

Fitting Planar Graphs on Planar Maps

Md. J. Alam¹, M. Kaufmann², S. G. Kobourov¹, T. Mchedlidze³

¹ Department of Computer Science, University of Arizona, USA

² Institute for Informatics, University of Tübingen, Germany

³ Institute for Theoretical Informatics, Karlsruhe Institute of Technology, Germany

Abstract. Graph and cartographic visualization have the common objective to provide intuitive understanding of some underlying data. We consider a problem that combines aspects of both by studying the problem of fitting planar graphs on planar maps. After providing an NP-hardness result for the general decision problem, we identify sufficient conditions so that a fit is possible on a map with rectangular regions. We generalize our techniques to non-convex rectilinear polygons, where we also address the problem of efficient distribution of the vertices inside the map regions.

1 Introduction

Visualizing geographic maps may require showing relational information between entities within and between the map regions. We study the problem of fitting such relational data on a given map. In particular, we consider the problem of *fitting planar graphs on planar maps*, subject to natural requirements, such as avoiding edge crossings and ensuring that edges between points in the same region remain in that region.

Fitting planar graphs on planar maps is related to cluster planarity [2, 3, 12]. In cluster-planar drawing we are given the graph along with a clustering and the goal is to find disjoint regions in the plane for the clusters for a valid plane realization of the given graph. The realization is *valid* if all the vertices in a given cluster are placed in their corresponding region, and there are no edge-crossings or edge-region crossings (i.e., edges crossing a region in the map more than once).

In our setting (fitting graphs on maps), we are given both the graph and the regions embedded in the plane, and must draw the clusters in their corresponding regions. The regions form a proper partition of the plane, such that the adjacency between two clusters is represented by a common border between their corresponding regions.

1.1 Related Work

The concept of clustering involves the notion of grouping objects based on the similarity between pairs of objects. In graph theory, this notion is captured by a clustered graph. Clustering of graphs is used in information visualization [17], VLSI design [15], knowledge representation [18], and many other areas.

Feng *et al.* defined *c-planarity* as planarity for clustered graphs [13]; also see Section 2 for related definitions. For clustered graphs in which every cluster induces a connected subgraph, *c-planarity* can be tested in quadratic time. Without the connectivity condition, the complexity of testing *c-planarity* is still an open problem. Algorithms for creating regions in the plane in which to draw *c-planar* graphs have also been studied. Eades *et al.* [10] presented an algorithm for constructing *c-planar* straight-line drawings of *c-planar* clustered graphs in which each cluster is drawn as a convex region,

while Angelini *et al.* [1] show that every such c-planar clustered graph has a c-planar straight-line drawing where each cluster is drawn inside an axis-aligned rectangle.

Many visualizations take advantage of our familiarity with maps by producing map-like representations that show relations among abstract concepts. For example, treemaps [24], squarified treemaps [5] and news maps represent hierarchical information by means of space-filling tilings, allocating area in proportion to some metric. Concept maps [8] are diagrams showing relationships among concepts. Somewhat similar are cognitive maps and mind-maps that represent words or ideas linked to and arranged around a central keyword. GMap [17] uses the geographic map metaphor to visualize relational data by combining graph layout and graph clustering, together with the creation and coloring of regions/countries.

Also related is work on contact graphs, where vertices are represented by simple interior-disjoint polygons and adjacencies are represented by a shared boundary between the corresponding polygons. For example, every maximally planar graph has a contact representation with convex polygons with at most six sides, and six sides are also necessary [9]. Of particular interest are *rectilinear duals*, where the vertices are represented by simple (axis-aligned) rectilinear polygons. It is known that 8 sides are sometimes necessary and always sufficient [16, 22, 26]. If the rectilinear polygons are restricted to rectangles, the class of planar graphs that allows such *rectangular duals* is completely characterized [21, 25] and can be obtained via bipolar orientation of the graph [14]; see Buchsbaum *et al.* [6] and Felsner [11] for excellent surveys.

1.2 Our Contributions

We first consider the question of testing whether a given planar clustered graph fits on a given planar map and show that the decision problem is NP-hard, even in the case where the map is made of only rectangular regions and each region contains only one vertex. Then we provide sufficient conditions that ensure such a fit on a rectangular map. Finally, we generalize the fitting techniques to rectilinear maps with rectangles, L-shaped and T-shaped polygons. In particular, we describe an efficient algorithm for distributing vertices appropriately in the case of maps with L-shaped polygons.

2 Preliminaries

In this section we introduce definitions used throughout the paper and then describe the properties of clustered graphs considered in the paper.

Let $G = (V, E)$ be a planar graph, with vertex set V partitioned into disjoint sets $\mathcal{V} = \{V_1, \dots, V_k\}$. We call the pair $C = (G, \mathcal{V})$ a *planar clustered graph*. We consider the following partition of the edges of G that corresponds to the given partition of vertices $\mathcal{V} = \{V_1, \dots, V_k\}$. Let E_i , for each i , $1 \leq i \leq k$ be the set of edges in E between two vertices of V_i and let E_{inter} be the set of all the remaining edges in E . Note that $E = E_1 \cup E_2 \cup \dots \cup E_k \cup E_{inter}$. We call $G_i = (V_i, E_i)$, $1 \leq i \leq k$, a *cluster* of G , the edges of E_i , $1 \leq i \leq k$, the *intra-cluster edges* and the edges of E_{inter} the *inter-cluster edges*.

