Computing Cartograms with Optimal Complexity

Md. Jawaherul Alam[1]
University of Arizona,
Tucson, AZ, USA
mjalam@email.arizona.edu

Therese Biedl[2]
University of Waterloo,
Waterloo, ON N2L 3G1, Canada
biedl@uwaterloo.ca

Stefan Felsner[3]
Technische Universität Berlin,
Berlin, Germany
felsner@math.tu-berlin.de

Michael Kaufmann
Eberhard Karls Universität Tübingen,
Tübingen, Germany
mk@informatik.uni-tuebingen.de

Stephen G. Kobourov[1]
University of Arizona,
Tucson, AZ, USA
kobourov@cs.arizona.edu

Torsten Ueckerdt[3]
Technische Universität Berlin,
Berlin, Germany
ueckerdt@math.tu-berlin.de

Abstract

In a rectilinear dual of a planar graph vertices are represented by simple rectilinear polygons and edges are represented by side-contact between the corresponding polygons. A rectilinear dual is called a cartogram if the area of each region is equal to a pre-specified weight of the corresponding vertex. The complexity of a cartogram is determined by the maximum number of corners (or sides) required for any polygon. In a series of papers the polygonal complexity of such representations for maximal planar graphs has been reduced from the initial 40 to 34, then to 12 and very recently to the currently best known 10. Here we describe a construction with 8-sided polygons, which is optimal in terms of polygonal complexity as 8-sided polygons are sometimes necessary. Specifically, we show how to compute the combinatorial structure and how to refine the representation into an area-universal rectangular layout in linear time. The exact cartogram can be computed from the area-universal rectangular layout with numerical iteration, or can be approximated with a hill-climbing heuristic.

We also describe an alternative construction for Hamiltonian maximal planar graphs, which allows us to directly compute the cartograms in linear time. Moreover, we prove that even for Hamiltonian graphs 8-sided rectilinear polygons are necessary, by constructing a non-trivial lower bound example. The complexity of the cartograms can be reduced to 6 if the Hamiltonian path has the extra property that it is one-legged, as in outer-planar graphs. Thus, we have optimal representations (in terms of both polygonal complexity and running time) for Hamiltonian maximal planar and maximal outer-planar graphs.

---

* This research was initiated at the Dagstuhl Seminar 10461 on Schematization.
[1] Research funded in part by NSF grants CCF-0545743 and CCF-1115971.
[2] Research supported by NSERC.
[3] Research partially supported by EUROGIGA project GraDR and DFG Fe 340/7-2.

1 Introduction

There is a large body of work about representing planar graphs as contact graphs, i.e., graphs whose vertices are represented by geometrical objects with edges corresponding to two objects touching in some specified fashion. Typical classes of objects might be curves, line segments, or polygons. An early result is Koebe's 1936 theorem [18] that all planar graphs can be represented by touching disks.

In this paper, we consider contact representations of planar graphs, with vertices represented by simple interior-disjoint polygons and adjacencies represented by a non-trivial contact (shared boundary) between the corresponding polygons. We are specifically interested in the rectilinear weighted version where the vertices are represented by simple (axis-aligned) rectilinear polygons. This type of a representation is known as a *rectilinear dual* of the input planar graph.

In the weighted version, the input is a planar graph $G = (V, E)$ along with a weight function $w : V(G) \rightarrow \mathbb{R}^+$ that assigns a weight to each vertex of G. A rectilinear dual is called a *cartogram* if the area of each region is equal to the pre-specified weight of the corresponding vertex. Such representations have practical applications in cartography [24], geography [28] and sociology [15], but also in VLSI Layout and floor-planning [22]. Other applications can be found in visualization of relational data, where using the adjacency of regions to represent edges in a graph can lead to a more compelling visualization than just drawing a line segment between two points [4].

For rectilinear duals (unweighted) and for cartograms (weighted) it is often desirable, for aesthetic, practical and cognitive reasons, to limit the *polygonal complexity* of the representation, measured by the number of sides (or by the number of corners). Similarly, it is also desirable to minimize the unused area in the representation, also known as *holes* in floor-planning and VLSI layouts. A given rectilinear dual is *area-universal* if it can realize a cartogram with any pre-specified set of weights for the vertices of the graph without disturbing the underlying adjacencies and without increasing the polygonal complexity.

With these considerations in mind, we study the problem of constructing area-universal rectilinear duals and show how to compute cartograms with worst-case optimal polygonal complexity and without any holes.

1.1 Related Work

In our paper and in most of the other papers cited here, "planar graph" refers to an inner-triangulated planar graph with a simple outer-face; the former restriction is required if at most three rectilinear polygons are allowed to meet in a point and the latter restriction is customary to achieve that the union of all the polygons in the representation is a rectangle.

Rectilinear duals (unweighted) were first studied in graph theoretic context, and then with renewed interest in the context of VLSI layouts and floor planning. It is known that 8 sides are sometimes necessary and always sufficient [13, 21, 34].

The case when the rectilinear polygons are restricted to rectangles has been of particular interest and there are several (independent) characterizations of the class of planar graphs that allows such *rectangular duals* [19, 20, 30]. A historical overview and a summary of the state of the art in the rectangle contact graphs literature can be found in Buchsbaum *et al.* [4].

In the above results on rectilinear duals and rectangular duals, the areas of the polygons are not considered; that is, these results deal with the unweighted version of the problem. The weighted version dates back to 1934 when Raisz described rectangular cartograms [24]. Algorithms by van Kreveld and Speckmann [31] and Heilmann *et al.* [14] yield representations with touching rectangles but the adjacencies may be disturbed and there may also be a small distortions of the weights. Recently, Eppstein *et al.* [11] characterized the class of planar graphs that have area-universal rectangular duals. The construction of the actual cartogram, given the area-universal rectilinear dual and the weight function, can be accomplished using a result by Wimer *et al.* [33], which in turn requires numerical iteration.

The result of Eppstein *et al.* above is restricted to planar graphs that have rectangular duals. Going back to the more general rectilinear duals, leads to a series of papers where the main goal has been to reduce the

polygonal complexity while respecting all areas and adjacencies. De Berg *et al.* initially showed that 40 sides suffice [8]. This was later improved to 34 sides [17]. In a recent paper [3] the polygonal complexity was reduced to 12 sides and even more recently to 10 sides [1].

Side contact representations of planar graphs have also been studied without the restriction to rectilinear polygons. In the unweighted case 6-sided polygons are sometimes necessary and always sufficient [12]. The constructive upper bound relies on convex 6-sided polygons. In the weighted version, where the area of each polygon is prescribed, examples are known for which polygons with 7 sides are necessary [29]. This lower bound is matched by constructive upper bound of 7 sides if holes are allowed [2]. In the same paper it is shown that even allowing arbitrarily high polygonal complexity and holes of arbitrary size, there exist examples with prescribed areas which cannot be represented with convex polygons. If holes are not allowed then the best previously known polygonal complexity is 10, and it is achieved with rectilinear polygons [1].

1.2 Our Results

Recall that the known lower bound on the polygonal complexity even for unweighted rectilinear duals is 8while the best known upper bound is 10. Here we present the first construction that matches the lower bound. Specifically, our construction produces 8-sided area-universal rectilinear duals in linear time, and is thus optimal in terms of polygonal complexity. The exact cartogram can be computed from the area-universal rectangular layout with numerical iteration, or can be approximated with a hill-climbing heuristic.

For Hamiltonian maximal planar graphs we have an alternative construction which allows us to directly compute cartograms with 8-sided rectilinear polygons in linear time. Moreover, we prove that 8-sided rectilinear polygons are necessary by constructing a non-trivial lower bound example. If the Hamiltonian path has the extra property that it is one-legged, then we can reduce the polygonal complexity and realize cartograms with 6-sided polygons. This can be used to obtain 6-sided cartograms of maximal outer-planar graphs. Thus we have optimal (in terms of both polygonal complexity and running time) representations for Hamiltonian maximal planar and maximal outer-planar graphs.

2 Preliminaries

A *planar graph*, $G = (V, E)$, is one that has a drawing without crossing in the plane along with an embedding, defined via the cyclic ordering of edges around each vertex. A *plane graph* is a fixed planar embedding of a planar graph. It splits the plane into connected regions called *faces*; the unbounded region is the *outer-face* and all other faces are called *interior faces*. A planar (plane) graph is *maximal* if no edge can be added to it without violating planarity. Thus each face of a maximal plane graph is a triangle. A *Hamiltonian cycle* in a graph G is a simple cycle containing all the vertices of G. A graph G is called *Hamiltonian* if it contains a Hamiltonian cycle.

