A CENSUS OF GRAPH-DRAWING ALGORITHMS BASED ON GENERALIZED TRANSVERSAL STRUCTURES

OLIVIER BERNARDI*, ÉRIC FUSY†, AND SHIZHE LIANG+

Abstract. We present two graph drawing algorithms based on the recently defined *grand-Schnyder woods*, which are a far-reaching generalization of the classical Schnyder woods. The first is a straight-line drawing algorithm for plane graphs with faces of degree 3 and 4 with no separating 3-cycle, while the second is a rectangular drawing algorithm for the dual of such plane graphs.

In our algorithms, the coordinates of the vertices are defined in a global manner, based on the underlying grand-Schnyder woods. The grand-Schnyder woods and drawings are computed in linear time.

When specializing our algorithms to special classes of plane graphs, we recover the following known algorithms:

- He's algorithm for rectangular drawing of 3-valent plane graphs, based on transversal structures,
- Fusy's algorithm for the straight-line drawing of triangulations of the square, based on transversal structures,
- Bernardi and Fusy's algorithm for the orthogonal drawing of 4-valent plane graphs, based on 2-orientations,
- Barrière and Huemer's algorithm for the straight-line drawing of quadrangulations, based on separating decompositions.

Our contributions therefore provide a unifying perspective on a large family of graph drawing algorithms that were originally defined on different classes of plane graphs and were based on seemingly different combinatorial structures.

1. Introduction

In the rich literature on graph drawing algorithms (see for instance [BETT99, NR04] and references therein), two of the most classical and popular types of drawings are *straight-line drawings* (in which edges are represented by straight-line segments) and *orthogonal drawings* (where edges are represented by a sequence of axis-aligned segments).

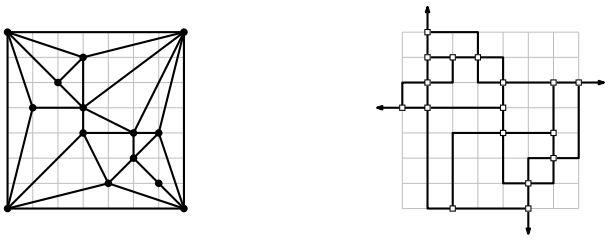


Figure 1. Straight-line drawing of a plane graph with faces of degree 3 and 4 (left), and rectangular drawing of its dual (right).

Date: December 3, 2025.

*Department of Mathematics, Brandeis University, Waltham MA, USA, bernardi@brandeis.edu.
†LIGM/CNRS, Université Gustave Eiffel, Champs-sur-Marne, France, eric.fusy@univ-eiffel.fr.
+Department of Mathematics, Brandeis University, Waltham MA, USA, Shizhe1011@brandeis.edu.

1.1. **New drawing algorithms based on grand-Schnyder woods.** In this article we present two new drawing algorithms: a straight-line drawing algorithm for simple plane graphs with faces of degree 3 or 4 with no separating cycle of length 3, and an orthogonal drawing algorithm for their duals. Graph drawings obtained using these algorithms are shown in Figure 1. In these algorithms, the coordinates of the vertices are defined globally, either in terms of face-counts in certain regions or in terms of increasing functions over some bipolar orientations. The combinatorial structures and the drawings can be computed in linear time.

Our drawing algorithms are based on a recently discovered class of combinatorial structures [BFL24]. These structures are called *grand-Schnyder structures* in honor of their relation to the classical *Schnyder woods* underlying a most famous drawing algorithm [Sch90]. Precisely, for every integer $d \geq 3$, a class of combinatorial structures called *d-grand-Schnyder* (*d-GS* for short), was defined for *d-maps*, that is, connected plane graphs such that the outer face has degree d and inner faces have degree at most d. It was shown in [BFL24] that a d-map admits a d-GS structure if and only if its non-facial cycles have length at least d. The classical Schnyder woods correspond to 3-GS structures on triangulations.

In this article, the algorithms are based on 4-GS structures. Interestingly, when specializing 4-GS structures to the case of the triangulations of the square, one recovers a combinatorial structure used in another classical graph drawing algorithm, namely the *regular edge labelings* used by He to define a rectangular drawing algorithm for 3-valent plane graphs [He93]. Since then, regular edge labelings have been rediscovered and popularized under the name *transversal structures* which we shall adopt in the present article. In a different direction, when specializing 4-GS structures to quadrangulations one recovers the separating decompositions (or equivalently, 2-orientations) which have also found applications to graph drawing algorithms: straight-line drawings and orthogonal drawings [BH12, BF12], as well as visibility representations and segment contact representations [RT86, TT86, TT89].

As we will later explain, our orthogonal drawing algorithm interpolates between He's algorithm [He93] (for 3-valent plane graphs) and Bernardi and Fusy's algorithm [BF12] (for 4-valent plane graphs), while our straight-line drawing algorithm interpolates between Fusy's algorithm [Fus09] (for triangulations) and Barrière and Huemer's algorithm [BH12] (for quadrangulations). This reveals the deep hidden commonality between these 4 algorithms (in addition to relating them to other Schnyder-wood based algorithms [Sch90, Fel01]). Our proofs are also uniform, and make use of a simple local planarity criteria which could be of independent interest (Lemma 3.7).

Before describing our results further, we give a quick review of the 4 drawing algorithms in [BH12, BF12, Fus09, He93].

1.2. **A review of four classical graph-drawing algorithms.** In this subsection we review the following graph drawing algorithms:

- the straight-line drawing algorithm for irreducible triangulations of the square due to Fusy [Fus09], and the (bendless) orthogonal drawing algorithm for their duals due to He [He93],
- the straight-line drawing algorithm for simple quadrangulations due to Barrière and Huemer [BH12], and the orthogonal drawing algorithm for their duals due to Bernardi and Fusy [BF12].

We start by defining the combinatorial structures underlying these algorithms: transversal structures and separating decompositions.

We call *planar map* a planar drawing of a connected planar graph, considered up to continuous deformation. A *triangulation of the square* is a planar map such that every inner face has degree 3, while the outer face has degree 4. Such a planar map is called *irreducible* if it

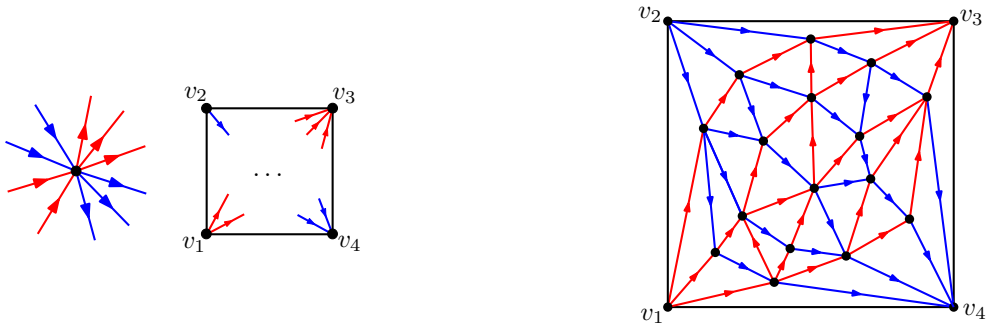


FIGURE 2. Left: The local conditions (at inner vertices and at outer vertices) for transversal structures. Right: Example of a transversal structure on a triangulation of the square.

has no cycle of length less than 4 except for the contour of the inner faces. It was shown by He [He93] that irreducible triangulations of the square can be endowed with some combinatorial structures that he named *regular edge-labelings*. These structures were later rediscovered by Fusy [Fus09], who used the name *transversal structures* that we adopt below. Consider an irreducible triangulation of the square G, with its outer vertices labeled v_1, v_2, v_3, v_4 in clockwise order. A *transversal structure* for G is an orientation and bicoloration of its inner edges — in blue and red — satisfying the local conditions shown in Figure 2:

(1) At each inner vertex the incident edges form 4 groups in clockwise order: outgoing red, outgoing blue, incoming red, incoming blue.
(2) The edges incident to the outer vertices v_1, v_2, v_3, v_4 are respectively outgoing red, outgoing blue, incoming red, and incoming blue.

It is known that these local conditions imply the global condition that the spanning subgraph G_r formed by the red edges and the 4 outer edges directed from v_1 to v_3 is an acyclic orientation with a unique source v_1 and a unique sink v_3; and similarly the spanning subgraph G_b formed by the blue edges and the 4 outer edges directed from v_2 to v_4 is an acyclic orientation with a unique source v_2 and a unique sink v_4. See Figure 2 for an example.

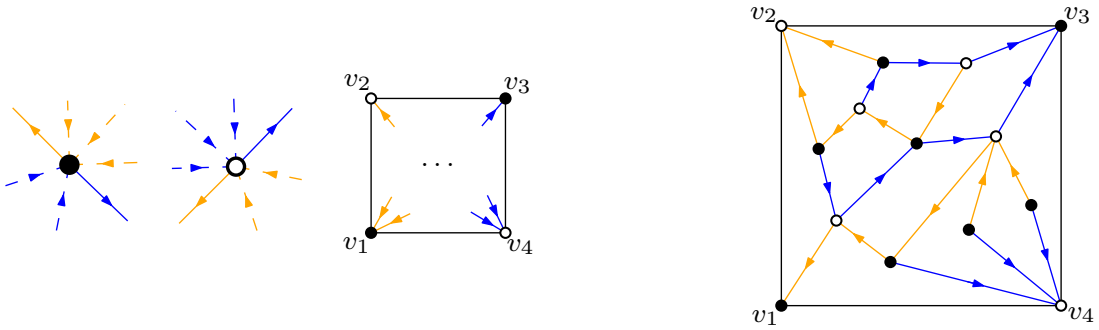


FIGURE 3. Left: The local conditions, at inner vertices and at outer vertices, for separating decompositions. Right: Example of a separating decomposition on a simple quadrangulation.

A *(simple) quadrangulation* is a planar map (without loops nor double edges) such that every face has degree 4. It was shown in [DFOdMR95] that simple quadrangulations can be endowed with combinatorial structures called *separating decompositions*. Consider a simple quadrangulation G with its outer vertices labeled v_1, v_2, v_3, v_4 in clockwise order, and consider the proper bicoloration of its vertices in black and white such that v_1 is black (quadrangulations are bipartite). A *separating decomposition* for G is an orientation and bicoloration of its inner edges — in blue and orange — satisfying the local conditions shown in Figure 3:

(1) At every black (resp. white) inner vertex there is a single outgoing blue edge e and a single outgoing orange edge e', and all the incident edges between e and e' in clockwise order are blue (resp. orange) while all the incident edges between e' and e are orange (resp. blue).
(2) The edges incident to the outer vertices v_1, v_2, v_3, v_4 are all incoming; they are orange at v_1, v_2 and blue at v_3, v_4.

It is shown in [DFOdMR95] that the subgraph T_o formed by the inner vertices, the orange edges, and the outer edge (v_1, v_2) is a tree, with edges oriented toward $\{v_1, v_2\}$; by convention it is considered as rooted at v_2. Similarly the subgraph T_b formed by the inner vertices, the blue edges and the outer edge (v_3, v_4) is a tree, rooted at v_4.

The two combinatorial structures described above have been used to define straight-line drawing algorithms using face-counting operations. These algorithms were introduced respectively in [Fus09] for transversal structures, and in [BH12] for separating decompositions. We will now describe these algorithms before turning to the dual picture.

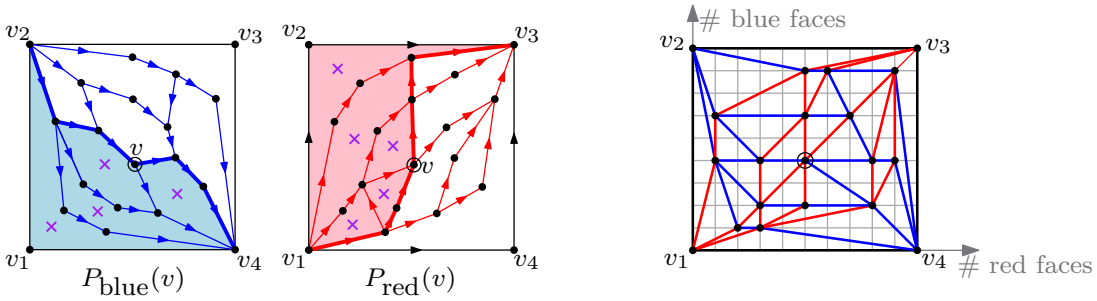


FIGURE 4. The straight-line drawing obtained using the transversal structure shown in Figure 2. The blue and red separating paths are shown for the circled vertex v. For this vertex, the blue path $P_b(v)$ has 4 faces on its right in the blue map, and the red path has 5 faces on its left in the red map. Accordingly, v has coordinates $(5, 4)$ in the drawing. Placing similarly all the inner vertices yields the planar straight-line drawing shown on the right. Since the red and blue maps have 10 and 9 inner faces respectively, the grid has size 10×9.

We start by describing the drawing algorithm from [Fus09] which is illustrated in Figure 4. Let G be an irreducible triangulation of the square endowed with a transversal structure. For every inner vertex v of G we will define some blue and red *separating paths*, represented in Figure 4, which are used to determine the placement of v. The *leftmost outgoing blue path* of an inner vertex v is the path $P^{\text{out}}_{\text{blue}}(v)$ starting at v and taking the leftmost outgoing blue edge at each step until reaching v_4, while the *rightmost incoming blue path* is the path $P^{\text{in}}_{\text{blue}}(v)$ starting at v and taking the rightmost incoming blue edge at each step until reaching v_2. We use the notation P^- for the reversal of a directed path P, and define the *blue separating path* at v as the blue path $P_{\text{blue}}(v) = P^{\text{out}}_{\text{blue}}(v) \cup P^{\text{in}}_{\text{blue}}(v)^-$ going from v_2 to v_4. We define similarly

the *leftmost outgoing red path* $P^{\text{out}}_{\text{red}}(v)$ (going from v to v_3), the *rightmost incoming red path* $P^{\text{in}}_{\text{red}}(v)$ (going from v to v_1), and the *red separating path* $P_{\text{red}}(v) = P^{\text{out}}_{\text{red}}(v) \cup P^{\text{in}}_{\text{red}}(v)^-$. In the straight-line drawing of G given by [Fus09], the vertices are placed on a grid of size $f_r \times f_b$, where f_r, f_b are the numbers of inner faces in the red and blue maps G_r, G_b:

- the outer vertices of G are placed at the 4 corners of the grid with v_1 at the origin,
- each inner vertex v is placed at (x_v, y_v), where x_v is the number of inner faces of G_r on the left of $P_{\text{red}}(v)$, and y_v is the number of inner faces of G_b on the right of $P_{\text{blue}}(v)$.

An example is given in Figure 4.

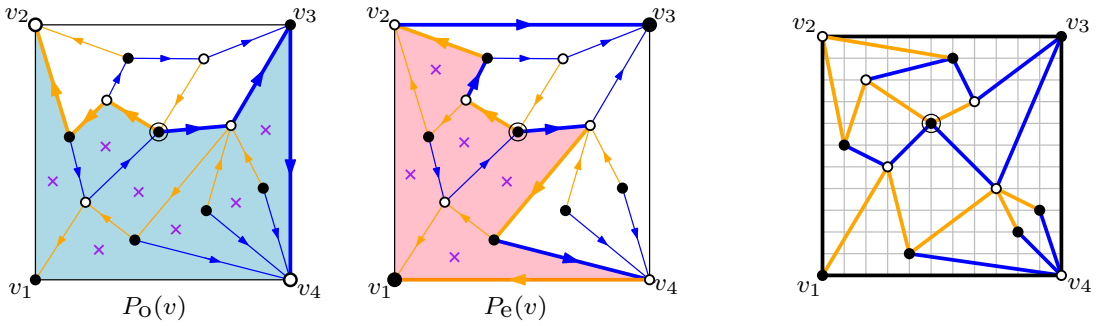


FIGURE 5. The straight-line drawing obtained using the separating decomposition shown in Figure 3. The even (resp. odd) separating path of the circled vertex is shown, with 5 faces on its left (resp. 7 faces on its right). Accordingly the circled vertex has coordinates $(5, 7)$ in the drawing. Placing similarly all the inner vertices yields the planar straight-line drawing shown on the right. Since there are 11 inner faces, the grid has size 11×11.

Next we describe the algorithm from [BH12], which is illustrated in Figure 5. Let G be a simple quadrangulation endowed with a separating decomposition. For each inner vertex v we define an odd and an even separating path as illustrated in Figure 5. For all $i \in \{1, 2, 3, 4\}$ we define $P_i(v)$ as the directed path from v to v_{i+3} obtained by following the outgoing arcs as follows:

- for $i = 1$ (resp. $i = 3$) we follow the outgoing blue (resp. orange) arc at every vertex,
- for $i = 2$ (resp. $i = 4$) we follow the outgoing blue (resp. orange) arc at every black vertex, and the outgoing orange (resp. blue) arc at every white vertex.

In the above definition of $P_i(v)$ the arcs (v_2, v_3) and (v_3, v_4) are blue, and the arcs (v_4, v_1) and (v_1, v_2) are orange. It is shown in [FHKO11] that with these conventions, the path $P_i(v)$ is always well-defined and ends at v_{i+3} for all i [FHKO11]. The *odd separating path* at v is the path $P_o(v) := P_1(v) \cup P_3(v)^-$ going from v_2 to v_4, while the *even separating path* at v is the path $P_e(v) := P_4(v) \cup P_2(v)^-$ going from v_1 to v_3. The separating paths are always simple paths as shown in [FHKO11]. In the straight-line drawing of G given by [BH12], the vertices are placed on a grid of size $f \times f$, where f is the number of inner faces:

- the outer vertices of G are placed at the 4 corners of the grid with v_1 at the origin,
- each inner vertex v is placed at (x_v, y_v), where x_v is the number of inner faces of G on the left of $P_e(v)$, and y_v is the number of inner faces of G on the right of $P_o(v)$.

An example is given in Figure 5.

Let us highlight the similarity between the algorithms in [Fus09] and [BH12]. As already noted in [BH12] (building upon the *0,1-labelings* of [FHKO11]), there is a natural way to label the inner corners of a separating decomposition with labels in $\{1, 2, 3, 4\}$ using the rule

represented in Figure 6, and the paths $P_1(v), \dots, P_4(v)$ associated to an inner vertex v can be defined in terms of this labeling: at each inner vertex the path $P_i(v)$ takes the outgoing edge separating the labels i and $i+1$. Similarly, there is a natural way to label the inner corners of a transversal structure using the rules represented in Figure 6, and the blue and red separating paths can be defined in terms of these labelings: at each inner vertex the path $P^{\text{out}}_{\text{blue}}(v)$ (resp. $P^{\text{in}}_{\text{red}}(v)$, $P^{\text{in}}_{\text{blue}}(v)$, $P^{\text{out}}_{\text{red}}(v)$) takes the edge separating the labels 1 and 2 (resp. 2 and 3, 3 and 4, 4 and 1).

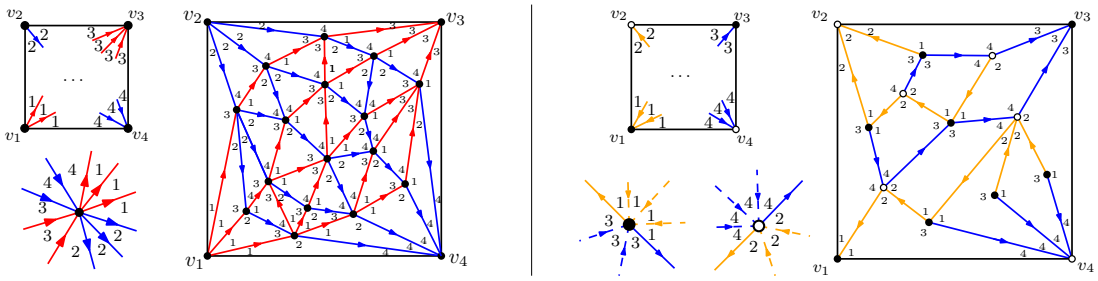


FIGURE 6. Left: Corner labeling associated to a transversal structure (rule for defining the labels around outer and inner vertices, and an example). Right: Corner labeling associated to a separating decomposition (rule for defining the labels around outer vertices and around black and white inner vertices, and an example).

We now turn our attention to the dual picture, and review two orthogonal drawing algorithms based on the same pair of combinatorial structures [BF12, He93]. An *orthogonal drawing* of a planar graph G' is a drawing such that each edge is a sequence of axis-aligned segments, a bend being a turning point on an edge. An orthogonal drawing is called *one-bend* if every edge has at most one bend. All orthogonal drawing algorithms to be reviewed and introduced in the paper produce one-bend drawings (upon not drawing the root vertex, see below).

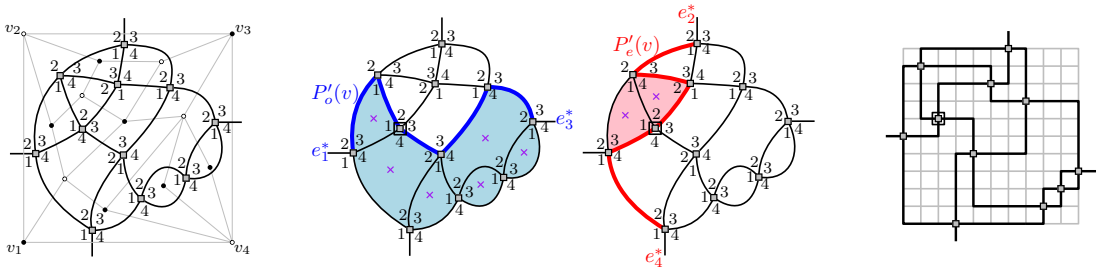


FIGURE 7. Orthogonal drawing from [BF12] obtained using the (dual of the) separating decomposition shown in Figure 3. The odd and even separating paths of the circled vertex v are indicated. The even path $P'_e(v)$ has 2 faces on its left, while the odd path $P'_o(v)$ has 6 faces on its right, hence the circled vertex has coordinates $(2, 6)$ in the drawing. Placing similarly all the dual vertices (excluding v_∞) yields the orthogonal drawing shown on the right.

We first describe orthogonal drawing algorithm from [BF12] for the dual of simple quadrangulations. Let G be a simple quadrangulation, let G^* be the dual of G, and let v_∞ be the

vertex of G^* dual to the outer face of G. We call v_∞ the *root vertex* of G^*. The drawing of $G' := G^* \setminus v_\infty$ is again based on a face-counting procedure. For $i \in \{1,2,3,4\}$ we define e_i^* as the edge of G^* dual to the outer edge $\{v_i, v_{i+1}\}$ of G, and we let v_i^* be the non-root endpoint of e_i^*. Recall that, given a separating decomposition of G, one can associate some labels to the inner corners of G as indicated in Figure 6. By duality, this gives a labeling of the corners of G^* not incident to v_∞. This labeling is such that the labels $1,2,3,4$ appear in clockwise order around each non-root vertex. See Figure 7 for an example. For $i \in \{1,2,3,4\}$, let $P'_i(v)$ be the path starting at v and taking the edge separating the corners of labels i and $i+1$ at each vertex, until reaching v_i^*. The *even separating path* at v is the simple directed path $P'_e(v) := P'_2(v) \cup P'_4(v)^-$ from v_4^* to v_2^* (where we again use the notation P^- to indicate the path obtained by reversing P). The *odd separating path* at v is the path $P'_o(v) := P'_3(v) \cup P'_1(v)^-$ from v_1^* to v_3^*. In the orthogonal drawing of G' from [BF12], the coordinates (x, y) of each non-root vertex v are such that x (resp. y) is the number of inner faces of G' on the left of $P'_e(v)$ (resp. on the right of $P'_o(v)$). Each edge is then drawn as the union of two axis-aligned segments in the unique way such that, at each vertex v, the path $P'_1(v)$ (resp. $P'_2(v)$, $P'_3(v)$, $P'_4(v)$) leaves v toward the left (resp. top, right, bottom). Equivalently, the corner of label 1 (resp. 2,3,4) occupies the south-west (resp. north-west, north-east, south-east) quadrant at v. If G' has n inner faces, then the grid size is $n \times n$. An example is given in Figure 7.

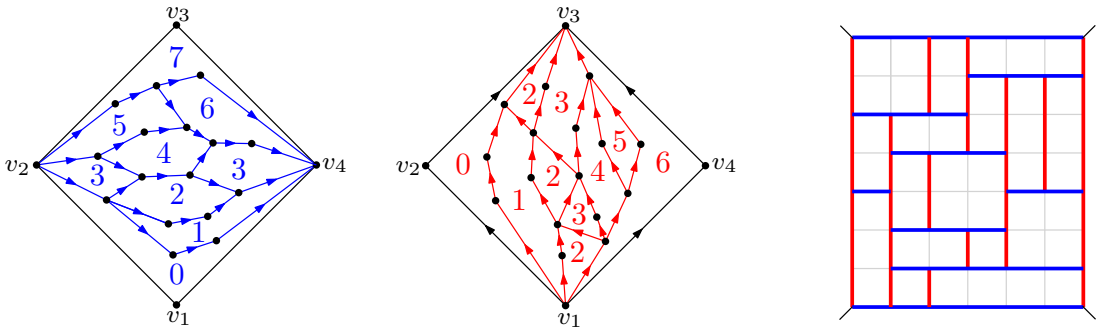


FIGURE 8. Rectangular drawing obtained using increasing functions on the transversal structure shown in Figure 2. The increasing functions of G_b and G_r are indicated on the left, and the resulting rectangular drawing of G' is shown on the right.

Last but not least, we recall the rectangular drawing algorithm from [He93] based on transversal structures. Recall that a *rectangular drawing* is an orthogonal drawing, such that edges have no bend, and all faces (including the outer face) have a rectangular shape. Let G be an irreducible triangulation of the square, let G^* be its dual, and let v_∞ be the vertex of G^* dual to the outer face of G. We consider a transversal structure of G, and define a rectangular drawing of $G' := G^* \setminus v_\infty$ using some "increasing functions" on the faces of the blue and red maps, as represented in Figure 8. An *increasing function* for G_r (resp. G_b) is an assignment of values in $\mathbb{R}$ to the inner faces such that every inner edge has a smaller (resp. larger) label on the left side than on the right side. Every vertex $v \in G'$ is dual to a triangular inner face f of G which is contained in a face $f_r(v)$ of G_r and in a face $f_b(v)$ of G_b. In the rectangular drawing of G' from [He93], the coordinates (x, y) of v are such that x (resp. y) is the label of the face $f_r(v)$ (resp. $f_b(v)$). If face-labels are in $\mathbb{N}$, then one obtains a rectangular drawing on a grid. The choice of increasing function for G_r (resp. G_b) yielding the smallest grid-width (resp. grid-height) is the one where the label of the leftmost (resp. rightmost) inner face is 0, and the

label of each other inner face is 1 plus the maximum of the labels of the faces that are adjacent from the left (resp. right); equivalently, each face is labeled by the length of a longest path in the dual bipolar orientation. This optimal choice of increasing functions is used in Figure 8.

1.3. **Outline of the article.** In this article we present two new drawing algorithms that make use of the *grand-Schnyder woods* introduced in [BFL24]. Our straight line drawing algorithm applies to the class of *4-adapted maps*, which are the planar maps such that the outer face has degree 4, the inner faces have degree at most 4 and the non-facial cycles have length at least 4. Our orthogonal drawing algorithm applies to the dual of 4-adapted maps.

It was shown in [BFL24] that any 4-adapted map admits a *4-grand-Schnyder wood* (or*4-GS wood* for short), which can be computed in linear time. The definition of 4-GS woods will be given in Section 2. Note that the irreducible triangulations of the square are the 4-adapted maps having only inner faces of degree 3, while the simple quadrangulations are the 4-adapted maps having inner faces of degree 4. Hence, irreducible triangulations of the square and simple quadrangulations form two subclasses of 4-adapted maps. Moreover, as we will explain in Section 2.2, the transversal structures of a triangulation of the square are in bijection with the 4-GS woods of that triangulation. As for separating decompositions, they correspond bijectively to a special class of 4-GS woods of quadrangulations satisfying some parity conditions (the *even 4-GS woods*). So the concept of 4-GS woods allows one to interpolate between the concept of transversal structures used in [Fus09, He93] and the concept of separating decompositions used in [BH12, BF12].

Regarding straight-line drawings, we present a "face-counting" algorithm for 4-adapted maps, where the coordinates of each vertex are determined as the number of faces in certain "regions" associated to that vertex. When specialized to irreducible triangulations of the square or to simple quadrangulations, the face counting algorithm yields the two algorithms from [Fus09] and from [BH12] respectively.[1] We then present a more general straight-line drawing algorithm for 4-adapted maps, where the coordinates of each vertex are obtained using *increasing functions* on some bipolar orientations associated to the 4-GS wood. We show that the increasing function algorithm can be performed in linear time, and that it leads to a grid size that is less or equal to the grid size of the face-counting algorithm. We also discuss some strategies for further decreasing the grid-size.

Regarding orthogonal drawings, we first give a "face-counting" algorithm for the dual of 4-adapted maps (where the coordinates of each vertex are defined as the number of faces in certain "regions" associated to that vertex). We then present a more general "increasing function" drawing algorithm. The face-counting algorithm recovers the algorithm from [BF12] when applied to the dual of quadrangulations, while the increasing-function algorithm recovers the algorithm from [He93] when applied to the dual of triangulations. We show that the increasing function algorithm can be performed in linear time, and we then discuss strategies to minimize the number of bends and to decrease the grid-size. The orthogonal drawings produced by our algorithm are *right-chiral*: an edge starting at a 3-valent vertex is either straight or has a right-bend (in particular edges between 3-valent vertices are straight). We show that the dual of a 4-map admits a right-chiral drawing if and only if it is 4-adapted, in which case we give an algorithm to construct a right-chiral drawing achieving the minimum number of bends possible.