The *cluster-graph* of a clustered graph $C = (G, \mathcal{V})$ is the graph $G_C = (\mathcal{V}, \mathcal{E})$, where the edge $(V_i, V_j) \in \mathcal{E}$, $1 \leq i, j \leq k$, $i \neq j$ if there exists an edge (u, w) in G so that $u \in V_i$ and $w \in V_j$. A clustered graph $C = (G, \mathcal{V})$ is said to be *connected* (resp. *biconnected*) if each of G_i , $1 \leq i \leq k$, is a connected (resp. *biconnected*) graph.

A drawing of a planar clustered graph $C = (G, \mathcal{V})$ is a planar straight-line drawing of G where each cluster G_i is represented by a simply-connected closed region R_i such that R_i contains only the vertices of G_i and the drawing of each edge e between two vertices of G_i is completely contained in R_i . An edge e and a region R have an *edge-region crossing* if the drawing of e crosses the boundary of R more than once. A drawing of a planar clustered graph C is *c-planar* if there is no edge crossing or edge-region crossing. If C has a c-planar drawing then we say that it is *c-planar*.

A *polygonal map* M is a set of interior-disjoint polygons on a plane. A *dual graph* G_M of M is a graph that contains one vertex for each polygon of M . Two vertices of G_M are connected by an edge if the corresponding polygons have a non-trivial common boundary. Given a planar graph G_M , a polygonal map M is called a *contact map* of G_M if G_M represents the dual graph of M . Let $C = (G, \mathcal{V})$ be a planar clustered graph. A polygonal map M which represents a contact map of the cluster-graph G_C is said to be *compatible* with C . Notice that this definition yields a correspondence between the clusters of C and polygons of M . In this paper we are interested in determining whether each cluster G_i of C can be drawn with straight-line edges inside its corresponding polygon in M , so that there is no edge crossing and no edge-region crossing. In case such a drawing exists we say that planar clustered graph C has a *straight-line planar fitting*, or just *planar fitting* on map M .

It is natural to consider all planar graphs, regardless of the clustering they come with. We preview the construction of a straight-line planar fitting and isolate the problem we are interested in. Recall that, by the definition of a planar fitting, each cluster has to be drawn inside a polygon, and there should be no edge crossings and no edge-region crossings. This implies that a clustered graph that has a planar fitting is also c-planar, so we consider only c-planar graphs. Unfortunately, the characterization of c-planar graphs is still an open problem. Thus we restrict ourselves to clustered graphs for which we know that c-planarity can be efficiently tested. We use the results of Feng et al. [13] who provide a polynomial-time algorithm to test whether a *connected* clustered graph is c-planar. Thus, in the rest of the paper we consider only *connected c-planar graphs*.

3 Fitting on a Rectangular Map

Here we consider the problem of deciding whether a connected c-planar graph G has a straight-line planar fitting on a given compatible rectangular map M . We first show that such a fitting does not always exist. To construct the counterexample we use a *wheel map*, which contains four “thin rectangles” that surrounds an inner rectangle; see Fig. 1.

Intuitively, the notion of a thin rectangle will be clear in the following constructions from the way it is used, but to be more precise, we formally define it. A *thin rectangle* is one whose larger side is at least 4 times its smaller side, i.e., it has aspect ratio at least 4. A thin rectangle is *horizontal* if its smaller side is its height; otherwise it is *vertical*. We assume all four thin rectangles in a wheel map have the same size (same length of larger sides, same length of smaller sides).

Let $\{V_1, \dots, V_k\}$ be the set of clusters of G and let (v_i, v_j) be an edge of G such that

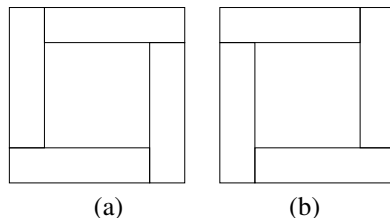


Fig. 1. Wheel maps cw (a), ccw (b).

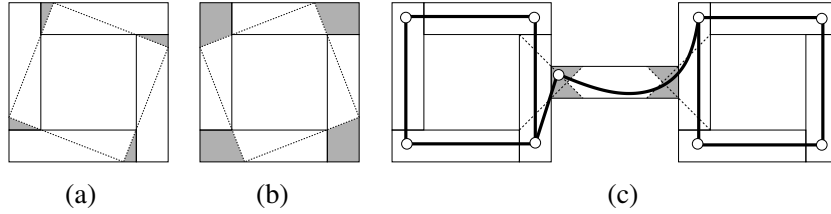


Fig. 2. (a)–(b) Proximity regions for the doors in a wheel map, (c) a clustered graph and a map with no fit (the visibility regions of the bridge are highlighted).