A set P of closed simple interior-disjoint polygons with an isomorphism $\mathcal{P} : V \rightarrow P$ is a *polygonal contact representation* of a graph if for any two vertices $u, v \in V$ the boundaries of $\mathcal{P}(u)$ and $\mathcal{P}(v)$ share a non-empty line-segment if and only if (u, v) is an edge. Such a representation is known as a *rectilinear dual* of the input graph if polygons in P are rectilinear. In the weighted version the input is the graph G, along with a weight function $w : V(G) \rightarrow \mathbb{R}^+$ that assigns a weight to each vertex of G. A rectilinear dual is called a *cartogram* if the area of each polygon is equal to the pre-specified weight of the corresponding vertex. We define the *complexity of a polygon* as the number of sides it has. A common objective is to realize a given graph and a set of weights, using polygons with minimal complexity.

2.1 Canonical Orders and Schnyder Realizers

Next we briefly summarize the concepts of a "canonical order" of a planar graph [10] and that of a "Schnyder realizer" [27]. Let $G = (V, E)$ be a maximal plane graph with outer vertices u, v, w in clockwise order. Then we can compute in linear time [7] a *canonical order* or *shelling order* of the vertices $v_1 = u$, $v_2 = v$, v_3, ..., $v_n = w$, which is defined as one that meets the following criteria for every $4 \leq i \leq n$.

- The subgraph $G_{i-1} \subseteq G$ induced by v_1, v_2, ..., v_{i-1} is biconnected, and the boundary of its outer face is a cycle C_{i-1} containing the edge (u, v).
- The vertex v_i is in the exterior face of G_{i-1}, and its neighbors in G_{i-1} form an (at least 2-element) subinterval of the path $C_{i-1} - (u, v)$.

A *Schnyder realizer* of a maximal plane graph G is a partition of the interior edges of G into three sets $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$ of directed edges such that for each interior vertex v, the following conditions hold:

- v has out-degree exactly one in each of $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$,
- the counterclockwise order of the edges incident to v is: entering $\mathcal{S}_1$, leaving $\mathcal{S}_2$, entering $\mathcal{S}_3$, leaving $\mathcal{S}_1$, entering $\mathcal{S}_2$, leaving $\mathcal{S}_3$.

Schnyder proved that any maximal plane graph has a Schnyder realizer and it can be computed in $O(n)$ time [27]. The first condition implies that $\mathcal{S}_i$, for $i = 1, 2, 3$ defines a tree rooted at exactly one exterior vertex and containing all the interior vertices such that the edges are directed towards the root. Denote by $\Phi_k(v)$ the parent of vertex v in tree T_k. The following well-known lemma shows a profound connection between canonical orders and Schnyder realizers.

Lemma 2.1 *Let G be a maximal plane graph. Then the following hold.*

(a) A canonical order of the vertices of G defines a Schnyder realizer of G, where the outgoing edges of a vertex v are to its first and last predecessor (where "first" is w.r.t. the clockwise order around v), and to its highest-numbered successor.

(b) A Schnyder realizer with trees S_1, S_2, S_3 defines a canonical order, which is a topological order of the acyclic graph $\mathcal{S}_1^{-1} \cup \mathcal{S}_2^{-1} \cup \mathcal{S}_3$, where $\mathcal{S}_k^{-1}$ is the tree $\mathcal{S}_k$ with the direction of all its edges reversed.

3 Cartograms with 8-Sided Polygons

In this section we show that 8-sided polygons are always sufficient and sometimes necessary for a cartogram of a maximal planar graph. Our algorithm for constructing 8-sided area-universal rectilinear duals has three main phases. In the first phase we create a contact representation of the graph G, where each vertex of G is represented by an upside-down **T**, i.e., a horizontal segment and a vertical segment. Figures 1(a)-(b) show a maximal planar graph and its contact representation using **T**'s, where the three ends of each **T** are marked with arrows. In the second phase we make both the horizontal and vertical segments of each **T** into thin polygons with λ thickness for some $\lambda > 0$. We then have a contact representation of G with T-shaped polygons as illustrated in Figure 1(c). In the third phase we remove all the unused area in the representation by assigning each (rectangular) hole to one of the polygons adjacent to it, as illustrated in Figure 1(d). We show that the resulting representation is an area-universal rectilinear dual of G with polygonal complexity 8, as illustrated in Figure 1(e).

3.1 Constructing Contact Representation with T's

Our contact representation with **T**'s is similar to the approach described by de Fraysseix *et al.* [9].

Let G be a planar graph. As mentioned earlier, we may assume that G is internally triangulated and has a simple outer-face. If need be, we can add two vertices (which we later choose as v_1 and v_2) and connect them to the outer-face to ensure that the graph is maximal. Now let v_1, v_2, v_3, ..., v_n be a canonical order of the vertices in G with corresponding Schnyder trees $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$ rooted at v_1, v_2 and v_n, Add to $\mathcal{S}_1$ the two edges (v_2, v_1) and (v_n, v_1) oriented towards v_1 and add to $\mathcal{S}_2$ the edge (v_n, v_2) oriented towards v_2. In what follows, we sometimes identify vertex v_i with its canonical label i.

We assign to vertex i the **T**-shape T_i consisting horizontal and vertical segments h_i and b_i. Begin by placing T_1 and T_2 so that h_1 is placed at $y = 1$, h_2 is placed at $y = 2$, the topmost points of both b_1 and b_2 have y-coordinate $n + 1$ and the leftmost point of the h_2 touches b_1. Next the algorithm iteratively constructs the contact representation by defining T_k so that h_k is placed at $y = k$ and the topmost point of b_k has y-coordinate $\Phi_3(k)$ for $3 \leq k < n$. After the k-th step of the algorithm we have a contact representation

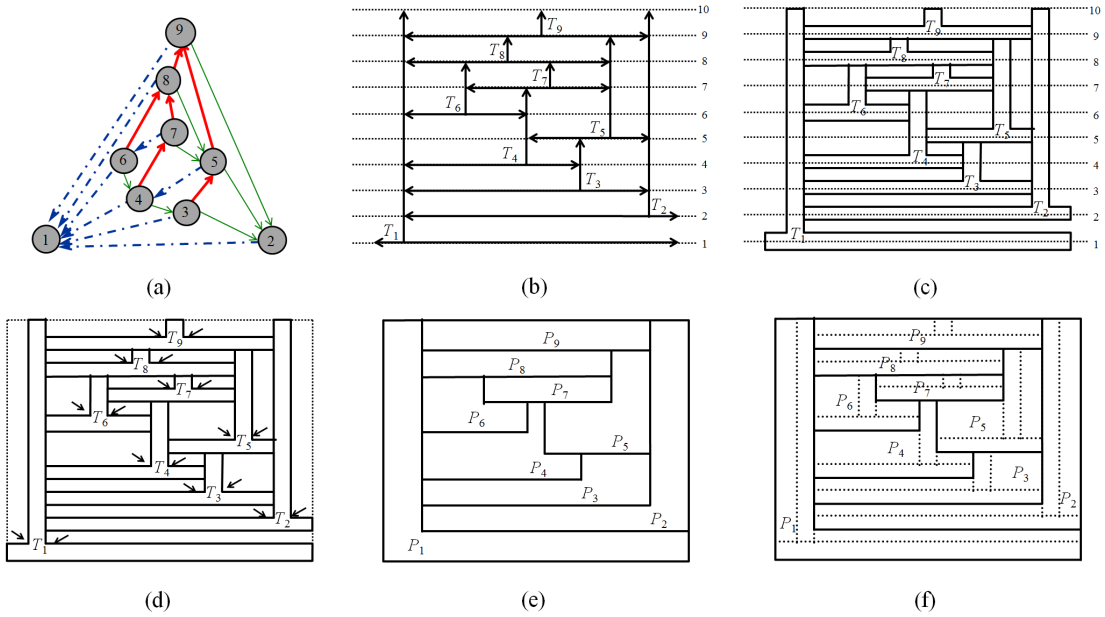


Figure 1: Construction of a rectilinear dual of a maximal planar graph with 8-sided polygons.

of G_k, and we maintain the invariant that the order of the vertical segments with non-empty parts in the half-plane $y > k$ corresponds to the same circular order of the vertices along $C_k - (v_1, v_2)$.