The article is organized as follows. In Section 2 we define the combinatorial structures relevant for our drawing algorithms: 4-GS labelings, 4-GS woods, and some associated bipolar orientations (in the primal and in the dual). In Section 3 we present our straight-line drawing algorithms for 4-adapted maps: we start by presenting a face-counting algorithm (Section 3.1)

[1]For triangulations of the square, we more precisely recover the algorithm from [Fus09] in its compact version (where every grid-line is occupied by at least one vertex).

before presenting an increasing-function algorithm (Section 3.2). We then prove the correctness of the algorithms (based on a local correctness criterion, Lemma 3.7), analyze the time-complexity and grid size, and present an optimization for reducing the grid size (Sections 3.3 to 3.6). In Section 4 we present our orthogonal drawing algorithms for the dual of 4-adapted maps. We start by presenting a face-counting algorithm (Section 4.1), before presenting an increasing-function algorithm (Section 4.2). We prove the correctness of the algorithms in Section 4.3, before discussing the relation with He's algorithm, discussing additional properties of the drawing, explaining how to deal with vertices of degree 2, analyzing the time-complexity and the grid size, and presenting optimizations for reducing the grid size (Sections 4.4 to 4.9). Finally in Section 5 we present some variations on the face-counting algorithms (Section 5.1), we discuss a way to triangulate 4-adapted maps while preserving 4-adaptedness (Section 5.2), and we give an algorithm for minimizing the number of bends when drawing the duals of 4-adapted maps (Section 5.3).

2. Combinatorial structures

In this section we lay out some basic definitions about planar maps, and then define the combinatorial structures which underlie our drawing algorithms. These combinatorial structures have been introduced in [BFL24] under the name *grand-Schnyder structures*. Precisely, we will be using 4-grand-Schnyder structures and their duals.

2.1. **Planar maps.**

Our *graphs* are finite and undirected. A *simple graph* is a graph without loops nor multiple edges. A *planar map* is a connected planar graph drawn in the plane without edge crossing, considered up to continuous deformation. A *corner* of a planar map is the sector delimited by 2 consecutive edges around a vertex. A vertex or face has degree k if it is incident to k corners. An *arc* of a graph is an edge together with a chosen orientation of this edge (so each edge gives rise to two *opposite* arcs). The endpoints of an arc a are called the *initial* and *terminal* vertices of a. We write $a = (u, v)$ to indicate that u and v are the initial and terminal vertices of the arc a respectively. The infinite face of a planar map is called the *outer face*, while the others are called *inner faces*. We also call *outer* the vertices, edges, arcs and corners incident to the outer face, and *inner* the others.

A *3,4-angulation of the square* is a planar map such that the inner faces have degree 3 or 4, and the outer face has degree 4 and is incident to 4 distinct outer vertices. We canonically denote the outer vertices of such a planar map by v_1, v_2, v_3, v_4, in clockwise order around the outer face. A 3,4-angulation of the square is called *adapted* if any simple cycle which is not the contour of a face has length at least 4. Note that adapted 3,4-angulations are necessarily simple. Note also that a planar map with every face of degree 4 is adapted if and only if its girth is 4 (where *girth* means the minimal length of cycles).

Recall that the *dual* G^* of a planar map G is obtained as follows:

- we draw a vertex v_f of G^* inside every face f of G and call v_f the *dual vertex* of f,
- and we draw an edge e^* of G^* connecting v_f and v_g across each edge e separating the faces f and g of G, and call e^* the *dual edge* of the *primal edge* e.

The *root vertex* of G^* is the vertex v_∞ dual to the outer face of G.

If G is a 3,4-angulation of the square, then its dual graph G^* is a planar map having vertices of degree 3 or 4, with a designated root vertex v_∞ of degree 4. We call such a map a *rooted 3,4-map*. A rooted 3,4-map G^* is called *dual-adapted* if G is adapted. Equivalently, a rooted 3,4-map G^* is dual-adapted if the deletion of less than 4 edges does not disconnect G^* into 2 subgraphs having more than 1 vertex each. Note that dual-adaptedness is less demanding than *4-edge connectivity* since vertices of degree 3 are allowed.

2.2. **Grand-Schnyder structures.** In this subsection we recall some definitions and results from [BFL24] about 4-GS labelings and 4-GS woods.

Let G be a 3,4-angulation of the square. A *4-labeling* of G is an assignment of a *label* in $\{1,2,3,4\}$ to each inner corner of G; see for example Figure 10(a). A 4-GS labeling of G is a 4-labeling satisfying certain local conditions represented in Figure 9. These conditions are best expressed in terms of *jumps*. Consider a 4-labeling of G. Let c and c' be two inner corners of G, and let i and i' be their respective labels. The *label jump* from the corner c to the corner c' is defined as the integer δ in $\{0,1,2,3\}$ such that $i+\delta \equiv i'$ modulo 4 (in other words, the label jump δ is $i'-i$ if $i'-i \geq 0$, and $i'-i+4$ otherwise).

For an inner vertex v of G, the *sum of clockwise jumps* around v is the sum of label jumps between consecutive corners in clockwise order around v. Similarly, the *sum of clockwise jumps* around a face f is the sum of label jumps between consecutive corners in clockwise order around f.

Definition 2.1. *Let G be a 3,4-angulation of the square. As usual, we assume that the outer vertices of G are denoted by $v_1,\dots,v_4$ in clockwise order around the outer face. A* 4-GS labeling *of G is a 4-labeling satisfying the following conditions.*

- *(L0) For all $i \in \{1,2,3,4\}$, all the inner corners incident to v_i have label i.*
- *(L1) For every inner vertex or face of G, the sum of clockwise jumps around this vertex or face is 4.*
- *(L2) The label jump from a corner to the next corner around a face are always strictly positive.*
- *(L3) Let e be an inner edge incident to a face f of degree 3. Let c and c' be the consecutive corners incident to e in clockwise order around f, and let v be the vertex incident to c'. The label jump δ from c to c' and the label jump ϵ from c' to the next corner in clockwise order around v satisfy $\delta+\epsilon \geq 2$.*

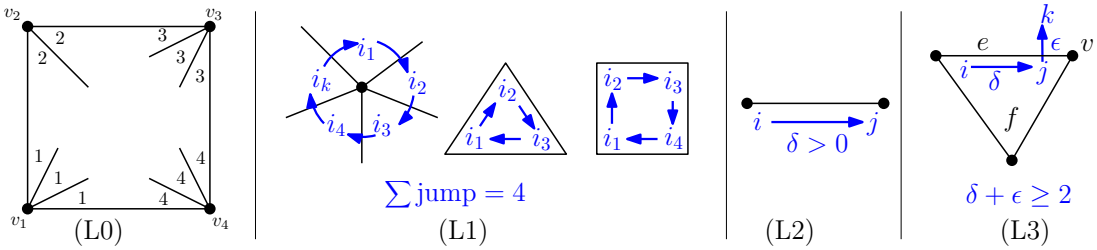


FIGURE 9. Conditions defining 4-GS labelings.

It was shown in [BFL24] that a 3,4-angulation of the square G admits a 4-GS labeling if and only if it is adapted. Moreover, in this case, a 4-GS labeling can be constructed in linear time (number of operations linear in the number of vertices of G). Let us state a few additional properties of 4-GS labelings.

Remark 2.2.
- Conditions (L1) and (L2) imply that around any inner face of degree 4 the labels are 1,2,3,4 in clockwise order. For an inner face of degree 3, the labels are $i, i+1, i+2$ (modulo 4) in clockwise order for some $i \in \{1,2,3,4\}$. In particular the label jumps along edges are either 1 or 2.
- The sum of label jumps counterclockwise around any inner edge e is equal to 4. This property has been established in [BFL24]. Note that this implies that the label jumps around vertices are either 0, 1 or 2.

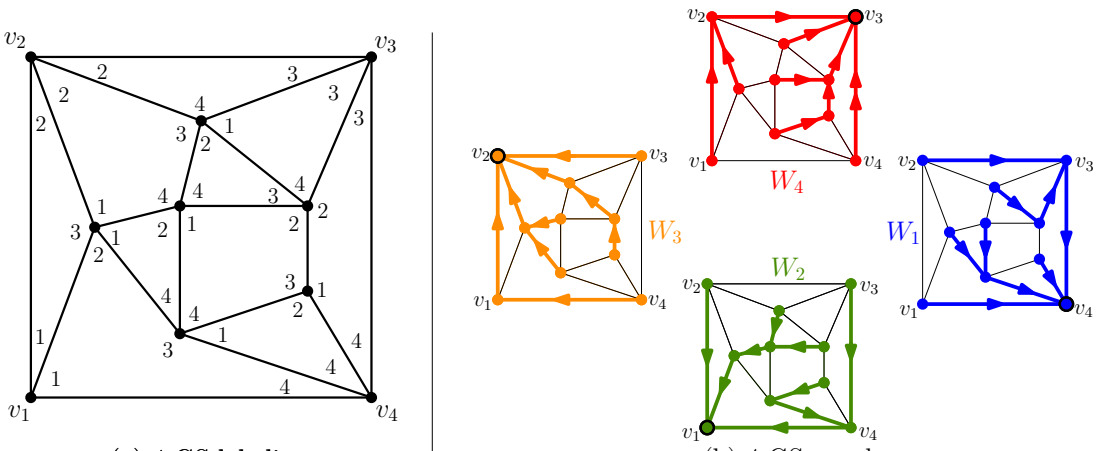
We now come back to the relation between transversal structures, separating decompositions, and 4-GS structures. As explained in the introduction, there is a natural way to associate a 4-labeling to a transversal structure, or to a separating decomposition. We start with the case of transversal structures. Let G be an irreducible (i.e. adapted) triangulation of the square. Given a transversal structure $\mathcal{T}$ of G, we define a labeling $\mathcal{L}$ of the inner corners of G by the rule represented in Figure 6(left):

- for all $i \in \{1, 2, 3, 4\}$, the corners incident to the outer vertex v_i are labeled i,
- a corner incident to an inner vertex v is labeled 1 (resp. 2,3,4) if the edge preceding c clockwise around v is outgoing red (resp. outgoing blue, incoming red, incoming blue).

It was shown in [BFL24] that this gives a bijection between the set of transversal structures of G and the set of 4-GS labelings of G. We now turn to separating decompositions. Let G be a simple (i.e. adapted) quadrangulation with its outer vertices labeled $v_1, \dots, v_4$, and its inner vertices properly bicolored in black and white such that v_1 is black. Given a separating decomposition $\mathcal{S}$ of G, we define a labeling of the inner corners of G by the rule represented in Figure 6(right):

- for all $i \in \{1, 2, 3, 4\}$, the corners incident to the outer vertex v_i are labeled i,
- a corner incident to a black (resp. white) inner vertex v is labeled 1 (resp. 4) if it is between the outgoing orange arc and the outgoing blue arc clockwise around v, and it is labeled 3 (resp. 2) otherwise.

This way of defining a corner labeling was given in [BH12]. As shown in [BF12, BFL24], it gives a bijection between the separating decompositions of G and the *even* 4-GS labelings of G, that is, the 4-GS labelings such that the labels around black vertices are odd and the labels around white vertices are even.



(a) 4-GS labeling (b) 4-GS wood

FIGURE 10. (a) A 4-GS labeling $\mathcal{L}$ of an adapted 3,4-angulation of the square. (b) The corresponding 4-GS wood $\mathcal{W} = (W_1, W_2, W_3, W_4)$.

Next, we define the 4-GS wood associated to a 4-GS labeling. Let G be a 3,4-angulation of the square. For each arc a of G there are four corners incident to a. The corner at the left of an arc a around its initial vertex is called the *left-initial corner of* a. The *right-initial*, *left-terminal*, and *right-terminal* corners of a are defined similarly. Given a 4-GS labeling $\mathcal{L}$ of G, and an inner arc a, we denote by left-init(a) (resp. right-init(a), left-term(a), right-term(a)), the label of the left-initial (resp. right-initial, left-terminal, right-terminal) corner of a. This is represented in

Figure 11(a). For i, j in $\{1, 2, 3, 4\}$, we denote by $[i : j[$ the set of integers $\{i, i+1, \dots, j-1\}$ modulo 4. Concretely, if $i = j$ then $[i : j[= \emptyset$; if $i < j$ then $[i : j[= \{k \mid i \le k < j\}$; and if $i > j$ then $[i : j[= \{k \mid i \le k \le 4 \text{ or } 1 \le k < j\}$.

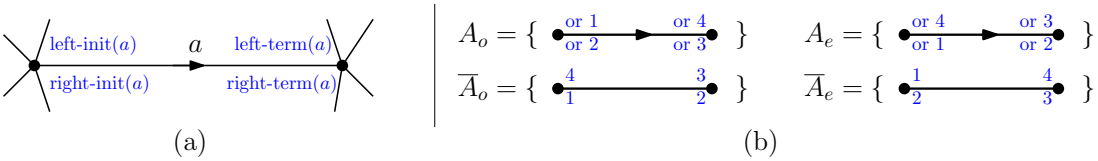


FIGURE 11. (a) The labels around an arc. (b) The inner arcs in the acyclic orientation A_o and A_e, and the inner edges not in their support.

Given a 4-GS labeling $\mathcal{L}$ of G, we define four oriented subgraphs $W_1, \dots, W_4$ as follows. For all $k \in \{1, 2, 3, 4\}$, the subgraph W_k has the same vertices as G and set of arcs

$$W_k = \{a \text{ inner arc} \mid k \in [\text{left-init}(a), \text{right-init}(a)[\} \cup \{(v_{k+1}, v_{k+2}), (v_{k+2}, v_{k+3}), (v_k, v_{k+3})\}.$$

The oriented subgraphs $W_1, \dots, W_4$ corresponding to the 4-GS labeling in Figure 10(a) are represented in Figure 10(b). We call $\mathcal{W} = (W_1, \dots, W_4)$ a *4-GS wood* of G, by analogy to the woods defined by Schnyder for triangulations [Sch89]. It was shown in [BFL24] that for all $k \in \{1, 2, 3, 4\}$, the subgraph W_k is an oriented spanning tree rooted at v_{k+3}, with every arc in the tree oriented from child to parent. Moreover, in the rooted tree W_k, the vertices v_k and v_{k+1} are leaves (i.e. they are not incident to any inner edge of G in W_k).[2]

2.3. **Bipolar orientations.** In this subsection we explain how to use 4-GS structures to define several bipolar orientations which play a key role in the drawing algorithms. Recall that a *bipolar orientation* is an orientation of a graph which is acyclic and has a unique *source* (vertex only incident to outgoing arcs) and a unique *sink* (vertex only incident to incoming arcs). Given a 4-GS labeling $\mathcal{L}$ of G, we define the *odd* and *even bipolar orientations* B_o and B_e as follows:

$$B_o = W_1 \cup W_3^- \text{ and } B_e = W_4 \cup W_2^-.$$

In the above definition, the symbol W_k^- indicates the oriented subgraph obtained from W_k by reversing every arc. The bipolar orientations B_o and B_e are indicated in Figure 12(a).

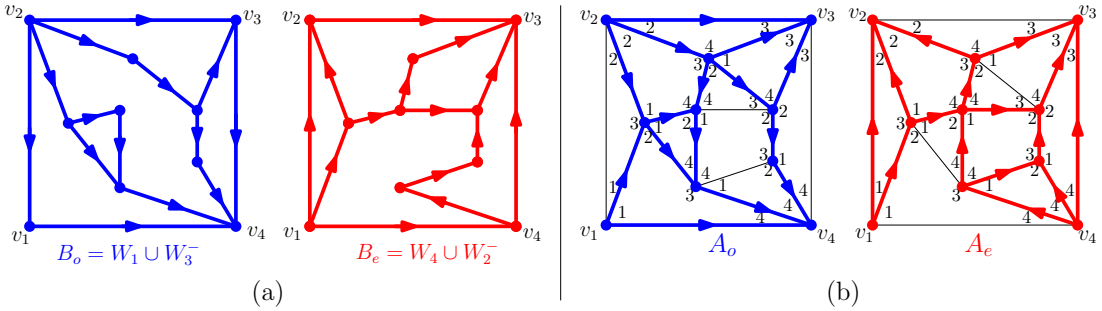


FIGURE 12. (a) Odd and even bipolar orientations. (b) Odd and even acyclic orientations.

[2]We mention that there is a small change of convention in the choice of the root of W_k in the present article compared to [BFL24]: in [BFL24] the root of W_k was defined to be v_k.

It is easy to check that neither B_o nor B_e contains any edge oriented both ways, that is, if an arc a belongs to B_o (resp. B_e) then the opposite arc $-a$ does not belong to B_o (resp. B_e). Hence B_o and B_e are oriented subgraphs of G. Furthermore we will prove below that B_o and B_e are not usurping their name, as these orientations are indeed bipolar.

Next we define the *odd* and *even acyclic orientations* A_o and A_e. These are the oriented subgraphs of G having the same vertices as G and the following sets of arcs:

$$\begin{aligned} A_o &= \{a \text{ inner arc} \mid \text{left-init}(a) = 1 \text{ or right-init}(a) = 2 \text{ or right-term}(a) = 3 \text{ or left-term}(a) = 4\} \\ &\quad \cup \{(v_1, v_4), (v_2, v_3)\}, \\ A_e &= \{a \text{ inner arc} \mid \text{left-init}(a) = 4 \text{ or right-init}(a) = 1 \text{ or right-term}(a) = 2 \text{ or left-term}(a) = 3\} \\ &\quad \cup \{(v_1, v_2), (v_4, v_3)\}. \end{aligned}$$

This definition of A_o and A_e is illustrated in Figure 11(b). An example is given in Figure 12(b). It is easy to see that A_o and A_e do not contain any edge oriented both ways, hence they are oriented subgraphs of G. Observe also that B_o is contained in $A_o \cup \{(v_2, v_1), (v_3, v_4)\}$, and B_e is contained in $A_e \cup \{(v_1, v_4), (v_2, v_3)\}$. The situation is represented in Figure 13.

Remark 2.3. It is easy to see that an inner arc a of G is in A_o if and only if $2 \in [\text{right-init}(a) : \text{right-term}(a)[$ or $4 \in [\text{left-term}(a) : \text{left-init}(a)[$. This is to be compared to the definition of B_e: an inner arc a is in B_e if and only if $2 \in [\text{right-term}(a) : \text{left-term}(a)[$ or $4 \in [\text{left-init}(a) : \text{right-init}(a)[$. Similarly, a is in A_e if $1 \in [\text{right-init}(a) : \text{right-term}(a)[$ or $3 \in [\text{left-term}(a) : \text{left-init}(a)[$, while it is in B_o if $1 \in [\text{left-init}(a) : \text{right-init}(a)[$ or $3 \in [\text{right-term}(a) : \text{left-term}(a)[$. We will comment further on this similarity in Section 2.4.

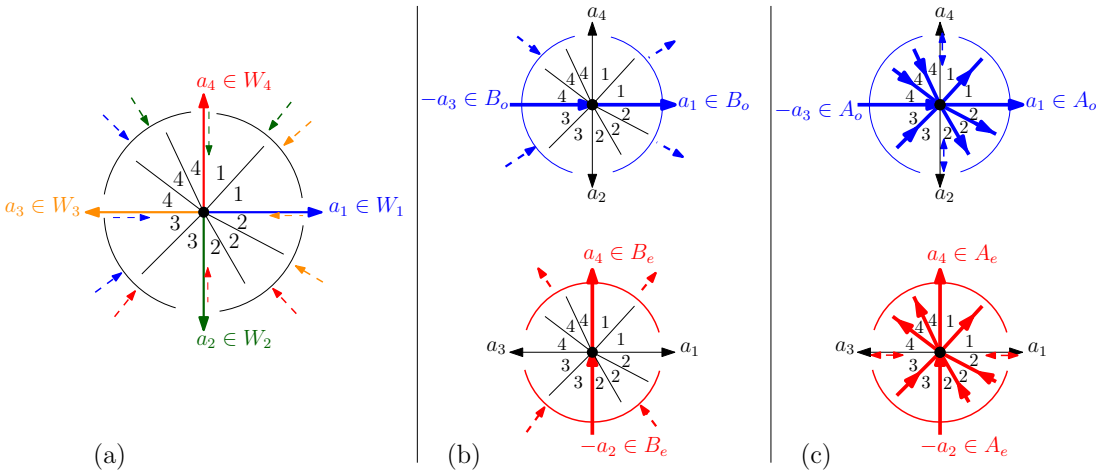


FIGURE 13. (a) The arcs of $W_1, \ldots, W_4$ around a vertex v, superimposed with the corner labels around v. The arc a_i is the outgoing arc of W_i at v. The arcs $a_1, \ldots, a_4$ appear on this clockwise order around v, but we may have $a_i = a_{i+1}$. The possible incoming arcs belonging to the edges between a_i and a_{i+1} are indicated: the edges can bear incoming arcs from W_{i+2} or W_{i+3} (or both, or neither). Meanwhile the arc opposite to a_i may belong to W_{i+1} (b) The arcs of B_o and B_e around v. For instance the incoming arcs of B_o at v must appear strictly between a_2 and a_4 in clockwise order around v. (c) The arcs of A_o and A_e around v.

Let $\overline{A_o}$ (resp. $\overline{A_e}$) be the set of edges of G which are not in the support of A_o (resp. A_e). The set $\overline{A_o}$ contains the outer edges $\{v_1, v_2\}$ and $\{v_3, v_4\}$ together with some inner edges, and it is easy to see that these inner edges are exactly those supporting the arcs a satisfying

$$\text{left-init}(a) = 4 \text{ and } \text{right-init}(a) = 1 \text{ and } \text{right-term}(a) = 2 \text{ and } \text{left-term}(a) = 3.$$

Furthermore it is easy to see that $\overline{A_o}$ does not contain any cycle. Indeed, if there was such a cycle in $\overline{A_o}$, then it would support a cycle of W_4, which is impossible since W_4 is a tree. Similarly $\overline{A_e}$ has no cycle, hence both $\overline{A_o}$ and $\overline{A_e}$ are forests. One can therefore consider the oriented planar map $\widetilde{A}_o = A_o/\overline{A_o}$ (resp. $\widetilde{A}_e = A_e/\overline{A_e}$), that is, the oriented map obtained from G by orienting the edges in A_o (resp. A_e) and contracting the edges in $\overline{A_o}$ (resp. $\overline{A_e}$).

Lemma 2.4. *Let G be a 3,4-angulation of the square endowed with a 4-GS labeling. Let $B_o, B_e, A_o, A_e, \widetilde{A}_o, \widetilde{A}_e$ be the oriented planar maps defined as above. Then $\widetilde{A}_o$ and $\widetilde{A}_e$ are bipolar orientations. Consequently, A_o and A_e are acyclic orientations, and B_o, B_e are bipolar orientations.*

Proof. We start by showing that $\widetilde{A}_o$ is acyclic. Suppose the contrary. For each simple directed cycle of $\widetilde{A}_o$ we consider the region of the plane it encloses. Let C be a directed cycle of $\widetilde{A}_o$ whose enclosed region is minimal (not containing a smaller such region). By minimality of C, there is no directed path in $\widetilde{A}_o$ starting and ending on C and staying strictly in the region enclosed by C. Since every inner vertex of $\widetilde{A}_o$ is incident to at least one incoming and at least one outgoing arc, the region enclosed by C does not contain any arc (otherwise, one could extend this arc into a directed path in both direction, so as to construct a directed path starting and ending on C and staying in the region enclosed by C). Hence the arcs of C are all incident to a single inner face of G. However, it is easy to check that for any inner face f of G, the set of arcs of $\widetilde{A}_o$ incident to f does not form a directed cycle. Indeed any inner face f of G is incident to at least one arc of A_o oriented clockwise around f (since f has a corner labeled 2 or a corner labeled 3, or both) and to at least one arc of A_o oriented counterclockwise around f (since f has a corner labeled 1 or a corner labeled 4, or both). This gives a contradiction and proves that $\widetilde{A}_o$ is acyclic. Furthermore, we claim the the unique source (resp. sink) of $\widetilde{A}_o$ is the vertex resulting from the contraction of the outer edge $\{v_1, v_2\}$ (resp. $\{v_3, v_4\}$). Indeed any inner vertex of G is incident to at least one incoming arc and one outgoing arc (since the inner arcs of $B_o = W_1 \cup W_3^-$ are in A_o), hence this also holds for any inner vertex of $\widetilde{A}_o$. Hence $\widetilde{A}_o$ is a bipolar orientation.

Since $\widetilde{A}_o$ is acyclic, the orientation A_o is also acyclic (because if A_o contained a directed cycle, then its contraction $\widetilde{A}_o$ would also contain one). In fact $A_o \cup \{(v_2, v_1), (v_3, v_4)\}$ is also acyclic (for the same reason). Hence the orientation B_o, which is contained in $A_o \cup \{(v_2, v_1), (v_3, v_4)\}$, is also acyclic. Furthermore B_o is bipolar since its unique sink (resp. source) is the outer vertex v_2 (resp. v_4).

Similarly, $\widetilde{A}_e$ is bipolar, A_e is acyclic, and B_e is bipolar (all the definitions coincide with the odd case up to increasing the indices by 1 modulo 4). □

2.4. **Dual Grand-Schnyder structures.** We now recall from [BFL24] some basic facts about *dual 4-GS structures*.

Let G be a 3,4-angulation of the square. Let G^* be the rooted 3,4-map dual to G, and let v_∞ be its root vertex. The edges incident to v_∞ are denoted by $e_1^*, \dots, e_4^*$, where e_i^* is the edge of G^* dual to the outer edge $\{v_i, v_{i+1}\}$ of G. The edges $e_1^*, \dots, e_4^*$ are called *root edges*, and all the other edges are called *non-root edges*. The faces $f_1^*, \dots, f_4^*$ in counterclockwise order around v_∞ are called *root-faces*, where f_i^* is dual to the outer vertex v_i of G. All the other faces are called *non-root faces*.

Definition 2.5. *Let G^* be a rooted 3,4-map. A* dual 4-GS labeling *of G^* is a labeling of its non-root corners in $\{1,2,3,4\}$ satisfying the following conditions.*

- *(L0*) For all $i \in \{1,2,3,4\}$, the corners incident to the root-face f_i^* have label i.*
- *(L1*) Around every non-root face or non-root vertex, the sum of clockwise jumps is 4.*
- *(L2*) Consecutive corners around a vertex have distinct labels.*
- *(L3*) Let a be an non-root arc whose initial vertex has degree 3. Let c, c', c'' be the left-initial, right initial and right terminal corners of e. The label jump δ from c to c' and the label jump δ from c' to c'' satisfy $\delta + \epsilon \geq 2$.*

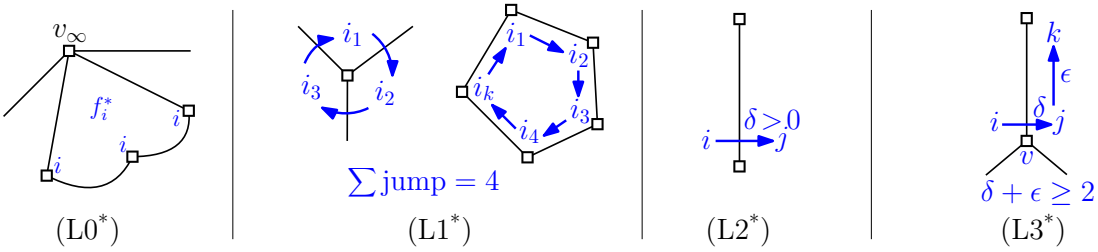


FIGURE 14. Conditions defining dual 4-GS labelings.

The conditions defining dual 4-GS labelings are illustrated in Figure 14. Observe that if G is a 3,4-angulation, then a 4-GS labeling of G induces a dual 4-GS labeling of G^* as represented in Figure 15(a), since Conditions (L0*)–(L3*) are dual to Conditions (L0)–(L3).

