

Vertex Contact Representations of Paths on a Grid

Nieke Aerts¹ Stefan Felsner¹

¹Diskrete Mathematik,
Technische Universität Berlin, Germany

Abstract

We study Vertex Contact representations of Paths on a Grid (VCPG). In such a representation, the vertices of G are represented by a family of interiorly disjoint grid-paths on a square grid. Adjacencies are represented by contacts between an *endpoint* of one grid-path and an *interior point* of another grid-path. Defining $u \rightarrow v$ if the path of u ends on the path of v , we obtain an orientation on G from a VCPG. To control the bends of the grid paths the orientation is not enough. We therefore consider pairs (α, ψ) : a 2-orientation α and a flow ψ in the angle graph. The 2-orientation describes the contacts of the ends of a grid-path and the flow describes the behavior of a grid-path between its two ends. We give a necessary and sufficient condition for such a pair (α, ψ) to be realizable as a VCPG.

Using realizable pairs, we show that every planar (2,2)-tight graph admits a VCPG with at most 2 bends per path and that this bound is tight. In a similar way, we show that simple planar (2,1)-sparse graphs have a 4-bend representation and simple planar (2,0)-sparse graphs have 6-bend representation.

Submitted: March 2015	Reviewed: July 2015	Revised: October 2015	Reviewed: November 2015	Revised: December 2015
	Accepted: December 2015	Final: December 2015	Published: December 2015	
	Article type: Regular paper	Communicated by: C. D. Tóth		

We acknowledge support by German Science Foundation (DFG) grant Fe 340/7-2.

An extended abstract of this paper was presented at 40th International Workshop on Graph-Theoretic Concepts in Computer Science, in Le Domaine de Chalès, France [1].

E-mail addresses: aerts@math.tu-berlin.de (Nieke Aerts) felsner@math.tu-berlin.de (Stefan Felsner)

1 Introduction

Outline of results. In this paper, we consider the question whether a graph G admits a VCPG, i.e., a Vertex Contact representation of Paths on a Grid, where the grid is the square grid. In such a representation, the vertices are represented by a family of interiorly disjoint grid-paths with distinct endpoints. An *endpoint* of one grid-path coincides with an *interior point* of another grid-path if and only if the two represented vertices are adjacent. The shared point is not a bend-point of either of the grid-paths. We denote such a contact by *proper contact*. It follows that a graph admitting such a representation has to be planar.

A VCPG induces a unique orientation of the edges of G : Orienting the edge uv as $u \rightarrow v$ if the grid-path of u ends on grid-path of v we obtain an orientation of G . As each grid-path has two ends, in the induced orientation each vertex has outdegree at most 2. We denote such an orientation simply *2-orientation*. The existence of a 2-orientation implies that the density of graphs admitting a VCPG is rather restricted, in technical terms they have to be $(2, 0)$ -sparse.

Every 2-orientation of a planar graph induces a VCPG (Lemma 8). However, a 2-orientation of G defines the representation of the edges in a VCPG, but not how the grid-paths behave (e.g. how many bends a grid-path has). To control the behavior of the grid-paths between its endpoints, we introduce a flow network in the angle graph (Section 3).

To obtain a full combinatorial description of a VCPG, we consider a pair (α, ψ) : a 2-orientation α in the graph and a flow ψ in the angle graph. The main contribution of this paper is a necessary and sufficient condition for such a pair (α, ψ) to be realizable as a VCPG. We will then use such realizable pairs to give bounds on the number of bends needed to represent certain classes of graphs.

When the number of bends of each path is at most k , we call the representation B_k -VCPG and when every path has precisely k bends, we speak about *strict* B_k -VCPG. We will show that there are graphs admitting a B_1 -VCPG but no strict B_1 -VCPG and we construct a planar $(2,2)$ -tight graph that does not admit a B_1 -VCPG (Section 4.1). We show that planar $(2,2)$ -tight graphs admit a B_2 -VCPG (Section 4.2) and planar $(2,0)$ -tight graphs admit a B_6 -VCPG (Section 4.3).