Consider inserting T_k for v_k. The neighbors v_{k_1}, v_{k_2}, ..., v_{k_d} of v_k in G_{k-1} form a subinterval of $C_{k-1} - (v_1, v_2)$ and hence the corresponding vertical segments are also in the same order in the half-plane $y > k - 1$ of the representation of G_{k-1}. Since $v_k = \Phi_3(v_{k_i})$ for $1 < i < d$ (Lemma 2.1), the topmost points of the corresponding vertical segments have y-coordinate k. As v_{k_1} and v_{k_d} are the parents of v_k in $\mathcal{S}_1$ and $\mathcal{S}_2$, the x-coordinates of b_{k_1} and b_{k_d} define the x-coordinates of the two endpoints of h_k. Let these coordinates be x_l and x_r; then h_k is placed between the two points (x_l, k), (x_r, k) and b_k is placed between the two points (x_m, k), $(x_m, \Phi_3(k))$ with $x_l + 1 < x_m < x_r - 1$. Finally for $k = n$, we place T_n so that h_n touches b_1 to the left and b_2 to the right and the topmost point of b_n has y-coordinate $n + 1$.

We note here that this representation can be computed in linear time so that all coordinates are integers by pre-computing a topological order π of $\mathcal{S}_1^{-1} \cup \mathcal{S}_2$; then h_k is the segment $[\pi(\Phi_1(k)), \pi(\Phi_2(k))] \times k$ and b_k is the segment $\pi(k) \times [k, \Phi_3(k)]$.

3.2 λ-Fattening of T_i's

Let Γ' be the contact representation of G using **T**'s obtained above. In this phase of the algorithm, we "fatten" **T**'s so that each vertex is represented by a T-shaped polygon. We replace each horizontal segment h_i by an axis-aligned rectangle H_i which has the same width as h_i, and whose top (bottom) side is $\lambda/2$ above (below) h_i, for some $0 < \lambda$, as illustrated in Figure 2(a). Similarly, we replace each vertical segment b_i by an axis-aligned rectangle B_i which has the same height as b_i and whose left (right) side is $\lambda/2$ to the left (right) of b_i. We call this process λ-*fattening* of T_i. Note that this process creates intersections of H_i with B_i, $B_{\Phi_1(i)}$ and $B_{\Phi_2(i)}$ and intersection of B_i with $H_{\Phi_3(i)}$. We remove these intersections by replacing H_i by $H_i - B_{\Phi_1(i)} - B_{\Phi_2(i)}$ and replacing B_i by $B_i - H_i - H_{\Phi_3(i)}$. The resulting layout is a contact representation Γ'' of G where each vertex v_i of G is represented by the T-shaped polygon $H_i \cup B_i$.

3.3 Removing unused area

In this step, we begin with the λ-fat T-shaped polygonal layout, Γ'', from above and assign each (rectangular) hole to a polygon adjacent to it. We start by placing an axis-aligned rectangle of minimum size that encloses Γ''. This creates five new bounded holes. Note that all these holes are rectangles, and each of them

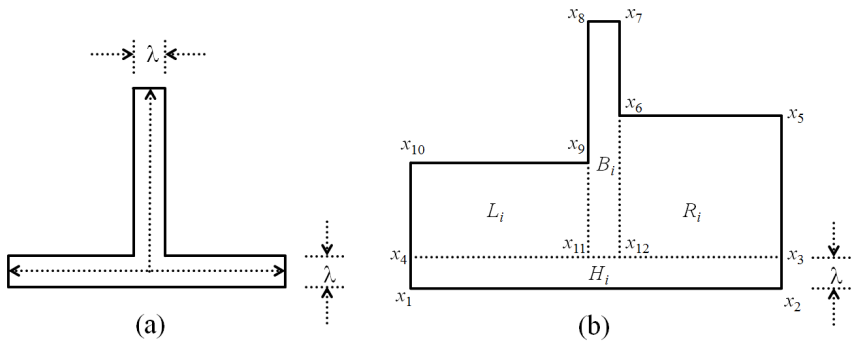


Figure 2: (a) λ-fattening of T, and (b) subdividing a T-shaped polygon into four rectangles.

is bounded at the bottom by H_i for some vertex v_i. We assign each hole to this vertex. This assigns at most two holes to each vertex v_i: one hole L_i to the left of B_i, and one hole R_i to the right of B_i. Now for each vertex v_i, define $P_i = T_i \cup L_i \cup R_i$. It is easy to see that P_i is an 8-sided rectilinear polygon since the left side of L_i has the same x-coordinate as the left side of H_i and the right side of R_i has the same x-coordinate as the right side of H_i. Thus we have a rectilinear dual, Γ, of G where each vertex v_i is represented by P_i.

We note here that the coordinates of P_i could be computed directly, without going through T-shapes and λ-fattening, using the values $\Phi_k(v_i)$ for $k = 1, 2, 3$ and a topological order π of $\mathcal{S}_1^{-1} \cup \mathcal{S}_2$. In particular, we can compute the representation in linear time, and all coordinates are integers of size $O(n)$.

3.4 Area-Universality

A rectilinear dual Γ is *area-universal* if any assignment of areas to its polygons can be realized by a combinatorially equivalent layout. Eppstein *et al.* [11] studied this concept for the case when all the polygons are rectangles and the outer-face boundary is also a rectangle (which they call a *rectangular layout*). They gave a characterization of area-universal rectangular layouts using the concept of "maximal line-segment". A *line-segment* of a layout is the union of inner edges of the layout forming a consecutive part of a straight-line. A line-segment that is not contained in any other line-segment is maximal. A maximal line-segment s is called *one-sided* if it is part of the side of at least one rectangular face, or in other words, if the perpendicular line segments that attach to its interior are all on one side of s.

Lemma 3.1 *[11] A rectangular layout is area-universal if and only if each maximal segment in the layout is one-sided.*

No such characterization is known when some faces are not rectangles. Still we can use the characterization from Lemma 3.1 to show that the rectilinear dual obtained by the algorithm from the previous section is area-universal, with the following Lemma.

Lemma 3.2 *Let Γ be the rectilinear dual obtained by the above algorithm. Then Γ is area-universal.*

Proof: To show the area-universality of Γ, we divide all the polygons in Γ into a set of rectangles such that the resulting rectangular layout is area-universal. Specifically, we divide each polygon P_i into four rectangles H_i B_i, L_i and R_i (as defined in the previous subsection) by adding three auxiliary segments: one horizontal and two vertical, as illustrated in Figure 2(b). Any horizontal segment s not on the bounding box belongs to some H_i (either top or bottom), and expanding it to its maximum it ends at $B_{\Phi_1(v_i)}$ on the left and $B_{\Phi_2(i)}$ on the right. So s is one-sided since it is a side of the rectangle H_i. Any vertical segment s not on the bounding box belongs to some B_i (either left or right), and expanding it to its maximum it ends at H_i on the bottom and $H_{\Phi_3(i)}$ on the top. So s is one-sided since it is a side of the rectangle B_i.

Now given any assignment of areas $w : V \to \mathbb{R}^+$ to the vertices V of G, we split $w(v_i)$ arbitrarily into four parts and assign the four values to its four associated rectangles. Since Γ^* is area-universal, there exists a rectilinear dual of G that is combinatorially equivalent to Γ for which these areas are realized. Figure 1(f) illustrates the rectangular layout obtained from the rectilinear dual in Figure 1(e). □

So for any area-assignment, the rectilinear dual that we found can be turned into a combinatorially equivalent one that respects the area requirements. This proves our main result for maximal planar graphs. Omitting v_1 and v_2 from the drawing still results in a cartogram where the union of all polygons is a rectangle, so the result also holds for all planar graphs that are inner triangulated and have a simple outer-face.

Recall that the lower bound on the complexity of polygons in any rectilinear dual (and hence in any cartogram) is 8, as proven by Yeap and Sarrafzadeh [34]. The algorithm described in this section, along with this lower bound leads to our main theorem.