Given a dual 4-GS labeling $\mathcal{L}^*$ one can define some subsets of arcs $W_1^*, ..., W_4^*$ using a similar rule as in Section 2.2: for all $k \in \{1,2,3,4\}$,

$$W_k^* = \{a \text{ arc of } G^* \mid k \in [\text{left-init}(a), \text{right-init}(a)[\}.$$

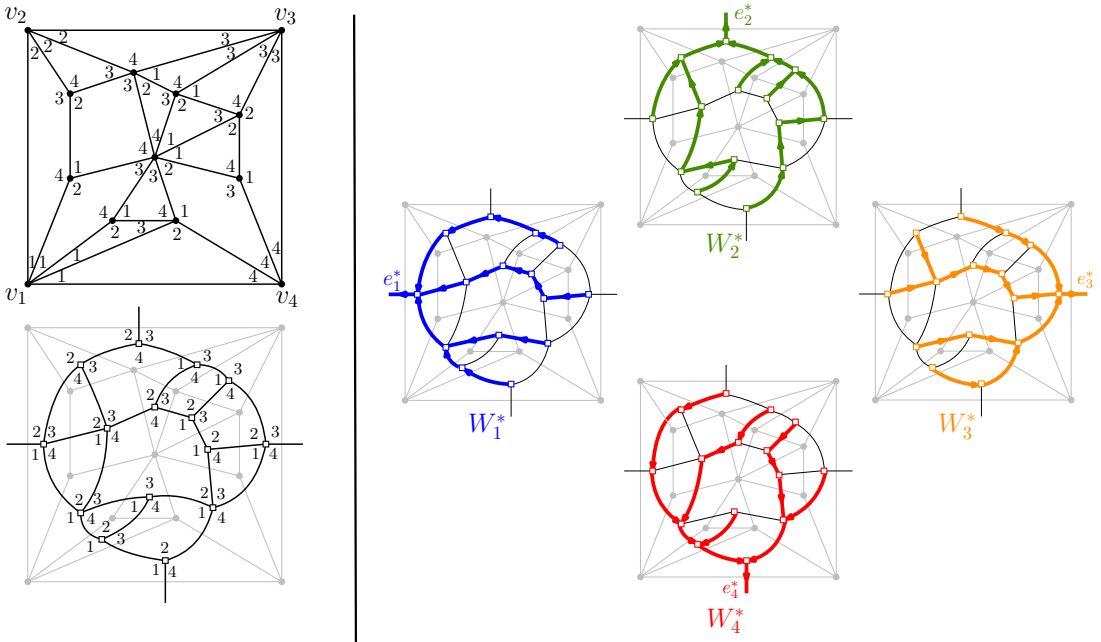
An example is represented in Figure 15(b). It was shown in [BFL24] that for all $k \in \{1,2,3,4\}$, the oriented subgraph W_k^* is a spanning tree oriented toward the root vertex v_∞ and containing the root-edge e_k^*. The tuple $\mathcal{W}^* = (W_1^*, ..., W_4^*)$ is called a *dual 4-GS wood*.

Given a dual 4-GS wood $\mathcal{W}^* = (W_1^*, ..., W_4^*)$, we define some bipolar orientations and acyclic orientations. The *odd* and *even dual bipolar orientations* are defined as

$$B_o^* = W_3^* \cup (W_1^*)^- \quad \text{and} \quad B_e^* = W_2^* \cup (W_4^*)^-.$$

An example is represented in Figure 16(a).

As we now explain, the orientations B_o^* and B_e^* are actually the dual of the bipolar contractions $\widetilde{A}_e$ and $\widetilde{A}_o$ defined in Section 2.2. This is represented in Figure 17. For an arc a of a planar map G, the *dual arc* a^* is the arc of G^* crossing a from left to right. The *dual* of an oriented planar map A is the oriented planar map consisting of all the dual arcs. Let $\mathcal{L}$ be a 4-GS labeling of a 3,4-angulation of the square G, and let $\mathcal{L}^*$ be the corresponding dual 4-GS labeling of G^*. It is easy to see, considering Remark 2.3, that the arcs in B_o^* (resp. B_e^*) are exactly the dual of the bipolar contractions $\widetilde{A}_e$ (resp. $\widetilde{A}_o^-$). Hence B_o^* (resp. B_e^*) is the dual of the oriented map $\widetilde{A}_e$ (resp. $\widetilde{A}_o^-$). The fact that $\widetilde{A}_o$ and $\widetilde{A}_e$ are bipolar orientations implies that B_o^*, B_e^* (or rather the orientation obtained from B_o^*, B_e^* by disconnecting the root-edges at the root vertex) are also bipolar orientations, as a consequence of duality for bipolar orientations [DFOdMR95].



(a) Primal and dual labelings (b) Dual wood

FIGURE 15. (a) A 4-GS labeling $\mathcal{L}$ (top) and the dual labeling $\mathcal{L}^*$ (bottom). (b) The corresponding dual 4-GS wood $\mathcal{W}^* = (W_1^*, W_2^*, W_3^*, W_4^*)$.

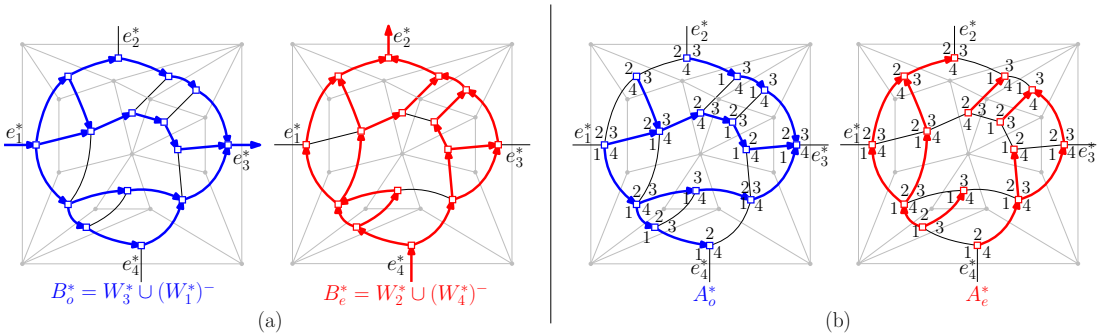


FIGURE 16. (a) Odd and even dual bipolar orientations. (b) Odd and even dual acyclic orientations, which will be defined in Section 4.2.

3. Straight-line drawing of planar maps with faces-degree at most 4

In this section we define and analyze straight-line drawing algorithms for adapted 3,4-angulations of the square. We first define a "face-counting" algorithm generalizing the algorithms from [Fus09, BH12], and then a more general (hence more compact) "increasing functions" algorithm.

3.1. Face-counting drawing algorithm for 3,4-angulations.

In this subsection we define a straight-line drawing algorithm for 3,4-angulations of the square based on 4-GS structures.

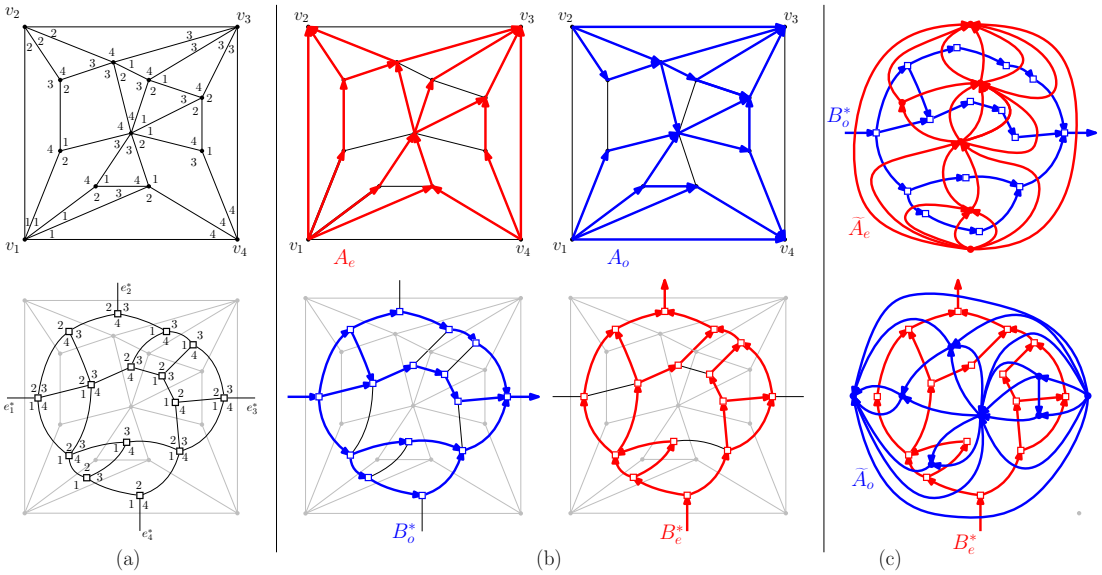


FIGURE 17. (a) A 4-GS labeling $\mathcal{L}$ (top) and its dual labeling $\mathcal{L}^*$ (bottom). (b) The acyclic orientations A_e and A_o (top) and the dual bipolar orientations B_o^* and B_e^*. (c) Duality relation between $\widetilde{A}_e$ and B_o^* (top) and between $\widetilde{A}_o^-$ and B_e^* (bottom).

Let G be an adapted 3,4-angulation of the square. Let $\mathcal{L}$ be a 4-GS labeling of G, and let W_1, W_2, W_3, W_4 be the corresponding 4-GS trees (which are rooted at v_2, v_3, v_4, v_1 respectively). Recall from Section 2.3 that the *odd and even bipolar orientations* associated to $\mathcal{L}$ are defined as $B_o = W_1 \cup W_3^-$ (which has source v_2 and sink v_4) and $B_e = W_4 \cup W_2^-$ (which has source v_1 and sink v_3). For each vertex v of G, and for $i \in \{1,2,3,4\}$ we define the path $P_i(v)$ as the directed path from v to the root v_{i-1} in the rooted tree W_i. We also define the *odd and even paths* for v as

$$P_o(v) = P_1(v) \cup P_3(v)^- \subset B_o \text{ and } P_e(v) = P_4(v) \cup P_2(v)^- \subset B_e,$$

where $P_i(v)^-$ is the directed path of G obtained by reversing $P_i(v)$. Note that $P_o(v)$ is a directed path from v_2 to v_4 in the bipolar orientations B_o, hence this path is simple (does not contain any cycle). Similarly, $P_e(v) \subset B_e$ is a simple directed path from v_1 to v_3.

We define the *face-counting coordinates* of v as $(\ell_e(v), r_o(v))$, where $r_o(v)$ is the number of inner faces of B_o on the right of the directed path $P_o(v)$, and $\ell_e(v)$ is the number of inner faces of B_e on the left of the directed path $P_e(v)$. The face-counting coordinates are represented in Figure 18(a).

Theorem 3.1. *Let G be an adapted 3,4-angulation of the square and let $\mathcal{L}$ be a 4-GS labeling of G. Placing each vertex v of G at its face-counting coordinates $(\ell_e(v), r_o(v)) \in \mathbb{R}^2$, and drawing each edge of G as a segment, yields a planar straight-line drawing of G.*

We will prove Theorem 3.1 in Section 3.3 as a corollary of a more general result.

We now compare the face counting algorithm of Theorem 3.1 to the algorithms from [Fus09] and [BH12] described in the introduction. We start with the algorithm from [Fus09]. Let G be an irreducible triangulation of the square. As explained in Section 2.2, there is a natural bijection between the transversal structures of G and the 4-GS labelings of G, which is given by

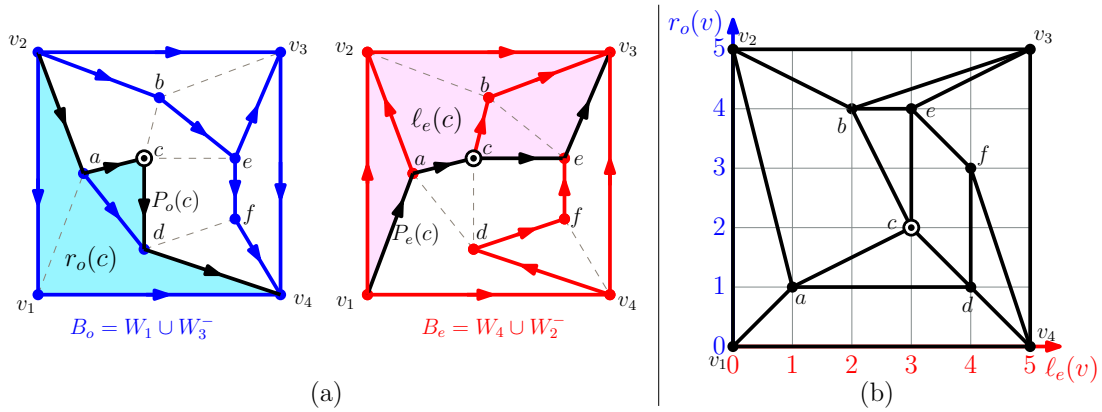


FIGURE 18. (a) The paths and regions used to compute the face-counting coordinates of a vertex. For the vertex c we get $\ell_e(c) = 3$ and $r_o(c) = 2$. (b) The drawing of G obtained by the face-counting algorithm.

the labeling rule of Figure 6(left). Moreover it is clear from the definitions that for any inner vertex v, the odd path $P_o(v)$ of $\mathcal{L}$ is equal to the blue separating path $P_{\text{blue}}(v)$ of $\mathcal{T}$, and the even path $P_e(v)$ of $\mathcal{L}$ is equal to the red separating path $P_{\text{red}}(v)$ of $\mathcal{T}$. Note also that the odd bipolar orientation $B_o = W_1 \cup W_3^-$ of $\mathcal{L}$ is contained in the blue map G_b of $\mathcal{T}$, and the even bipolar orientation $B_e = W_4 \cup W_2^-$ is contained in the red map G_r. Precisely, B_o is made of the edges of the blue map G_b which are on a blue separating path (for some inner vertex), while B_e is made of the edges of the red map G_r which are on a red separating path. Therefore, when applying the face-counting algorithm of Section 3.1 to an irreducible triangulation, one gets the x-coordinate (resp. y-coordinate) of each inner vertex v by counting the faces at the left of the red separating path $P_{\text{red}}(v)$ (resp. right of the blue separating path $P_{\text{blue}}(v)$) in the submap B_e of G_r (resp. submap B_o of G_b). Hence our algorithm gives an optimization of the algorithm based on transversal structures presented in the introduction. This optimization is precisely the one given in [Fus09, Section 5.2] (compact version of the algorithm where every grid-line is occupied by at least one vertex).

Next we discuss the face-counting algorithm from [BH12] based on separating decompositions. Let G be a simple quadrangulation. As explained in Section 2.2, there is a natural bijection between the separating decompositions of G and the even 4-GS labelings of G, which is given by the labeling rule of Figure 6(right). Let $\mathcal{S}$ be a separating decomposition, and let $\mathcal{L}$ be the associated 4-GS labeling. It is clear from the definitions that the blue and orange trees T_b and T_o of $\mathcal{S}$ are equal to the trees W_1 and W_3 of $\mathcal{L}$. Hence every edge of G belongs to the odd bipolar orientation $B_o = W_1 \cup W_3^-$. Similarly, every edge of G belongs to the even bipolar orientation $B_e = W_4 \cup W_2^-$. Note also that, for any inner vertex v and any $i \in \{1, 2, 3, 4\}$, the path $P_i(v)$ defined from the separating decomposition $\mathcal{S}$ is equal to the path from v to v_i in the tree W_i. Therefore, the separating paths $P_o(v)$ and $P_e(v)$ defined from $\mathcal{S}$ are equal to the odd and even paths defined from $\mathcal{L}$, and the placement of vertices by the face-counting algorithm in [BH12] based on $\mathcal{S}$ coincides with the placement from the face-counting algorithm based on $\mathcal{L}$.

In conclusion, the straight-line drawing algorithm from Theorem 3.1 interpolates between the compact version of the algorithm from [Fus09], and the face-counting algorithm from [BH12].

Remark 3.2. It is shown in [BH12] that the face-counting algorithm can be supplemented by a compaction procedure that significantly reduces the grid-size. It is an open question to

determine how to best extend this compaction procedure to the more general setting of the present article.

3.2. **Increasing function drawing algorithm for 3,4-angulations.** In this section we present a drawing algorithm for adapted 3,4-angulations of the square which generalizes the one presented in the previous subsection.

Let G be an adapted 3,4-angulation of the square, and let $\mathcal{L}$ be a 4-GS labeling of G. Recall from Section 2.3 the definition of the *odd and even acyclic orientations* A_o and A_e of G. By Lemma 2.4 the oriented planar maps $\widetilde{A}_o = A_o/\overline{A_o}$ and $\widetilde{A}_e = A_e/\overline{A_e}$ are bipolar orientations. We call them the *odd and even bipolar contractions* of G. Recall that the set of edges $\overline{A_o}$ and $\overline{A_e}$ are forests of G (i.e. acyclic subgraphs). Hence the faces of $\widetilde{A}_o$ (resp. $\widetilde{A}_e$) are in 1-to-1 correspondence with the faces of G. We will now triangulate some of the faces of $\widetilde{A}_o$ and $\widetilde{A}_e$ of degree 4, in order to construct the "augmented" bipolar orientations $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$. This is represented in Figure 19.

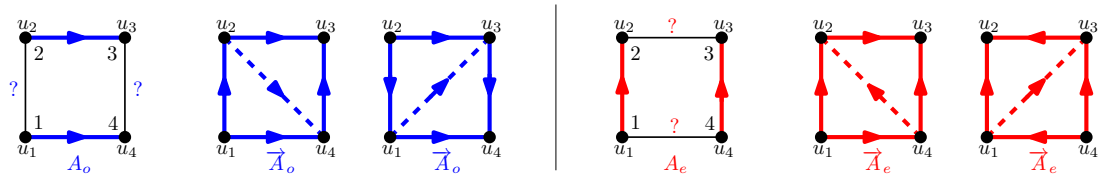


FIGURE 19. Adding diagonal arcs to some of the quadrangular faces of A_o (left) and A_e (right) in order to construct the augmented bipolar contractions $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$.

Let f be a face of G of degree 4, and let u_1, u_2, u_3, u_4 be the vertices of G incident to the corners of f labeled $1, 2, 3, 4$. By definition of A_o (see, for example, Figure 11(b)), the two arcs (u_2, u_3) and (u_1, u_4) are in A_o; The situation is illustrated in Figure 19(left).

- If the arcs (u_1, u_2) and (u_4, u_3) are in A_o, then we call (u_2, u_4) an *odd diagonal arc* of G.
- If the arcs (u_2, u_1) and (u_3, u_4) are in A_o, then we call (u_1, u_3) an *odd diagonal arc* of G.

The *augmented odd bipolar contraction* is the oriented map $\overrightarrow{A}_o$ obtained from A_o by adding all the odd diagonal arcs and then contracting the edges in $\overline{A_o}$. Note that $\overrightarrow{A}_o$ is obtained from $\widetilde{A}_o$ by adding diagonal arcs in some of its faces of degree 4 as represented in Figure 19(left).

Similarly, the two arcs (u_1, u_2) and (u_4, u_3) around the face f are in A_e; see Figure 19(right).

- If the arcs (u_1, u_4) and (u_2, u_3) are in A_e, then we call (u_4, u_2) an *even diagonal arc* of G.
- If the arcs (u_4, u_1) and (u_3, u_2) are in A_e, then we call (u_1, u_3) an *even diagonal arc* of G.

The *augmented even bipolar contraction* is the oriented map $\overrightarrow{A}_e$ obtained from A_e by adding all the even diagonal arcs and contracting the edges in $\overline{A_e}$. Note that $\overrightarrow{A}_e$ is obtained from $\widetilde{A}_e$ by adding diagonal arcs in some of its faces of degree 4 as represented in Figure 19(right). As explained in Remark 3.6 below, the oriented maps $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ are indeed bipolar orientations.

Let V be the vertex set of G. An $\overrightarrow{A}_o$*-increasing* function for G is a function $h : V \to \mathbb{R}$ that satisfies the following 2 conditions:

- $h(u) = h(v)$ for every edge $\{u, v\}$ in $\overline{A_o}$,
- $h(u) < h(v)$ for every arc (u, v) in A_o and every odd diagonal arc (u, v)

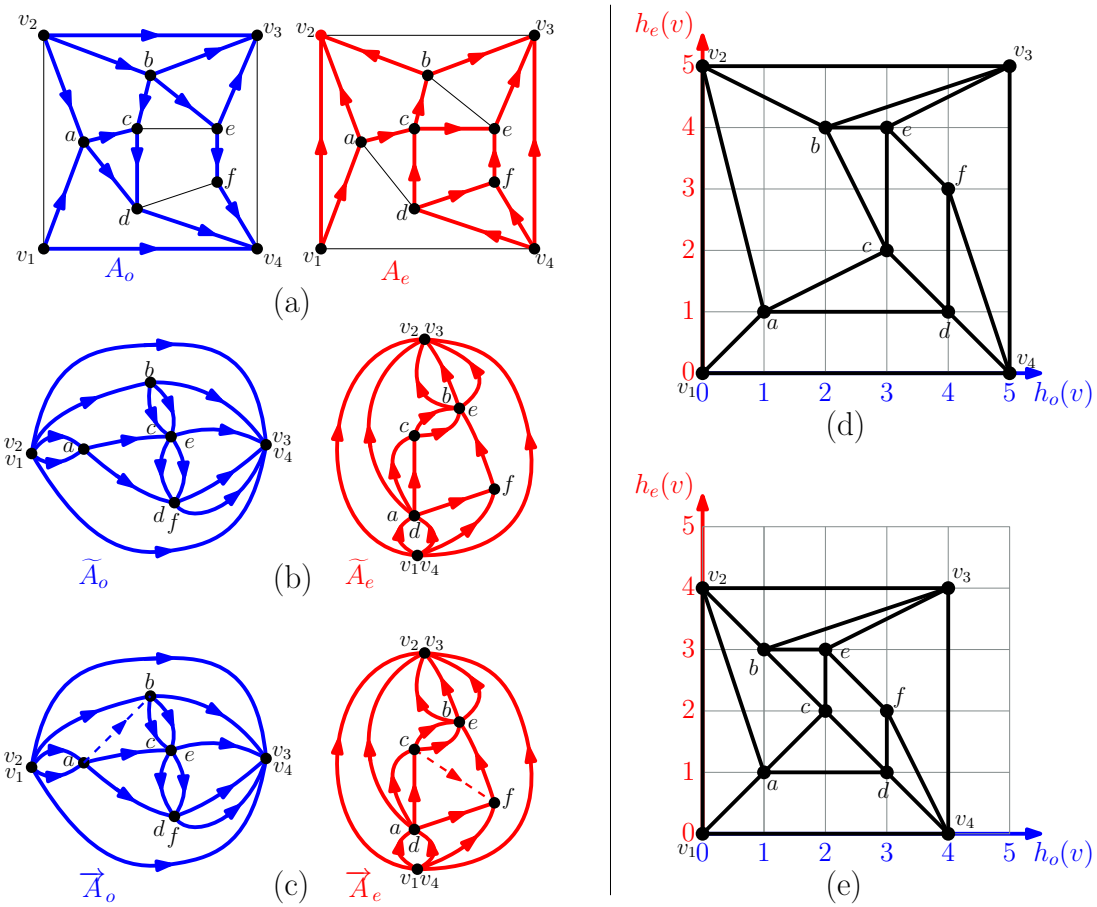


FIGURE 20. (a) The acyclic orientations. (b) The bipolar contractions. (c) The augmented bipolar contractions. (d) The drawing obtained by choosing some increasing function h_o and h_e. (e) The drawing obtained by choosing quasi-increasing functions (as defined in Section 3.6) h_o and h_e.

Such functions can be identified with functions defined on the vertices of $\overrightarrow{A}_o$ which are strictly increasing along each arc of $\overrightarrow{A}_o$. Similarly, an $\overrightarrow{A}_e$*-increasing* function for G is a function $h : V \to \mathbb{R}$ such that $h(u) = h(v)$ for every edge $\{u, v\}$ in $\overline{A_e}$, and $h(u) < h(v)$ for every arc (u, v) in A_e and every even diagonal arc (u, v).

Theorem 3.3. *Let G be an adapted 3,4-angulation of the square, let $\mathcal{L}$ be a 4-GS labeling, and let $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ be the corresponding augmented bipolar contractions of G. Let h_o be an $\overrightarrow{A}_o$-increasing function, and let h_e be an $\overrightarrow{A}_e$-increasing function. Placing each vertex v of G at coordinates $(h_o(v), h_e(v)) \in \mathbb{R}^2$, and drawing each edge of G as a segment, yields a planar straight-line drawing of G.*

Example 3.4. Theorem 3.3 is illustrated in Figure 20(d). For the example in that figure, an $\overrightarrow{A}_o$-increasing function is any function h_o defined on $V = \{v_1, v_2, v_3, v_4, a, b, c, d, e, f\}$ such that

$$h_o(v_1) = h_o(v_2) < h_o(a) < h_o(b) < h_o(c) = h_o(e) < h_o(d) = h_o(f) < h_o(v_3) = h_o(v_4),$$

and an $\overrightarrow{A}_e$-increasing function is any function $h_e : V \to \mathbb{R}$ such that

$$h_e(v_1) = h_e(v_4) < h_e(a) = h_e(d) < h_e(c) < h_e(f) < h_e(b) = h_e(e) < h_e(v_2) = h_e(v_3).$$

Such functions h_o, h_e are used in Figure 20(d).

The proof of Theorem 3.3 will be given in Section 3.3. We will also give a slightly stronger result in Section 3.6, which allows for more compact grid drawings such as the one shown in Figure 20(e).

As we now explain, Theorem 3.3 implies Theorem 3.1 about the face-counting algorithm. Indeed we will prove the following result.

Lemma 3.5. *Let G be a 3,4-angulation of the square endowed with a 4-GS labeling. Let $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ be the augmented bipolar contractions of G. The face-counting coordinates $(\ell_e(v), r_o(v))$ defined in Section 3.1 are such that the function ℓ_e is $\overrightarrow{A}_o$-increasing and the function r_o is $\overrightarrow{A}_e$-increasing.*

Remark 3.6. Lemma 3.5 shows that there exist some $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$-increasing functions. This shows that $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ are acyclic orientations. It is also clear that these orientations have a unique source and unique sink (since this is the case for $\widetilde{A}_o$ and $\widetilde{A}_e$), hence $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ are bipolar orientations. This could also have been shown by the same arguments as in the proof of Lemma 2.4, upon observing that the contours of the new faces of degree 3 (created by the insertion of diagonals) are acyclic.

Proof of Lemma 3.5. Let us prove that the face-counting function ℓ_e is $\overrightarrow{A}_o$-increasing. There are three facts to prove:
(a) if $\{u, v\}$ is an edge in $\overline{A_o}$, then $\ell_e(u) = \ell_e(v)$,
(b) if (u, v) is an arc in A_o, then $\ell_e(u) < \ell_e(v)$,
(c) if (u, v) is a diagonal arc of $\overrightarrow{A}_o$, then $\ell_e(u) < \ell_e(v)$.
Recall that $\ell_e(v)$ is defined as the number of faces of the bipolar orientation B_e on the left of the path $P_e(v) = P_4(v) \cup P_2(v)^-$.

We start by proving (a). The statement is clear for outer edges, so we now consider an inner edge $e = \{u, v\} \in \overline{A_o}$. Recall that the corners incident to e are labeled as indicated in Figure 11(b), so we have $(u, v) \in W_4$ and $(v, u) \in W_2$. In this case the paths $P_e(u)$ and $P_e(v)$ are equal, hence $\ell_e(u) = \ell_e(v)$.

Next, we prove (b). The statement is clear for outer arcs, so we now consider an inner arc $a = (u, v)$ in A_o. The condition satisfied by the labels of the corners incident to the arc a is indicated in Figure 11(b). Let us first suppose that left-init$(a) = 1$. In this case left-term$(a) \in \{3, 4\}$. We contend that the path $P_e(u)$ stays (weakly) to the left of $P_e(v)$. The situation is represented in Figure 21(a). Consider the path $P = P_2(u)^- \cup \{a\} \cup P_4(v)$. The label of the corners around a implies that the first arc of $P_4(u)$ is strictly to the left of P as indicated in Figure 21(a). Moreover, $P_4(u)$ cannot cross P. Indeed, $P_4(u)$ and $P_2(u)$ have no vertex in common except u (otherwise there would be a directed cycle in the bipolar orientation B_e); and if $P_4(u)$ and $P_4(v)$ have a common vertex then they merge after that vertex. Hence $P_4(u)$ stays on the left of P. Thus $P_e(u)$ stays to the left of P. Similarly, $P_2(v)$ starts weakly to the right of P, hence it stays weakly to the right of P. Thus $P_e(v)$ stays to the right of P. In conclusion $P_e(u)$ is to the left of P which is to the left of $P_e(v)$. This gives the desired inequality $\ell_e(u) \leq \ell_e(v)$ for the number of faces on the left of $P_e(u)$ and $P_e(v)$. The inequality is actually strict since $P_e(u) \neq P$. This completes the proof for the case left-init$(a) = 1$.

A similar proof can be given in the cases right-init$(a) = 2$, or left-term$(a) = 3$, or right-term$(a) = 4$ which are represented in Figure 21(b). Indeed in all these cases the label condition implies that

a is on the right of $P_e(u)$ and on the left of $P_e(v)$, which is enough to prove that $\ell_e(u) < \ell_e(v)$ by reasoning as above.

Lastly we prove (c). There are two types of diagonal arcs, as represented in Figure 21(c). In both cases, the diagonal $a = (u, v)$ is to the right of $P_e(u)$ and to the left of $P_e(v)$. Hence, as before we can see that the path $P = P_2(u)^- \cup \{a\} \cup P_4(v)$ is at the right of $P_e(u)$ and at the left of $P_e(v)$, which implies $\ell_e(u) < \ell_e(v)$ as desired. This completes the proof that the function ℓ_e is $\overrightarrow{A}_o$-increasing.