Related work. Graphs that can be represented as intersection graphs of grid-paths without bends, i.e., by vertical and horizontal segments are known as Grid Intersection Graphs (GIG). This is a well studied class of bipartite graphs. Contact graphs of vertical and horizontal segments, i.e., B_0 -VCPG graphs or contact 2DIR graphs, are exactly the bipartite planar graphs [6, 11]. In fact, 2-orientations¹ of maximal bipartite planar graphs are in bijection with separating decompositions of this graph and separating decomposition induce a segment contact representation by segments in two directions (cf. [5, 6, 8, 11]).

¹With two non-adjacent vertices of out-degree 0 on the outer face.

Intersection graphs of grid-paths are called VPG graphs. VPG graphs were introduced by Asinowski et al. [3]. When the number of bends of each path is at most k , we denote the representation by B_k -VPG. From the result that every planar graph has a contact representation by T-shapes [7], it follows that every planar graph has an intersection representation by grid-paths where each path has at most three bends [3]. To obtain the B_3 -VPG, every T-shape is replaced by a grid-path that follows the T-shape in such a way that every point is covered on at least one side. Asinowski et al. conjectured this bound to be tight, i.e., there exists a planar graph for which three bends are necessary. Chaplick and Ueckerdt disproved this by showing that every planar graph admits a B_2 -VPG [4]. It is still open to decide whether this is tight. Chaplick and Ueckerdt conjectured that every planar graph admits a B_1 -VPG [4].

In a contact representation by grid-paths (as well as segments, pseudosegments, etc.), the vertices are represented by a family of internally disjoint objects. An *endpoint* of one object coincides with an *interior point* of another object if and only if the two represented vertices are adjacent. Kobourov, Ueckerdt and Verbeek show that all planar (2,3)-tight graphs admit an L-contact representation [12], i.e., a strict B_1 -VCPG (a sketch of their algorithm is given in Section 4.1). It follows that every (2,3)-sparse graph also admits a strict B_1 -VCPG. There are graphs that are not (2,3)-sparse but admit a strict B_1 -VCPG, e.g. K_4 . Kobourov et al. stated that every graph that admits a B_1 -VCPG is planar and (2,2)-sparse and they asked whether these conditions are also sufficient.

Chaplick and Ueckerdt show that triangle-free planar graphs, i.e., (2,4)-sparse planar graphs, admit a contact representation of $\{L, \Gamma, l, -\}$ -shapes [4]. In other words, triangle-free planar graphs admit a B_1 -VCPG with only two possible orientations of the one-bend paths. Alam et al. investigate which (2,0)-sparse graphs have a contact representation of circular arcs [2].

Another related area of research is *orthogonal graph drawing*. Here one wants to draw a given graph in the classical setting, i.e., the vertices are points in the plane and the edges are grid-paths. In orthogonal graph drawing, there have been many results on minimizing the number of bends. Note that in this setting vertices have degree at most 4, or as a workaround, the vertices can be represented as boxes. An early result of Tamassia gives an algorithm to obtain an orthogonal drawing with minimal bend number of a planar graph in polynomial time [14]. The algorithm preserves the embedding of the input graph. Optimizing the number of bends per path has received much attention too. Schäffter [13] gives an algorithm to draw 4-regular graphs in a grid with at most two bends per edge (which is tight when not restricted to planar graphs). For orthogonal drawings without degree restriction, Fößmeier, Kant and Kaufmann have shown that every plane graph has an orthogonal drawing preserving the embedding with at most one bend per edge [9].

Outline of the paper. In Section 2, we define sparse and tight graphs and show that every (2,0)-sparse graph has a 2-orientation. In Section 3, we intro-

duce a flow network. We then give the necessary and sufficient condition for a pair, a 2-orientation and a flow, to be realizable as a VCPG. In Section 4, we apply realizable pairs to give bounds on the number of bends in a VCPG.