$v_i \in V_i, v_j \in V_j, 1 \leq i, j \leq k$. Then there exists a common boundary between the polygons representing V_i and V_j in M . Call the common boundary the *door* for the edge (v_i, v_j) . Consider a wheel map W and its dual G which has a simple clustering: each vertex constitutes a cluster. For the rest of the section we often assume that a wheel map is associated with this clustered graph. With this consideration in mind, each thin rectangle of W contains two doors, one for each incident thin rectangle. We define the *entry door* to be the one which contains a complete side of the rectangle, and the *exit door* to be the one that contains a complete side of a neighboring thin rectangle. We call a wheel map a *clockwise (cw) wheel* when going from the entry door to the exit door in each rectangle requires a clockwise walk through the wheel; see Fig. 1(a). A *counterclockwise (ccw) wheel* is defined analogously; see Fig. 1(b).

We now define the notion of “proximity region” for each door inside each thin rectangle of a wheel map. Given that the thin rectangles of a wheel map are of the same size, using basic geometry we can define inside each thin rectangle a triangular region of maximum area where these two conditions hold: (i) the triangular region inside each thin rectangle contains the entry door of this rectangle; and (ii) for each point inside the triangle of one of the four thin rectangles, there exists a point inside the triangle of each other rectangle such that the visibility line between the two points inside any pair of neighboring rectangles passes through the corresponding door. We call these triangular regions the *proximity regions* of the corresponding entry doors; see Fig. 2(a). For each exit door we can analogously define a quadrangular *proximity region*; see Fig 2(b). Next we state a simple observation that follows from these definitions:

Observation 1. *Let W be a wheel map and let G be its dual graph. In a straight-line planar fitting of G the vertices in the thin rectangles either all lie inside the proximity region of the entry doors, or they all lie in the proximity region of the exit doors. There exists a straight-line planar fitting in each case.*

Proof: The sufficiency follows from the definition of proximity regions. The necessity follows from the fact that the proximity regions for the entry and the exit doors inside each thin rectangle are disjoint since the aspect ratio of the thin rectangles is ≥ 4 . \square

The next lemma shows that fitting a planar clustered graph on a compatible map is not always possible.

Lemma 1. *There exist a planar clustered graph $C = (G, \mathcal{V})$ and a compatible rectangular map M , so that there is no straight-line planar fitting of C on M .*

Proof: Consider a rectangular map M made of two wheel maps (of the same size) joined together by a thin horizontal rectangle, called a *bridge*; see Fig. 2(c). We choose

the height of the bridge to be at most the length of the smaller sides in the thin rectangles of the wheels and we attach the bridge in the middle of the neighboring two vertical rectangles. For each of these two thin rectangles, the *visibility region* of the bridge is the set of points in it from which the visibility line to at least one point in the proximity region of either of the entry or exit door of the rectangle passes through the door between the bridge and the rectangle. We choose the length of the bridge to be long enough, such that the two visibility regions for the two rectangles do not overlap.

Let G be the dual of M : two 4-cycles connected by a path of length two. Assume a clustering of G where each vertex constitutes a separate cluster. Then G has no straight-line planar fitting on M . If G had a straight-line planar fitting Γ on M , by Observation 1, all the vertices inside the thin rectangles of both the wheels must be placed in the proximity regions of the doors in Γ . But then, there is no feasible position for the vertex that represents the bridge since the two visibility regions of the bridge do not overlap. \square

3.1 Fitting is NP-Hard

We show that deciding if a given planar clustered graph has a planar fitting inside a given map is NP-hard, even for rectangular maps, with a reduction from Planar-3-SAT which is known to be NP-complete [23]. *Planar-3-SAT* is defined analogously to 3-SAT with the additional restriction that the *variable-clause bipartite graph* G_F for a given formula F is planar. There is an edge (x_i, A_j) in G_F if and only if x_i or \bar{x}_i appears in A_j . Knuth and Raghunathan [20] showed that one can always find a crossing-free drawing of the graph G_F for a Planar-3-SAT instance, where the variables and clauses are represented by rectangles, with all the variable-rectangles on a horizontal line, and with vertical edge segments representing the edges connecting the variables to the clauses. The problem remains NP-complete when such a drawing is given.

Theorem 1. *Let $C = (G, \mathcal{V})$ be a planar clustered graph with a rectangular map M , compatible with C . Deciding if C admits a straight-line planar fitting on M is NP-hard.*

Proof: We reduce an instance of Planar-3-SAT to an instance (C, M) of our problem. Let $F := A_1 \wedge \dots \wedge A_m$ be an instance of a Planar-3-SAT, where each literal in each clause A_i is a variable (possibly negated) from $U = \{x_1, \dots, x_n\}$. Let Γ_F be the given planar rectilinear drawing for this instance, as defined in [20]. From Γ_F we first construct the rectangular map M , then take G as the dual of M , where each vertex constitutes a separate cluster. We represent each literal by a wheel map (of the same size) in M : a positive (negative) literal is a cw (ccw) wheel. From the two possible vertex configurations inside each wheel we take the one in which the corresponding literal assumes a true value when the vertices inside the thin rectangles of the wheel lie in the proximity region of the exit doors and the literal assumes a false value when they lie in the proximity region of the entry doors. Unlike in Γ_F , we use a distinct wheel for each literal in each clause. For each variable x , we draw the wheels for all the (positive and negative) literals for x appearing in different clauses in a left-to-right order, according to the ordering of the edges incident to the corresponding vertices in Γ_F . To maintain consistency, we ensure that a true (false) value for a literal x implies a true (false) value for every other instance of x and a false (true) value for each instance of \bar{x} . This is done by means of thin rectangular bridges between two consecutive literals; see Fig. 3. The size of the bridges is chosen equal to the thin rectangles in the wheels.