The proof that the function r_o is $\overrightarrow{A}_e$-increasing is almost identical and left to the reader. □

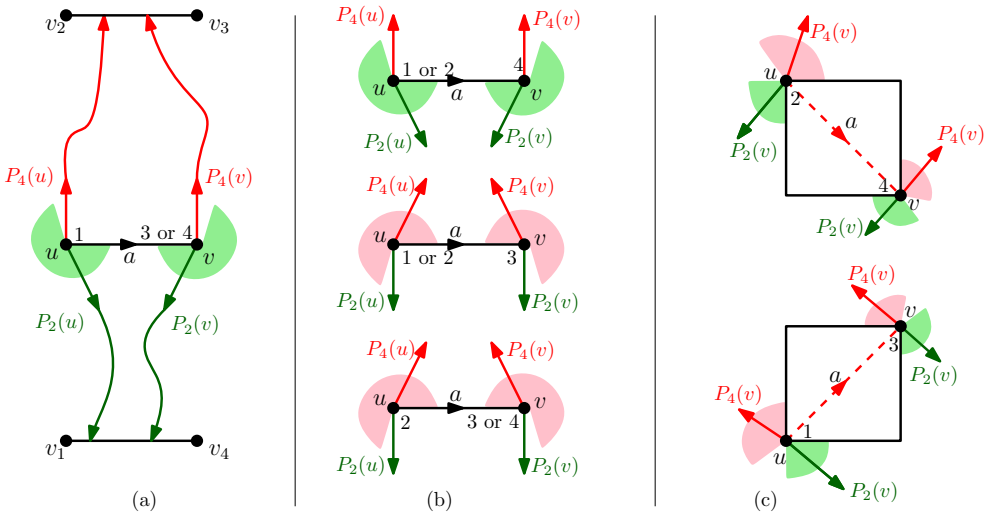


FIGURE 21. Proof of Lemma 3.5. (a) An arc $a = (u, v)$ such that left-init$(a) = 1$. The clockwise order of a, $P_2(u)$ and $P_4(u)$ around u is indicated, with the range for position of the first arc of $P_2(u)$ indicated by a green sector. Similarly the clockwise order of a, $P_2(v)$ and $P_4(v)$ around v is indicated. (b) Local clockwise order of the relevant paths around an arc $a = (u, v)$ satisfying either left-term$(a) = 4$ (top) right-init(a) (middle) or right-term$(a) = 3$ (bottom). (c) Local clockwise order of the relevant paths around a diagonal arc $a = (u, v)$.

3.3. **Proof of planarity.** In this subsection we prove Theorem 3.3. We will rely on a general planarity criterion that we now state. It is well-known and appears in different forms in the literature, for instance as Lemma 4.4 in [dV03], with references. We provide a short proof for completeness.

Let G be a 2-connected planar map (for instance, an adapted 3,4-angulation of the square). Note that the contour of every face of G is a simple cycle. A *piecewise-linear drawing* of a graph G is a drawing of G in the plane where every edge is represented as a (non-empty) self-avoiding concatenation of segments. The drawing is called *valid* for a face f of G if the curve corresponding to the clockwise contour[3] of f is a polygonal simple curve in the plane whose interior is on its right.

[3] *Clockwise* means that the contour is traversed with f on its right if f is an inner face, and with f on its left if f is the outer face.

Lemma 3.7. *Let G be a 2-connected planar map. Consider a piecewise-linear drawing of the graph underlying G. If the drawing is valid for every face of G (including the outer face), then the drawing is planar (and corresponds to the planar map G).*

Proof. Consider a piecewise-linear drawing of a 2-connected planar map G, which is valid for every face. Let F be the set of faces of G. For a face $f \in F$, let P_f be the polygon representing f in the drawing, and let Σ_f be the set of points corresponding to the vertices incident to f. If f is an inner face (resp. is the outer face), we define $N_f : \mathbb{R}^2 \backslash \Sigma_f \to \{0, \frac{1}{2}, 1\}$ to be the indicator function for f, assigning value 1 to the points in the interior (resp. exterior) of P_f, value $1/2$ to the points on the boundary of P_f (excluding Σ_f), and value 0 to the points in the exterior (resp. interior) of P_f. Let Σ be the set of points in the plane that are images of vertices of G, and let N be the function on $\mathbb{R}^2 \backslash \Sigma$ defined as $N := \sum_{f \in F} N_f$. As the drawing is valid for every face, the function N is locally constant, hence it is constant. Since it is equal to 1 for a point in the drawing of the outer face far enough so that it avoids the polygons representing inner faces, it is constant equal to 1. Moreover, at each vertex v, the fact that the drawing is valid for all faces ensures that the set of faces in clockwise order around v forms a fan that wraps k times (i.e., of total angle $2k\pi$), for a positive integer k. Since $N = 1$ in the neighborhood of v, we must have $k = 1$. Similarly, no two vertices can be at the same point, nor a vertex intersect an edge, nor two edges intersect each other (any of these situations would imply that $N \geq 2$ in a neighborhood). This proves the planarity of the drawing. □

In Section 4 we will also need the following complementary lemma:

Lemma 3.8. *Let G be a 2-connected planar map, endowed with a piecewise-linear drawing. Let $C = (C(t))_{t \in [0,1]}$ be the directed closed curve given by the drawing of the outer face contour of G traversed clockwise. A point $p = C(t)$ is called* outer-right-visible *if there is a ray starting from p to the right side of C (locally at p) and reaching to infinity without meeting C again. If the drawing is valid for every* inner *face of G, then there is no outer-right-visible point.*

Proof. Assume the drawing is valid at every inner face, and consider a ray γ starting on the right side of C at a point $p = C(t)$ and reaching to infinity. Let p' be the farthest point on γ that belongs to the drawing of an edge or vertex. Clearly, after leaving p' the ray γ cannot visit any point on an edge or vertex, nor in an area corresponding to the drawing of an inner face. Hence, p' cannot be the image of a point on an inner edge, nor the image of an inner vertex v (for this we need the validity for inner faces, which ensures that the inner faces incident to v form a fan that wraps k times around v in the drawing, for some positive integer k). Hence $p' \in C$. It only remains to show that $p' \neq p$. If p belongs to the drawing of an edge of C, then this edge is on the contour of an inner face f whose drawing is valid, hence γ starts within the interior of f and will meet the contour of f again, thus $p' \neq p$. Suppose now that p corresponds to the drawing of a vertex v of C. In this case, the drawing of the inner faces incident to v (which are all valid) form a fan that covers the right side of C at p (and possibly wraps around p several more times). Hence γ starts in the interior of an inner face, or in the drawing of an inner edge. In every case we conclude that $p' \neq p$ which completes the proof. □

Remark 3.9. The condition that the drawing is valid at every inner face is not enough to avoid self-intersections of the outer contour, as illustrated in Figure 22, where it can be seen that the drawing has no outer-right-visible point.

We now use Lemma 3.7 to prove Theorem 3.3. Let G be an adapted 3,4-angulation of the square, and let V be its vertex set. We call a function $\alpha : V \to \mathbb{R}^2$ a *vertex-placement* for G. A vertex placement α induces a *straight-line drawing* of G (in which each edge is represented by a segment). We say that α is *valid* for a face f, if the straight-line drawing of f obtained

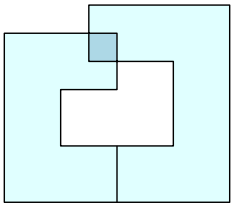


FIGURE 22. A piecewise-linear drawing of a planar map which is valid for its two inner faces but not planar: its outer contour intersects itself.

from this vertex placement is valid. Let $\mathcal{L}$ be a 4-GS labeling of G, let $h_o : V \to \mathbb{R}$ be an $\overrightarrow{A}_o$-increasing function, and let $h_e : V \to \mathbb{R}$ be an $\overrightarrow{A}_e$-increasing function. In order to prove Theorem 3.3 it suffices to prove that the vertex placement $\alpha : v \mapsto (h_o(v), h_e(v))$ is valid for every face of G.

It is clear that α is valid for the outer face of G. Indeed $\alpha(v_1) = (\min(h_o), \min(h_e))$, $\alpha(v_2) = (\min(h_o), \max(h_e))$, $\alpha(v_3) = (\max(h_o), \max(h_e))$, and $\alpha(v_4) = (\max(h_o), \min(h_e))$.

Next we show that α is valid for every inner face of degree 4. Such a face f has its corners labeled 1, 2, 3 and 4 in clockwise order in $\mathcal{L}$. Let u_1, u_2, u_3 and u_4 be the vertices incident to the corners labeled 1, 2, 3 and 4 respectively. We claim that

(1)
$$\max(h_o(u_1), h_o(u_2)) < \min(h_o(u_3), h_o(u_4)) \text{ and } \max(h_e(u_1), h_e(u_4)) < \min(h_e(u_2), h_e(u_3)).$$

Let us justify the inequality for h_o. The situation is represented in Figure 23(a). Since h_o is $\overrightarrow{A}_o$-increasing, and the arcs (u_1, u_4) and (u_2, u_3) are in A_o, we get $h_o(u_1) < h_o(u_4)$ and $h_o(u_2) < h_o(u_3)$. It remains to check the inequalities $h_o(u_1) < h_o(u_3)$ and $h_o(u_2) < h_o(u_4)$. If the edge $\{u_1, u_2\}$ belongs to $\overline{A_o}$, then $h_o(u_1) = h_o(u_2)$ hence these inequalities hold. Similarly if $\{u_3, u_4\} \in \overline{A_o}$, then the inequalities hold. So we need only consider the situation where all the edges incident to f are in the support of A_o. If either of the arcs (u_1, u_2) or (u_4, u_3) is in A_o then we get $h_o(u_1) < h_o(u_3)$ from these arcs, while if neither of these arcs are in A_o then the diagonal arc (u_1, u_3) belongs to $\overrightarrow{A}_o$ which ensures $h_o(u_1) < h_o(u_3)$; see Figure 19. Similarly, one checks $h_o(u_2) < h_o(u_4)$ in all cases. The proof of the inequality concerning h_e is similar.

By (1) there is a vertical line separating the placements of u_1, u_2 from the placement of u_3 and u_4, and an horizontal line separating the placements of u_1, u_4 from the placement of u_2 and u_3 as indicated in Figure 23(a). This easily implies that the vertex placement α is valid for the face f.

It remains to show that the placement α is valid for every inner face f of G of degree 3. There are 4 possibilities for the labels of the corners of f: it can be missing the label 1, 2, 3 or 4. Suppose for concreteness that the missing label is 4, and let u_1, u_2 and u_3 be the vertices incident to the corners of f labeled 1, 2 and 3, respectively. We claim that

$$h_o(u_1) \le h_o(u_2) < h_o(u_3) \text{ and } h_e(u_1) < h_e(u_3) \le h_e(u_2). \tag{2}$$

Let us justify the inequalities for h_o. The situation is represented in Figure 23(b). First, the arc (u_2, u_3) is in A_o (since right-term$(u_2, u_3) = 3$), hence $h_o(u_2) < h_o(u_3)$. Second, Condition (L3) of 4-GS labelings implies that the label left-term(u_1, u_2) is not equal to 2, hence this label must be ether 3 or 4. Therefore, there are only 3 possibilities for the labels at the left of the arc (u_1, u_2):

- either left-init$(u_1, u_2) = 4$ and left-term$(u_1, u_2) = 3$, in which case $\{u_1, u_2\} \in \overline{A_o}$ and $h_o(u_1) = h_o(u_2)$;

- or left-init$(u_1, u_2) = 1$ and left-term$(u_1, u_2) = 3$, in which case $(u_1, u_2) \in A_o$ and $h_o(u_1) < h_o(u_2)$;
- or left-init$(u_1, u_2) = 1$ and left-term$(u_1, u_2) = 4$, in which case $(u_1, u_2) \in A_o$ and $h_o(u_1) < h_o(u_2)$.

Hence $h_o(u_1) \leq h_o(u_2)$ in all cases. The inequalities for h_e is justified in a similar manner after observing that Condition (L3) of 4-GS labelings implies left-term$(u_2, u_3) \in \{1, 4\}$. By (2), the placement of vertices are as represented at the bottom of Figure 23(b), which clearly implies that the vertex placement α is valid for f.

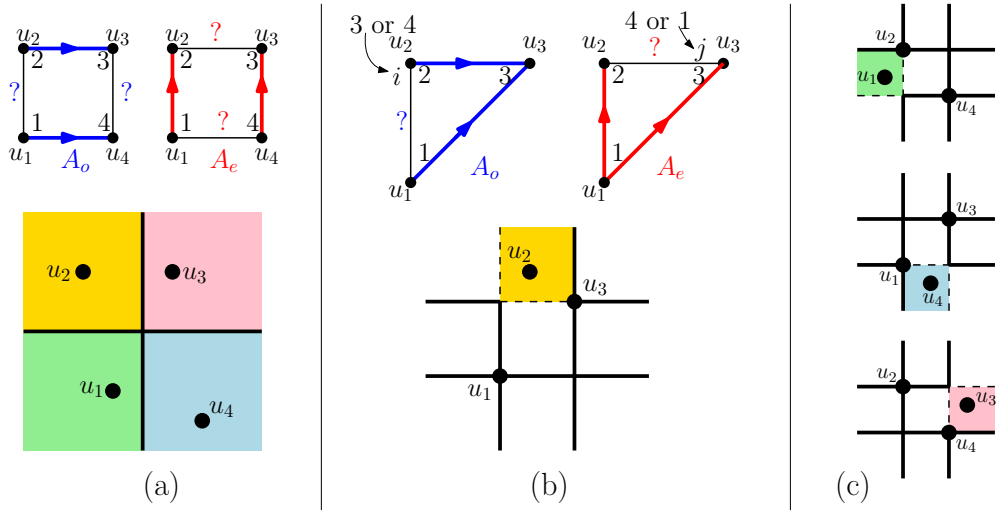


FIGURE 23. Proof of planarity. (a) Top: orientations A_o and A_e for a face f of degree 4. Bottom: Vertex placement for the face f of degree 4. (b) Top: orientations A_o and A_e for a face f of degree 3 having corner labels 1, 2 and 3. Vertex placement for the face f. The shaded region corresponds to the placement of vertex u_2 (the dotted lines indicates the weak inequalities). (c) Vertex placement for a face of degree 3 with missing corner label 3 (top), 2 (middle), or 1 (bottom).

A similar reasoning applies to the faces of degree 3 with other labeling situations as represented in Figure 23(c): compared to the case treated above the geometric constraints for the case of a face of degree 3 with missing label 3 (resp. 2, 1) is obtained by a rotation of 90 degrees (resp. 180 degrees, 270 degrees). This completes the proof that the vertex placement α is valid for all inner faces. Hence, by Lemma 3.7, α induces a planar drawing. This proves Theorem 3.3.

Remark 3.10. Any drawing of a planar map G using the increasing function algorithm of Theorem 3.3 completely determines the 4-GS labeling used to obtain the drawing. Indeed, from the above discussion on the different types of faces of G and the corresponding vertex placements, it is not hard to see that the labels of the corners inside each inner face can be determined from the placement of the vertices incident to this face.

3.4. **Time complexity of the increasing function drawing algorithm.** In this subsection we discuss the time complexity of the increasing function drawing algorithm presented in Section 3.2. We say that an algorithm for 3,4-angulations can be performed in *linear time* if it can

be done in a number of arithmetic operations which is linear in the number of vertices. Note that the numbers of edges and faces of a 3,4-angulation are linear in the number of vertices by the Euler relation combined with the face-edge incidence relation.

Let $G = (V, E)$ be an adapted 3,4-angulation of the square. We aim to draw G with vertices placed at integer coordinates. Using the increasing function drawing algorithm for this task amounts to:

(i) computing a 4-GS labeling $\mathcal{L}$, and the corresponding augmented bipolar contractions $\overrightarrow{A}_o$, $\overrightarrow{A}_e$, and

(ii) computing some integer-valued $\overrightarrow{A}_o$-increasing and $\overrightarrow{A}_e$-increasing functions h_o and h_e.

For step (ii) we aim for a small grid size, so we want the increasing function h_o, h_e to be *tight*. Precisely, an $\overrightarrow{A}_o$-increasing function h_o is called *tight* if it takes non-negative integer values and minimizes $\max(h_o)$ among such functions. Tightness for h_e is defined similarly.

Step (i) can be completed in linear time. Indeed, a linear time algorithm for finding a 4-GS labeling $\mathcal{L}$ was given in [BFL24]. Moreover it is easy to see that computing the orientations $\overrightarrow{A}_o$, $\overrightarrow{A}_e$ from $\mathcal{L}$ can be done in linear time.

We now show that Step (ii) can also be completed in linear time. Given an acyclic directed graph D, we call *source-level* of a vertex v of D the maximal length of the directed paths ending at v (so that the sources of D are at source-level 0). As we now explain, the source-level of every vertex in a digraph $D = (V_D, E_D)$ can be computed in time linear in $|V_D| + |E_D|$. Recall that the *topological ordering algorithm* on D returns a total order $\mathcal{O}$ of the vertex set V_D such that every arc of D is oriented from the smaller vertex to the greater vertex for the order $\mathcal{O}$. It is known that the topological ordering algorithm is linear in $|V_D| + |E_D|$. Once the total order $\mathcal{O}$ is fixed, one can compute the source-level of every vertex, by treating vertices in the order given by the total order $\mathcal{O}$ (from smallest to greatest): for every vertex v the source-level of v is 0 if v is a source and

$$\text{source-level}(v) = \max\{\text{source-level}(u) + 1 \mid (u, v) \in E_D\}$$

otherwise. This computation is also clearly linear in $|V_D| + |E_D|$.

Consider now the function $h_o : V \to \mathbb{N}$ defined by setting $h_o(v)$ to be the source-level of the vertex corresponding to v in $\overrightarrow{A}_o$ for all $v \in V$. It is easy to see that $h_o(v)$ is $\overrightarrow{A}_o$-increasing (recall that $\overrightarrow{A}_o$-increasing functions can be identified with functions on the vertex set of $\overrightarrow{A}_o$ which are increasing along the arcs of $\overrightarrow{A}_o$). Moreover, the increasing function h_o is tight. Indeed, any $\overrightarrow{A}_o$-increasing function $f : V \to \mathbb{N}$ satisfies

$$\forall v \in V,\ h_o(v) \leq f(v).$$

This property is easily shown by induction on the source-level of v (or rather the source-level of the vertex corresponding to v in $\overrightarrow{A}_o$). Indeed we can identify h_o and f with some functions h'_o and f' on the vertex set V' of $\overrightarrow{A}_o$, and we want to show $h'_o(v) \leq f'(v)$ for all $v \in V'$. For the source v_0 of $\overrightarrow{A}_o$, one has $h'_o(v_0) = 0 \leq f'(v_0)$. And for any vertex $v \in V'$ at source-level $k > 0$, there is an arc (u, v) of $\overrightarrow{A}_o$ such that u is at source-level $k - 1$, hence the induction hypothesis gives $h'_o(v) = h'_o(u) + 1 \leq f'(u) + 1 \leq f'(v)$, which completes the proof. Hence h_o is a tight $\overrightarrow{A}_o$-increasing function which can be computed in linear time.

Similarly the function $h_e(v) : V \to \mathbb{N}$ defined by setting $h_e(v)$ to be the source-level of the vertex corresponding to v in $\overrightarrow{A}_e$ is a tight $\overrightarrow{A}_e$-increasing function which can be computed in linear time. We summarize our findings as follows:

Proposition 3.11. *The increasing function algorithm for drawing an adapted 3,4-angulation of the square can be completed in linear time in the number of vertices.*

3.5. **Bounds on the grid size.** In this subsection we discuss the size of the grid needed for our straight-line grid-drawing algorithms.

Let G be an adapted 3,4-angulation of the square, and let $\mathcal{L}$ be a 4-GS labeling. Let h_o, h_e be the tight increasing functions computed by the increasing-function algorithm. By definition the grid required for the vertex placement has size $m_o \times m_e$, where $m_o = \max(h_o)$ and $m_e = \max(h_e)$. We want to give some upper bounds on m_o and m_e.

Since the face-counting algorithm presented in Section 3.1 produces some increasing functions ℓ_e and r_o, we get the following upper bounds for the dimension of the grid: $m_o \leq \max(\ell_e)$ and $m_e \leq \max(r_o)$. By definition, $\max(r_o)$ is the number of inner faces in the bipolar orientation B_o. Using the Euler relation, one gets $\max(r_o) = \mathbf{e}_o - \mathbf{v} + 1$, where $\mathbf{v}$ is the number of inner vertices of G, and $\mathbf{e}_o$ is the number of inner edges in $B_o = W_1 \cup W_3^-$. Next, we observe that $\mathbf{e}_o = 2\mathbf{v} - \mathbf{d}_o$, where $\mathbf{d}_o$ is the number of inner edges of G carrying both an arc in W_1 and an arc in W_3. Thus $\max(r_o) = \mathbf{v} + 1 - \mathbf{d}_o$. Similarly $\max(\ell_e) = \mathbf{v} + 1 - \mathbf{d}_e$ where $\mathbf{d}_e$ is the number of inner edges of G carrying both an arc in W_2 and an arc in W_4. We summarize:

Proposition 3.12. *Let G be an adapted 3,4-angulation of the square, and let $\mathcal{L}$ be a 4-GS labeling. The increasing function algorithm provides a grid drawing of G on a grid of size $m_o \times m_e$, with $m_o \leq \boldsymbol{v} + 1 - \boldsymbol{d}_e$ and $m_e \leq \boldsymbol{v} + 1 - \boldsymbol{d}_o$, where $\boldsymbol{d}_o$ (resp. $\boldsymbol{d}_e$) is the number of inner edges of G carrying both an arc in W_1 and an arc in W_3 (resp. an arc in W_2 and an arc in W_4).*

In the case where G is a triangulation of the square, $\mathbf{d}_e + \mathbf{d}_o$ is the number $\mathbf{e}_2$ of inner edges having 2 colors in the 4-GS woods (because each arc has at most one color for any 4-GS structure on any triangulation of the square). Let $\mathbf{e}_0, \mathbf{e}_1, \mathbf{e}_2$ be the number of inner edges having $0, 1, 2$ colors respectively, and let $\mathbf{e} = \mathbf{e}_0 + \mathbf{e}_1 + \mathbf{e}_2$ be the total number of inner edges. The Euler relation (combined with the incidence relation between faces and edges) gives $\mathbf{e} = 3\mathbf{v} + 1$, and the fact that each inner vertex is incident to one outgoing arc of each color gives $\mathbf{e}_1 + 2\mathbf{e}_2 = 4\mathbf{v}$. This implies $\mathbf{e}_2 = \mathbf{v} - 1 + \mathbf{e}_o$, and $m_o + m_e = 2\mathbf{v} + 2 - \mathbf{e}_2 = \mathbf{v} + 3 - \mathbf{e}_0 \leq \mathbf{v} + 3$. Thus, for a triangulation of the square, the half-perimeter of the grid drawing is strictly less than the number of vertices.

3.6. **Reduction of the grid size.** In this subsection we discuss some optimizations of the increasing function drawing algorithm aimed at reducing the grid size. These improvements will be obtained by relaxing some requirements related to diagonal arcs for the increasing functions h_o, h_e.

In order to state which requirements can be relaxed, we first need to classify the diagonal arcs of $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$. We refer to Figure 19 for the definition of the diagonal arcs. For a diagonal arc a of A_o or A_e, we denote by $f(a)$ the inner face of G containing a. Each of the 4 edges of G incident to the face $f(a)$ can be either *missing* if they are in $\overline{A_o} \cup \overline{A_e}$, or *converging* if they are oriented the same way in A_o and A_e, or *diverging* if they are oriented in opposite directions in A_o and A_e. We let missing(a), converging(a) and diverging(a) be the number of missing, converging and diverging edges incident to $f(a)$. Note that missing$(a) \in \{0, 1, 2\}$ and missing(a) + converging(a) + diverging$(a) = 4$.

We call all the diagonal arcs of $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ *weak*, except for the diagonal arcs a of $\overrightarrow{A}_e$ such that converging$(a) = 4$ or diverging$(a) = 4$ which are called *strict*. We call a diagonal arc a *mandatory* unless $|\text{converging}(a) - \text{diverging}(a)| = \text{missing}(a)$. It is easy to check that a diagonal arc a is non-mandatory if and only if it satisfies one of the following conditions:

(i) missing$(a) = 0$ and converging$(a) = 2$,
(ii) missing$(a) = 1$ and converging$(a) \in \{1, 2\}$,
(iii) missing$(a) = 2$.

Indeed, when missing$(a) = 0$ the condition $|\text{converging}(a) - \text{diverging}(a)| = 0$ is equivalent to converging$(a) = 2$, (since converging(a) + diverging$(a) = 4$), while when missing$(a) = 2$ one can check that converging$(a) \in \{0, 2\}$ (see Figure 19 and observe that the missing edges are either both in $\overline{A_o}$ or both in $\overline{A_e}$) hence $|\text{converging}(a) - \text{diverging}(a)| = 2$ (since converging(a) + diverging$(a) = 2$).

Note that a strict diagonal arc is always mandatory.

Example 3.13. For the diagonal arc $a \in \overrightarrow{A}_o$ on the left of Figure 20(c) one has missing$(a) = 0$, converging$(a) = 1$, and diverging$(a) = 3$, hence this arc is weak and mandatory. For the diagonal arc $b \in \overrightarrow{A}_e$ on the right of Figure 20(c) one has missing$(b) = 2$, converging$(a) = 0$, and diverging$(a) = 2$, hence this arc is weak and non-mandatory.

We say that a function $g : V \to \mathbb{R}$ is *quasi $\overrightarrow{A}_o$-increasing* (resp. *quasi $\overrightarrow{A}_e$-increasing*) if it satisfies the following 3 conditions:

- $g(u) = g(v)$ for every edge $\{u, v\}$ in $\overline{A_o}$ (resp. $\overline{A_e}$),
- $g(u) \leq g(v)$ for every weak mandatory diagonal arc (u, v) in $\overrightarrow{A}_o$ (resp. $\overrightarrow{A}_e$),
- $g(u) < g(v)$ for every arc (u, v) in A_o (resp. A_e) and every strict mandatory diagonal arc (u, v) in $\overrightarrow{A}_o$ (resp. $\overrightarrow{A}_e$).

Clearly any $\overrightarrow{A}_o$-increasing function is quasi $\overrightarrow{A}_o$-increasing (and similarly for $\overrightarrow{A}_e$). We can now state a slightly stronger version of Theorem 3.3.

Theorem 3.14. *Let G be an adapted 3,4-angulation of the square, let $\mathcal{L}$ be a 4-GS labeling, and let $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ be the corresponding augmented bipolar contractions of G. Let h_o be a quasi $\overrightarrow{A}_o$-increasing function and h_e be a quasi $\overrightarrow{A}_e$-increasing function. Placing each vertex v of G at coordinates $(h_o(v), h_e(v)) \in \mathbb{R}^2$ gives a planar straight-line drawing of G.*

Theorem 3.14 is illustrated in Figure 20(e). In that figure the vertex placement is given by some quasi increasing functions h_o and h_e. Note that in that example, the function h_o is not $\overrightarrow{A}_o$-increasing since $h_o(a) = h_o(b)$ and the function h_e is not $\overrightarrow{A}_e$-increasing since $h_e(c) = h_o(f)$.

Proof of Theorem 3.14. Our proof of Theorem 3.14 relies on the same line of arguments as for Theorem 3.3. Compared to Theorem 3.3, some requirements associated to the diagonal arcs of $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ have been relaxed, and we need to check that the proof remains valid. In the proof of Theorem 3.3, the requirements associated to a diagonal arc a for a placement function $\alpha = (h_o, h_e)$ are used solely to prove that the placement α is valid for the face $f(a)$ of G. Hence, we only need to prove the following two claims.

(i) For any non-mandatory diagonal arc $a = (u, v)$, the placement α is valid for $f(a)$ even if the requirement on h_o or h_e associated to a is dropped.

(ii) For any weak diagonal arc $a = (u, v)$, the placement α is valid for $f(a)$ even if the strict inequality requirement on h_o or h_e associated to a is replaced by a weak inequality.

Let us start by proving Claim (i). Let a be a non-mandatory arc of $\overrightarrow{A}_o$, and let u_1, u_2, u_3, u_4 be the vertices of G incident to the corners of $f(a)$ labeled 1,2,3,4 respectively (see again Figure 19). Recall that (u_2, u_3) and (u_1, u_4) are in A_o. There are two cases to check: either $a = (u_1, u_3)$ or $a = (u_2, u_4)$. Let us start by assuming that $a = (u_1, u_3)$ as in Figure 24(b). In this situation we know

$$\{(u_2, u_1), (u_1, u_4), (u_2, u_3), (u_3, u_4)\} \subseteq A_o \quad \text{and} \quad \{(u_1, u_2), (u_4, u_3)\} \subseteq A_e,$$

and since a is non-mandatory there are only four possible configurations for the orientation of $f(a)$ in $\overrightarrow{A}_e$, which are represented in Figure 24:

- $(u_1, u_4) \in A_e$ and $(u_2, u_3) \in A_e$ (this gives missing$(a) = 0$ and converging$(a) = 2$),

- or $(u_1, u_4) \in A_e$ and $\{u_2, u_3\} \in \overline{A_e}$ (this gives missing$(a) = 1$ and converging$(a) = 1$),
- or $\{u_1, u_4\} \in \overline{A_e}$ and $(u_2, u_3) \in \overrightarrow{A}_e$ (this gives missing$(a) = 1$ and converging$(a) = 1$),
- or $\{u_1, u_4\} \in \overline{A_e}$ and $\{u_2, u_3\} \in \overline{A_e}$ (this give missing$(a) = 2$ and converging$(a) = 0$).

In all four configurations, the quasi increasing functions h_o and h_e satisfy

$$h_o(u_2) < h_o(u_1), h_o(u_3) < h_o(u_4) \quad \text{and} \quad h_e(u_1) \leq h_e(u_2), h_e(u_4) \leq h_e(u_3). \tag{3}$$

These inequalities (together with the observation $h_e(u_1) < h_e(u_3)$) are enough to ensure that the placement $\alpha = (h_o, h_e)$ is valid for the face $f(a)$ as claimed; see Figure 24(c).

