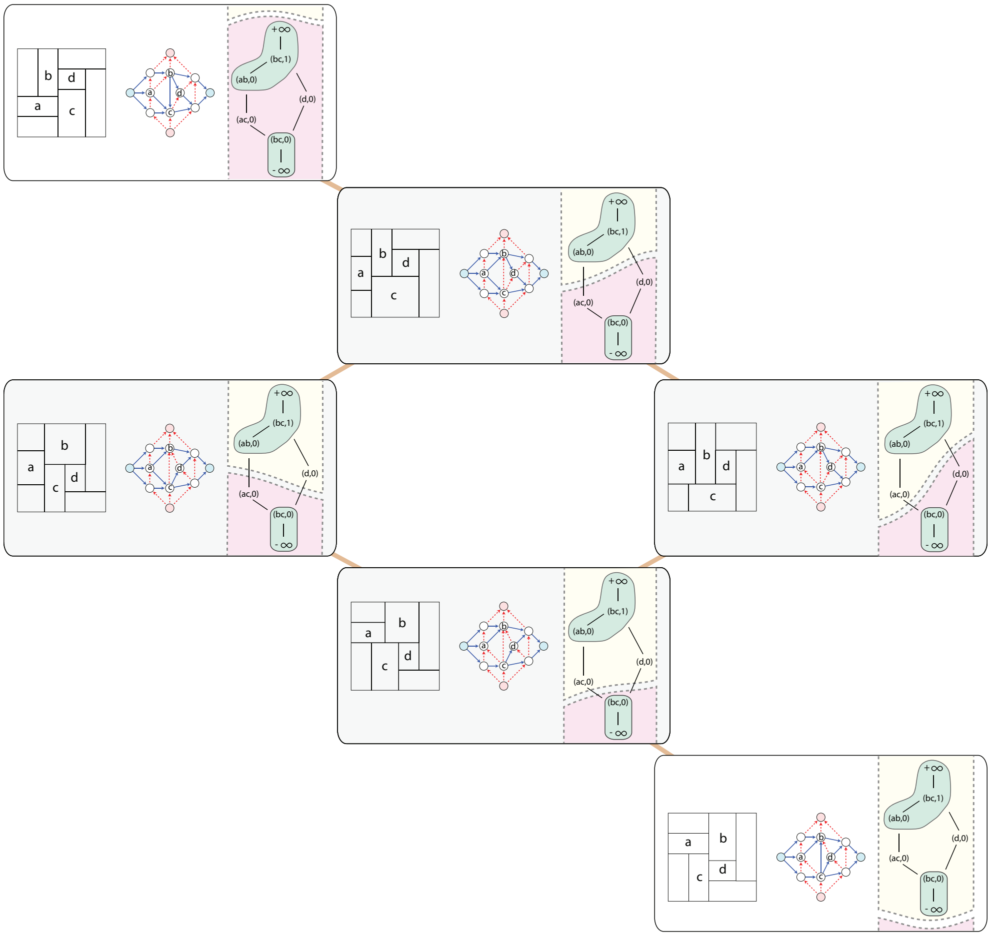

FIG. 17. *The family of rectangular layouts dual to a given extended graph $E(\mathcal{G})$ satisfying the constraints that the edge between rectangles a and b must be vertical (cannot be colored red) and that the edge between rectangles b and c must be horizontal (cannot be colored blue). The green regions depict strongly connected components of the associated quasi order Q. The four central shaded elements of the lattice correspond to layouts satisfying the constraints. Color is available only in the online version.*

type we could force one of the three rectangles at the T-junction to be the one with the 180-degree angle.