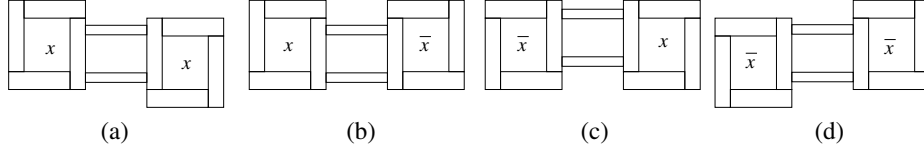


Fig. 3. Representation of variables.

For each clause $A = (x \vee y \vee z)$ of F , with the corresponding vertex lying above the variables in Γ_F , we draw vertical rectangles l_x^A , l_y^A and l_z^A from the topmost rectangles T_x , T_y , T_z of the wheels for x , y and z , respectively, attached at the end that is not shared by other thin rectangles. (The case when the vertex lies below the variables in Γ_F is analogous.) We place l_x^A , l_y^A , l_z^A so that they are completely visible from all the points in the proximity of the exit doors of T_x , T_y , T_z , respectively. We choose the length of the rectangles l_x^A , l_y^A , l_z^A so that not all points inside them are visible from any point of the proximity region of the entry doors of T_x , T_y , T_z , respectively; see Fig. 5.

We then draw a rectangle R for the clause and attach these three thin rectangles l_x^A , l_y^A and l_z^A to R . For z we attach the vertical rectangle l_z^A to the bottom of R , while for each of x and y , we attach horizontal rectangles h_x^A , h_y^A to R that also touch the vertical rectangles l_x^A and l_y^A coming from x or y , respectively. A point p in rectangle h_x^A (h_y^A) is *reachable* from a point q inside T_x (T_y) if there exists a point r inside l_x^A or l_y^A such that the two lines pr and rq pass through the doors between the corresponding rectangles. The *reachable region* of h_x^A (h_y^A) is the set of all points that are reachable from a point inside the proximity region of the entry door of T_x (T_y).

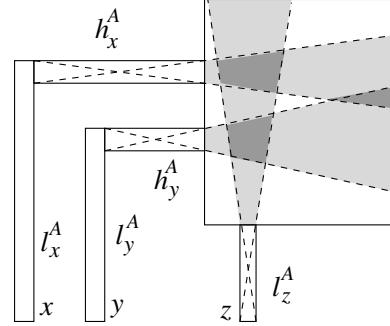


Fig. 4. Clause representation.

Similarly a point p inside l_z^A is reachable from a point q inside rectangle T_z of the wheel for z if the line pq passes through the door between the two rectangles. The *reachable region* of l_z^A is the set of all points reachable from a point inside the proximity region of the entry door of T_z . Choose the lengths for the horizontal rectangles h_x^A , h_y^A and the vertical rectangles l_x^A , l_y^A , l_z^A so that the reachable regions inside them do not coincide with the entire inside of these rectangles. For this purpose it is sufficient that the sizes of these rectangles are comparable to the sizes of the rectangles inside the wheels. Thus the sizes of all the wheels and other rectangles are polynomial in the size of the Planar-3-SAT instance.

Next we attach the thin rectangles h_x^A , h_y^A , l_z^A to R in such a way that the areas visible from the reachable regions of h_x^A , h_y^A , l_z^A do not have a common intersection, while every pair of them does have a common intersection; see Fig 4. We now observe that the size of R does not have to be too big to ensure this. Specifically, we attach l_z^A to the left half of the bottom line of R and choose the height of R small enough so that the visible area from its reachable region is only in the left half of R . We also adjust the vertical distance between the horizontal rectangles h_x^A and h_y^A and adjust the width of R so that the areas visible from the reachable regions of h_x^A and h_y^A do not intersect in the left half of R but they do intersect in the right half. This can be achieved if for

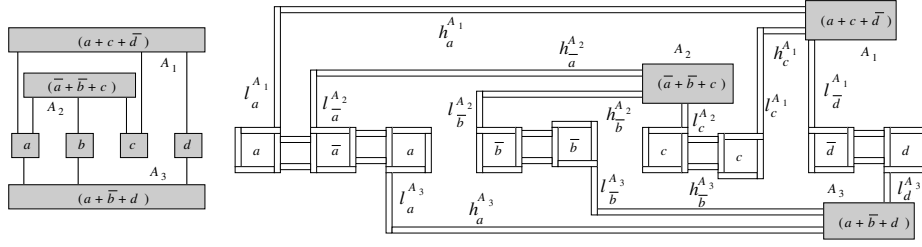


Fig. 5. Planar 3-SAT instance and corresponding map fitting instance.

example we take the vertical distance between h_x^A and h_y^A to be a constant multiple of their height, while we take the width of R to be a constant multiple of the length of h_x^A , h_y^A . Finally we fill all the unused regions in the map with additional rectangles to get the final map M . Since the sizes of all the rectangles are constant multiples of each other and the total size is polynomial in the size of the Planar-3-SAT instance, the coordinates for the map can be chosen to be polynomial in the size of the Planar-3-SAT instance.