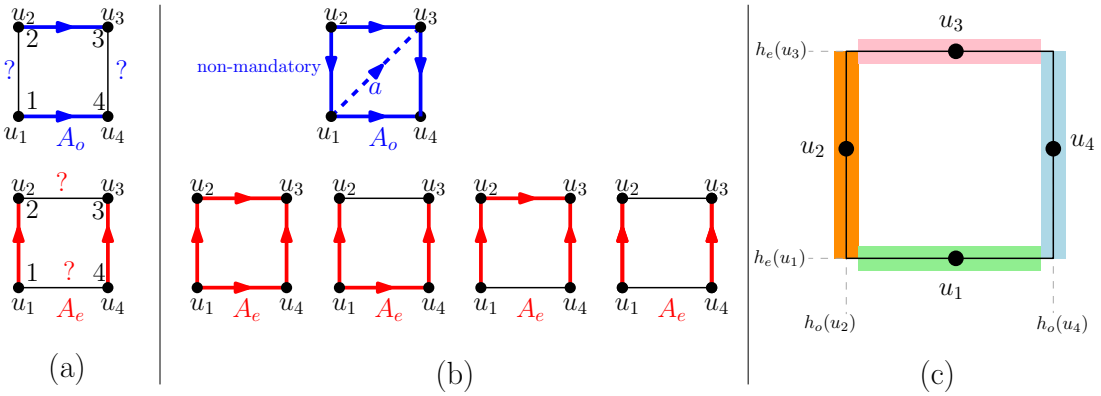


FIGURE 24. Proof of Claim (i) for a non-mandatory arc a of $\overrightarrow{A}_o$. (a) The face $f(a)$ and the arcs which are necessarily present in A_o and A_e. (b) The possible configurations for A_o and A_e assuming that $a = (u_1, u_3)$ is a non-mandatory arc a of $\overrightarrow{A}_o$. (c) Vertex placement α for the face $f(a)$ given by the inequalities in (3): the shaded regions indicate where each vertex could be placed by α. In every case the placement α is valid for $f(a)$.

Similarly, in the situation $a = (u_2, u_4)$ we have

$$\{(u_1, u_2), (u_2, u_3), (u_1, u_4), (u_4, u_3)\} \subseteq A_o \quad \text{and} \quad \{(u_1, u_2), (u_4, u_3)\} \subseteq A_e,$$

and either $(u_1, u_2) \in A_e$ or $\{u_1, u_2\} \in \overline{A_e}$ and either $(u_4, u_3) \in A_e$ or $\{u_4, u_3\} \in \overline{A_e}$. This gives

$$h_o(u_1) < h_o(u_2), h_o(u_4) < h_o(u_3) \quad \text{and} \quad h_e(u_4) \leq h_e(u_1), h_e(u_3) \leq h_e(u_2),$$

which (together with the observation $h_e(u_4) < h_e(u_2)$) is enough to ensure that the placement α is valid for the face $f(a)$. The situation where a is a non-mandatory arc of $\overrightarrow{A}_e$ is similar.

It remains to prove Claim (ii). Let a be weak diagonal arc of $\overrightarrow{A}_o$ or $\overrightarrow{A}_e$ and let u_1, u_2, u_3, u_4 be the vertices of G incident to the corners of $f(a)$ labeled 1,2,3,4 respectively. Since we have already proved Claim (i) we can assume that any diagonal arc in the face $f(a)$ is mandatory, hence the quasi increasing functions h_o and h_e satisfy

$$h_o(u_1), h_o(u_2) \leq h_o(u_3), h_o(u_4) \quad \text{and} \quad h_e(u_1), h_e(u_4) \leq h_e(u_2), h_e(u_3).$$

In fact, there are 3 possible situations for h_o (since $h_o(u_1) < h_o(u_4)$ and $h_o(u_2) < h_o(u_3)$):

- (a_o) either $h_o(u_1), h_o(u_2) < h_o(u_3), h_o(u_4)$,
- (b_o) or $h_o(u_2) < h_o(u_1) = h_o(u_3) < h_o(u_4)$ (which can happen if $a = (u_1, u_3)$ is a weak diagonal arc of $\overrightarrow{A}_o$),

(c_o) or $h_o(u_1) < h_o(u_2) = h_o(u_4) < h_o(u_3)$ (which can happen if $a = (u_2, u_4)$ is a weak diagonal arc of $\overrightarrow{A}_o$).

Similarly, there are 3 possible situations for h_e:

(a_e) either $h_e(u_1), h_e(u_4) < h_e(u_2), h_e(u_3)$,

(b_e) or $h_e(u_4) < h_e(u_1) = h_e(u_3) < h_e(u_2)$ (which can happen if $a = (u_1, u_3)$ is a weak diagonal arc of $\overrightarrow{A}_e$),

(c_e) or $h_e(u_1) < h_e(u_2) = h_e(u_4) < h_e(u_3)$ (which can happen if $a = (u_4, u_2)$ is a weak diagonal arc of $\overrightarrow{A}_e$).

It is easy to see that if (a_o) or (a_e) holds, then the placement α is valid for $f(a)$ as illustrated in Figure 25(b). If both (b_o) and (c_e) hold, then the placement α is valid for $f(a)$ as illustrated in Figure 25(c). Similarly, if both (b_e) and (c_o) hold, then the placement α is valid for $f(a)$.

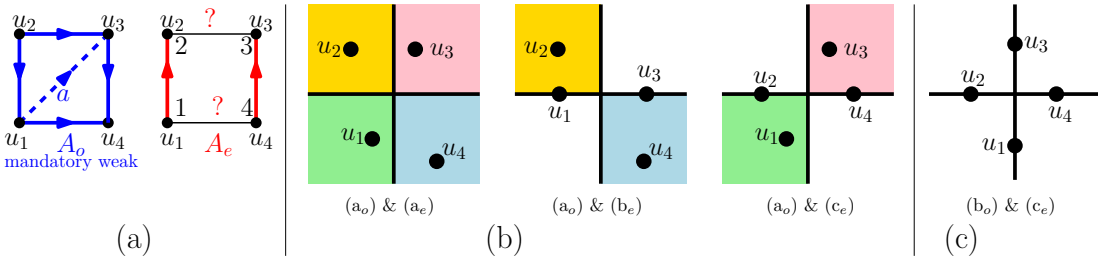


FIGURE 25. Proof of Claim (ii) for a weak diagonal arc $a = (u_1, u_3)$ of $\overrightarrow{A}_o$. (a) The face $f(a)$ and the arcs which are necessarily present in A_o and A_e. (b) Vertex placement α for the face $f(a)$ assuming (a_o) holds. (c) Vertex placement for $f(a)$ assuming both (b_o) and (c_e) hold.

The only remaining situations are when both (b_o) and (b_e) hold, or both (c_o) and (c_e) hold. However these situations are incompatible with our convention for weak diagonal arcs. For instance, in order for both (b_o) and (b_e) to hold, the arc (u_1, u_3) needs to be a diagonal arc of both $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$; but this implies converging$(a) = 4$ so that (u_1, u_3) is a strict diagonal arc of $\overrightarrow{A}_e$, which contradicts (b_e). This completes the proof of Claim (ii), and Theorem 3.14. □

Let us call *optimized* increasing function drawing algorithm, the grid drawing algorithm obtained from Theorem 3.14. Just as the increasing function algorithm discussed in Section 3.4, this optimized version can be performed in linear time. Indeed, starting for an adapted 3,4-angulation of the square G, the computation of the augmented contractions $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ can be done in linear time as explained above. Removing the non-mandatory diagonal arcs, and identifying the weak diagonal arcs can clearly be done in linear time. It then remains to compute "tight" quasi increasing functions h_o and h_e.

Given an acyclic directed graph $D = (V_D, E_D)$ with arcs marked as either *weak* or *strict*, we call *strict source-level* of a vertex $v \in V_D$, the maximal number of strict arcs on directed paths ending at v. The strict source-level of every vertex can be computed in time linear in $|V_D| + |E_D|$. Indeed, one can run the *topological ordering algorithm* on D to get a total order $\mathcal{O}$ on the vertex set V_D (such that every arc of D is oriented from the smaller vertex to the greater vertex for the order $\mathcal{O}$). Then, one can compute the strict source-level of every vertex, by treating vertices in the order given by the total order $\mathcal{O}$ (from smallest to greatest): for every vertex $v \in V_D$ the strict source-level of v is 0 if v is a source, and

$$\text{strict-level}(v) = \max\big(\{\text{strict-level}(u) \mid (u,v) \in E_D \text{ weak}\} \cup \{\text{strict-level}(u)+1 \mid (u,v) \in E_D \text{ strict}\}\big)$$

otherwise.

It is easy to see that the strict source-levels for the digraphs $\overrightarrow{A}_o$ and $\overrightarrow{A}_e$ (after removal of the non-mandatory diagonal arcs) give a tight quasi $\overrightarrow{A}_o$-increasing function h_o and a tight quasi $\overrightarrow{A}_e$-increasing function h_e. Hence we get the following result.

Proposition 3.15. *The optimized increasing function algorithm for drawing an adapted 3,4-angulation of the square can be completed in linear time.*

4. Orthogonal drawing of planar maps with vertex-degree at most 4

In this section we turn our attention to the dual picture: we will define orthogonal drawing algorithms for dual-adapted rooted 3,4-maps, which are the maps dual to the adapted 3,4-angulations of the square.

Recall that a *planar orthogonal drawing* of a graph is a planar drawing where each edge is represented as a sequence of horizontal and vertical line segments. For convenience (and in view of displaying drawings with at most one bend per edge), we adopt the convention of not drawing the root-vertex v_∞, as in [BF12, FKV14].

Definition 4.1. *Let G^* be a rooted 3,4-map, let v_∞ be its root vertex, and let $e_1^*, \dots, e_4^*$ be the root edges. A* suspended planar orthogonal drawing *for G^*, or* SPO drawing *for short, is a planar orthogonal drawing of the map obtained from G^* by disconnecting the edges $e_1^*, \dots, e_4^*$ from v_∞, in which the edges $e_1^*, \dots, e_4^*$ are drawn as dangling arrows in the outer face whose directions are* west*,* north*,* east *and* south *respectively*[4]*.*

For instance, the drawing of Figure 26(b) is an SPO drawing.

The algorithms we present in this section are based on *dual 4-GS structures*, which were introduced in [BFL24]. We first define a "face-counting" SPO drawing algorithm which generalizes Bernardi and Fusy's algorithm [BF12], and then a more general (and more compact) "increasing functions" SPO drawing algorithm, which also generalizes He's algorithm [He93].

4.1. Face-counting drawing algorithm for 3,4-maps.

In this subsection we define our first SPO drawing algorithm for rooted 3,4-maps. In this algorithm, the coordinates of each vertex are obtained by counting the number of faces in certain regions of the odd and even dual bipolar orientations. We shall refer to this algorithm as the *face-counting SPO drawing algorithm.*

Let G^* be a dual-adapted rooted 3,4-map. Let $\mathcal{L}^*$ be a dual 4-GS labeling of G^* satisfying the additional requirement that no edge is *fully-colored*, that is, belongs to all four trees of the dual 4-GS wood $\mathcal{W}^* = \{W_1^*, W_2^*, W_3^*, W_4^*\}$ associated with $\mathcal{L}^*$. We will explain the necessity of this additional requirement in Remark 4.5. For now let us briefly justify the existence of such a structure.

Lemma 4.2. *Any dual-adapted rooted 3,4-map admits a dual 4-GS labeling without any fully-colored edge. Moreover such a structure can be found in linear time.*

Proof. Via duality, it suffices to prove the following equivalent property: any adapted 3,4-angulation of the square G admits a 4-GS labeling $\mathcal{L}$ without any uncolored edge (edge not belonging to any of the trees of the associated 4-GS wood). It is shown in [BFL24], that the set of 4-GS woods of G admits the structure of a distributive lattice. This lattice structure is easier to define using a different incarnation of 4-GS structures called *4-GS angular orientations.* We refer the reader to [BFL24] for the definition of 4-GS angular orientations and the correspondence with 4-GS woods. The relevant information here is that if $\mathcal{L}$ has an uncolored edge e,

[4]This type of orthogonal drawing is called *anchored* in [FKV14] (which considers non-necessarily planar drawings).

then the associated angular orientation $\mathcal{A}$ has a counterclockwise cycle (made of the 4 "star edges" surrounding e). Moreover if $\mathcal{L}$ is the maximal element of the distributive lattice, then $\mathcal{A}$ has no counterclockwise cycle, hence $\mathcal{L}$ has no uncolored edge. Lastly, it is shown in [BFL24], that the maximal element $\mathcal{L}$ can be constructed in linear time. □

Recall from Section 2.4 that

$$B_o^* = W_3^* \cup (W_1^*)^- \quad \text{and} \quad B_e^* = W_2^* \cup (W_4^*)^-$$

are bipolar orientations. For each non-root vertex v, we denote by $P_i^*(v)$ the directed path from v to the root vertex v_∞ in the tree W_i^*, and define the *odd and even dual paths* for v as

$$P_o^*(v) = P_3^*(v) \cup P_1^*(v)^- \subseteq B_o^* \text{ and } P_e^*(v) = P_2^*(v) \cup P_4^*(v)^- \subseteq B_e^*.$$

Note that $P_o^*(v)$ is a simple directed path whose support contains the root-edges e_3^* and e_1^*, and $P_e^*(v)$ is a simple directed path whose support contains the root-edges e_2^* and e_4^*.

We define the *face-counting coordinates* of v as $(\ell_e^*(v), r_o^*(v))$, where $\ell_e^*(v)$ is the number of inner faces of B_e^* on the left of $P_e^*(v)$, and $r_o^*(v)$ is the number of inner faces of B_o^* on the right of $P_o^*(v)$. The face-counting coordinates are represented in Figure 26(a).

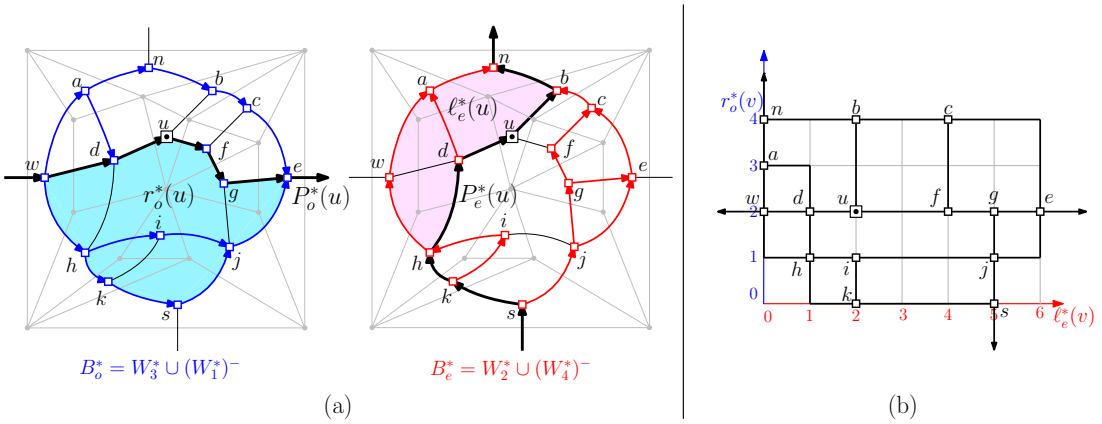


FIGURE 26. (a) The paths and regions considered to compute the face-counting coordinates of the vertex u. Here $\ell_e^*(u) = 2$ and $r_o^*(u) = 2$. (b) The face-counting drawing of G^*.

We now need to specify how to draw the edges of G^*. For a point A in $\mathbb{R}^2$, we denote by $\gamma_1(A)$ (resp. $\gamma_2(A)$, $\gamma_3(A)$, $\gamma_4(A)$) the half-line starting from A going in the negative x-direction (resp. positive y-direction, positive x-direction, negative y-direction). See Figure 27(a).

Definition 4.3. *Let G^* be a dual-adapted rooted 3,4-map endowed with a dual 4-GS labeling $\mathcal{L}^*$. We consider the coloring of the arcs of G^* given by the 4-GS wood $(W_1^*, W_2^*, W_3^*, W_4^*)$, and recall that every non-root arc has at least one color in $\{1,2,3,4\}$.*

Given a vertex placement $\alpha : V' \to \mathbb{R}^2$, where V' is the set of non-root vertices of G^*, the $\mathcal{L}^*$-rule *for drawing each non-root edge $e = \{u, v\}$ is as follows:*

- *If e has exactly two colors, and these colors i, j are of different parity, then the edge e is drawn as a bent-edge. Precisely, if the arc (u, v) has color i and the arc (v, u) has color j, then e is drawn as the union of line segments $[\alpha(u), X] \cap [X, \alpha(v)]$, where X is the intersection of the rays $\gamma_i(\alpha(u))$ and $\gamma_j(\alpha(v))$ (assuming the intersection X is non empty); see Figure 27(b).*
- *In all the other situation, e is drawn as a straight edge.*

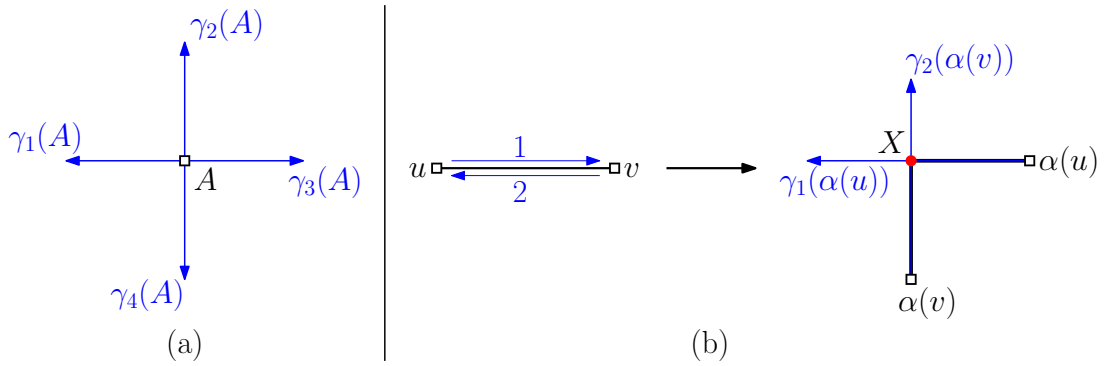


FIGURE 27. (a) The four rays $\gamma_1(A), \gamma_2(A), \gamma_3(A), \gamma_4(A)$ emanating from a point A. (b) The bent-edge drawn according to the $\mathcal{L}^*$-rule for an edge $e = \{u, v\}$ having arc (u, v) colored 1 and arc (v, u) colored 2.

The *face-counting drawing* of G^* (based on $\mathcal{L}^*$) is defined as follows.

- Each non-root vertex v is placed at the face-counting coordinates $(\ell_e^*(v), r_o^*(v))$.
- Each non-root edge e is drawn according to the $\mathcal{L}^*$-rule (Definition 4.3).
- The root edges $e_1^*, e_2^*, e_3^*, e_4^*$ are drawn as four dangling arrows whose directions are west, north, east, and south, respectively.

An example is represented in Figure 26(b). Of course, one needs to justify that it is possible to draw the edges according to the $\mathcal{L}^*$-rule (and that the straight edges are all either horizontal or vertical). In fact, we will prove the following stronger result.

Theorem 4.4. *Let G^* be a dual-adapted rooted 3,4-map and let $\mathcal{L}^*$ be a dual 4-GS labeling of G^* such that no edge is fully-colored. The face-counting drawing of G^* based on $\mathcal{L}^*$ is an SPO drawing.*

We will prove Theorem 4.4 after presenting a more general algorithm in the next subsection.

Remark 4.5. Now we explain why the requirement that the dual 4-GS structure does not have fully-colored edges is necessary: if one uses a dual 4-GS labeling $\mathcal{L}^*$ having a fully colored edge $e = \{u, v\}$, then the face-counting coordinates of u and v are equal, hence the face-counting drawing is not planar.

To close this subsection, we compare the face-counting SPO drawing algorithm of Theorem 4.4 to the algorithm from [BF12] to draw the dual G^* of a simple quadrangulation G. Recall from the introduction that this algorithm relies on a corner labeling $\mathcal{L}^*$ of G^* obtained from a separating decomposition $\mathcal{S}$ of G. As explained in Section 2.2, the separating decomposition $\mathcal{S}$ corresponds to an even 4-GS labeling $\mathcal{L}$ of G, and it clear from the definition that the corner labeling $\mathcal{L}^*$ is the dual of $\mathcal{L}$. For each non-root vertex v, the path $P_i'(v)$ used by the algorithm in [BF12] is equal to the path $P_i^*(v)$ in the tree W_i^*, and the odd and even separating paths $P_o'(v)$ and $P_e'(v)$ are equal to the odd and even dual paths $P_o^*(v)$ and $P_e^*(v)$. Therefore, the placement of vertices by the algorithm in [BF12] coincides with the placement from the face-counting algorithm based on the dual 4-GS labeling. Furthermore, the rule for drawing edges in [BF12] also coincides with the rule described before. (A consequence of the evenness of $\mathcal{L}$ property is that every non-root edge of G^* has exactly two colors of different parity hence is drawn as a bent edge.) In conclusion, the algorithm of [BF12] is the same as the face-counting algorithm from Theorem 4.4 restricted to the dual of simple quadrangulations endowed with even dual 4-GS labelings.

4.2. **Increasing function drawing algorithm for 3,4-maps.** In this subsection we present another SPO drawing algorithm for dual-adapted rooted 3,4-maps. We shall call this algorithm the *increasing function SPO drawing algorithm*. This algorithm generalizes the face-counting algorithm presented in the previous subsection, hence can provide some improvement on the grid size. It relies on some acyclic orientations that we now present.

Let G^* be a dual-adapted rooted 3,4-map, and let $\mathcal{L}^*$ be a dual 4-GS labeling (whose associated dual 4-GS wood may have fully-colored edges). The *odd* and *even dual acyclic orientations* A_o^* and A_e^* are defined as:

$$\begin{aligned} A_o^* =& \{a \text{ non-root arc} \mid 4 \in [\text{right-init}(a), \text{right-term}(a)[\text{ or } 2 \in [\text{left-term}(a), \text{left-init}(a)[\\ & \text{or } (\text{right-init}(a), \text{right-term}(a), \text{left-term}(a), \text{left-init}(a)) = (1,1,3,3)\} \\ A_e^* =& \{a \text{ non-root arc} \mid 3 \in [\text{right-init}(a), \text{right-term}(a)[\text{ or } 1 \in [\text{left-term}(a), \text{left-init}(a)[\\ & \text{or } (\text{right-init}(a), \text{right-term}(a), \text{left-term}(a), \text{left-init}(a)) = (4,4,2,2)\} \end{aligned}$$

The definition of A_o^* and A_e^* is represented in Figure 28 (top row), and a concrete example is represented in Figure 16(b). One can check that A_o^* is a subset of the acyclic orientation B_o^*, and A_e^* is a subset of the acyclic orientation B_e^*, hence they are acyclic.

Let $\overline{A_o^*}$ be the set of edges not in the support of A_o^*, and let $\overline{A_e^*}$ be the set of edges not in the support of A_e^*. From the definition of A_o^* and A_e^* and the conditions for dual 4-GS labelings, it is not hard to see that

$$\begin{aligned} \overline{A_o^*} =& \{e \text{ non-root edge} \mid \text{one side of } e \text{ has labels } \{1,2\} \text{ or } \{3,4\} \\ & \text{or the four labels around } e \text{ are } \{2,2,4,4\}\}, \\ \overline{A_e^*} =& \{e \text{ non-root edge} \mid \text{one side of } e \text{ has labels } \{2,3\} \text{ or } \{4,1\} \\ & \text{or the four labels around } e \text{ are } \{1,1,3,3\}\}. \end{aligned}$$

The definition of $\overline{A_o^*}$ and $\overline{A_e^*}$ is also represented in Figure 28 (middle and bottom rows). We claim that $\overline{A_o^*}$ (resp. $\overline{A_e^*}$) do not contain cycles. To see this, first note that by construction $\overline{A_o^*}$ (resp. $\overline{A_e^*}$) is a subset of the support of B_e^* (resp. B_o^*). Moreover, it is easy to check that a cycle in $\overline{A_o^*}$ (resp. $\overline{A_e^*}$) always supports a directed cycle in B_e^* (resp. B_o^*), which is impossible since we have established in Section 2.4 that B_e^* and B_o^* are bipolar orientations.

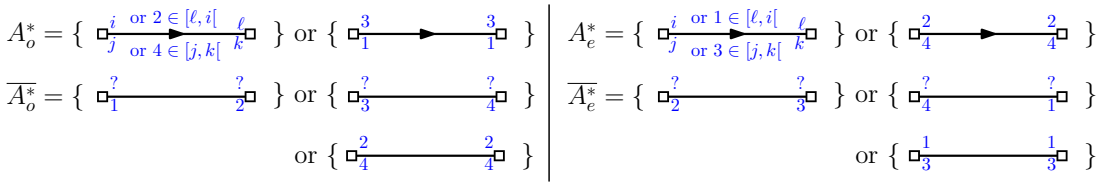


FIGURE 28. The non-root arcs in the dual acyclic orientations A_o^* and A_e^*, and the non-root edges not in their support.

We now define the dual augmented bipolar contractions in two steps. The *odd* (resp. *even*) *dual augmented bipolar contraction* $\overrightarrow{A}_o^*$ (resp. $\overrightarrow{A}_e^*$) is obtained from A_o^* (resp. A_e^*) by the following procedure:

(i) For each non-root face of G^*, add an *odd* (resp. *even*) *diagonal arc* from the last corner labeled 4 (resp. 3) clockwise around f to the last corner labeled 2 (resp. 1) clockwise around f, if both corners exist.

(ii) Contract all the edges in $\overline{A_o^*}$ (resp. $\overline{A_e^*}$).

This process is represented in Figure 29(a)-(b). Note that Step (ii) does not depend on Step (i) and can be executed independently. In fact, it is convenient to introduce some separate notations for the structures obtained by performing Step (ii) only: we define $\widetilde{A}^*_o := A^*_o/\overline{A^*_o}$ and $\widetilde{A}^*_e := A^*_e/\overline{A^*_e}$, and call them the *odd* and *even bipolar contractions*. We will prove later that $\widetilde{A}^*_o$, $\widetilde{A}^*_e$, $\overrightarrow{A}^*_o$ and $\overrightarrow{A}^*_e$ are bipolar orientations.

An $\overrightarrow{A}^*_o$ (resp. $\overrightarrow{A}^*_e$)-*increasing* function for G^* is a function h^* defined on the non-root vertices of G^* that satisfies the following 2 conditions:

- $h^*(u) = h^*(v)$ for every edge $\{u, v\}$ in $\overline{A^*_o}$ (resp. $\overline{A^*_e}$),
- $h^*(u) < h^*(v)$ for every arc (u, v) in A^*_o (resp. in A^*_e) and every odd (resp. even) diagonal arc (u, v).

Such functions can be identified with functions defined on vertices of $\overrightarrow{A}^*_o$ (resp. $\overrightarrow{A}^*_e$) that are strictly increasing along arcs.

Theorem 4.6. *Let G^* be a dual-adapted rooted 3,4-map, let $\mathcal{L}^*$ be a dual 4-GS labeling, and let $\overrightarrow{A}^*_o$ and $\overrightarrow{A}^*_e$ be the corresponding augmented dual bipolar contractions of G^*. Let h^*_o be an $\overrightarrow{A}^*_o$-increasing function, and let h^*_e be an $\overrightarrow{A}^*_e$-increasing function. Placing each non-root vertex v at coordinates $(h^*_o(v), h^*_e(v)) \in \mathbb{R}^2$, and drawing each non-root edge of G^* using the $\mathcal{L}^*$-rule (Definition 4.3), yields an SPO drawing of G^*.*

An SPO drawing based on increasing functions is represented in Figure 29. One can compare this drawing to the one in Figure 26 obtained by the face-counting algorithm on the same map, and observe that the increasing-function drawing is slightly more compact.

Theorem 4.6 about the increasing function algorithm actually implies Theorem 4.4 about the face-counting algorithm. This is because face-counting coordinates are increasing functions as stated below.

Lemma 4.7. *Let G^* be a dual-adapted rooted 3,4-map, and let $\mathcal{L}^*$ be a dual 4-GS labeling such that no edge is fully-colored. The face-counting coordinates $(\ell^*_e(v), r^*_o(v))$ defined in the previous subsection are such that the function ℓ^*_e is $\overrightarrow{A}^*_o$-increasing and the function r^*_o is $\overrightarrow{A}^*_e$-increasing.*

Proof. We prove the statement for the function ℓ^*_e (the situation for r^*_o is similar). We need to show three facts:

(a) If $\{u, v\} \in \overline{A^*_o}$, then $\ell^*_e(u) = \ell^*_e(v)$.
(b) If $(u, v) \in A^*_o$, then $\ell^*_e(u) < \ell^*_e(v)$.
(c) If (u, v) is a diagonal arc in $\overrightarrow{A}^*_o$, then $\ell^*_e(u) < \ell^*_e(v)$.

We prove (a) first. One can easily check that any edge in $\overline{A^*_o}$ bears a pair of opposite arcs of colors 2 and 4. Hence $P^*_e(u) = P^*_e(v)$ and $\ell^*_e(u) = \ell^*_e(v)$.