Theorem 3.1 *Eight-sided polygons are always sufficient and sometimes necessary for a cartogram of an inner triangulated planar graph with a simple outer-face.*

3.5 Feature Size and Supporting Line Set

In addition to optimal polygonal complexity, we point out here a practical feature of the 8-sided area-universal rectilinear layout constructed with our algorithm. Earlier constructions, e.g., [3, 8], often rely on "thin connectors" to maintain adjacencies, whereas our construction does not. Moreover, we have the freedom to choose how to divide the area assigned to any vertex v_i among the four rectangles associated with it. This flexibility makes it possible to achieve other desired properties. In particular, we can show that the minimum feature size can be made as large as $1/2\sqrt{A} \cdot \min_{v \in V(G)} w(v)$ (where A is the sum of the weights), and that this is worst-case optimal. We can also reduce the number of supporting lines in the cartogram from the $3n$, in the construction above, to only $2n$. Details are in the appendix.

3.6 Computing the Cartogram

The proof of Lemma 3.1 implies an algorithm for computing the final cartogram. Splitting the T-shaped polygons into four rectangles and distributing the weights on these rectangles yields an area-universal rectangular dual. This combinatorial structure has to be turned into an actual cartogram, i.e., into a layout respecting the given weights. Wimer *et al.* [33] gave a formulation of the problem which combines flows and quadratic equations. Eppstein *et al.* [11] indicated that a solution can be found with a numerical iteration. Alternate methods also exist, based on non-linear programming [26], geometric programming [23], and convex programming [6]. Heuristic hill-climbing schemes converge much quicker and can be used in practice, at the expense of small errors [5, 16, 32].

We implemented the entire algorithm, along with a force-directed heuristic to compute the final cartogram. We treat each region as a rectilinear "room" containing an amount of "air" equal to the weight assigned to the corresponding vertex. We then simulate the natural phenomenon of air pressure applied to the "walls", which correspond to the line segment borders in our layout. At each iteration, we consider the segment that feels the maximum pressure and let it move in the appropriate direction. This process is guaranteed to converge with the desired cartogram, but in practice we stop when the (cartographic) errors fall below a certain threshold. Illustrations and experiments are in the appendix.

4 Cartograms for Hamiltonian Graphs

In this section we show that 8-sided polygons are always sufficient and sometimes necessary for a cartogram of a *Hamiltonian* maximal planar graph. We first give a direct linear-time construction with 8-sided regions without relying on numerical iteration or heuristics, as discussed in the previous section. We then prove that this is optimal by showing that 8 sides are necessary, with a non-trivial lower bound example.

4.1 Sufficiency of 8-sided Polygons

Let $v_1, \ldots, v_n$ be a Hamiltonian cycle of a maximal planar graph G. Consider a plane embedding of G with the edge (v_1, v_n) on the triangular outer-face. The Hamiltonian cycle splits the plane graph G into two outer-planar graphs which we call the *left graph* G_l and *right graph* G_r. Edges on the Hamiltonian cycle belong to both graphs. The naming is with respect to a planar drawing Γ of G in which the vertices

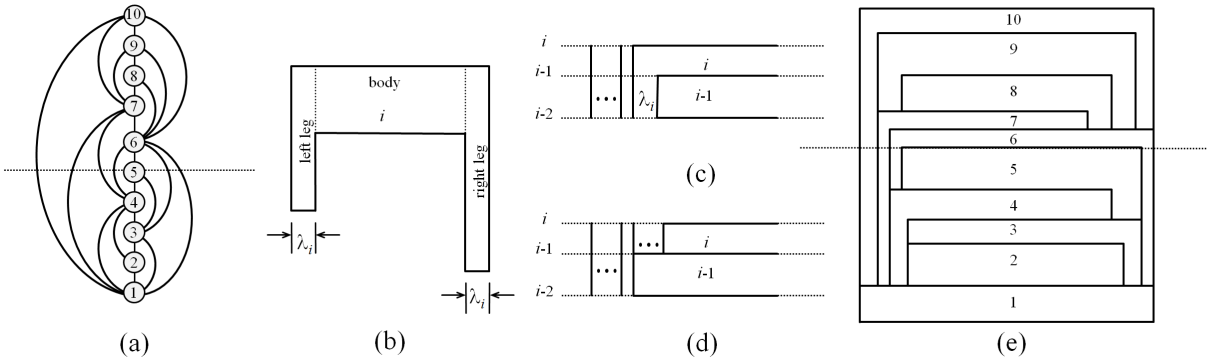


Figure 3: (a) A Hamiltonian maximal planar graph G, (b) an 8-sided polygon for vertex i, (c)–(d) illustration for the algorithm to construct a cartogram of G, (e) a cartogram of G with 8-sided polygons.

$v_1, \ldots, v_n$ are placed in increasing order along a vertical line, and the edges are drawn with y-monotone curves with leftmost edge (v_1, v_n); see Figure 3(a).

Lemma 4.1 *Let $G = (V, E)$ be a Hamiltonian maximal planar graph and let $w : V \to \mathbb{R}^+$ be a weight function. Then a cartogram with 8-sided polygons can be computed in linear time.*

Proof: Let $v_1, \ldots, v_n$ be a Hamiltonian cycle and Γ be the drawing defined above with (v_1, v_n) on the outer-face. Suppose R is a rectangle of width W and height H where $W \times H = A = \sum_{v \in V} w(v)$. Each vertex v_i will be represented as the union of three rectangles, the *left leg*, the *body* B_i, and *right leg* of v_i. We set the width of the legs to $\lambda_i = w(v_i)/(2H + W)$; see Figure 3(b).

Our algorithms places vertices $v_1, \ldots, v_n$ in this order, and also reserves vertical strips for legs of all vertices that have earlier neighbors. More precisely, let $\mathcal{L}_j$ be all vertices v_k with an edge (v_i, v_k) in G_l for which $i \leq j < k$. Similarly define $\mathcal{R}_j$ with respect to edges in G_r. In the drawing Γ, $\mathcal{L}_j$ are those vertices above v_j for which the horizontal ray left from v_j crosses an incident edge.

We place vertices $v_1, \ldots, v_j$ with the following invariant: The horizontal line through the top of B_j intersects, from left to right: (a) a vertical strip of width λ_k for each $v_k \in \mathcal{L}_j$, in descending order, (b) a non-empty part of the top of B_j, and (c) a vertical strip of width λ_k for each $v_k \in \mathcal{R}_j$, in ascending order.

We start by placing B_1 as a rectangle that spans the bottom of R. At the left and right end of the top of B_1, we reserve vertical strips of width λ_k for each vertex in $\mathcal{L}_1$ and $\mathcal{R}_1$, respectively.

To place B_i, $i > 1$, first locate the vertical strips reserved for v_i in previous steps (since $v_i \in \mathcal{L}_{i-1}$ and $i \in \mathcal{R}_{i-1}$, there always are such strips, though they may have started only at the top of B_{i-1}). Since vertical strips are in descending/ascending order, the strips for v_i are the innermost ones. Let B_i be a rectangle just above B_{i-1} connecting these strips. Choose the height of B_i so large that it, together with the left and right leg inside the strips, has area $w(v_i)$; we will discuss soon why this height is positive.

Finally, at the top left of the polygon of v_j we reserve a new vertical strip of width λ_k for each vertex k that is in $\mathcal{L}_i - \mathcal{L}_{i-1}$. Similarly reserve strips for vertices in $\mathcal{R}_i - \mathcal{R}_{i-1}$. Using planarity, it is easy to see that vertices in $\mathcal{L}_i - \mathcal{L}_{i-1}$ must have smaller indices than vertices in $\mathcal{L}_{i-1}$, and so this can be done such that the order required for the invariant is respected.

Clearly this algorithm takes linear time and constructs 8-gons of the correct area. To see that it creates contacts for all edges, consider an edge (v_i, v_k) with $i < k$ in G_l (edges in G_r are similar.) By definition $k \in \mathcal{L}_i$. If $v_k \in \mathcal{L}_i - \mathcal{L}_{i-1}$, then we reserved a vertical strip for v_k when placing v_i. This vertical strip is used for the left leg of v_k, which hence touches v_i. Otherwise ($v_k \notin \mathcal{L}_i - \mathcal{L}_{i-1}$) we have $v_k \in \mathcal{L}_{i-1}$. At the time that v_{i-1} was placed, there hence existed a vertical strip for v_k. There also was a vertical strip for $v_i \in \mathcal{L}_{i-1}$. These two strips must be adjacent, because by planarity (and edge (v_i, v_k)) there can be no vertex v_j with $i < j < k$ in $\mathcal{L}_{i-1}$. So these strips create a contact between the two left legs of v_i and v_k.