Lemma 2. F is satisfiable if and only if G has a straight-line planar fitting on M .

Proof: Assume first that there exists a straight-line planar fitting Γ of G on M . We show that F is satisfiable, i.e., there is a truth assignment for all the variables of F such that for each clause $A = (x, y, z)$ of F , at least one of x, y and z is true. Let W_x be the wheel for x . If the vertices in W_x are placed inside the proximity regions of the entry doors, then by construction of M , the vertex in the horizontal rectangle h_x^A is placed inside the reachable region. Thus this vertex can see only the highlighted visible area in Fig. 4 inside the rectangle R for A . However if the vertices in W_x are placed in the proximity regions of the exit doors, then the vertex in the horizontal rectangle h_x^A can be placed outside the reachable region so that it can see the entire interior of R . This is true for each of the three literals. Since the visible areas of the three literals have no common intersection, the vertices in the wheel for at least one of x, y and z must be placed in the proximity region of the exit door. We make each such literal true. This assignment has no conflict, because of the way the wheels for a particular variable are attached to each other. Furthermore, this assignment satisfies F .

Conversely if F is satisfiable, for each clause $A = (x, y, z)$ of F , at least one of x, y, z is true. Without loss of generality, assume that x is true. Place the vertices in the wheel of x in the proximity regions of the exit doors. Then the vertex in the h_x^A can be placed outside the reachable region and it can see the entire interior of R . Place the vertex for R in the intersection of the areas visible from reachable regions of h_y^A and l_z^A . This ensures that we can place the vertices in the wheel for y and z in the proximity regions of either the entry doors or the exit doors and we are still able to place the vertices in rectangles h_y^A, l_y^A, l_z^A so that all the straight-line edges create no area-region crossings. This yields the desired straight-line planar fitting of G on M . \square

The proof of Lemma 2 completes the NP-hardness proof. Fig. 5 illustrates a 3-SAT formula, its Planar-3-SAT realization with the conditions of [20], and the corresponding instance for the map fitting problem (rectangles filling up the holes are not shown). \square

Note that Bern and Gilbert [4] and recently Kerber [19] obtained NP-completeness results using similar techniques. In particular, Bern and Gilbert [4] consider the problem of drawing the dual on top of the drawing of a plane graph G , such that each dual vertex

lies in the corresponding face of G , while each dual edge is drawn as a straight-line segment that crosses only its corresponding primal edge. They show that this problem is NP-complete and the techniques used are similar to ours, as this problem can be thought of as a special case of fitting a clustered graph on a map, where each cluster consists of a single vertex. However, we consider the more restricted class of rectangular maps instead of the generic drawing of a planar graph, and hence the NP-completeness of our problem is not implied by [4]. Kerber [19] considers the problem of embedding the dual on top of a primal partition of the d -dimensional cube into axis-aligned simplices and proved that this problem is NP-complete. In 2D, this problem is also a special case of our problem, with the exception that in Kerber's setting edge-region crossings are allowed. Thus the result in [19] also does not imply our results.

4 Sufficient Conditions for Fitting

We showed in Lemma 1 that not every c -planar connected graph admits a planar straight-line fitting on a compatible map even if each cluster is a single vertex. The counterexample relies on two facts: (1) there is a vertex in some cluster (the bridge) that is connected to vertices in two different clusters (the wheels); (2) its cluster-graph contains two cycles. By considering graphs that do not have at least one of the above characteristics we show planar straight-line fittings are always possible. In this sense the following two lemmas give tight sufficient conditions for graphs to admit planar straight-line fittings.

Lemma 3. *Let $C = (G, \mathcal{V})$ be a biconnected c -planar graph. Let M be a rectangular map compatible with C . If for each vertex v of G , all the vertices adjacent to v through an inter-cluster edge lie in the same cluster, C has a straight-line planar fitting on M .*