Any internal T-junction of a layout for $E(\mathcal{G})$ (dual to a triangle of $\mathcal{G}$) has 12 potential orientations: each of its three rectangles can be the one with the 180-degree angle, and with that choice fixed there remain four choices for the orientation of the junction. In terms of the regular edge labeling, any triangle of $\mathcal{G}$ may be colored and oriented in any of 12 different ways. For a given covering pair (x, i) and (y, j), let $C^{i,j}_{x,y}$ denote the set of edges between pairs $(x, i+4k)$ and $(y, j+4k)$ for all possible integer values of k, together with an edge from $-\infty$ to $(y, 0)$ if $j \bmod 4 = 0$, and an edge from $(x, i+4k)$ to $+\infty$ if $i+4k$ is the largest value of i' such that (x, i') belongs to P. Any T-junction is associated with 12 of these edge sets, as there are three ways of choosing a pair of adjacent flippable items and four ways of choosing values of i and j (mod 4) that lead to covering pairs. Including any one of these edge sets in the constraint graph C corresponds to forbidding one of the 12 potential orientations of the T-junction.

Thus, Lemma 22 and its corollaries may be applied without change to dual graphs $E(\mathcal{G})$ with junction orientation constraints as well as edge orientation constraints, as long as $E(\mathcal{G})$ has no nontrivial separating four-cycles.

7.4. Constrained layouts for unconstrained dual graphs. Proper graphs with nontrivial separating four-cycles still have finite distributive lattices of layouts, but it is no longer possible to translate orientation constraints into equivalences between members of an underlying partial order. The reason is that, for a graph without trivial separating four-cycles, the orientation of a feature of the layout changes only for a flip involving that feature, so that the orientation may be determined from the flip count mod 4. For more general graphs the orientation of a feature is changed not only for flips directly associated with that feature, but also for flips associated with larger four-cycles that contain the feature, so the flip count of the feature no longer determines its orientation. For this reason we again treat general proper graphs by decomposing them into minimal separation components with respect to separating four-cycles and piecing together solutions found separately within each of these components.

We use the representation of a graph as a tree of minimal separation components in our search for constrained layouts for $\mathcal{G}$. We first consider each such minimal component separately for every possible mapping of vertices of C to $\{l, t, r, b\}$ (we call these mappings the *orientation* of $E(\mathcal{G})$). Different orientations imply different flipping values of forbidden labels for the given constraint function, since the flipping numbers are defined with respect to the orientation of $E(\mathcal{G})$. Bearing that in mind we are going to test the graph $E(\mathcal{G})$ for existence of a constrained layout in the following way.

For each piece in a bottom-up traversal of the decomposition tree and for each orientation of the corners of the piece:

1. Find the partial order P describing the layouts of the piece.
2. Translate the orientation constraints within the piece into a constraint graph on the augmented partial order $A(P)$.
3. Compute the strongly connected components of the union of $A(P)$ with the constraint graph, and form a binary relation that is a subset of Q and that includes all covering relations in Q by finding the components containing each pair of elements in each covering relation in P.
4. Translate the existence or nonexistence of a layout into a constraint on the label of the corresponding degree-four vertex in the parent piece of the decomposition. That is, if the constrained layout for a given orientation of $E(\mathcal{G}')$ does not exist, forbid (in the parent piece of the decomposition) the label of the degree-four vertex corresponding to that orientation.

If the algorithm above confirms the existence of a constrained layout, we may list all layouts satisfying the constraints as follows. For each piece in the decomposition tree, in top-down order:

1. List all lower sets of the corresponding quasi order Q.
2. Translate each lower set into a layout for that piece.
3. For each layout and each child of the piece in the decomposition tree, recursively list the layouts in which the child's corner orientation matches the labeling of the corresponding degree-four vertex of the outer layout.
4. Glue the inner and outer layouts together.