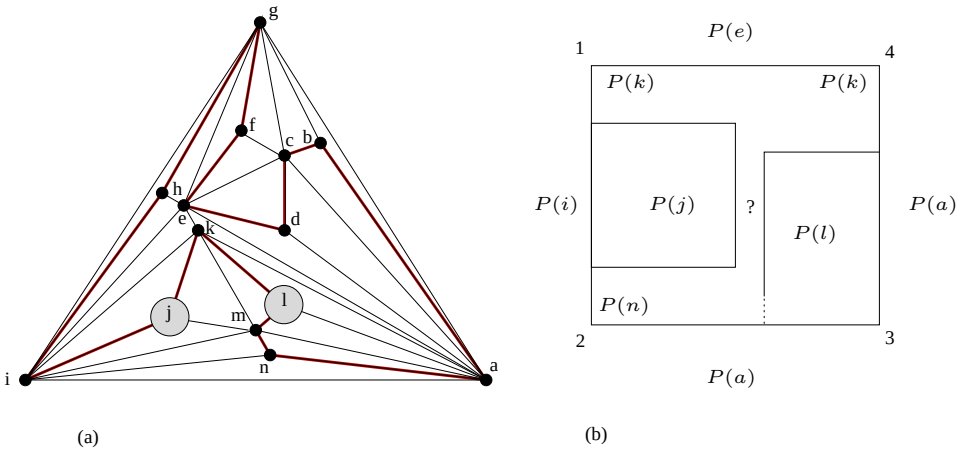


Figure 4: (a) A maximal planar Hamiltonian graph with a weight function that requires at least one 8-sided polygon in any cartogram. (b) Illustration for the proof of Lemma 4.2.

We now discuss the choice of $\lambda_i = w(v_i)/(2H + W)$. Each leg of v_i has height $\leq H$ and width λ_i, hence area $\leq H\lambda_i$. Then the body B_i has area $\geq w(v_i) - 2H\lambda_i$ and width $\leq W$, hence height $\geq (w(v_i) - 2H\lambda_i)/W = \lambda_i$. It follows that B_i has positive height. Also all vertical strips fit: after placing vertex v_i, we have a strip of width λ_k for each vertex v_k in $\mathcal{L}_i$ and $\mathcal{R}_i$, and these strips use width

$$\sum_{v_k \in \mathcal{L}_i} \frac{w(v_k)}{2H+W} + \sum_{v_k \in \mathcal{R}_i} \frac{w(v_k)}{2H+W} \leq \frac{2\sum_{v_k \in V - \{v_i\}} w(v_k)}{2H+W} \leq \frac{2(A - w(v_i))}{2H+W} \leq \frac{2A}{2H} - \frac{2w(v_i)}{2H+W} = W - 2\lambda_i.$$

Hence B_i has width $\geq 2\lambda_i$ and the polygon of v_i has minimum feature size λ_i. □

This algorithm also guarantees a minimum feature size for the cartogram: $\min_{v_i \in V} \lambda_i = \frac{w_{min}}{2H+W}$, where $w_{min} = \min_{v \in V} w(v_i)$. Choosing $W = \sqrt{2A}$ and $H = \sqrt{A/2}$, yields minimum feature size $\frac{w_{min}}{2\sqrt{2}\sqrt{A}}$.

4.2 Necessity of 8-sided Polygons

While it was known that 8-sided rectilinear polygons are necessary for general planar graphs [25], here we show that 8-sided rectilinear polygons are necessary even for Hamiltonian maximal planar graphs.

Lemma 4.2 *Consider the Hamiltonian maximal planar graph $G = (V, E)$ in Figure 4(a). Define $w(j) = w(l) = D$ and $w(v) = \delta$ for $v \in V \setminus \{j, l\}$, where $D \gg \delta$. Then any cartogram of G with weight function w requires at least one 8-sided polygon.*

Proof: Assume for a contradiction that G admits a cartogram Γ with respect to w such that the polygons $\{P(v)\}$ used in Γ have complexity at most 6. Observe that if $\{u, v, x\}$ is some separating triangle in G, i.e., three mutually adjacent vertices whose removal disconnect the graph, then the region R_{uvw} used for the inside of the separating triangle contains at least one reflex corner of the polygon $P(u)$, $P(v)$, or $P(x)$. The 5-vertex set $\{a, c, e, g, i\}$ in G is the union of the five separating triangles $\{a, c, g\}$, $\{a, c, e\}$, $\{c, e, g\}$, $\{a, e, i\}$, and $\{e, g, i\}$ with disjoint interiors. Since all the polygons in Γ are either 4-sided or 6-sided, the union of the polygons for these five vertices has at most five reflex corners and hence each of the five separating triangles above contains the only reflex corner of the polygon for a, c, e, g, or i. In particular, the outer boundary of R_{aei} contains exactly one reflex corner from one of $P(a)$, $P(e)$ and $P(i)$, and hence it is a rectangle, say 1234. By symmetry, we may assume that the reflex corner of $P(i)$ is *not* used for R_{aei}.

The 4-vertex set $\{a, i, k, m\}$ is the disjoint union of three separating triangles $\{a, k, m\}$, $\{k, i, m\}$, $\{i, a, m\}$ whose interiors are vertices l, j and n, respectively. Since the reflex corner of $P(i)$ is not used for R_{aei}, it also cannot be used for any of these separating triangles. Hence each of $P(j)$, $P(l)$ and $P(n)$ contains exactly one reflex corner from $P(a)$, $P(k)$ and $P(m)$. In particular, rectangle 1234 (which is R_{aei})

must contain the reflex corner of $P(a)$. We also can conclude that $P(j)$, $P(l)$ and $P(n)$ are all rectangles, since there are no additional reflex corners available to accommodate additional convex corners from $P(j)$, $P(l)$ and $P(n)$.

Assume the naming in Figure 4(b) is such that edge 12 belongs to $P(i)$, edges 23 and 34 belong to $P(a)$ and edge 41 belongs to $P(e)$. By the adjacencies, $P(k)$ must occupy corners 1 and 4 and $P(n)$ must occupy corner 2, while corner 3 (which is the reflex corner of $P(a)$) could belong to n or l.

Now consider the rectangles $P(j)$ and $P(l)$. If D is sufficiently big, then these two rectangles each occupy almost half of rectangle 1234. Therefore, either their x-range or their y-range must overlap in their interior. Assume their y-range overlaps, the other case is similar. Which polygon should occupy the area that is between $P(j)$ and $P(l)$ horizontally? It cannot be k, because $P(k)$ contains corners 1 and 2 and hence would obtain 2 reflex angles from $P(j)$ and $P(l)$. So it must be $P(m)$, since n is not adjacent to j and l. But $P(m)$ must also separate $P(n)$ from both $P(j)$ and $P(l)$. Regardless of whether n or l occupies corner 3, this is not possible without two reflex vertices for m. Therefore either the areas are not respected or some polygon must have 8 sides. □

Lemma 4.1 together with Lemma 4.2 yield the following theorem.

Theorem 4.1 *Eight-sided polygons are always sufficient and sometimes necessary for a cartogram of a Hamiltonian maximal planar graph.*

5 Cartograms with 6-sided Polygons

Here we study cartograms with rectilinear 6-sided polygons. We first note that these are easily constructed for outer-planar graphs. Then we generalize this technique to other maximal planar Hamiltonian graphs.

5.1 Maximal Outer-planar Graphs

Our algorithm from Lemma 4.1 naturally gives drawings of maximal outer-planar graphs that use 6-sided polygons. Another linear-time algorithm for constructing a cartogram of a maximal outer-planar graph with 6-sided rectilinear polygons is also described in [1], however, our construction based on Lemma 4.1 is much simpler. Any maximal outer-planar graph G can be made into a maximal Hamiltonian graph by duplicating G and gluing the copies together at the outer-face such that $G_l = G = G_r$. (This graph has double edges, but the algorithm in Lemma 4.1 can handle double edges as long as one copy is in the left and one in the right graph.) Create the drawing based on Lemma 4.1 with all vertices having double the weight, and cut it in half with a vertical line. This gives a drawing of G with 6-sided rectilinear polygons as desired.