Proof: Let Γ be a c -planar drawing of C . Let G_1, G_2, \dots, G_k be the clusters of C and let $\mathcal{V} = \{V_1, V_2, \dots, V_k\}$ be the corresponding vertex partition. For each rectangle R_i , $1 \leq i \leq k$, of M representing the cluster G_i , denote by O_i the ellipse inscribed in R_i . We first place the vertices on the outer boundary of G_i in Γ on O_i as follows. Consider two adjacent rectangles R_i and R_j in M . Let $v_{i_1}, \dots, v_{i_r} \in V_i$ and $v_{j_1}, \dots, v_{j_s} \in V_j$ be the vertices of V_i and V_j , incident to the inter-cluster edges between these two clusters, taken in the order they appear on the outer boundary of G_i and G_j , respectively. Define p_i, p'_i and p_j, p'_j to be points of O_i and O_j , respectively, such that the straight-line segments $\overline{p_i p_j}$ and $\overline{p'_i p'_j}$ cross the common border of R_i and R_j , without crossing each other. Place the vertices v_{i_1}, \dots, v_{i_r} of V_i and v_{j_1}, \dots, v_{j_s} of V_j on O_i and O_j , between points p_i, p'_i and p_j, p'_j , respectively, so that all the inter-cluster edges between these vertices cross the common border of R_i and R_j . Repeat the above procedure for each pair of adjacent rectangles in M . Since each vertex thus placed is adjacent to a unique cluster, its position is uniquely defined. For each cluster G_i , $1 \leq i \leq k$, we have thus placed some vertices on the outer boundary of G_i in Γ on the ellipse O_i . Distribute the remaining vertices of the boundary of G_i on O_i , so that the order of the vertices is the same as in the boundary of G_i . Since the resulting drawing of the outer boundary of G_i is convex and G_i is biconnected, apply the algorithm for drawing a graph with a prescribed convex outer face [7] to complete the drawing of each cluster. \square

Lemma 4. *Let $C = (G, \mathcal{V})$ be a biconnected c -planar graph. Let M be a rectangular map compatible with C . If each connected component of cluster-graph G_C contains at most one cycle, then C has a straight-line planar fitting on M .*

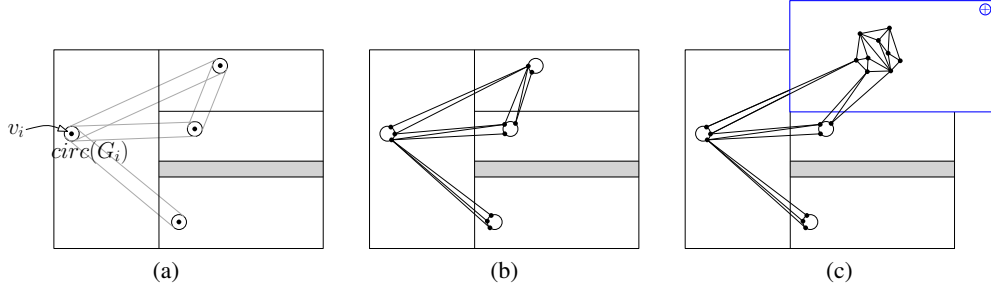


Fig. 6. (a) Drawing of G_C , each edge is represented by a strip of width $\varepsilon > 0$. (b) Placing the boundary vertices of the clusters on the corresponding circles. (c) Step 4 of the proof of Lemma 4.

Proof Sketch: Assume that each connected component of G_C contains at most one cycle. Let v_1, \dots, v_k be the vertices of G_C that represent clusters G_1, \dots, G_k respectively. To complete the proof we go through the following steps:

- (1) We show that G_C has a planar fitting on M .
- (2) We blow up the drawing of G_C , so that the edges of G_C are represented by strips of width $\varepsilon > 0$, without creating edge-region crossings; see Fig. 6(a). For each vertex v_i of G_C , we draw a small circle $\text{circ}(G_i)$ centered at the intersection of the strip-edges that are adjacent to v_i .
- (3) We draw the boundary of G_i on the circle $\text{circ}(G_i)$, $i = 1, \dots, k$, so that the inter-cluster edges, when drawn with straight-line segments, intersect neither the boundaries of the clusters, nor each other; see Fig. 6(b).
- (4) Since the boundary of each G_i is a convex polygon and G_i is biconnected, we can apply the algorithm for drawing a graph with a prescribed convex outer face [7] to complete the drawing of the clusters; see Fig. 6(c).

While steps (2) and (4) are straight-forward, steps (1) and (3) need to be proved. We provide a detailed proof of step (1) below. Step (3) is intuitively easy, however, its exact proof is quite technical; we omit the proof for this step here due to the space constraints.

Step (1). We show that $G_C = (V_C, E_C)$ has a straight-line planar fitting on M . Consider first the case when G_C is a tree and let $v_1 \in V_C$ be the root of G_C . We prove that even if the position of v_1 is fixed in its corresponding rectangle R_1 , we can place the remaining vertices of G_C in their corresponding rectangles so that the resulting straight-line drawing is a planar fitting of G_C on M . Let v_2, \dots, v_f be the children of v_1 and let R_2, \dots, R_f be the corresponding rectangles of M . Place v_2, \dots, v_f inside R_2, \dots, R_f , respectively so that the straight-line edges (v_1, v_i) , $2 \leq i \leq f$ cross the common boundary of R_1 and R_i . Continue with the children of v_2, \dots, v_f , recursively.