THEOREM 6. *The existence of a constrained layout for a proper graph $\mathcal{G}$ can be found in polynomial time in $|\mathcal{G}|$. The set of all constrained layouts for the graph can be found in polynomial time per layout.*

Proof. By Lemma 7, the partial order P describing the layouts of each piece has a number of elements and covering pairs that is quadratic in the number of vertices in the dual graph of the piece, and a description of this partial order in terms of its covering pairs may be found in quadratic time. The strongly connected component calculation within the algorithm takes time linear in the size of P, and therefore the overall algorithm for testing the existence of a constrained layout takes time $O(n^2)$, where n is the number of vertices in the given dual graph. ∎

7.5. Finding area-universal constrained layouts. In section 6 we presented a fixed-parameter tractable algorithm for the problem of searching for an area-universal layout. We can use a similar approach to search for an area-universal layout with constrained orientations. Within each piece of the separation decomposition, we consider $2^{O(k^2)}$ sets of stretched pairs in P, as before. However, to test one of these sets, we perform a monotonic sequence of flips in $J(Q)$, at each point either flipping an element of Q that contains the upper element of a pair that should be stretched, or performing a flip that is a necessary prerequisite to flipping such an upper element. Eventually, this process will reach either an area-universal layout for the piece or the top element of the lattice; in the latter case, no area-universal layout having that pattern of stretched pairs exists. By testing all sets of stretched pairs, we may find whether an area-universal layout matching the constraints exists for any corner coloring of any piece in the separation decomposition. These constrained layouts for individual pieces can then be combined by the same tree traversal of the separation decomposition tree as discussed earlier, due to Corollary 1. This fixed-parameter tractable algorithm also runs in $2^{O(K^2)}n^{O(1)}$ time.

8. Layouts with given dual spanning trees. Rinsma [23] considered the question of finding a cartogram for a given weight vector, such that the dual graph $\mathcal{G}$ has a given tree $\mathcal{T}$ as its spanning tree. She showed that, by a simple layout process in which the root of $\mathcal{T}$ is placed at the bottom of a layout and recursively constructed layouts for its children are placed above it, such a cartogram can always be found. However, her layouts are not, in general, area-universal. For instance, in the layout shown in the center of Figure 18, produced by her algorithm, the line segment with rectangles D and F to its left and with rectangles G and H to the right is not one-sided, showing that the tree in this example leads to a nonarea-universal layout

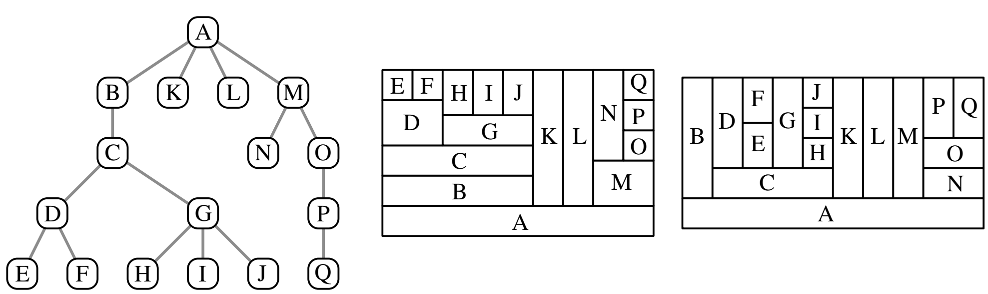

FIG. 18. *A dual spanning tree $\mathcal{T}$, Rinsma's nonarea-universal layout, and our area-universal layout for $\mathcal{T}$.*

according to her algorithm.

However, a simple modification of Rinsma's layout process can be used to generate area-universal layouts that have the given tree as a spanning tree of the dual. The method produces layouts in which the root of a tree either covers the entire bottom edge of the layout or the entire left edge of the layout. For a given tree, to find a layout with the root at the bottom, use the same algorithm recursively to generate layouts for each subtree rooted at a child of the root with the child at the left, and place these subtree layouts in left-to-right order above the bottom root rectangle. Symmetrically, to find a layout with the root on the left, use the same algorithm recursively to generate layouts for each subtree rooted at a child with the child on the bottom, and place these subtree layouts in bottom-to-top order to the right of the root rectangle. Thus, for a given tree, the layouts with the root at the bottom and with the root at the left are mirror images of each other, as reflected across a line with slope one. The area-universal layout resulting from this algorithm for the same example tree is shown on the right of Figure 18.