5.2 One-Legged Hamiltonian Cycles

We now aim to find more maximal Hamiltonian graphs which have cartograms with 6-sided polygons. In a Hamiltonian cycle $v_1, \ldots, v_n$, call vertex v_j *two-legged* if it has a neighbor v_i^l in G_l with $i^l < j - 1$ and also a neighbor v_i^r in G_r with $i_r < j - 1$. Call a Hamiltonian cycle *one-legged* if none of its vertices is two-legged. In the construction from Lemma 4.1, the polygon of v_j obtains a reflex vertex on both sides only if it has a neighbor below v_{j-1} on both sides, or in other words, if it is two-legged. Hence we have:

Lemma 5.1 *Let $G = (V, E)$ be a maximal planar graph with a one-legged Hamiltonian cycle and let $w : V \to R^+$ be a weight function. Then a cartogram with 6-sided polygons can be computed in linear time.*

It is a natural question to characterize graphs that have such Hamiltonian cycles. Given a Hamiltonian cycle $v_1, \ldots, v_n$ we fix a plane embedding of G with outer triangle $\{v_1, v_k, v_n\}$. The following lemma has an straight-forward proof (given in the appendix) using the concepts involved.

Lemma 5.2 *Let $v_1, \ldots, v_n$ be a Hamiltonian cycle in a maximal plane graph G with (v_1, v_n) on the outer triangle. Define $w_i := v_{n-i+1}$. Then the following conditions are equivalent:*

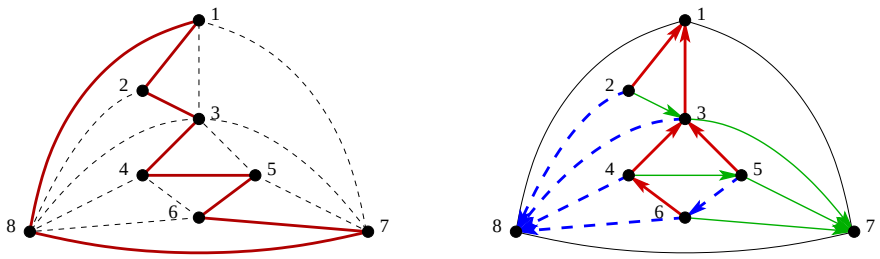


Figure 5: A graph with a one-legged Hamiltonian cycle and the corresponding Schnyder realizer.

(a) The Hamiltonian cycle is one-legged.

(b) For $i = 2, \ldots, n$, edge (v_{i-1}, v_i) is an outer edge of the graph G_i induced by by $v_1, v_2, \ldots, v_i$ (with the induced embedding.)

(c) v_{n-1} is an outer vertex and vertex v_i has at least two neighbors with a larger index for $i = 1, \ldots, n-2$.

(d) $w_1, \ldots, w_n$ is a canonical ordering for G.

(e) G admits a Schnyder realizer $(\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ in which w_1, w_2 and w_n are the roots of $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_3$, respectively and every inner vertex is a leaf in $\mathcal{S}_1$ or $\mathcal{S}_2$.

Figure 5 shows an example of a maximal plane graph with a one-legged Hamiltonian cycle, the corresponding canonical ordering, and the Schnyder realizer.

Once we have a one-legged Hamiltonian cycle, we can build a 6-sided cartogram via Lemma 5.1 in linear time. Alternately we could obtain from it a Schnyder wood, rooted such that every vertex is a leaf in $\mathcal{S}_1$ or $\mathcal{S}_2$, and hence obtain a 6-sided cartogram via the algorithm in Section 3. However, we prefer the construction of Lemma 5.1 due to its linear runtime.

We know that not every Hamiltonian maximally planar graph admits a one-legged Hamiltonian cycle; for example, the graph in Figure 4 does not even admit a cartogram with 6-gons. However, we believe that some non-trivial subclasses of Hamiltonian maximally planar graphs are also one-legged Hamiltonian. In particular, we have the following conjecture:

Conjecture 5.1 *Every 4-connected maximal planar graph has a one-legged Hamiltonian cycle.*

Note that by Lemma 5.2, the conjecture is equivalent to asking whether every 4-connected maximal graph has a Hamiltonian cycle such that taking the vertices in this order gives a canonical ordering. Such a result might be of use for other graph problems as well.

6 Conclusion and Open Problems

We presented a cartogram construction for maximal planar graphs with optimal polygonal complexity. For the precise realization of the actual cartogram this approach requires numerical iteration. Even though the simple heuristic works well in practice, a natural open problem is whether everything can be computed with an entirely combinatorial linear-time approach.

We also presented such an entirely combinatorial linear-time construction for Hamiltonian maximal planar graphs and showed that the resulting 8-sided cartograms are optimal. Finally, we showed that if the graph admits a one-legged Hamiltonian cycle (for example outer-planar graphs), only 6 sides are needed. It remains to identify larger classes of planar graphs which are one-legged Hamiltonian and thus have 6-sided cartograms. We conjecture that 4-connected maximal planar graphs have this property.

All of the constructions in this paper yield area-universal rectilinear duals with optimal polygonal complexity. While Eppstein *et al.* [11] characterized area-universal *rectangular* layouts, a similar characterization remains an open problem for general area-universal *rectilinear* layouts.

References

[1] M. J. Alam, T. Biedl, S. Felsner, A. Gerasch, M. Kaufmann, and S. G. Kobourov. Linear-time algorithms for proportional contact graph representations. Technical Report CS-2011-19, University of Waterloo, 2011. To appear at ISAAC 2011.

[2] M. J. Alam, T. Biedl, S. Felsner, M. Kaufmann, and S. G. Kobourov. Proportional contact representations of planar graphs. In *Proceedings of the 19th International Symposium on Graph Drawing (GD 2011)*, 2011.

[3] T. Biedl and L. E. Ruiz Velázquez. Orthogonal cartograms with few corners per face. In *Proc. Data Structures and Algorithms Symposium (WADS'11)*, volume 6844 of *Lecture Notes in Computer Science*, pages 98–109. Springer, 2011.

[4] A. L. Buchsbaum, E. R. Gansner, C. M. Procopiuc, and S. Venkatasubramanian. Rectangular layouts and contact graphs. *ACM Transactions on Algorithms*, 4(1), 2008.

[5] I. Cederbaum. Analogy between vlsi floorplanning problems and realisation of a resistive network. *IEE Proceedings, Part G, Circuits, Devices and Systems*, 139(1):99–103, 1992.

[6] T. Chen and M. K. H. Fan. On convex formulation of the floorplan area minimization problem. In *International Symposium on Physical Design*, pages 124–128, 1998.

[7] M. Chrobak and T. Payne. A linear-time algorithm for drawing planar graphs. *Inform. Process. Lett.*, 54:241–246, 1995.

[8] M. de Berg, E. Mumford, and B. Speckmann. On rectilinear duals for vertex-weighted plane graphs. *Discrete Mathematics*, 309(7):1794–1812, 2009.

[9] H. de Fraysseix, P. O. de Mendez, and P. Rosenstiehl. On triangle contact graphs. *Combinatorics, Probability and Computing*, 3:233–246, 1994.

[10] H. de Fraysseix, J. Pach, and R. Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990.

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

[12] E. R. Gansner, Y. Hu, M. Kaufmann, and S. G. Kobourov. Optimal polygonal representation of planar graphs. In *9th Latin Am. Symp. on Th. Informatics (LATIN)*, pages 417–432, 2010.

[13] X. He. On floor-plan of plane graphs. *SIAM Journal of Computing*, 28(6):2150–2167, 1999.

[14] R. Heilmann, D. A. Keim, C. Panse, and M. Sips. Recmap: Rectangular map approximations. In *10th IEEE Symp. on Information Visualization (InfoVis 2004)*, pages 33–40, 2004.

[15] D. H. House and C. J. Kocmoud. Continuous cartogram construction. In *Proceedings of the conference on Visualization (VIS '98)*, pages 197–204, 1998.

[16] T. Izumi, A. Takahashi, and Y. Kajitani. Air-pressure model and fast algorithms for zero-wasted-area layout of general floorplan. *IEICE Transaction on Fundamentals of Electronics, Communications and Computer Sciences, Special Section on Discrete Mathematics and Its Applications*, E81–A(5):857–865, 1998.

[17] A. Kawaguchi and H. Nagamochi. Orthogonal drawings for plane graphs with specified face areas. In *4th Conf. on Theory and Applications of Models of Comp.*, pages 584–594, 2007.

[18] P. Koebe. Kontaktprobleme der konformen Abbildung. *Berichte über die Verhandlungen der Sächsischen Akademie der Wissenschaften zu Leipzig. Math.-Phys. Klasse*, 88:141–164, 1936.

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

[20] S. M. Leinwand and Y.-T. Lai. An algorithm for building rectangular floor-plans. In *21st Design Automation Conference*, pages 663–664. IEEE Press, 1984.