For (b), suppose $(u, v) \in A^*_o$. Given that $\mathcal{L}^*$ has no edge with incident corners labeled $\{1, 1, 3, 3\}$, it is easy to see that the arc (u, v) does not have color 4, and the opposite arc (v, u) does not have color 2. Hence $P^*_e(u) \neq P^*_e(v)$. Moreover, $P^*_e(u)$ is weakly on the left of $P^*_e(v)$. The situation is represented in Figure 30(a). In particular, $P^*_e(u)$ and $P^*_e(v)$ cannot cross each other: any common vertex of $P^*_e(u)$ and $P^*_e(v)$ is either a common vertex of $P^*_4(u)$ and $P^*_4(v)$ or a common vertex of $P^*_2(u)$ and $P^*_2(v)$. But $P^*_2(u)$ and $P^*_2(v)$ (and similarly, $P^*_4(u)$ and $P^*_4(v)$) merge once they have a common vertex. This implies $\ell^*_e(u) < \ell^*_e(v)$.

For (c), recall that in $\overrightarrow{A}^*_o$, a diagonal arc inside a non-root face is from the last corner with label 4 to the last corner with label 2, if both corners exist. Let us refer to the two vertices incident to these two corners as u and v. We claim that $P^*_e(u) \neq P^*_e(v)$ and $P^*_e(u)$ is weakly on the left of $P^*_e(v)$. Indeed, $P^*_2(u)$ leaves the face at u, while $P^*_4(u)$ follows the boundary of the face until the first corner with label 4; $P^*_4(v)$ leaves the face at v, while $P^*_2(v)$ follows the boundary

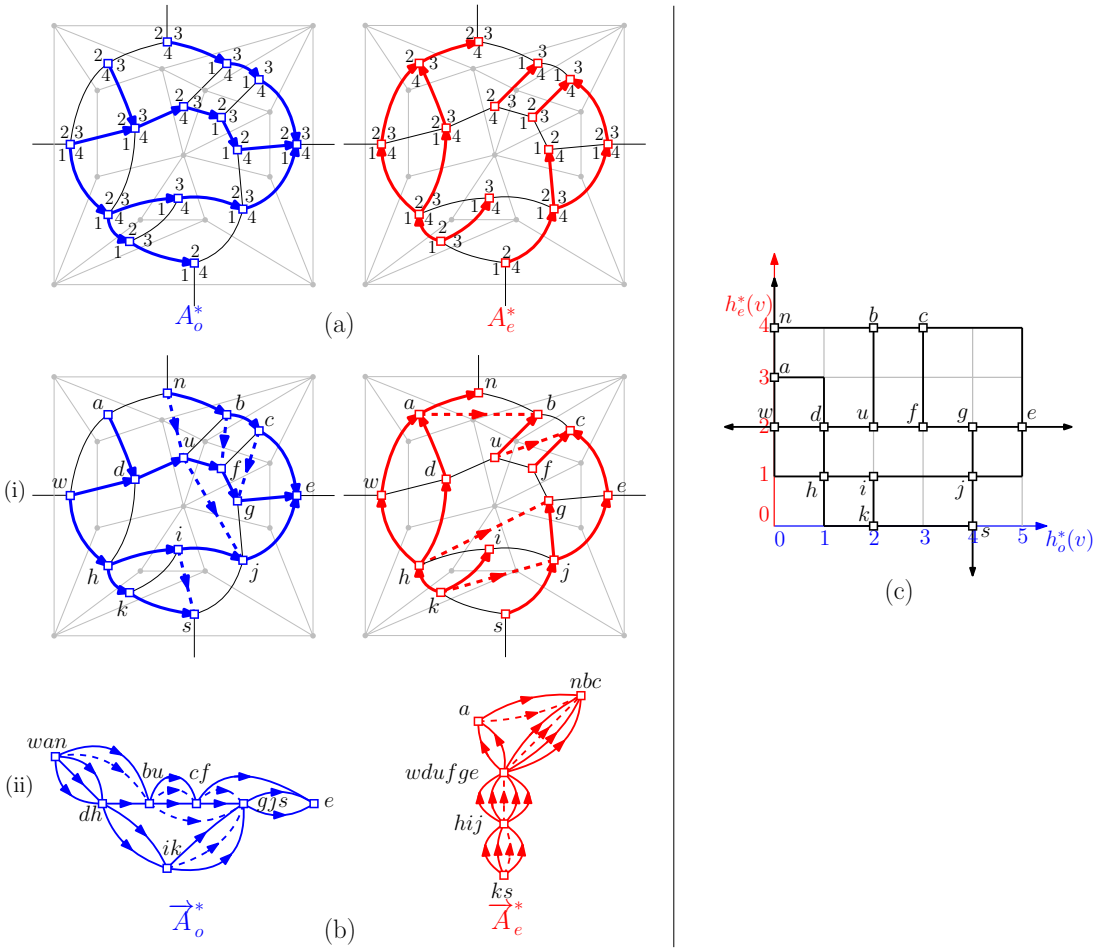


FIGURE 29. (a) The dual acyclic orientations. (b) The augmented dual bipolar contractions. (c) The drawing obtained by choosing some tight increasing functions h_o^* and h_e^*.

of the face until the first corner with label 2. This situation is presented in Figure 30(b). This implies $\ell_e^*(u) < \ell_e^*(v)$, which completes the proof that ℓ_e^* is $\overrightarrow{A}_o^*$-increasing. □

Next we show that the orientations $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$ are acyclic (hence admit increasing functions).

Lemma 4.8. *The maps $\widetilde{A}_o^*$, $\widetilde{A}_e^*$, $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$ are bipolar orientations.*

Proof. We will prove the property for $\widetilde{A}_o^*$ and $\overrightarrow{A}_o^*$ (the proof for $\widetilde{A}_e^*$ and $\overrightarrow{A}_e^*$ is similar). In order to prove that $\overrightarrow{A}_o^*$ is acyclic, we first show that the face-counting function ℓ_e is "almost" $\overrightarrow{A}_o^*$-increasing. Precisely, by reasoning as in the proof of Lemma 4.7, the following properties are easy to show:

(a') If $\{u, v\} \in \overline{A_o^*} \cup \{\text{edges with labels } \{1,1,3,3\}\}$, then $\ell_e^*(u) = \ell_e^*(v)$.
(b') If $(u, v) \in A_o^* \backslash \{\text{edges with labels } \{1,1,3,3\}\}$, then $\ell_e^*(u) < \ell_e^*(v)$.
(c) If (u, v) is a diagonal arc in $\overrightarrow{A}_o^*$, then $\ell_e^*(u) < \ell_e^*(v)$.

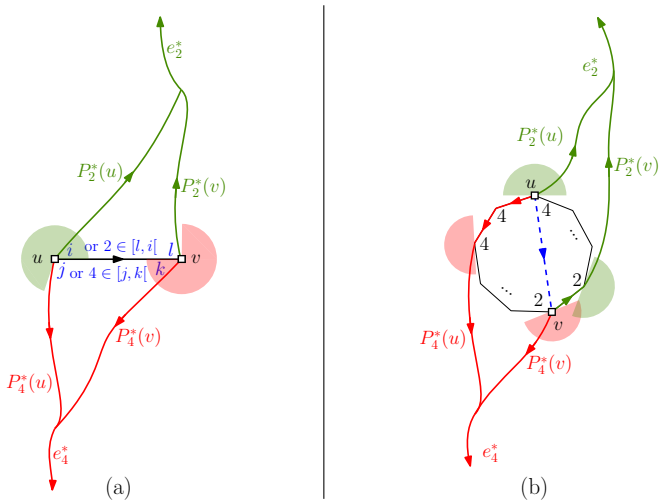


FIGURE 30. (a) An arc $a = (u, v) \in A_o^*$. The range for position of the first arc of $P_2^*(u)$ is indicated by the green sector, and the range for position of the first arc of $P_4^*(v)$ is indicated by the red sector. (b) A non-root face, on which u is incident to the last corner with label 4, and v is incident to the last corner with label 2. The range for position of the first arc of $P_2^*(u)$ is indicated by the green sector, and the range for position of the first arc of $P_4^*(v)$ is indicated by the red sector.

Supposing for contradiction that $\overrightarrow{A}_o^*$ has a directed cycle C, the above properties imply that ℓ_e^* is constant on C, and C consists solely of fully-colored edges with labels $\{1, 1, 3, 3\}$. This implies that C is contained in the tree W_3^* (and in fact every tree of the dual 4-GS wood), which is a contradiction. Hence $\overrightarrow{A}_o^*$ is acyclic. Since $\overrightarrow{A}_o^*$ is obtained from $\widetilde{A}_o^*$ by adding some edges and identifying some vertices, $\widetilde{A}_o^*$ must also be acyclic.

It remains to show that $\widetilde{A}_o^*$ and $\overrightarrow{A}_o^*$ have a unique source and a unique sink. For this it suffices to show that for any non-root vertex v of G^*, the vertex $\widetilde{v}$ of $\widetilde{A}_o^*$ corresponding to v is incident to an ingoing and an outgoing arc in $\widetilde{A}_o^*$ (hence also in $\overrightarrow{A}_o^*$). Observe that for every non-root vertex v of G^*, the directed path $P_3^*(v) \cup P_1^*(v)^-$ is contained in $A_o^* \cup \overline{A_o^*}$, which implies that the vertex $\widetilde{v}$ of $\widetilde{A}_o^*$ is on a directed path (hence is incident to an ingoing and an outgoing arc in $\widetilde{A}_o^*$). This completes the proof. □

We conclude this subsection by discussing the relationship between $\widetilde{A}_o^*$ and $\widetilde{A}_e^*$ and the primal bipolar orientations B_o and B_e.

Remark 4.9. Given G and $\mathcal{L}$ as above, we define the *odd* and *even augmented bipolar orientations* as the following oriented submaps of G:

$$B'_o = B_o \cup U_o \text{ and } B'_e = B_e \cup U_e$$

where

$$U_o = \{a \text{ inner arc} \mid (\text{left-init}(a), \text{right-init}(a), \text{right-term}(a), \text{left-term}(a)) = (2, 2, 4, 4)\}$$
$$U_e = \{a \text{ inner arc} \mid (\text{left-init}(a), \text{right-init}(a), \text{right-term}(a), \text{left-term}(a)) = (1, 1, 3, 3)\}$$

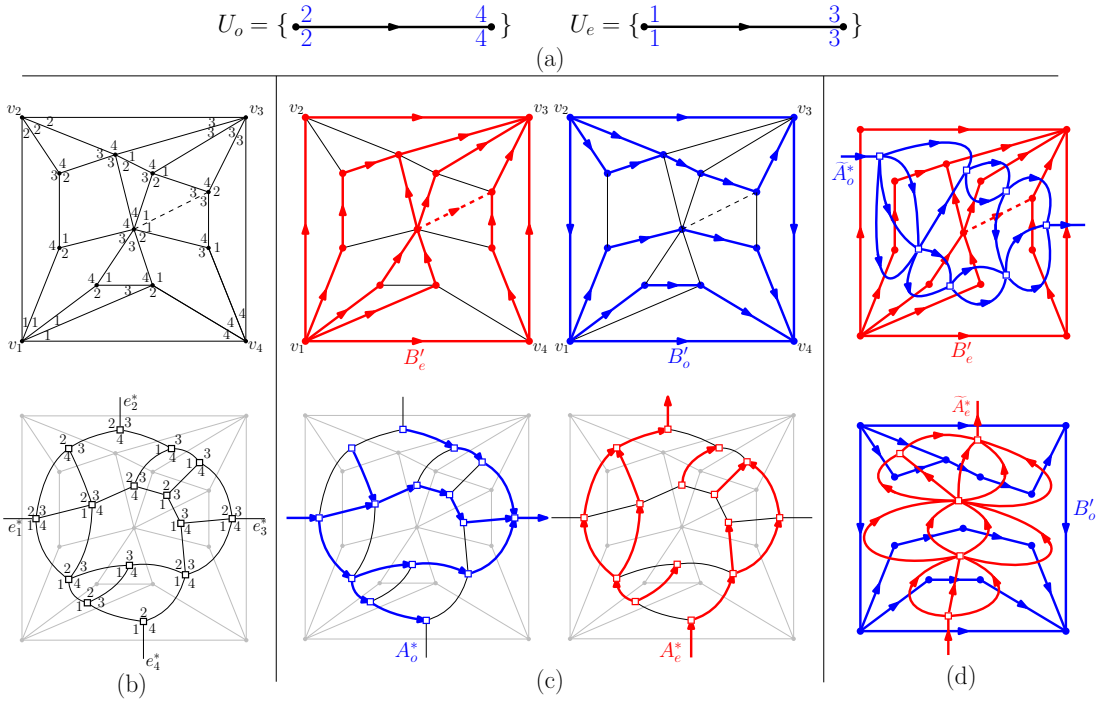


FIGURE 31. (a) The inner arcs in U_o and U_e. (b) A 4-GS labeling $\mathcal{L}$ (top), where the uncolored edge is presented as a dashed line, and its dual labeling $\mathcal{L}^*$ (bottom). (c) The augmented bipolar orientations B'_e and B'_o (top) and the dual acyclic orientations A^*_o and A^*_e (bottom). (c) Duality relations between $\widetilde{A}^*_o$ and B'_e (top) and between $(\widetilde{A}^*_e)^-$ and B'_o (bottom).

The definition of U_o and U_e is illustrated in Figure 31(a). Note that when the primal map G is an irreducible triangulation, the augmented bipolar orientations B'_o and B'_e associated to a 4-GS labeling $\mathcal{L}$ of G are exactly the blue and red submaps G_b and G_r of the transversal structure corresponding to $\mathcal{L}$.

It easily follows from the definition that the bipolar orientation $\widetilde{A}^*_o$ is the *dual* of B'_e, while the bipolar orientation $\widetilde{A}^*_e$ is the *dual* of $(B'_o)^-$. This is represented in Figure 31. This can be compared with the duality between $\widetilde{A}_o$ (resp. $\widetilde{A}_e$) and $(B^*_e)^-$ (resp. B^*_o) represented in Figure 17. These two duality relations are summarized in Figure 32.

4.3. **Proof of planarity.** In this subsection we prove Theorem 4.6.

Let G^* be a dual-adapted rooted 3,4-map, and let $\mathcal{L}^*$ be a dual 4-GS labeling on G^*. Let G' be the submap $G^*\backslash\{v_\infty\}$, and let V' be the set of non-root vertices of G. Let $h^*_o : V' \to \mathbb{R}$ be an $\overrightarrow{A}^*_o$-increasing function, and let $h^*_e : V' \to \mathbb{R}$ be an $\overrightarrow{A}^*_e$-increasing function.

We want to show that the vertex placement $\alpha : v \mapsto (h^*_o(v), h^*_e(v))$ induces an SPO drawing of G^*. The proof again relies on the planarity criterion established in Section 3.3 (see Lemmas 3.7 and 3.8). Recall from Section 3.3 that a drawing of G^* is called *valid* for a face f of G^* if the curve corresponding to the clockwise contour of f is a simple curve in the plane whose interior is on its right.

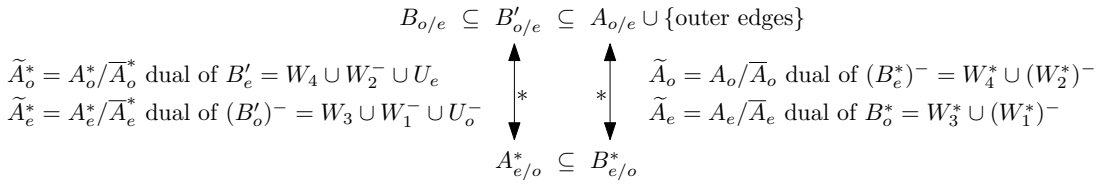


FIGURE 32. Duality relations for the bipolar orientations underlying the increasing function algorithms. One uses the orientations $\widetilde{A}_o$ and $\widetilde{A}_e$ (which are the dual of $(B^*_e)^-$ and B^*_o) for drawing 3,4-angulations, and the orientations $\widetilde{A}^*_o$ and $\widetilde{A}^*_e$ (which are the dual of B'_e and $(B'_o)^-$) for drawing 3,4-valent maps.

Lemma 4.10. *The vertex placement α is such that each non-root edge can be drawn according to the $\mathcal{L}^*$-rule (Definition 4.3). It produces an orthogonal drawing of G' which is valid for every inner face of G'.*

Proof. Let f be an inner face of G'. The labels of the corners inside f in clockwise order form an interval of 1, an interval of 2, an interval of 3 and an interval of 4 (where at most two of these intervals can be empty). We claim that the face f will be embedded as represented in Figure 33 (the shape of the embedding depends on which labels appears inside f).

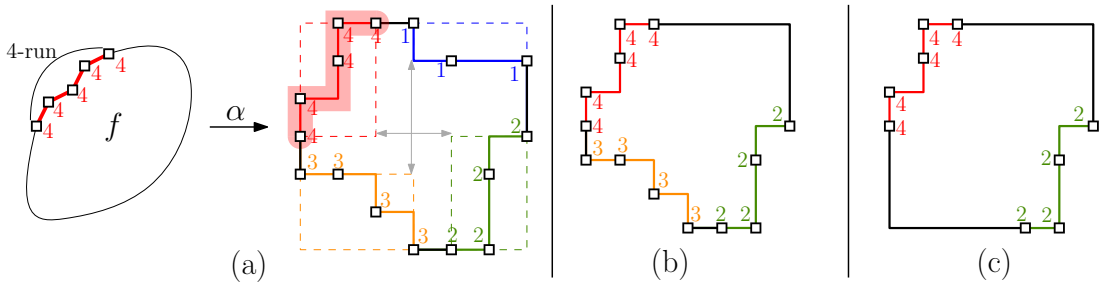


FIGURE 33. (a) The embedding of an inner face f with no missing labels. The different runs are indicated by different colors. (b) The embedding of an inner face f missing the label 1. (c) The embedding of an inner face f missing labels 1 and 3.

For $i \in \{1, 2, 3, 4\}$, we call *i-run* the sequence of arcs having f on their right, with the two labels on their right equal to i. We consider the drawing of the arcs in the i-run. Let us consider $i = 4$ for concretness. Let $a = (u, v)$ be an arc in the 4-run of f. It is easy to see that the pair of labels $(j, k) = (\text{left-init}(a), \text{left-term}(a))$ on the left of a can be either $(3, 1)$, $(3, 2)$, $(2, 1)$ or $(2, 2)$. These situations are represented in Figure 34(a).

- If $(j, k) = (3, 1)$, then a has color 3 while $-a$ has color 4, so $a \in A^*_o \cap A^*_e$. In this case the $\mathcal{L}^*$-rule for drawing a can be applied and a is drawn as a bent edge going "right-up".
- If $(j, k) = (3, 2)$, then a has color 3 while $-a$ has colors 1 and 4, so $a \in A^*_o \cap \overline{A^*_e}$. In this case the $\mathcal{L}^*$-rule for drawing a can be applied and a is drawn as an horizontal edge going "right".
- If $(j, k) = (2, 1)$ (resp. $(2, 2)$), then a has colors 2 and 3 while $-a$ has color 4 (resp. colors 1 and 4), so $a \in \overline{A^*_o} \cap A^*_e$. In this case the $\mathcal{L}^*$-rule for drawing a can be applied and a is drawn as a vertical edge going "up".

In conclusion, the 4-run of f goes (weakly) right-up. Symmetrically, the 1-run goes down-right, the 2-run goes left-down, and the 3-run goes up-left.

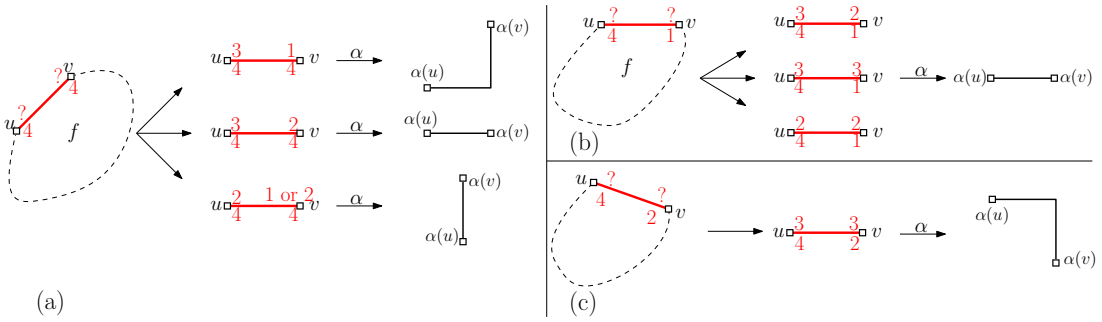


FIGURE 34. (a) Embedding of edges with labels $\{4, 4\}$ inside f. (b) Embedding of edges with labels $\{4, 1\}$ inside f. (c) Embedding of edges with labels $\{4, 2\}$ inside f.

Next we consider the edges between runs. Consider an arc a having f on its right with labels $(\text{right-init}(a), \text{right-term}(a)) = (4, 1)$. Then, the pair of labels $(j, k) = (\text{left-init}(a), \text{left-term}(a))$ on the left of a can be either $(3, 2)$, $(3, 3)$ or $(2, 2)$. In every case the $\mathcal{L}^*$-rule for drawing a can be applied and a is drawn as an horizontal edge going "right". This is represented in Figure 34(b). Symmetrically,

- if a is an arc having f on its right with labels $(\text{right-init}(a), \text{right-term}(a)) = (1, 2)$, then it is drawn down,
- if a is an arc having f on its right with labels $(\text{right-init}(a), \text{right-term}(a)) = (2, 3)$, then it is drawn left,
- if a is an arc having f on its right with labels $(\text{right-init}(a), \text{right-term}(a)) = (3, 4)$, then it is drawn up.

Finally, an arc a having f on its right with labels $(\text{right-init}(a), \text{right-term}(a)) = (4, 2)$ will be drawn as a bent edge going "right then down". This is represented in Figure 34(c). Symmetric results olds for an arc with label $(\text{right-init}(a), \text{right-term}(a)) = (1, 3)$ or $(2, 4)$ or $(3, 1)$.

To summarize, every non-root edged of G^* can be drawn following the $\mathcal{L}^*$-rule which produces an orthogonal drawing of G'. Moreover, for each inner face f, the drawing of the clockwise contour of f is as follows: the 1-run drawn going down-right, a down segment, the 2-run drawn left-down, a left segment, the 3-run drawn up-left, an up segment, the 4-run going right-up, a right segment.

It is clear from the properties stated above that consecutive runs of f (that is, a i-run and a $(i + 1)$-run) cannot intersect one another. Hence the drawing is valid for the face f unless opposite runs (that is, a i-run and a $(i + 2)$-run) intersect each other. Now, the function h_o^* is increasing along odd diagonal arcs inside f (which goes from the end of the 4-run to the end of the 2-run), hence the 4-run is strictly to the left of the 2-run; and the function h_e^* is increasing along even diagonal arcs inside f (which goes from the end of the 3-run to the end of the 1-run), hence the 3-run is strictly below the 1-run. In conclusion, the drawing of f is as illustrated in Figure 33, and it is valid. □

In order to finish the proof of Theorem 4.6, we need to show that the orthogonal drawing of G' is valid for the outer face.

Consider the vertices on the outer boundary of G' that are adjacent to the root vertex v_∞ in G^*. Let us denote the vertex incident to the root-edge e_1^* (resp. e_2^*, e_3^*, e_4^*) by v_1^* (resp. v_2^*, v_3^*, v_4^*). The vertices $v_1^*, v_2^*, v_3^*, v_4^*$ divide the clockwise outer contour of G' into four parts. We refer to the part from v_i^* to v_{i+1}^* as the *outer* $(i+1)$*-run*, as all the labels on the outer face of this part are $(i+1)$. Some of the outer runs can be empty since we could have $v_i^* = v_{i+1}^*$; however we cannot have $v_i^* = v_{i+2}^*$ unless G' consists of a single vertex.

It is not hard to see that, for any pair of increasing functions h_o^* and h_e^*, the four runs of the outer contour satisfy some specific inequalities about h_o^* and h_e^*, which translate into the following orientation properties:

- The outer 2-run is directed up-right.
- The outer 3-run is directed right-down.
- The outer 4-run is directed down-left.
- The outer 1-run is directed left-up.

In other words, the shapes of the four outer runs are similar to the shapes of the four runs of a generic inner face (with index of the run offset by 2). See Figure 35(a) (left) for an example. This alone however does not guarantee that the drawing of the outer face of G' is valid. The difference is that in the case of inner faces, the condition on diagonal arcs ensure that opposite runs do not intersect. For the outer face however, not only are such diagonal arc requirements non-existent, but also the outer contour may have cut vertices in the first place.

Fortunately, the shapes of the four runs imply that all cut vertices, if they exist, must belong to one of the two pairs of opposite outer runs, say the outer 3-run and the outer 1-run. The cut vertices divide G' into a chain of 2-connected subgraphs, that we call *2-blocks* of G (consecutive 2-blocks intersect at a cut vertex). It is clear from the orientation property of the outer-face of G' that the drawing of G' is planar if and only if the drawing of each 2-block is planar. An example is illustrated in Figure 35(a) (right).

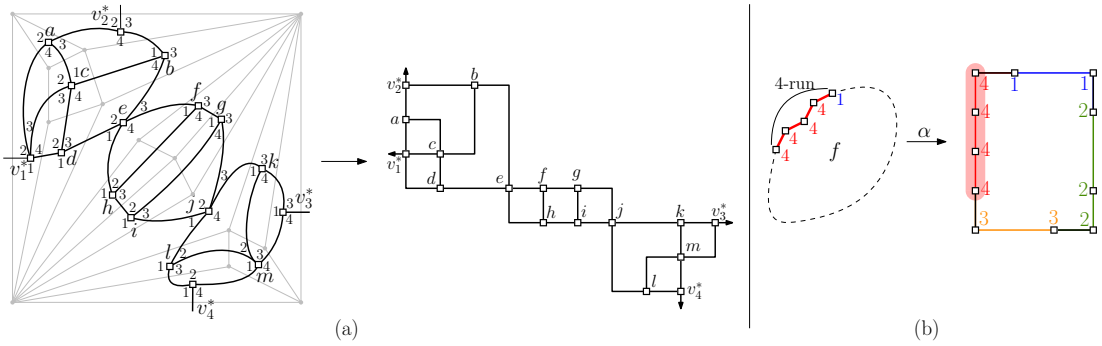


FIGURE 35. (a) Left: A dual 4-GS labeling $\mathcal{L}^*$, with two cut vertices e and j. They divide G' into three 2-blocks. Right: The drawing obtained by choosing some increasing functions. (b) The embedding of a face by He's algorithm.

Let H be a 2-block of G. Recall that the drawing of each inner face of H is valid by Lemma 4.10. Hence, if we can show that the drawing of the outer face of H is valid, then we can invoke Lemma 3.7 to conclude that the drawing of H is planar. Let us suppose for contradiction that the drawing of the outer face of H is not valid. Given the orientation property of the runs, this implies that the boundary of H must have some outer-right-visible point (recall the definition from Lemma 3.8 in Section 3.3). This is impossible according to Lemma 3.8. Therefore the outer face of H is valid, and by Lemma 3.7, the drawing of H is planar.

Since the drawing of every 2-block of G is planar, the drawing of the whole graph G' is planar. It is also clear that the placements of v_i^*'s are extremal in their respective directions, and hence the four dangling arrows can be added to the outer face, completing a valid SPO drawing of G^*. This completes the proof of Theorem 4.6.

4.4. **Comparison to He's algorithm.** We now compare the increasing function algorithm of Theorem 4.6 to He's algorithm from [He93]. He's algorithm is based on a transversal structure on an irreducible triangulation G and a pair of increasing functions on the faces of the red and blue submaps G_r and G_b. As explained in Section 2.2, each transversal structure also corresponds to a 4-GS labeling, and Remark 4.9 points out that G_r and G_b are equal to the augmented bipolar orientations B'_e and B'_o. Given the duality relations between the dual bipolar contractions $\widetilde{A}^*_o, \widetilde{A}^*_e$ and B'_e, B'_o, it is easy to see that the increasing functions on the faces of G_r and G_b, which are used in He's algorithm, naturally corresponds to a pair of increasing functions on the vertices of $\widetilde{A}^*_o$ and $\widetilde{A}^*_e$. We claim that such functions are also increasing functions on $\overrightarrow{A}^*_o$ and $\overrightarrow{A}^*_e$, because they automatically increase along the diagonal arcs. To see this, we can investigate the shape of faces in the drawing. First, note that an edge connecting two vertices of degree 3 cannot have exactly two colors with different parity, and hence every edge of G^* is drawn with no bend. Next, inside each inner face of G^*, all four labels must appear, and with an analysis as in Section 4.3, one can observe that the shape of the four runs around an inner face of G^* degenerates to something simple. Namely,

- the 1-run is drawn as a sequence of edges directed horizontally to the right,
- the 2-run is drawn as a sequence of edges directed vertically downward,
- the 3-run is drawn as a sequence of edges directed horizontally to the left,
- the 4-run is drawn as a sequence of edges directed vertically upward.

Consequently, each inner face f of G^* is drawn as a rectangle, with the four corners of that rectangle corresponding to the last vertices incident to a corner of label 1,2,3,4 respectively in clockwise order around f. See Figure 35(b) for an illustration. Hence the functions used in He's algorithm already ensures strict increments along diagonal arcs. In short, the algorithm of [He93] is the same as the increasing function algorithm from Theorem 4.6 restricted to the case where all non-rooted vertices have degree 3.

4.5. **Properties of the SPO drawings.** In this subsection we discuss some additional properties of the SPO drawings obtained by the increasing function algorithm. In previous subsections, we have investigated the shapes of the embedded faces, and we also remarked that an edge connecting two degree 3 vertices must be straight. Here we investigate some further orientation properties of embedded edges. These properties will provide some intuitions for the bend-minimization method discussed in Section 5.3.