2 Preliminaries: on $(2, \ell)$ -sparse graphs

Let $G = (V, E)$ be a planar graph and let R be a VCPG of G . When we speak about the embedding of G , we mean the embedding R . In R , the vertices of G are represented by internally disjoint grid-paths with distinct endpoints. We denote the grid-path of $v \in V$ by p_v . Each grid-path has two endpoints. When two grid-paths intersect, the intersection point is an endpoint for precisely one of the grid-paths. An endpoint of p_v coincides with an interior point of p_u precisely when u and v are adjacent. If an endpoint does not coincide with another grid-path, we call it a *free end*. The number of free ends is denoted by ℓ . A free end in an interior face can be extended to represent an edge, which is considered a dummy edge. As interior free ends can be seen as dummy edges, we assume that R only has free ends in the outer face. Later, in Lemma 2, we will see that free ends in the outer face reduce the number of bends needed to close the outer face. Therefore, the assumption that all free ends are in the outer face is also attractive.

If a graph $G = (V, E)$ is given and $W \subseteq V$, then we let $E[W]$ be the set of edges induced by W in G . We start with an easy proposition which gives an upper bound on the number of edges in a VCPG.

Proposition 1 *If $G = (V, E)$ admits a VCPG then:*

$$\forall W \subseteq V : |E[W]| \leq 2|W|. \quad (1)$$

Proof: Each edge is represented as a proper contact between two grid-paths representing two vertices. For one of the vertices this contact point is an endpoint of its grid-path. This vertex is the representative for the edge. Each vertex can be a representative for at most two edges. Therefore the number of edges induced by a set $W \subseteq V$ is at most twice the number of vertices. \square

If every grid-path has at most one bend, then there must be at least two free ends in the outer face². Therefore, the number of edges of a B_1 -VCPG is at most twice the number of vertices minus two.

Definition 1 *Let $G = (V, E)$ a graph and $k, \ell \geq 0$ integers. If*

$$\forall W \subseteq V : |E[W]| \leq k|W| - \ell,$$

then G is called (k, ℓ) -sparse. Where if $k < \ell$, the cardinality of W should be at least $\lceil \ell/k \rceil$. If G is (k, ℓ) -sparse and

$$|E| = k|V| - \ell$$

holds, then G is called (k, ℓ) -tight.

²This should be quite evident. A formal proof can be given by adding up the changes in direction, see the proof of Lemma 2 and in particular Equation 3.

Graphs that admit a VCPG must be planar and (2,0)-sparse (Proposition 1). In this paper we focus on (2,0)-tight graphs, (2,1)-tight graphs and (2,2)-tight graphs. When we speak about (2, ℓ)-tight or sparse graphs, we consider ℓ = 0, 1 or 2. Alam et al. [2] have shown that every plane (2, ℓ)-sparse graph *H*, is a subgraph of a plane (2, ℓ)-tight graph *G*. Hence, it follows that our focus on tight graphs is not a restriction.

2.1 A 2-Orientation Representing Edges

Defining $u \rightarrow v$ if p_u ends on p_v , we obtain an orientation on *G* from a VCPG. Every vertex has outdegree at most two in this orientation. The vertices with outdegree less than two are precisely those for which the grid-path has free ends. Such an orientation is denoted by 2-orientation.

A VCPG is not completely described by a plane graph with a 2-orientation. For example, the VCPGs in Figure 1 induce the same 2-orientation on the same plane graph, but the number of bends in the representations is different.

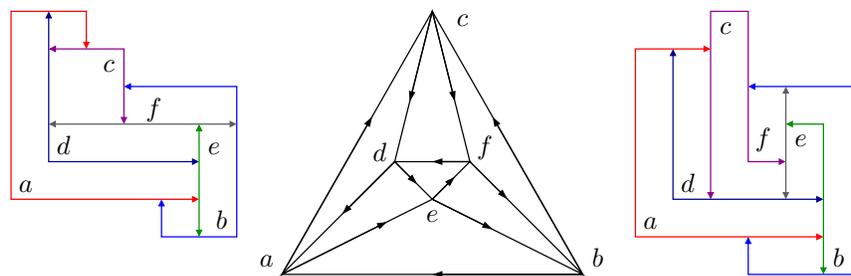


Figure 1: Two VCPGs of the octahedron, which induce the same 2-orientation (shown in the middle) but the number of bends differs.