THEOREM 7. *For any tree $\mathcal{T}$ the algorithm described above finds an area-universal layout, having $\mathcal{T}$ as a spanning tree of the dual, in time linear in the size of $\mathcal{T}$.*

Proof. At each level of the recursion, each child is placed adjacently to the root of its subtree, so $\mathcal{T}$ is a spanning tree of the dual, and the algorithm clearly runs in linear time. Each maximal segment of the layout, other than the outer boundaries of the root rectangle, either separates the root of a subtree from its children or one child subtree from the next child subtree. If the segment separates the root of a subtree from its children, it forms a side of the root rectangle, and if it separates one child subtree from the next, it forms a side of the root of the second subtree. Thus, each maximal segment is the side of a rectangle and hence the layout is one-sided. The result follows by Theorem 2. ∎

9. Conclusions and open problems. We presented a simple necessary and sufficient condition for a rectangular layout to be area-universal. We also described how to find a layout that is equivalent or order-equivalent to a given layout and that realizes a given weight function as a cartogram. For a given graph $\mathcal{G}$, we presented an algorithm to find a one-sided and hence area-universal layout dual to $\mathcal{G}$. Additionally we give algorithms to find layouts (one-sided or not) with certain orientation constraints. Unlike much past work on rectangular layouts, we did not restrict our attention to sliceable layouts, dual graphs without separating four-cycles, or other such special cases.

There remain several questions for further investigation. An important problem in the generation of rectangular layouts with special properties, one that has resisted our lattice-theoretic approach, is the generation of sliceable layouts. If we are given a graph $\mathcal{G}$, can we determine whether it is the graph of a sliceable layout in polynomial time? Also, even though we have an algorithm for finding area-universal rectangular cartograms, it is not fully polynomial, and it would be of interest to find faster algorithms or determine whether it is NP-complete to test whether an area-universal cartogram exists for a given dual graph. And although our algorithms for finding orientation-constrained layouts (excluding one-sidedness) are polynomial time, there seems no reason intrinsic to the problem for them to take as much time as they do: can we achieve subquadratic time bounds for finding orientation-constrained layouts, perhaps by using an algorithm based more on the special features of the problem and less on general ideas from lattice theory? Finally, if an area-universal cartogram does not exist, but we are given an area assignment or a range of area assignments, can we efficiently find a layout realizing this assignment or assignments? Past work on related problems suggests that such problems might be difficult [1].

Moving beyond layouts, there are several other important combinatorial constructions that may be represented using finite distributive lattices, notably the set of matchings and the set of spanning trees of a planar graph, and certain sets of orientations of arbitrary graphs [19]. It would be of interest to investigate whether our approach of combining the underlying partial order of a lattice with a constraint graph produces useful versions of constrained matching and constrained spanning tree problems, and whether other algorithms that have been developed in the more general context of distributive finite lattices [20] might fruitfully be applied to lattices of rectangular layouts. Recently, the same framework of reducing to a partial order has successfully been used to construct adjacency-preserving spatial treemaps [5].

Acknowledgments. B. Speckmann would like to thank O. Aichholzer, T. Hackl, and B. Vogtenhuber for stimulating discussions concerning the topic of this paper. The authors would also like to thank an anonymous referee for helpful comments.