There is a particular type of edges we are interested in. We say that a non-root arc $a = (u, v)$ is *special* if we have

$$(\text{right-init}(a), \text{right-term}(a), \text{left-term}(a), \text{left-init}(a)) = (k, k, k+2, k+3)$$

for some $k \in \{1, 2, 3, 4\}$. If one arc of an edge e is special, then we also call e *special*. See Figure 36(a) for an example.

Special edges satisfy some unusual properties. For one thing, from Condition (L3*) of the definition of dual 4-GS labelings, one can easily see that if an arc $a = (u, v)$ is special, then u must be a vertex of degree 4 and v of degree 3. Furthermore, using the terminology of i-runs introduced in Subsection 4.3, one can easily verify that each run around a face contains at most one special edge, and if there is one then it has to be the last edge on that run (in clockwise order).

We now state the orientation properties of edges in our drawings.

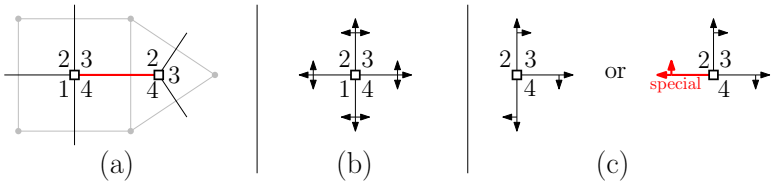


FIGURE 36. (a) A special edge with labels $\{4, 4, 2, 3\}$. (b) Orientation properties around vertices of degree 4. (c) Orientation properties around vertices of degree 3 with incident labels $\{2, 3, 4\}$ depending on whether it is incident to a special edge or not (the red arc is part of a special edge).

Proposition 4.11. *An SPO drawing obtained by the increasing function algorithm satisfies the following properties:*

- *An arc whose initial vertex has degree 3 never turns left: it is either drawn as a straight segment, or it turns right.*
- *An arc with a unique color k is drawn so that it starts by going in direction k.*
- *An arc with colors k and $k+1$ is drawn so that it starts by going in direction k, unless it is part of a special edge, in which case it starts by going in direction $k+1$.*

Observe that the edge properties listed in Proposition 4.11 imply that the situation around vertices is as illustrated in Figure 36(b-c).

Proof. Let us prove the first property. First, by $\mathcal{L}^*$-rule for drawing edges, an arc with more than one color will be drawn as a straight segment. Now suppose an arc $a = (u, v)$ has a unique color k, and u has degree 3. For it to turn left, the opposite arc $-a$ needs to have a unique color $k+1$, but this violates Condition (L3*) of dual 4-GS labelings.

The later two properties can be verified using a similar analysis as in previous subsections. □

4.6. **On allowing vertices of degree 2.** In this subsection we explain how to draw graphs having some vertices of degree 2.

For an arbitrary graph with a vertex v of degree 2, we call *erasing* v the operation of deleting v and replacing the 2 incident edges $\{v, u_1\}$ and $\{v, u_2\}$ by a single edge $\{u_1, u_2\}$. See figure 37(a).

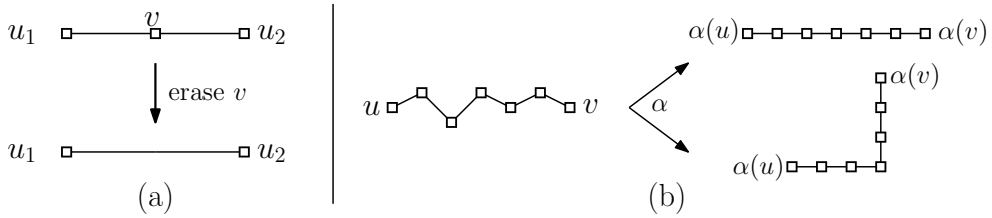


FIGURE 37. (a) Erasing a vertex of degree 2. (b) Two possible embeddings of a chain with five vertices of degree 2.

Now, let G° be a planar map whose vertex degrees are in $\{2, 3, 4\}$, with a designated root vertex v_∞. For simplicity let us suppose v_∞ is not incident to any vertex of degree 2. Let G^* be the planar map obtained by erasing all the vertices of degree 2. Then every non-root edge of G^* corresponds to a path of edges in G°.

Assuming that G^* is a dual-adapted rooted 3,4-map, we claim that we can easily obtain a grid SPO drawing for G°. This takes two steps:

(1) Compute a dual 4-GS labeling on G^*, and run the increasing function algorithm with an additional requirement: if a non-root edge e of G^* corresponds to a path in G° with N_e edges, then e needs to be drawn with total length at least N_e (either as a straight segment or as a bent-edge).
(2) Insert the vertices of degree 2 on the edges of G^* they belong to. Due to the additional requirement in Step (1), they all can be placed at grid points.

The additional requirement in Step (1) is easy to achieve: for each edge e of G^*, record the length N_e of the corresponding path in G°. Then, when computing the increasing functions, make sure that the increments along e in $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$ sum to at least N_e. This ensures that the total length of e in the drawing is at least N_e. See Figure 37(b) for an illustration.

4.7. **Time complexity.** In this subsection we discuss the time complexity and the grid size that can be achieved by the increasing function orthogonal drawing algorithm.

Let G^* be a dual-adapted rooted 3,4-map. To perform the increasing function orthogonal drawing algorithm one needs to:

(i) compute a dual 4-GS labeling $\mathcal{L}^*$, and the corresponding orientations $\overrightarrow{A}_o^*$, $\overrightarrow{A}_e^*$, and
(ii) compute some integer-valued $\overrightarrow{A}_o^*$-increasing and $\overrightarrow{A}_e^*$-increasing functions h_o^* and h_e^* that are *tight* (recall the definition from Section 3.4).

Step (i) can be completed in linear time since it was proved in [BFL24] that 4-GS labeling could be completed in linear time. Step (ii) can be completed using a similar procedure as in the primal case, which is elaborated in Section 3.4. We summarize these claims into the following statement.

Proposition 4.12. *The increasing function orthogonal drawing algorithm for drawing a dual-adapted rooted 3,4-map can be completed in linear time in the number of vertices.*

We mention that the time complexity remains linear when allowing for vertices of degree 2 using the method discussed in Section 4.6.

4.8. **Bound on the grid size.** In this subsection we discuss the size of the grid needed for the orthogonal grid-drawing algorithms.

Let G^* be a dual-adapted rooted 3,4-map, and let $\mathcal{L}^*$ be a dual 4-GS labeling. Denote by $m_o^* \times m_e^*$ the grid size resulting from the tight increasing functions associated to $\mathcal{L}^*$. Note that m_o^* is the maximal length of directed paths in $\overrightarrow{A}_o^*$, and m_e^* is the maximal length of directed paths in $\overrightarrow{A}_e^*$. In general, it seems difficult to estimate m_o^* and m_e^*. However, an obvious bound is $m_o^* \leq \mathbf{v}(\overrightarrow{A}_o^*)-1$, where $\mathbf{v}(\overrightarrow{A}_o^*)$ is the number of vertices of $\overrightarrow{A}_o^*$, and similarly $m_e^* \leq \mathbf{v}(\overrightarrow{A}_e^*)-1$, where $\mathbf{v}(\overrightarrow{A}_e^*)$ is the number of vertices of $\overrightarrow{A}_e^*$.

Now, recall from Section 4.2 the duality relations between $\widetilde{A}_o^*$ and $B'_e = B_e \cup U_e$, and between $(\widetilde{A}_e^*)^-$ and $B'_o = B_o \cup U_o$ (see Remark 4.9). This implies $m_o^* \leq \mathbf{f}(B'_e) - 1$ and $m_e^* \leq \mathbf{f}(B'_o) - 1$, where $\mathbf{f}(B'_e)$ and $\mathbf{f}(B'_o)$ are the number of inner faces of B'_e and B'_o. Since $B'_e = B_e \cup U_e$ and $B'_o = B_o \cup U_o$ we get

$$\mathbf{f}(B'_e) = \mathbf{f}(B_e) + |U_e| \text{ and } \mathbf{f}(B'_o) = \mathbf{f}(B_e) + |U_o|,$$

where $\mathbf{f}(B_e)$ is the number of inner faces of B_e, and $\mathbf{f}(B_o)$ is the number of inner faces of B_o.

It was shown in Section 3.5 that $\mathbf{f}(B_e) = \mathbf{v} + 1 - \mathbf{d}_e$, and $\mathbf{f}(B_o) = \mathbf{v} + 1 - \mathbf{d}_o$, where $\mathbf{v}$ is the number of inner vertices of the primal map G, and $\mathbf{d}_e$ (resp. $\mathbf{d}_o$) is the number of inner edges of G with colors $\{2, 4\}$ (resp. colors $\{1, 3\}$). Combining these results and expressing them in terms of parameters of G^* gives the following bound for the grid size.

Proposition 4.13. *Let G^* be a dual-adapted rooted 3,4-map, and let $\mathcal{L}^*$ be a dual 4-GS labeling. The increasing function orthogonal drawing algorithm produces a grid drawing of G^* on a grid of size $m_o^* \times m_e^*$, with*

$$m_o^* \leq \boldsymbol{f}^* - \mathbf{d}_o^* + \mathbf{u}_o^*, \text{ and } m_e^* \leq \boldsymbol{f}^* - \mathbf{d}_e^* + \mathbf{u}_e^*,$$

where $\boldsymbol{f}^*$ (resp. $\mathbf{d}_o^*, \mathbf{d}_e^*, \mathbf{u}_o^*, \mathbf{u}_e^*$) is the number of non-root faces (resp. edges of color $\{1,3\}$, edges of color $\{2,4\}$, fully-colored edges with labels $\{1,1,3,3\}$, fully-colored edges with labels $\{2,2,4,4\}$) of G^*.

In the case where G is a quadrangulation and the 4-GS labeling $\mathcal{L}$ is even, there are no edges of G^* with exactly two colors of the same parity nor any fully-colored edges, hence m_o^* and m_e^* are both bounded by $\mathbf{f}^*$. This bound matches the grid size of the Bernardi and Fusy's algorithm [BF12] (before the compaction step).

In the case where G is a triangulation, $\mathbf{d}_o^* + \mathbf{d}_e^*$ is equal to the number e_2 of edges of G bearing exactly 2 colors, while $\mathbf{u}_o^* + \mathbf{u}_e^*$ is equal to the number e_0 of edges of G bearing no colors, and the calculation in Section 3.5 gives $\mathbf{d}_o^* + \mathbf{d}_e^* = \mathbf{f}^* - 1 + \mathbf{u}_o^* + \mathbf{u}_e^*$. In this case, the bound on the half-perimeter of the drawing is simply $m_o^* + m_e^* \leq \mathbf{f}^* + 1$, which matches the bound given by He's algorithm [He93], a bound also achieved by the algorithm in [RNN98] with a different approach.

Remark 4.14. Our drawings have the property that every grid line is occupied by at least one vertex. In that case, letting $\mathbf{v}^*$ be the number of non-root vertices, and s_h (resp. s_v) be the number of straight horizontal (resp. vertical) edges, An easy argument (see [Bie96]) then ensures that the grid-width is at most $\mathbf{v}^* - 1 - s_v$ and the grid-height is at most $\mathbf{v}^* - 1 - s_h$. This matches the bounds for m_o^* and m_e^* from Proposition 4.13, since $\mathbf{v}^* - s_v = \mathbf{v}(\overrightarrow{A}_o^*)$ (contracted edges in $\overrightarrow{A}_o^*$ are those counted by s_v), and similarly $\mathbf{v}^* - s_h = \mathbf{v}(\overrightarrow{A}_e^*)$.

Remark 4.15. There is another well-known approach, due to Tamassia [Tam87, Theorem 1], to obtain a grid realization of a given (combinatorial) plane orthogonal drawing. It consists in refining the orthogonal drawing into a (combinatorial) rectangular drawing by suitable insertions of vertices and edges, and then applying He's algorithm [He93] to obtain a grid realization of the rectangular drawing, which yields a grid realization of the original orthogonal drawing. Since every grid-line is occupied by at least one vertex, the bounds of Proposition 4.13 also hold with this strategy. Our approach shows that one can directly apply an increasing function strategy in order to obtain a grid realization, without the preliminary refinement step.

We now state some further bounds on the number of straight edges and grid size in terms of parameters of the planar map solely.

Proposition 4.16. *Let G^* be a dual-adapted rooted 3,4-map, let $\mathbf{e}^*$ be its number of non-root edges, and let $\mathbf{v}_3^*$ (resp. $\mathbf{v}_4^*$) be its number of vertices of degree 3 (resp. non-root vertices of degree 4).*

For any dual 4-GS labeling $\mathcal{L}^*$, the increasing function orthogonal drawing algorithm produces a grid drawing of G^* whose number of straight edges is at least $\mathbf{v}_3^*/2$ and number of bent edges is at most $4\mathbf{v}_4^*$. Consequently, the semi-perimeter of the grid is at most $\min(\mathbf{e}^*, \mathbf{e}^* + 2\mathbf{v}_4^* - \mathbf{v}_3^* + 2)$.

Moreover, upon choosing $\mathcal{L}^*$ appropriately (linear time), the number of bent edges is at most $3\mathbf{v}_4^*$ and the semi-perimeter of the grid is at most $\min(\mathbf{e}^*, \mathbf{e}^* + \mathbf{v}_4^* - \mathbf{v}_3^* + 2)$.

Proof. The second and third items in Proposition 4.11 imply that, for every vertex of degree 3, the arc with two colors is on a straight edge. Hence the number of straight edges is at least $\mathbf{v}_3^*/2$. On the other hand, the first item in Proposition 4.11 ensures that every bent edge turns left from a non-root vertex of degree 4. Hence, the number of bent edges is at most $4\mathbf{v}_4^*$.

Then it follows from Biedl's bound (see Remark 4.14) that the semi-perimeter is at most

$$\min(2\mathbf{v}^* - 2 - \mathbf{v}_3^*/2, 2\mathbf{v}^* - 2 - \mathbf{e}^* + 4\mathbf{v}_4^*) = \min(\mathbf{e}^*, \mathbf{e}^* + 2\mathbf{v}_4^* - \mathbf{v}_3^* + 2),$$

where $\mathbf{v}^* = \mathbf{v}_4^* + \mathbf{v}_3^*$, and we used $\mathbf{e}^* = 2\mathbf{v}_4^* + \frac{3}{2}\mathbf{v}_3^* - 2$.

Finally, we observe that for any vertex of degree 4 with contribution 4 to the number of bent edges, one can reduce this contribution to 0 by decreasing by 1 (mod 4) the labels at the 4 corners around that vertex. These operations can be performed simultaneously at all such vertices, and yield a new dual 4-GS labeling for which each vertex of degree 4 has contribution at most 3 to the number of bends. Thus, the number of bent edges after these operations is at most $3\mathbf{v}_4^*$, and the grid semi-perimeter is at most $\min(\mathbf{e}^*, \mathbf{e}^* + \mathbf{v}_4^* - \mathbf{v}_3^* + 2)$. □

Remark 4.17. The bound $\mathbf{e}^*$ on the semi-perimeter is also satisfied by the algorithm of Biedl and Kant [BK98], which applies to the (larger) family of 2-connected planar maps with vertex degrees at most 4. However, edges can have up to 2 bends with that algorithm.

Remark 4.18. The bound $3\mathbf{v}_4^*$ for the number of bends of the drawing of G^* (in Proposition 4.16) shows that our algorithm produce a drawing at distance at most $3\mathbf{v}_4^*$ from the optimal bend-number over all plane orthogonal drawings of G^*. Having distance at most $3\mathbf{v}_4^*$ to the optimal bend-number also appears in the orthogonal drawing algorithm given in [BLK18] (building on [RNN99]) for 3-connected planar maps with vertex degrees at most 4 (a family which intersects ours but without inclusion of one into the other), where however the optimal bend number can itself be larger than $3\mathbf{v}_4^*$.

4.9. **Reduction of the grid size.** In this subsection we discuss two methods to reduce the grid size of the increasing function orthogonal drawing algorithm. Similarly as for the primal case, these improvements will be obtained by relaxing some constrains for the increasing functions h_o^*, h_e^*.

Remark 4.19. Throughout this section we work with a fixed dual 4-GS labeling. We note however that in [He93], the author discussed some methods to reduce the grid size by changing the transversal structure. We leave as an open question whether these methods can be generalized.

Relaxing the equalities for edges in $B_o^* \setminus A_o^*$ and $B_e^* \setminus A_e^*$: The first optimization we present is about relaxing some equalities for the increasing functions h_o^*, h_e^*. Recall that the increasing functions h_o^* (resp. h_e^*) are defined on all non-root vertices of G^*, and are constant along edges in $\overline{A_o^*}$ (resp. $\overline{A_e^*}$). We will now relax this condition for some of the edges in $B_o^* \cap \overline{A_o^*}$ (resp. $B_e^* \cap \overline{A_e^*}$). Recall that $A_o^* \subseteq B_o^*$ and $A_e^* \subseteq B_e^*$. A function g^* defined on the set of non-root vertices of G^* is called B_o^*-*increasing* (resp. B_e^*-*increasing*) if it satisfies the following conditions:

(i) $g^*(u) = g^*(v)$ for every non-root edge $\{u, v\}$ not in B_o^* (resp. not in B_e^*),
(ii) $g^*(u) \leq g^*(v)$ for every arc (u, v) in $B_o^* \setminus A_o^*$ (resp. $B_e^* \setminus A_e^*$),
(iii) $g^*(u) < g^*(v)$ for every arc (u, v) in A_o^* and (resp. A_e^*) and every odd (resp. even) diagonal arc (u, v).

The only difference between the definition of B_o^*-increasing functions and $\overrightarrow{A}_o^*$-increasing functions is that for $\overrightarrow{A}_o^*$-increasing functions the requirement $g^*(u) \leq g^*(v)$ in Condition (ii) would be replaced by $g^*(u) = g^*(v)$. Hence, any $\overrightarrow{A}_o^*$-increasing function is B_o^*-increasing. Similarly, any $\overrightarrow{A}_e^*$-increasing function is B_e^*-increasing. Hence replacing $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$-increasing functions by B_o^* and B_e^*-increasing functions in the increasing function algorithm could potentially reduce the grid size, as long as we can show this does not break planarity. We will indeed prove the following result.

Proposition 4.20. *Let G^* be a dual-adapted rooted 3,4-map and let $\mathcal{L}^*$ be a dual 4-GS labeling. Let g_o^* be a B_o^*-increasing function, and let g_e^* be a B_e^*-increasing function. Placing each non-root vertex v at coordinates $(g_o^*(v), g_e^*(v)) \in \mathbb{R}^2$ (and drawing each non-root edges following the rules specified below) yields an SPO drawing of G^*.*

Proposition 4.20 is illustrated in Figure 38. In that example, using B_o^* and B_e^*-increasing functions (instead of $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$-increasing functions) reduces the horizontal dimension of the drawing by 1.

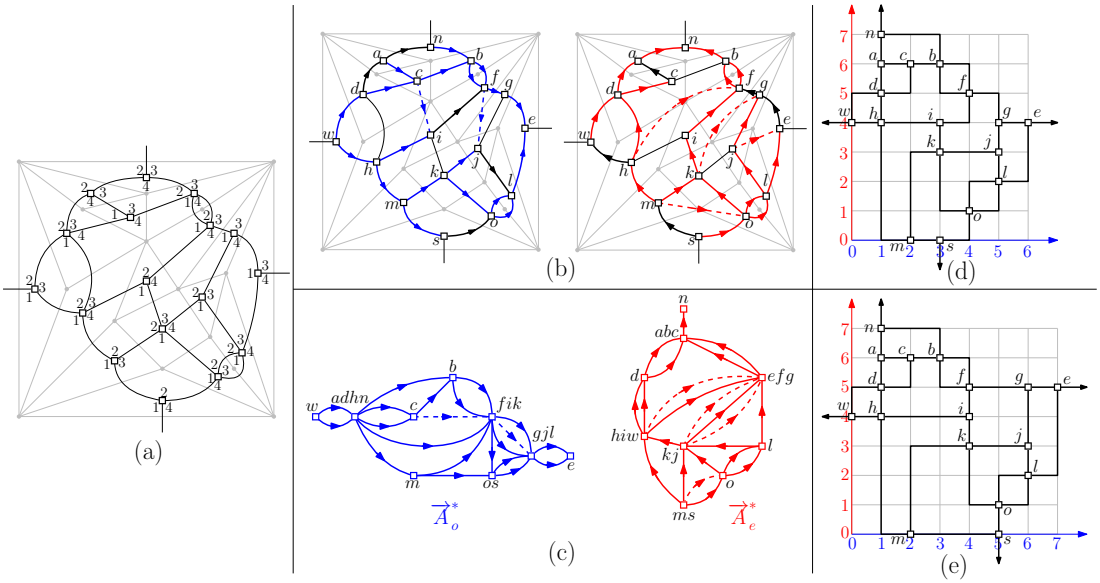


FIGURE 38. (a) A rooted 3,4-map with a dual 4-GS labeling. (b) The bipolar orientations B_o^* and B_e^* together with the diagonal arcs. (c) The augmented bipolar contractions $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$. (d) The drawing output using the B_o^* and B_e^*-increasing functions. (e) The drawing output using the $\overrightarrow{A}_o^*$ and $\overrightarrow{A}_e^*$-increasing functions.

Proof. The proof of Proposition 4.20 is similar to the one of Theorem 4.6. We need to check that the drawing is still valid with the relaxed condition on the coordinate functions.

Let f be a non-root face of G^*. Recall the definition of the i-runs of f from Section 4.3. Using a similar analysis as in Section 4.3, one can conclude that the 4-run is still drawn weakly right-up; the only difference is that arcs that were previously drawn as a straight segment (going either up or right) may now be drawn as a bent-edge going right-up. This is illustrated in Figure 39(a). This figure also gives the rule for drawing the edge in each situation. By symmetry, the 1-run (resp. 2-run, 3-run) is drawn weakly down-right, (resp. left-down, up-left).[5]

[5]We mention, that the rule for drawing fully-colored edges (for instance an edge that is on a 4-run of a face, and a 2-run of the opposite face) cannot be made fully symmetric: the bend has to be convex for one of the adjacent face, unlike the other edges on i-runs. However, the precise rule for drawing these edges is irrelevant for the planarity argument given below.

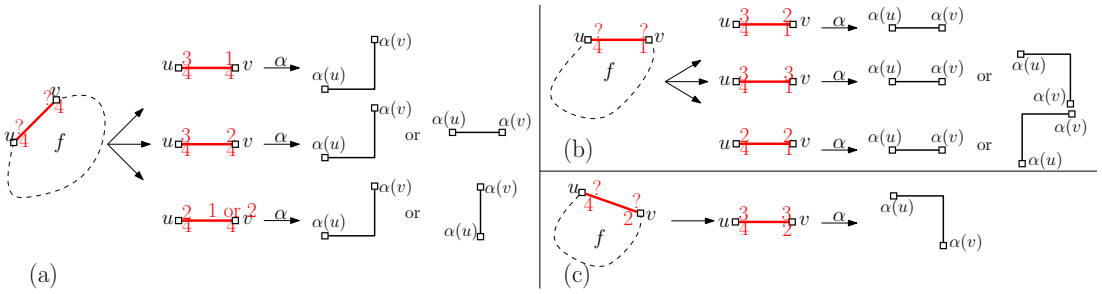


FIGURE 39. SPO drawing with B_o^*, B_e^* increasing coordinates. (a) Embedding of edges with labels $\{4,4\}$ inside a face f. (b) Embedding of edges with labels $\{4,1\}$ inside f. (c) Embedding of edges with labels $\{4,2\}$ inside f.

Moreover an edge connecting the 4-run and the 1-run of f (that is, an edge incident with corner labels $\{4,1\}$ around f) is drawn "rightward", that is, either right, or right-up or down-right. This is illustrated in Figure 39(b). This orientation property ensures that the 4-run and 1-run do not touch each other. By symmetry, consecutive runs cannot touch each other. Figure 39 also gives the rule for drawing the edge connecting the 4-run and the 1-run (if it exists), or the edge connecting the 4-run and the 2-run (if it exists). Note that the rule for drawing edges ensures that if an edge around f is incident to two distinct labels and has a bend, then this bend is convex with respect to f. Therefore these edges cannot intersect with the runs. Lastly, the condition on diagonal arcs prevent opposite runs (e.g. the 4-run and the 2-run) from touching each other. This implies that every inner face has a valid drawing.

We can also investigate the shape of the outer face. As before one can show that the (e_1^*, e_2^*)-run (resp. (e_2^*, e_3^*)-run, (e_3^*, e_4^*)-run, (e_4^*, e_1^*)-run) is directed right-up (resp. down-right, left-down, up-left). This is enough to conclude that the drawing of the outer face is valid by using the same reasoning as before. This implies that the overall drawing is planar. □

Relaxing the inequalities on some diagonal arcs: The second optimization we present is about weakening some inequalities for the B_o^* and B_e^*-increasing functions g_o^*, g_e^*. In the increasing function algorithm of Proposition 4.20, the increasing functions g_o^*, g_e^* are required to be strictly increasing along the diagonal arcs. We claim that some of these inequalities can be relaxed to weak inequalities.

Let G^* be a rooted 3, 4-map, and let $\mathcal{L}^*$ be a dual 4-GS labeling. We call *quasi B_o^*-increasing* a function g^* on the non-root vertices of G^* which satisfies all the conditions of B_o^*-increasing functions except that for odd diagonal edges (u, v) the condition $g^*(u) < g^*(v)$ is replaced by $g^*(u) \leq g^*(v)$. We define *quasi B_e^*-increasing* functions similarly.

Note that using quasi B_o^* and B_e^*-increasing coordinates (instead of strict B_o^* and B_e^*-increasing coordinates) could lead to a non-planar drawing. Indeed, recall that the condition on diagonal arcs was used to ensure the planarity of the drawing of each non-root face (precisely, these conditions ensure that opposite runs around a face do not intersect). But it is clear that in the example represented in Figure 40(a), the weakening of the inequality for the diagonal arc $(u, v) \in \overrightarrow{A}_o^*$ could lead to a violation of planarity (the 4-run could intersect the 2-run).

We now identify which diagonals arcs can be weakened. Consider an SPO drawing of a rooted 3, 4-map G^* obtained by some B_o^* and B_e^*-increasing functions (g_o^*, g_e^*). Let f be a non-root face of G^* containing a diagonal arc $(u, v) \in \overrightarrow{A}_o^*$ (so that the 4-run and the 2-run of f are non-empty). Let C_4 be the lowest point with the same abscissa as u on the drawing of

the 4-run of f, and let C_2 be the highest point with the same abscissa as v on the drawing of the 2-run of f. The points C_2 and C_4 are indicated in Figure 40. We emphasize that C_4 and C_2 are geometric features of the embedding of f, not features of f itself (in particular, they may correspond to actual vertices on f or just bends of the drawings of edges). It is easy to see that as long as C_4 is strictly above C_2, then the inequality $g_o^*(u) < g_o^*(v)$ induced by the diagonal arc (u, v) can be weakened (that is, replaced by the condition $g_o^*(u) \leq g_o^*(v)$). This is illustrated in Figure 40(b). In this case, we say that the diagonal arc (u, v) is $\overrightarrow{A}_o^*$*-weak*.

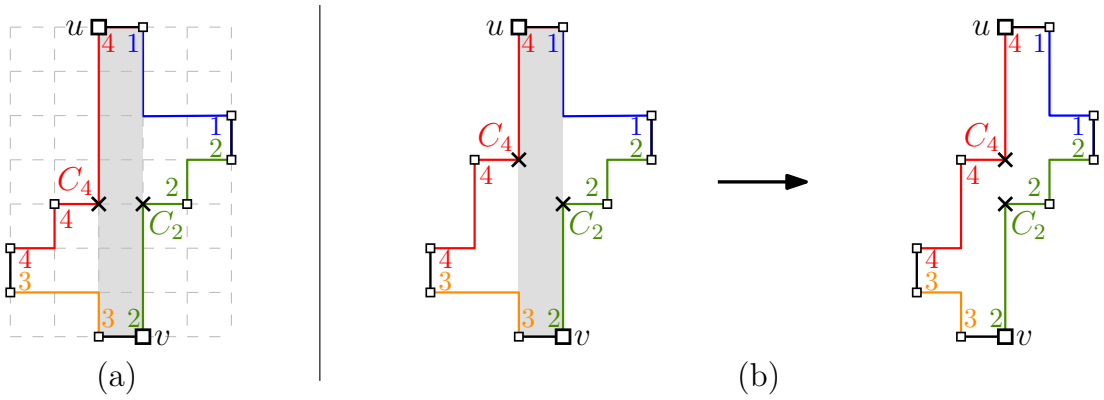


FIGURE 40. (a) A face for which the diagonal arc $(u, v) \in \overrightarrow{A}_o^*$ cannot be weakened. (b) If C_4 is strictly above C_2, then (u, v) can be weakened.

Note that the notion of $\overrightarrow{A}_o^*$-weakness for a diagonal arc in $\overrightarrow{A}_o^*$ actually depends on the B_e^*-increasing function g_e^*. Hence this type of optimization needs to be performed one coordinate at a time. Given an SPO drawing of G^* where the coordinate functions (g_o^*, g_e^*) are quasi B_o^*-increasing and B_e^*-increasing respectively, we can define $\overrightarrow{A}_e^*$*-weak* diagonals similarly as above: for a diagonal arc $(u, v) \in \overrightarrow{A}_e^*$ in a face f of G^* we consider the rightmost point C_3 with the same ordinate as u on the drawing of the 3-run of f and the leftmost point C_1 with same ordinate as v on the drawing of the 1-run of f, and say that the diagonal (u, v) is $\overrightarrow{A}_e^*$*-weak* if C_3 is strictly to the left of C_1. In this case the inequality $g_e^*(u) < g_e^*(v)$ can be weakened without breaking planarity.