[21] C.-C. Liao, H.-I. Lu, and H.-C. Yen. Compact floor-planning via orderly spanning trees. *Journal of Algorithms*, 48:441–451, 2003.

[22] J. Michalek, R. Choudhary, and P. Papalambros. Architectural layout design optimization. *Engineering Optimization*, 34(5):461–484, 2002.

[23] T.-S. Moh, T.-S. Chang, and S. L. Hakimi. Globally optimal floorplanning for a layout problem. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 43(9):713–720, 1996.

[24] E. Raisz. The rectangular statistical cartogram. *Geographical Review*, 24(3):292–296, 1934.

[25] I. Rinsma. Nonexistence of a certain rectangular floorplan with specified area and adjacency. *Environment and Planning B: Planning and Design*, 14:163–166, 1987.

[26] E. Rosenberg. Optimal module sizing in vlsi floorplanning by nonlinear programming. *Methods and Models of Operations Research*, 33:131–143, 1989.

[27] W. Schnyder. Embedding planar graphs on the grid. In *Proceedings of the 1st ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 138–148, 1990.

[28] W. Tobler. Thirty five years of computer cartograms. *Annals, Assoc. American Geographers*, 94:58–73, 2004.

[29] T. Ueckerdt. *Geometric Representations of Graphs with low Polygonal Complexity*. PhD thesis, Technische Universität Berlin, 2011.

[30] P. Ungar. On diagrams representing graphs. *J. London Math. Soc.*, 28:336–342, 1953.

[31] M. J. van Kreveld and B. Speckmann. On rectangular cartograms. *Computational Geometry*, 37(3):175–187, 2007.

[32] K. Wang and W.-K. Chen. Floorplan area optimization using network analogous approach. In *IEEE International Symposium on Circuits and Systems*, volume 1, pages 167 –170, 1995.

[33] S. Wimer, I. Koren, and I. Cederbaum. Floorplans, planar graphs, and layouts. *IEEE Transactions on Circuits and Systems*, 35(3):267 –278, 1988.

[34] K.-H. Yeap and M. Sarrafzadeh. Floor-planning by graph dualization: 2-concave rectilinear modules. *SIAM Journal on Computing*, 22:500–526, 1993.

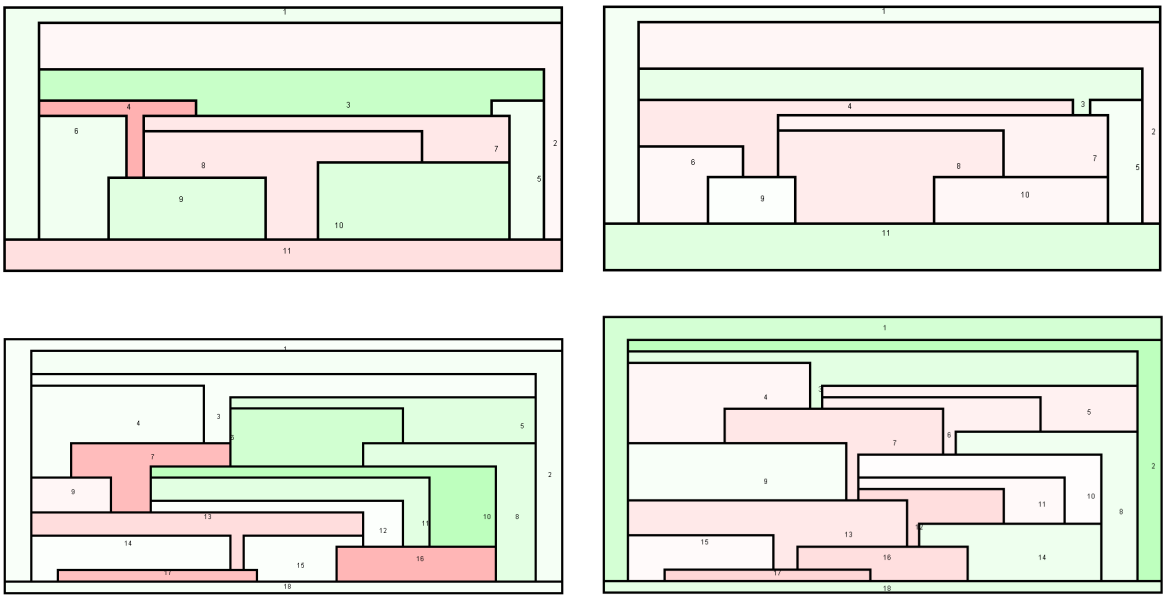


Figure 6: Input-output pairs: weights are assigned at random in the range $[10, 100]$, and the cartographic error in the output file is less than 1%. The colors indicate air-pressure: the greener a region is, the more it needs to shrink; the redder a regions is, the more it needs to grow.

Appendix

A.1 Feature Size and Supporting Line Sets

Here we consider the choice of weights for the 8-sided construction of Section 3 with the goal of maximizing feature size and minimizing the number of supporting lines.

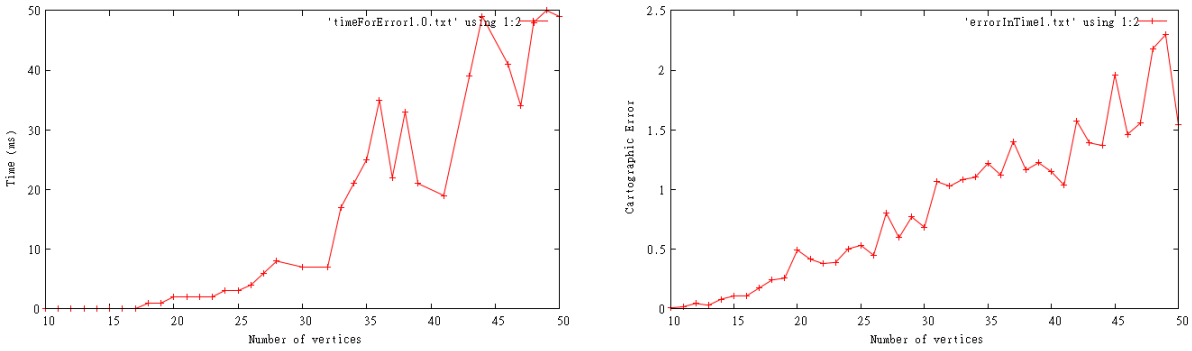
Specifically, let G be a maximal planar graph with a prescribed weight function $w : V(G) \to \mathbb{R}^+$. Choose W and H such that $W \times H = A = \sum_{v \in V(G)} w(v)$. We are interested in cartograms within a rectangle of width W and the height H. Define $w_{min} = min_{v \in V(G)} w(v)$.

Recall that each vertex v_i is represented by the union of at most four rectangles $H_i \cup B_i \cup R_i \cup L_i$, with H_i and B_i non-empty. We can distribute the weight of v_i arbitrarily among them. In particular, we can assign zero areas to the rectangles L_i and R_i and split its weight into two equal parts to H_i and B_i. In this layout each original vertex is represented by rectangles H_i and B_i whose union is some fattened T or L, and all the necessary contacts remain. Hence we can use this simplified layout to produce the cartogram.

The distribution of the weight of v_i in equal parts to H_i and B_i allows to bound the feature size. The height and width of each rectangle are bounded by H and W respectively. Its weight is at least $w_{min}/2$. Therefore, the height and width of each rectangle is at least $\frac{w_{min}}{2 \max\{W,H\}}$. Thus the minimum feature size of the cartogram is at least $\frac{w_{min}}{2 \max\{W,H\}}$. This is worst-case optimal, as the polygon with the smallest weight might need to reach from left to right and top to bottom in the representation. We may choose $W = H = \sqrt{A}$. Then the minimum feature size is $\frac{w_{min}}{2\sqrt{A}}$. Furthermore the rectangular layout based on the rectangles H_i and B_i alone yields a cartogram with at most $2n$ supporting lines, instead of the $3n$ supporting lines in the cartogram based on four rectangles per vertex.

A.2 Implementation and Experimental Results

The proof of Lemma 3.1 implies an algorithm for the actual computation of the cartogram. Splitting the T-shaped polygons into four rectangles and distributing the weights on these rectangles yields an area-universal rectangular dual. What remains is to manipulate the combinatorial structure of so as to achieve the desired areas.



(a) (b)

Figure 7: Experimental results for graphs with 10-50 vertices. Each sample point corresponds to 25 graphs. (a) Plotting the average time it takes to reach cartographic error of 1%. (b) Plotting the average cartographic error achieved in 1millisecond.