REFERENCES

[1] T. Biedl and B. Genc, *Complexity of Octagonal and Rectangular Cartograms*, Technical report CS-2005-13, School of Computer Science, University of Waterloo, Waterloo, ON, Canada, 2005.
[2] G. Birkhoff, *Rings of sets*, Duke Math. J., 3 (1937), pp. 443–454.
[3] G. Birkhoff and S. A. Kiss, *A ternary operation in distributive lattices*, Bull. Amer. Math. Soc., 52 (1947), pp. 749–752.
[4] M. Bruls, K. Huizing, and J. J. van Wijk, *Squarified treemaps*, in Proceedings of the Eurographics and IEEE TCVG Symposium on Visualization, Springer, 2000, pp. 33–42.
[5] K. Buchin, D. Eppstein, M. Löffler, M. Nöllenburg, and R. I. Silveira, *Adjacency-preserving spatial treemaps*, in Proceedings of the 12th Algorithms and Data Structures Symposium (WADS), 2011, pp. 159–170; available online at http://arxiv.org/abs/1105.0398.
[6] P. Dasgupta and S. Sur-Kolay, *Slicible rectangular graphs and their optimal floorplans*, ACM Trans. Design Automat. Electron. Systems, 6 (2001), pp. 447–470.
[7] C. F. Earl and L. J. March, *Architectural applications of graph theory*, Applications of Graph Theory, R. Wilson and L. Beineke, eds., Academic Press, London, 1979, pp. 327–355.
[8] D. Eppstein, *Arboricity and bipartite subgraph listing algorithms*, Inform. Process. Lett., 51 (1994), pp. 207–211.
[9] D. Eppstein, *Regular labelings and geometric structures*, in Proceedings of the 22nd Canadian Conference on Computational Geometry, 2010, pp. 125–130.
[10] D. Eppstein and E. Mumford, *Steinitz theorems for orthogonal polyhedra*, in Proceedings of the 26th ACM Symposium on Computational Geometry, 2010, pp. 429–438.
[11] S. Felsner, *Lattice structures from planar graphs*, Electron. J. Combin., 11 (2004), R15.
[12] É. Fusy, *Transversal structures on triangulations, with application to straight-line drawing*, in Proceedings of the 13th International Symposium on Graph Drawing (GD 2005), Lecture Notes in Comput. Sci. 3843, Springer, Berlin, 2006, pp. 177–188.
[13] É. Fusy, *Transversal structures on triangulations: A combinatorial study and straight-line drawings*, Discrete Math., 309 (2009), pp. 1870–1894.
[14] G. Kant and X. He, *Regular edge labeling of* 4*-connected plane graphs and its applications in graph drawing problems*, Theoret. Comput. Sci., 172 (1997), pp. 175–193.
[15] K. Koźmiński and E. Kinnen, *Rectangular duals of planar graphs*, Networks, 5 (1985), pp. 145–157.
[16] N. Megiddo, *Linear programming in linear time when the dimension is fixed*, J. ACM, 31 (1984), pp. 114–127.
[17] E. Mumford, *Drawing Graphs for Cartographic Applications*, Ph.D. thesis, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, 2008.
[18] R. Niedermeier, *Invitation to Fixed-parameter Algorithms*, Oxford Lecture Ser. Math. Appl. 31, Oxford University Press, Oxford, UK, 2006.

[19] J. Propp, *Lattice Structure for Orientations of Graphs*, Electronic preprint arxiv:math/0209005, 1993.

[20] J. Propp, *Generating random elements of finite distributive lattices*, Electron. J. Combin., 4 (1997), R15.

[21] E. Raisz, *The rectangular statistical cartogram*, Geograph. Rev., 24 (1934), pp. 292–296.

[22] I. Rinsma, *Nonexistence of a certain rectangular floorplan with specified areas and adjacency*, Environ. Plann. B, 14 (1987), pp. 163–166.

[23] I. Rinsma, *Rectangular and orthogonal floorplans with required rooms areas and tree adjacency*, Environ. Plann. B, 15 (1988), pp. 111–118.

[24] H. Tang and W. K. Chen, *Generation of rectangular duals of a planar triangulated graph by elementary transformations*, in Proceedings of the IEEE International Symposium on Circuits and Systems, Vol. 4, 1990, pp. 2857–2860.

[25] P. Ungar, *On diagrams representing maps*, J. London Math. Soc., 28 (1953), pp. 336–342.

[26] S. Wimer, I. Koren, and I. Cederbaum, *Floorplans, planar graphs, and layouts*, IEEE Trans. Circuits Syst., 35 (1988), pp. 267–278.

[27] G. K. H. Yeap and M. Sarrafzadeh, *Sliceable floorplanning by graph dualization*, SIAM J. Disc. Math., 8 (1995), pp. 258–280.