Assume now that each connected component of $G_C = (V_C, E_C)$ contains at most one cycle. We show how to draw a single connected component of G_C . Let $v_0, \dots, v_m \in V_C$ induce the unique cycle of G_C and let R_0, \dots, R_m be the rectangles that correspond to them, so that R_i and $R_{(i+1) \bmod (m+1)}$, $0 \leq i \leq m$, are adjacent. Place v_i , $0 \leq i \leq m$ inside R_i such that for any point $p \in R_{(i+1) \bmod (m+1)}$, segment $\overline{pv_i}$ crosses the common boundary of R_i and $R_{(i+1) \bmod (m+1)}$. (For example placing v_i right on the door suffices.) The removal of the edges of this cycle results in several trees. Root the trees at the vertices v_0, \dots, v_m , to which they are adjacent and apply the procedure described in the first part of the proof. This completes the construction. \square

Putting together the results in this section we obtain the following theorem:

Theorem 2. *Let $C = (G, \mathcal{V})$ be a biconnected c -planar graph. Let M be a rectangular map compatible with C . If (a) for each vertex v of G , all the vertices adjacent to v through an inter-cluster edge lie on the same cluster; or (b) each connected component of cluster-graph G_C contains at most one cycle, then C has a straight-line planar fitting on M . Moreover, there exist a c -planar graph C and a compatible map M which do not fulfill condition (a) and (b) and do not admit a planar straight-line fitting.*

5 Fitting Graphs on Rectilinear Maps

In this section we give a short overview of our results for more general rectilinear maps.

It is known that only a restricted class of planar graphs can be realized by rectangular maps. For general planar graphs, 8-sided polygons (T-shapes) are necessary and sufficient for contact maps [16]. In this section, we assume that the input is a rectilinear map, with rectangles, L- and T-shapes, together with a c -planar graph G with a cluster-graph G_C . The first condition that we require is that the subgraph induced by the inter-cluster edges is a matching. From Lemma 3 this condition is sufficient for rectangular maps. Now, we extend this to L-shaped and T-shaped polygons (maps). We impose several conditions under which we prove the existence of a fitting. Because of the presence of concave corners, we impose our second condition: none of the clusters contains a *boundary chord*, i.e., a non-boundary edge between two boundary vertices.

The idea is to apply the algorithm for drawing a graph with a prescribed convex outer face [7]. We partition the polygons into convex pieces. Since the polygons form a contact map, for each common boundary of adjacent polygons there is at least one edge between the corresponding clusters. Our last condition restricts this further: between any two adjacent clusters there exist at least two independent inter-cluster edges. We call a graph in which every pair of adjacent clusters has this property *doubly-interconnected*.

Note that the common boundary of two adjacent polygons contains at most two concave corners, one in each polygon. We place the vertices next to the common boundary on both sides of these concave corners. This ensures that the cycle spanned by the boundary vertices of the cluster is completely within the corresponding polygon and there are at most two concave corners along the cycle. Let a and b be the vertices at these corners; see Fig. 7. We choose a third boundary vertex c lying opposite a and b . Straight-line cuts between a, c and b, c define 3 convex regions. Now we compute an $(a \rightsquigarrow c)$ -path and a $(b \rightsquigarrow c)$ -path. These paths cannot have shortcuts, where a *shortcut* of a path P is an edge between vertices nonadjacent in P , so that we can place the vertices on these two paths on the two cuts between a, c and b, c . Note that such a path should not contain any other boundary vertex, already placed elsewhere.

The above strategy for T-shaped polygons can also be applied to L-shaped polygons, where the straight-line segment splitting this polygon into two convex parts is between the concave corner of the L-shaped polygon and its opposite convex corner of the neighboring polygon. Together, these yield the following theorem.

Theorem 3. *Let G be a doubly-interconnected and biconnected c -planar graph such that the inter-cluster edges of G form a matching and there is no boundary chord in any cluster. Then there exists a straight-line planar fitting of G on any compatible map with rectangular, L-shaped, or T-shaped polygons.*

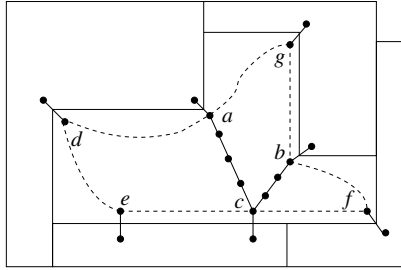


Fig. 7. Illustrating how vertices on the boundary of a T -shape are placed and how the cluster is partitioned by an $(a \rightsquigarrow c)$ -path and a $(b \rightsquigarrow c)$ -path, such that it fits into three adjacent convex polygons with the paths as common boundaries.

Note that any algorithm that distributes vertices inside non-convex regions while preserving cluster-planarity must distribute the vertices among the convex components of the regions. It is only natural to try to make such distribution balanced. We define a measure called “imbalance” which captures the difference between the geometric partition of the non-convex regions into convex regions (e.g., region area) and the partition of the clusters into subclusters (e.g., subcluster size) corresponding to the convex regions.

First we consider the distribution inside one L-shaped polygon partitioned into two pieces by a straight-line cut; then we use this result to minimize the maximum imbalance in all L-shaped polygons in the map. We prove that the global imbalance minimization problem can be solved optimally, using dynamic programming and min-max shortest paths. These techniques are interesting by themselves.