We treat each region as a rectilinear "room" containing an amount of "air" equal to the weight assigned to the corresponding vertex. We then simulate the natural phenomenon of air pressure applied to the "walls", which correspond to the line segment borders in our layout.

For each vertex v_i of G, the polygon P_i contains air with volume $w(v_i)$. If the area of P_i is A_i, then the pressure applied to each of the walls surrounding P_i is given by $\mathcal{P}(v_i) = \frac{w(v_i)}{A_i}$. In Section 3.4, we saw that the maximal segments of the layout are the two horizontal and the two vertical segment associated with each polygon. For each polygon, the horizontal segment other than the base is entirely inside the polygon, hence it feels no "pressure" on it. For each of the other three segments s for the polygon P_i, the "inward force" it feels is given by $\mathcal{F}(s) = \sum_{v_j \in V(s)} [\mathcal{P}(v_j) l_j] - \mathcal{P}(v_i) l_i$. Here $V(s)$ is the set of vertices other that v_i whose corresponding polygon touches the segment s and l_i (resp. l_j) denotes the length of s that is shared with P_i (resp. P_j). At each iteration, we consider the segment that feels the maximum pressure and let it move in the appropriate direction. The convergence of this scheme follows from [16]. Some sample input-output pairs are shown in Fig. 6; more examples and movies showing the gradual transformation can be found at `www.cs.arizona.edu/~mjalam/optocart`.

We ran a few simple experiments to test the heuristic for time and accuracy. In the first experiment we generated 5 graphs on n vertices with n in the range $[10 - 50]$ and assigned 5 random weight distributions with weights in the range $[10 - 100]$. Next we ran the heuristic until the cartographic error dropped below 1% and recorded the average time. All the averages were below 50 milliseconds, which confirms that good solutions can be found very quickly in practice; see Fig 7(a). In the second experiment we fixed the time allowed and tested the quality of the cartograms obtained within the time limit. Specifically, we generated 5 graphs on n vertices with n in the range $[10 - 50]$ and assigned 5 random weight distributions with weights in the range $[10 - 100]$. We allowed the program to run for 1 millisecond and recorded the average "cartographic error". Even with such a small time limit, the average cartographic error was under 2.5%; see Fig. 7(b). Here, the *cartographic error* for a cartogram of a planar graph $G = (V, E)$ is defined as in [31]: $max_{v \in V}(|A(v) - w(v)|/w(v))$, where $w(v)$ denotes the weight assigned to v and $A(v)$ denotes the area of the polygon representing v. All of the experiments were run on an Intel Core i3 machine with a 2.2GHz processor and 4GB RAM.

A.3 Omitted Proofs

Proof of Lemma 5.2. (a) $\Longleftrightarrow$ (b): For $i = 2, \ldots, n$ we argue that (a) vertex v_i is not two-legged if and only if (b) holds for i. Indeed, (v_{i-1}, v_i) is an inner edge in G_i if and only if there are boundary edges (v_i, v_j) and (v_i, v_k) with $j, k < i - 1$ in G_l and G_r, respectively. But then v_i is two-legged by definition.

(b) $\Longleftrightarrow$ (c): Since v_n is an outer vertex and $G_n = G$, (b) holds for $i = n$ if and only if v_{n-1} is an outer vertex. For $i = 2, \ldots, n-1$ we argue that (b) holds for i if and only if (c) holds for $i-1$. Let v_i^l, respectively v_i^r, denote the third vertex in the inner facial triangle containing the edge (v_{i-1}, v_i) in G_l, respectively G_r. (Both triangles exist, since (v_{i-1}, v_i) is an inner edge in G.) Now (v_{i-1}, v_i) is an inner edge in G_i if and only if both, v_i^l and v_i^r, have a smaller index than v_{i-1}, which in turn holds if and only if the index of *every* neighbor of v_{i-1}, different from v_i, is smaller than $i-1$.

(c) $\Longrightarrow$ (d): By (c) $\{w_1, w_2, w_n\} = \{v_n, v_{n-1}, v_1\}$ is the outer triangle of G, and moreover, $\tilde{G}_3$, which is induced by v_n, v_{n-1}, v_{n-2}, is a triangle. Hence the outer boundary of $\tilde{G}_3$ is a simple cycle C_3 containing the edge (w_1, w_2). In other words, the first condition of a canonical ordering is met for $i = 4$. Assuming (c) and the first condition for $i = 4, \ldots, n-1$, we show that the second and first condition holds for i and $i+1$, respectively. In the end, the second condition holds for $i = n$ since w_n is an outer vertex.

First note that w_i is in the exterior face of $\tilde{G}_{i-1}$ since w_n lies in the exterior face and the path $w_i, \ldots, w_n$ is disjoint from vertices in $\tilde{G}_{i-1}$ and the embedding is planar. By (c) w_i has at least two neighbors in $\tilde{G}_{i-1}$. If the neighbors would not form a subinterval of the path $C_{i-1} \setminus (w_1, w_2)$, there would be a non-triangular inner face in $\tilde{G}_i$, which contains a vertex w_j with $j > i$ in its interior. But then the path $w_j, \ldots, w_n$, which is disjoint from $\tilde{G}_i$, would start and end in an interior and the exterior face of $\tilde{G}_i$, respectively. This again contradicts planarity. Thus the second condition of a canonical ordering is satisfied for i. Moreover $\tilde{G}_i$ is internally triangulated, has a simple outer cycle C_i containing the edge (w_1, w_2). In other words, the first condition holds for $i+1$.

(d) $\Longrightarrow$ (c): Since $w_1, \ldots, w_n$ is a canonical ordering, (w_1, w_2) is an outer edge. In particular, $w_2 = v_{n-1}$ is an outer vertex. Clearly v_1 has at least two neighbors and every neighbor has a larger index, i.e., (c) holds for $i = 1$. Moreover, by the second condition of a canonical ordering every vertex $v_i = w_{n-i+1}$, for $i = 2, \ldots, n-2$, has at least two neighbors in $\tilde{G}_{n-i} = G \setminus G_i$, which is the subgraph induced by $v_n, \ldots, v_{i+1}$.

(d) $\Longrightarrow$ (e): Consider the Schnyder realizer $(\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ of G defined by the canonical order $w_1, \ldots, w_n$ according to Lemma 2.1. For $i = 3, \ldots, n-1$ the outer cycle C_i of $\tilde{G}_i$ consists of the edge (w_1, w_2), the $w_i w_1$-path P_1 in $\mathcal{S}_1$, and the $w_i w_2$-path P_2 in $\mathcal{S}_2$. Due to the counterclockwise order of edges in a Schnyder realizer, no vertex on P_1, respectively P_2, has an incoming inner edge in $\tilde{G}_i$ in $\mathcal{S}_2$, respectively $\mathcal{S}_1$. Thus considering only edges in $\tilde{G}_i$ every outer vertex in $\tilde{G}_i$, different from w_1, w_2, is a leaf in $\mathcal{S}_1$ or $\mathcal{S}_2$. When in the canonical ordering vertex w_{i+1} is attached to $\tilde{G}_i$, some vertices on C_i become inner vertices of $\tilde{G}_{i+1}$. Every inner edge in $\tilde{G}_{i+1}$, which was not an edge in $\tilde{G}_i$ is in $\mathcal{S}_3$. Thus every inner vertex in $\tilde{G}_i$ is a leaf in either $\mathcal{S}_1$ or $\mathcal{S}_2$.

(e) $\Longrightarrow$ (d): Consider a canonical ordering $w_1, w_2, \ldots, w_n$ of G defined by the Schnyder realizer $(\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ according to Lemma 2.1. Then $\{w_1, w_2, w_3\}$ is a triangle, hence C_3 consists of the edge (w_1, w_2), the $w_3 w_1$-path P_1 in $\mathcal{S}_1$, and the $w_3 w_2$-path P_2 in $\mathcal{S}_2$. For $i = 4, \ldots, n$ the vertex w_i is attached to $\tilde{G}_{i-1}$. If w_{i+1} would have no edge to w_i then the outgoing edge of w_i in $\mathcal{S}_1$ or $\mathcal{S}_2$ is connected an inner vertex in the $w_i w_2$-path or $w_i w_1$-path, respectively. But this vertex would then have an incoming edge in both, $\mathcal{S}_1$ and $\mathcal{S}_2$ – a contradiction. □