This suggests the following optimized SPO drawing algorithm:

(1) Compute a (strict) B_e^*-increasing function $\widetilde{g_e^*}$, and find all the $\overrightarrow{A}_o^*$-weak diagonal arcs.
(2) Compute a quasi B_o^*-increasing function g_o^*, which is weakly increasing along $\overrightarrow{A}_o^*$-weak diagonal arcs (and strictly increasing along the other diagonal arcs).
(3) Compute a quasi B_e^*-increasing function g_e^*, which is weakly increasing along $\overrightarrow{A}_e^*$-weak diagonal arcs (and strictly increasing along the other diagonal arcs).
(4) Draw the graph G^* using the coordinate function (g_o^*, g_e^*).

The above discussion shows that the result of this algorithm is planar (since the validity of the drawing of the inner faces is preserved, and the validity of the drawing of the outer face does not depend on diagonal arcs).

A drawing obtained using this optimized algorithm is represented in Figure 41. In this example, the weakening of diagonal edges inequality leads to a reduction of the vertical dimension by 1.

Remark 4.21. This optimization strategy is analogous to the reduction operations used in the 4-valent case [BF12].

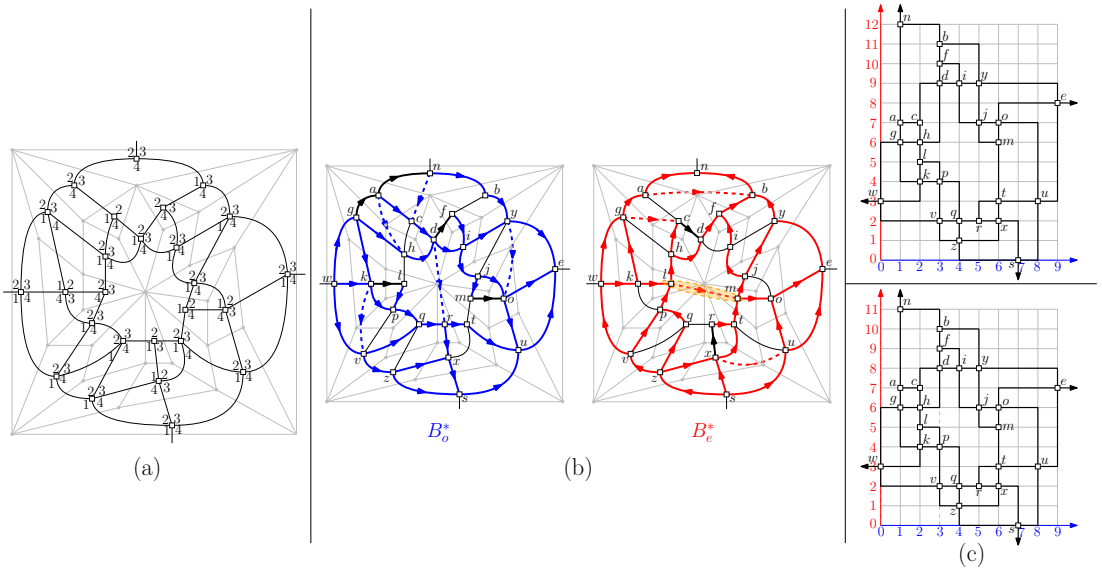


FIGURE 41. (a) A dual-adapted rooted 3,4-map with a 4-GS labeling. (b) The bipolar orientations B_o^* and B_e^* plus the diagonal arcs. Those edges in $B_o^* \cap \overline{A}_o^*(B_e^* \cap \overline{A}_e^*)$ are colored as black. The odd diagonal arc (l, m) is weak. (c) Top: The drawing output using strict B_o^* and B_e^*-increasing functions. Bottom: The drawing output using the weak B_o^* and B_e^*-increasing functions obtained using the algorithm outlined above.

5. Additional remarks and optimization

5.1. **Variants on the face counting algorithms.** The face-counting algorithms given in Sections 3.1 and 4.1 admit some variants that we now describe.

Consider the primal picture first. Let G be an adapted 3,4-angulation of the square and let $\mathcal{L}$ be a 4-GS labeling of G. In the face-counting algorithm of Section 3.1, each vertex v is placed at the point $(\ell_e(v), r_o(v))$, where $r_o(v)$ is the number of inner faces of B_o at the right of $P_o(v)$ and $\ell_e(v)$ is the number of inner faces of B_e at the left of $P_e(v)$. Given some positive weights associated to each inner face of B_o the B_e, one can consider the vertex placement $(\ell'_e(v), r'_o(v))$ where $r'_o(v)$ is the sum of the weights of the inner faces of B_o at the right of $P_o(v)$ and $\ell'_e(v)$ is the sum of the weights of the inner faces of B_e at the left of $P_e(v)$. We claim that this vertex placement lead to a planar drawing of G for any choice of positive weight. This generalizes Theorem 3.1 which corresponds to a weight of 1 for every face. Indeed, one can prove that $\ell'_e(v)$ is an $\overrightarrow{A}_o$-increasing function and $r'_o(v)$ is an $\overrightarrow{A}_e$-increasing function exactly as before; hence planarity is again a consequence of Theorem 3.3. Similarly one could define some other placement $(\ell''_e(v), r''_o(v))$ using vertex-counting instead, as they would still be a special case of the increasing function algorithm.

In the dual picture one can also consider variants of the face-counting algorithm by assigning weights to the faces of B_o^* and B_e^*; or one could define vertex-counting variants as all of these variants fall under the "increasing function" umbrella.

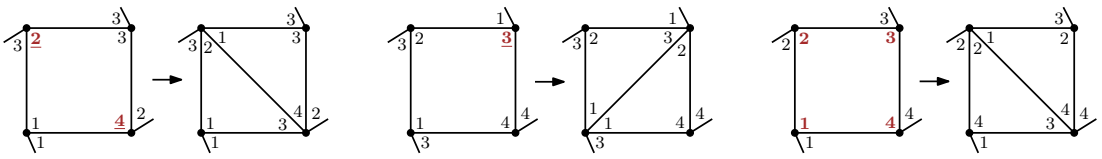


FIGURE 42. Some cases for the configuration of special/non-special labels in a quadrangular inner face, where special labels are bold colored, with the diagonal ones underlined.

Remark 5.1. The face-counting algorithm of Section 3.1 is somewhat reminiscent of Schnyder's famous drawing algorithm for triangulations [Sch90]. The relation is even closer for a variant of the algorithm that we now present. Let G be an adapted 3,4-angulation of the square with $\mathbf{f}$ inner faces, and let $\mathcal{L}$ be a 4-GS labeling of G. Consider the vertex placement given by $\alpha'(v) = (\ell'_e(v), r'_o(v))$, where $r'_o(v)$ is the number of inner faces of G at the right of $P_o(v)$ and $\ell'_e(v)$ is the number of inner faces of G at the left of $P_e(v)$. The analogue of Theorem 3.1 holds for this vertex placement. Indeed it corresponds to one of the face-weighted variants described above (where the weight of a face of B_o or B_e is the number of faces of G that it contains). Now, for a vertex v of G, we can consider the paths $P_1(v), \dots, P_4(v)$ in $W_1, \dots, W_4$ between v and v_4, v_1, v_2, v_3 respectively. These paths are non-crossing as proved in [BFL24], and denoting by $r_i(v)$ the number of inner faces between $P_{i-1}(v)$ and $P_{i+1}(v)$ one gets $r'_o(v) = r_2(v) + r_3(v)$ and $\ell'_e(v) = r_3(v) + r_4(v)$. Hence, in barycentric coordinates, one has

$$\alpha'(v) = \lambda_1(v)\, v_1 + \lambda_2(v)\, v_2 + \lambda_3(v)\, v_3 + \lambda_4(v)\, v_4,$$

where $v_1 = (0,0)$, $v_2 = (0, \mathbf{f})$, $v_3 = (\mathbf{f}, \mathbf{f})$, $v_4 = (\mathbf{f}, 0)$, and $\lambda_i(v) = \frac{r_i(v)}{\mathbf{f}}$ for all i. This barycentric vertex placement given by the number of inner faces between consecutive Schnyder wood paths is a direct analogue of Schnyder's algorithm [Sch90].

5.2. **Triangulating under the 4-connectivity constraint.** Given G, an adapted 3,4-angulation of the square, it is always possible to add a diagonal in each quadrangular inner face in such a way that the resulting triangulation of the square is adapted. Indeed, it is easy to see that for every quadrangular inner face of G, at least one of the two diagonals can be added without creating a separating 3-cycle. As we now explain, if G is endowed with a 4-GS labeling then this decision can be made very easily, and moreover the 4-GS labeling can be maintained all along, in time $O(1)$ for each edge addition, to end with an irreducible triangulation of the square endowed with a 4-GS labeling.

Let f be a quadrangular inner face of G. Let c be a corner incident to f, let e be the edge visited after c in clockwise order around f, and let c' be the corner incident to e opposite to c (i.e., at the other extremity and on the other side). The corner c is called *special* if the label-jump from c to c' is 1. A corner is called *diagonal* if it is special and preceded by a non-special corner in clockwise order around f. With c_1, c_2, c_3, c_4 the four corners around f (the index indicating the label), we choose a pair c_i, c_{i+2} of opposite corners such that the other corners c_{i+1} and c_{i-1} are not diagonal; such a pair exists since f can not have two consecutive diagonal corners.

We then perform the following operations, see Figure 42 for examples:

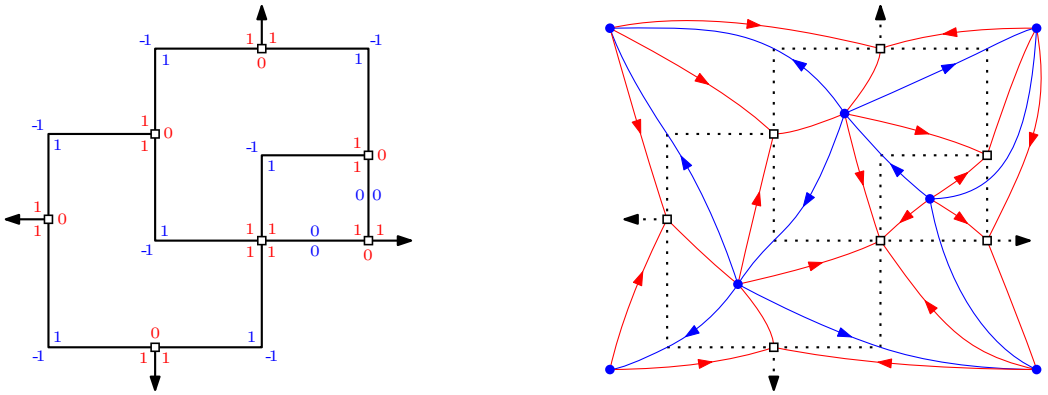
(1) decrease by 1 the labels of the special corners around f,
(2) insert a diagonal e in f connecting the vertex v at c_i and the vertex v' at c_{i+2}, which creates a new corner at v (resp. at v'), the new corner being considered as the one on the right of e, looking from v (resp. from v'),
(3) give label i (resp. $i+2$) to the new corner at v (resp. v').

The conditions of a 4-GS labeling are still satisfied, the crucial point being that the labels of special corners have been decreased in order for Condition (L3) to remain satisfied after adding the diagonal [6]. Moreover, with the notation of Section 3.5, any edge contributing to $\mathbf{d}_e$ will keep contributing to $\mathbf{d}_e$ all along subsequent edge-additions, and similarly for $\mathbf{d}_o$. Hence, the bounds on the grid width and height in Proposition 3.12 are weakly decreasing when adding diagonals by this process (however the actual optimized grid-size may increase).

Let us mention two related previous works. Biedl and Kant have given in [BKK97] a linear-time algorithm that, starting from a planar map with no separating triangle, adds edges incrementally while avoiding the creation of separating triangles, yielding a 4-connected triangulation of the original planar map. More closely related, in [Fus06] it is shown that, starting from a simple quadrangulation endowed with a separating decomposition, there is a simple process to add diagonals within faces to yield an irreducible triangulation of the square endowed with a transversal structure. What we describe here can be seen as an extension of this process to adapted 3,4-angulations of the square.

5.3. **Bend minimization.** In this subsection we explain how to minimize the number of bends in the orthogonal drawings obtained by our increasing function algorithm for rooted 3,4-maps. More precisely, given a dual-adapted rooted 3,4-map G^*, we want to determine a dual 4-GS structure which leads to a drawing with the least number of bends. For this, we will adapt a method pioneered by Tamassia [Tam87].

Definition 5.2. *An SPO drawing of a rooted 3,4-map G^* is* one-bend *if every edge has at most one bend. We use the abbreviation* BSPO *for such a drawing of G^*. Two BSPO drawings of G^* are called* equivalent *if they represent the same rooted planar map, and each half-edge has the same direction in both drawings. The* combinatorial type *of a BSPO drawing is the equivalence class for this relation.*



(a) (b)

FIGURE 43. (a): a BSPO drawing of a rooted 3,4-map G^*, with the encoding of the combinatorial type by a Tamassia assignment. (b): the encoding by a valid Tamassia flow: the arrows indicate the arcs of $G^{\boxtimes}$ for which the flow is 1.

[6] All edge-additions can actually be performed simultaneously, since the status special/non-special of a corner in a quadrangular face is not affected by label changes in other faces.

As shown in Figure 43(a), and following Tamassia [Tam87], the combinatorial type of a BSPO drawing can be encoded by assigning values in $\{-1, 0, 1\}$ at vertex-corners and at edge-sides, with value 1 (resp. 0, -1) indicating an angle of $\pi/2$ (resp. π, $3\pi/2$). It was shown in [Tam87] that such an assignment encodes a valid combinatorial type (non-empty equivalence class of drawings) if and only if the following conditions are satisfied:

- the sum of values at corners around each vertex v is $2\deg(v) - 4$,
- the sum of edge-side values at each edge is 0,
- the sum of values in each inner face is 4,
- with the outer contour partitioned by the dangling half-edges, the sum of values in each of the 4 portions is 1.

Note that these conditions are clearly necessary in order to be able to use the assignment to consistently propagate the directions among {west, north, east, south} to all half-edges after fixing the direction of the 4 dangling half-edges.

We call *Tamassia assignment* such an assignment of values to corners and edge-sides of G^*. A Tamassia assignment for G^* can equivalently be formulated as a flow on a related graph $G^{\boxtimes}$, as shown in Figure 43(b). Let $G^{\boxtimes}$ be the graph obtained from G^* by

- placing a *blue vertex* v_f in each face f of G^* and tracing a *blue edge* (joining 2 blue vertices) across each edge of G^*,
- adding a *red edge* between blue vertex v_f and each of the corner of the face f,
- deleting the edges of G^*.

Given an assignment of a *flow* in $\{0, 1\}$ to each arc of $G^{\boxtimes}$, we call *net flow* of a vertex v the sum of the flows of the outgoing arcs at v, minus the sum of the flows of the incoming arcs at v. We call *Tamassia flow* an assignment of a *flow* in $\{0, 1\}$ to each arc of $G^{\boxtimes}$ such that

- every vertex v of G^* has net flow $4 - 2\deg(v)$,
- every blue vertex v_f has net flow 4 if f is a non-root face, and net flow 1 otherwise.

It is easy to see that the Tamassia assignments are in bijection with the Tamassia flows such that if an arc of $G^{\boxtimes}$ has flow 1, then the opposite arc has flow 0; such Tamassia flows are called *valid*. We call *min-cost* a Tamassia flow minimizing the sum of the flow values of arcs on blue edges. Note that the min-cost Tamassia flows are valid and they correspond bijectively to the (combinatorial type of) BSPO drawings of G^* minimizing the number of bends. Note that $G^{\boxtimes}$ admits a Tamassia flow (which can be obtained in linear time by applying our drawing algorithm on G^*), hence a min-cost one can be found in time $O(n^{3/2}\log(n))$, where n is the number of vertices of G^* (see [GKHT15] and references therein).

In order to combine the above framework with our algorithm, we need to establish a correspondence between the combinatorial type of the BSPO of G^* and the dual 4-GS labelings of G^*.

As illustrated in Figure 44(a), there is a simple rule to associate to each BSPO drawing of G^* a *4-labeling* of G^*, that is, an assignment of a label in $\{1, 2, 3, 4\}$ to each corner of G^*: each corner is labeled according to the direction of the half-edge delimiting it on the right side, with labels $1, 2, 3, 4$ corresponding to the directions west, north, east and south respectively. This defines a map Φ from the set of combinatorial types of the BSPO of G^* to the set of 4-labelings of G^*.

We now consider some particular types of BSPO drawings.

Definition 5.3. *A BSPO drawing of a rooted 3,4-map is called* right-chiral *if it satisfies the following properties:*

(A1) no edge leaving a vertex of degree 3 turns left,
(A2) each dangling half-edge has a corner of angle $\pi/2$ on its right.

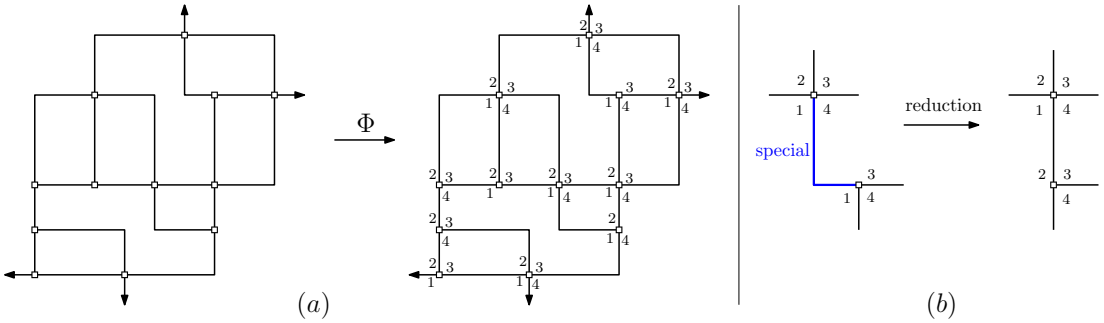


FIGURE 44. (a) Illustrating Φ: A right-chiral BSPO drawing and the corresponding dual 4-GS labeling. (b) Proof that if a dual GS-labeling has a special edge, then it does not correspond to a BSPO type with minimal number of bends.

Note that Condition (A2) is close to redundant: it is a consequence of (A1) when there is at least one vertex of degree 4 on the outer contour. Moreover, Condition (A1) ensures that the edges connecting two vertices of degree 3 have no bend. Note also that when all vertices have degree 4, any BSPO drawing is right-chiral.

Lemma 5.4. *Let G^* be a rooted 3,4-map. The mapping Φ is a bijection between the combinatorial types of right-chiral BSPO drawings of G^* and the dual 4-GS labelings of G^*.*

Proof. Let T be a combinatorial type of right-chiral BSPO drawing, and let $\mathcal{L}^* = \Phi(T)$. It is easy to check that the sum of label-jumps in clockwise order around each vertex is 4, and the sum of label-jumps in counterclockwise order around each edge is 4. Moreover, Property (A1) directly translates to the condition (L3*) at edges. It remains to show that there is no label-jump in each of the four outer faces, and that the sum of clockwise label-jumps in each inner face is 4. Let F_i be the set of inner faces, and F_o the set of 4 outer faces. For $f \in F_i \cup F_o$, let $J(f)$ be the sum of clockwise label-jumps around f, thus a multiple of 4. It is actually a positive multiple of 4 for $f \in F_i$.

Moreover, (A2) ensures that the clockwise label-jumps across the dangling half-edges are $+1$. These are called *root label-jumps*. Letting J_V be the sum of non-root clockwise label-jumps around vertices, we thus have $J_V = 4|V| - 4$. Letting J_E be the sum of counterclockwise label-jumps around edges, we have $J_E = 4|E|$. Each label-jump in counterclockwise around an edge either comes from a non-root label-jump around a vertex, or from a clockwise label-jump around a face. Hence

$$\sum_{f \in F_i \cup F_o} J(f) = J_E - J_V = 4|E| - 4|V| + 4 = 4|F_i|,$$

by the Euler relation. Since $J(f) \geq 4$ for $f \in F_i$, we conclude that $J(f) = 4$ for $f \in F_i$ and $J(f) = 0$ for $f \in F_o$. Hence, $\mathcal{L}^*$ is a dual 4-GS labeling.

Conversely, for a dual 4-GS labeling $\mathcal{L}^*$, applying the local rule of Φ allows to determine the directions of all half-edges, yielding a valid combinatorial type T of BSPO drawing. Condition (L3*) then guarantees that (A1) is satisfied, and Condition (A2) is also clearly satisfied, so that T is right-chiral. And by construction the two mappings are inverse of each other. □

As a consequence of Lemma 5.4, a rooted 3,4-map G^* admits a right-chiral BSPO drawing if and only if it admits a dual 4-GS labeling, and this holds if and only if G^* is dual-adapted[7] . In this case, one can easily adapt the approach described above in order to find a right-chiral BSPO drawing of G^* having a minimal number of bends. Let us call *right-chiral* a Tamassia flow of $G^{\boxtimes}$ such that the flow is 0 on every blue arc of $G^{\boxtimes}$ having a vertex of G^* of degree 3 on its right. Then the combinatorial types of right-chiral BSPO drawings of G^* are in bijection with the right-chiral valid flows of $G^{\boxtimes}$. In this bijection, the combinatorial types of right-chiral BSPO drawings of G^* having a minimal number of bends correspond to the min-cost right-chiral Tamassia flows (as before, min-cost minimizes the sum of the flow values of arcs on blue edges). Moreover such a min-cost right-chiral Tamassia flow can be found in time $O(n^{3/2}\log(n))$, where n is the number of vertices of G^*.

Finally, recall from Section 4.5 that given a dual 4-GS labeling, a non-root arc a is called *special* if

$$(\text{right-init}(a), \text{right-term}(a), \text{left-term}(a), \text{left-init}(a)) = (k, k, k+2, k+3)$$

for some $k \in \{1, 2, 3, 4\}$. We claim that the dual 4-GS labeling corresponding to a min-cost right-chiral Tamassia flow has no special arc. Indeed, if a dual 4-GS labeling $\mathcal{L}^*$ has a special arc a, then one can change one of the labels incident to a and get a new dual 4-GS labeling $\mathcal{L}^*$ which corresponds (via Φ) to a combinatorial type of right-chiral BSPO with one less bend. This is illustrated in Figure 44(b).

Since the dual 4-GS labeling $\mathcal{L}^*$ corresponding to a min-cost right-chiral Tamassia flow has no special arc, Proposition 4.11 ensures that the BSPO of G^* obtained using the increasing function drawing algorithm of Section 4.2 has combinatorial type equal to $\Phi^{-1}(\mathcal{L}^*)$. Thus the obtained BSPO has minimal number of bends among the left-chiral BSPOs of G^*. This is summarized below.

Proposition 5.5. *Let G^* be a dual-adapted rooted 3,4-map with n vertices, and let $\mathbf{T}(G^*)$ be the set of combinatorial types of right-chiral BSPO drawings of G^*. There is an algorithm with time complexity $O(n^{3/2}\log(n))$ for finding an element $T \in \mathbf{T}(G^*)$ with minimal number of bends. Moreover, letting $\mathcal{L}^* = \Phi(T)$ be the associated dual 4-GS labeling, the drawing algorithm of Section 4.2 applied to $\mathcal{L}^*$ yields a grid-drawing realization of T.*

Remark 5.6. Proposition 4.11 ensures that, for any dual 4-GS labeling $\mathcal{L}^*$, the drawing algorithm of Section 4.2 applied to $\mathcal{L}^*$ always yields a right-chiral BSPO drawing of G^* (although its combinatorial type of that BSPO may differ from $\Phi^{-1}(\mathcal{L}^*)$ if $\mathcal{L}^*$ has some special edges). Hence the optimal 4-GS labeling of G^* in Proposition 5.5 is also the one for which the algorithm of Section 4.2 yields the smallest number of bends (among all possible 4-GS labelings of G^*).

Conflict of interest statement: The authors have no conflict of interest to disclose in relation to this article.

Acknowledgments: The authors thank the anonymous referees for their many useful suggestions. Olivier Bernardi was partially supported by NSF Grant DMS-2154242. Éric Fusy was partially supported by the project ANR19-CE48-011-01 (COMBINÉ), and the project ANR-20-CE48-0018 (3DMaps).

[7]It is shown in [FKV14] that a graph with max-degree 4 and a marked vertex v_∞ of degree 4 admits a (possibly non-planar) one-bend suspended orthogonal drawing if and only if the following condition (P) is satisfied: for every vertex-subset S not containing v_∞, there are at most $2|S| - 2$ edges with both ends in S. For consistency one can check that, for a rooted 3,4 map G^*, being dual-adapted implies (P). Moreover, if G^* is 4-regular, then being dual-adapted is equivalent to (P).

References

[BETT99] G. Di Battista, P. Eades, R. Tamassia, and I.G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice-Hall, 1999.

[BF12] O. Bernardi and É. Fusy. Schnyder decompositions for regular plane graphs and application to drawing. *Algorithmica*, 62(3):1159–1197, 2012.

[BFL24] O. Bernardi, É. Fusy, and S. Liang. Grand Schnyder woods. *Annals of Combinatorics*, 2024. arXiv.2303.15630, `https://doi.org/10.1007/s00026-024-00729-8`.

[BH12] L. Barriere and C. Huemer. 4-labelings and grid embeddings of plane quadrangulations. *Discrete Mathematics*, 312(10):1722–1731, 2012.

[Bie96] T. Biedl. Optimal orthogonal drawings of triconnected plane graphs. In *5th Scandinavian Workshop on Algorithm Theory, SWAT*, volume 1097 of *Lecture Notes in Computer Science*, pages 333–344. Springer, Springer, 1996.

[BK98] T. Biedl and G. Kant. A better heuristic for orthogonal graph drawings. *Computational Geometry*, 9(3):159–180, 1998.

[BKK97] T. Biedl, G. Kant, and M. Kaufmann. On triangulating planar graphs under the four-connectivity constraint. *Algorithmica*, 19:427–446, 1997.

[BLK18] S. Bhatia, K. Lad, and R. Kumar. Bend-optimal orthogonal drawings of triconnected plane graphs. *AKCE Int. J. Graphs Comb.*, 15(2):168–173, 2018.

[DFOdMR95] H. De Fraysseix, P. Ossona de Mendez, and P. Rosenstiehl. Bipolar orientations revisited. *Discrete Applied Mathematics*, 56(2-3):157–179, 1995.

[dV03] É. Colin de Verdière. *Shortening of curves and decomposition of surfaces*. PhD thesis, Université Paris 7, 2003.

[Fel01] S. Felsner. Convex drawings of planar graphs and the order dimension of 3-polytopes. *Order*, 18:19–37, 2001.

[FHKO11] S. Felsner, C. Huemer, S. Kappes, and D. Orden. Binary Labelings for Plane Quadrangulations and their Relatives. *Discrete Mathematics & Theoretical Computer Science*, Vol. 12 no. 3, January 2011.

[FKV14] S. Felsner, M. Kaufmann, and P. Valtr. Bend-optimal orthogonal graph drawing in the general position model. *Comput. Geom.*, 47(3):460–468, 2014.

[Fus06] É. Fusy. Straight-line drawing of quadrangulations. In *Graph Drawing, 14th International Symposium (GD)*, volume 4372 of *Lecture Notes in Computer Science*, pages 234–239. Springer, 2006.

[Fus09] É. Fusy. Transversal structures on triangulations: A combinatorial study and straight-line drawings. *Discrete Math.*, 309:1870–1894, 2009.

[GKHT15] A.V. Goldberg, H. Kaplan, S. Hed, and R.E. Tarjan. Minimum cost flows in graphs with unit capacities. In *32nd International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 30 of *LIPIcs*, pages 406–419. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

[He93] X. He. On finding the rectangular duals of planar triangulated graphs. *SIAM Journal on Computing*, 22:1218–1226, 1993.

[NR04] T. Nishizeki and Md. S. Rahman. *Planar Graph Drawing*, volume 12 of *Lecture Notes Series on Computing*. World Scientific, 2004.

[RNN98] Md. S. Rahman, S.-I. Nakano, and T. Nishizeki. Rectangular grid drawings of plane graphs. *Comput. Geom.*, 10(3):203–220, 1998.

[RNN99] Md. S. Rahman, S.-I. Nakano, and T. Nishizeki. A linear algorithm for bend-optimal orthogonal drawings of triconnected cubic plane graphs. *J. Graph Algorithms Appl.*, 3(4):31–62, 1999.

[RT86] P. Rosenstiehl and R.E. Tarjan. Rectilinear planar layouts and bipolar orientations of planar graphs. *Discret. Comput. Geom.*, 1:343–353, 1986.

[Sch89] W. Schnyder. Planar graphs and poset dimension. *Order*, 5(4):323–343, 1989.

[Sch90] W. Schnyder. Embedding planar graphs on the grid. In *Proceedings of the First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 138–148. SIAM, 1990.

[Tam87] R. Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM Journal on Computing*, 16(3):421–444, 1987.

[TT86] R. Tamassia and I.G. Tollis. A unified approach a visibility representation of planar graphs. *Discret. Comput. Geom.*, 1:321–341, 1986.

[TT89] R. Tamassia and I.G. Tollis. Tessellation representations of planar graphs. In *Proc. 27th Allerton Conf. Commun. Control Comput.*, page 48857, 1989.