Theorem 4. *Let G be a connected c -planar graph, G_C be the cluster-graph of G and let M be a rectilinear map of G_C with six-sided polygons such that M represents the contact map of G_C . Then one can split the regions of M in $O(n^4)$ time into convex shapes and distribute the vertices and faces of the clusters within the regions, such that the maximum imbalance is minimized.*

6 Conclusion and Future Work

We showed that fitting planar graphs on planar maps is NP-hard. The proof involves skinny regions; it is natural to ask whether the problem becomes easier if all regions are “fat”. We presented necessary and sufficient conditions for the construction of planar straight-line fitting on rectangular map, for c -planar graphs with biconnected clusters. These conditions are tight, in the sense that violating them makes it possible to construct counterexamples. Relaxing the biconnectivity requirement is an open problem. Finally, we gave a rather restricted set of sufficient conditions for fitting planar graphs on maps with non-convex regions. It would be worthwhile to investigate whether these conditions can be relaxed. We gave an algorithm for finding a fitting with a “balanced distribution” of the vertices. Another interesting question is whether an exact bound on vertex resolution can be guaranteed.

References

1. P. Angelini, F. Frati, and M. Kaufmann. Straight-line rectangular drawings of clustered graphs. In *Symposium on Algorithms & Data Structures (WADS)*, pages 25–36, 2009.

2. G. D. Battista, G. Drovandi, and F. Frati. How to draw a clustered tree. *Journal of Discrete Algorithms*, 7(4):479–499, 2009.
3. G. D. Battista and F. Frati. Efficient c-planarity testing for embedded flat clustered graphs with small faces. *Journal of Graph Algorithms and Applications*, 13(3):349–378, 2009.
4. M. W. Bern and J. R. Gilbert. Drawing the planar dual. *Information Processing Letters*, 43(1):7–13, 1992.
5. M. Bruls, K. Huizing, and J. J. van Wijk. Squarified treemaps. In *Joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42, 2000.
6. A. Buchsbaum, E. Gansner, C. Procopiu, and S. Venkatasubramanian. Rectangular layouts and contact graphs. *ACM Transactions on Algorithms*, 4(1), 2008.
7. E. Chambers, D. Eppstein, M. Goodrich, and M. Löffler. Drawing graphs in the plane with a prescribed outer face and polynomial area. *Journal of Graph Algorithms and Applications*, 16(2):243–259, 2012.
8. J. W. Coffey, R. R. Hoffman, and A. J. Cañas. Concept map-based knowledge modeling: perspectives from information and knowledge visualization. *Information Visualization*, 5(3):192–201, 2006.
9. C. Duncan, E. Gansner, Y. Hu, M. Kaufmann, and S. G. Kobourov. Optimal polygonal representation of planar graphs. *Algorithmica*, 63(3):672–691, 2012.
10. P. Eades, Q.-W. Feng, X. Lin, and H. Nagamochi. Straight-line drawing algorithms for hierarchical graphs and clustered graphs. *Algorithmica*, 44(1):1–32, 2006.
11. S. Felsner. Rectangle and square representations of planar graphs. In J. Pach, editor, *Thirty Essays on Geometric Graph Theory*. Springer, 2012.
12. Q.-W. Feng, R. F. Cohen, and P. Eades. How to draw a planar clustered graph. In *Conference on Computing and Combinatorics (COCOON)*, pages 21–30, 1995.
13. Q.-W. Feng, R. F. Cohen, and P. Eades. Planarity for clustered graphs. In *European Symposium on Algorithms (ESA)*, pages 213–226, 1995.
14. É. Fusy. Transversal structures on triangulations: A combinatorial study and straight-line drawings. *Discrete Mathematics*, 309(7):1870–1894, 2009.
15. D. Harel. On visual formalisms. *Communications of the ACM*, 31(5):514–530, 1988.
16. X. He. On floor-plan of plane graphs. *SIAM Journal of Computing*, 28(6):2150–2167, 1999.
17. Y. Hu, E. R. Gansner, and S. G. Kobourov. Visualizing graphs and clusters as maps. *IEEE Computer Graphics and Applications*, 30(6):54–66, 2010.
18. T. Kamada. *Visualizing Abstract Objects and Relations*, volume 5 of *World Scientific Series in Computer Science*. 1989.
19. M. Kerber. Embedding the dual complex of hyper-rectangular partitions. *Journal of Computational Geometry*, 4(1):13–37, 2013.
20. D. E. Knuth and A. Raghunathan. The problem of compatible representatives. *SIAM Journal on Discrete Mathematics*, 5(3):422–427, 1992.
21. K. Koźmiński and E. Kinnen. Rectangular duals of planar graphs. *Networks*, 15:145–157, 1985.
22. C.-C. Liao, H.-I. Lu, and H.-C. Yen. Compact floor-planning via orderly spanning trees. *Journal of Algorithms*, 48:441–451, 2003.
23. D. Lichtenstein. Planar formulae and their uses. *SIAM Journal on Computing*, 11(2):329–343, 1982.
24. B. Shneiderman. Tree visualization with tree-maps: 2-D space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, 1992.
25. P. Ungar. On diagrams representing graphs. *Journal of London Mathematical Society*, 28:336–342, 1953.
26. K.-H. Yeap and M. Sarrafzadeh. Floor-planning by graph dualization: 2-concave rectilinear modules. *SIAM Journal on Computing*, 22:500–526, 1993.