It is clear that for a (2, ℓ)-tight graph with ℓ > 0, not every vertex has outdegree precisely 2. We also assume that the vertices that do not have outdegree 2 are on the outer face. With the next lemma, we show that for 0 ≤ ℓ ≤ 2 every (2, ℓ)-tight graph has such an orientation.

Lemma 1 *Every planar (2, ℓ)-tight graph has a 2-orientation. If 0 < ℓ ≤ 2, then for every embedding, there exists a 2-orientation such that all vertices with outdegree less than 2 are on the boundary of the outer face.*

Proof: Let $G = (V, E)$ be a planar (2, ℓ)-tight graph. Suppose there is a subset *W* of the vertices of *G* that has less than $2|W|$ incident edges. Then $G[V - W]$ must induce at least:

$$2|V| - \ell - (2|W| - 1) = 2|V - W| - \ell + 1$$

edges, which contradicts (2, ℓ)-sparseness. Hence, every subset *W* of the vertices of *G* has at least $2|W|$ incident edges.

Now we construct a bipartite graph B . The first color class, V_1 , consists of two copies of all but ℓ of the vertices of G . The remaining ℓ vertices are only added once. The second color class, V_2 , consists of the edges of G . The edge set of B is defined by the incidences in G : two vertices are connected if the corresponding vertex of G is an endpoint of the corresponding edge of G . We will show that $(2, \ell)$ -tightness of G implies that this bipartite graph has a perfect matching. A perfect matching defines a 2-orientation of G .

Let $U \subseteq V_1$ consist of n duplicate vertices, and m vertices without a copy, so $|U| = 2n + m$. Let $N_B(U)$ denote the set of neighbors of U in B . The neighbors are all edges in G , except for the edges between two vertices that are not in U . There are at most $2(|V| - n - m) + \ell$ such edges. Therefore,

$$|N_B(U)| \geq |E| - 2(|V| - n - m) + \ell \geq 2|V| - \ell - 2(|V| - n - m) + \ell = 2(n + m) \geq |U|.$$

Hence, Hall's marriage condition is satisfied for each subset V_1 . Since $|V_1| = |V_2|$, the graph B has a perfect matching and, hence, G has a 2-orientation.

To prove the second part of the lemma, we fix an embedding of G . The vertices that are not added twice to V_1 are chosen arbitrarily from the boundary of the outer face. The above argument implies the existence of a perfect matching and thus the desired 2-orientation. \square

Remark. Lemma 1 holds for all $(2, \ell)$ -tight graphs, not only simple and planar $(2, \ell)$ -tight graphs. In the proof, neither simplicity nor planarity is used. The only difference is that in a non-planar graph the vertices with outdegree less than 2 should be selected differently, because "being on the outer face" is meaningless.

3 A Combinatorial Characterization of VCPGs

3.1 A Feasible Flow Representing Bends

From here on, we consider graphs that are $(2, \ell)$ -tight, plane and 2-connected. Note that any $(2, \ell)$ -tight graph can easily be extended to a 2-connected $(2, \ell)$ -tight graph by adding an appropriate number of degree 2 vertices.

To describe the behavior of the bends of a grid-path, we introduce a flow network. The construction is inspired by Tamassia's flow network [14]. A *flow network* \mathcal{N} is a graph that has two distinct sets of vertices that are the *sources* and the *sinks* of the network. The sources and the sinks in our case, as well as in the network designed by Tamassia, are the faces of the graph (see Figure 2). For a face of a graph G , the need of a convex angle of a bend is presented as having a negative demand³, i.e., it is a source. Similarly, needing a concave angle of a bend is presented as having a positive demand, i.e., it is a sink.

The network will be the angle graph of G . Formally, the *angle graph* arises from G by taking the union of the vertices and faces of G as the vertices of

³We could also speak about excess, however, for simplicity we have chosen to only use the notion demand throughout this work.

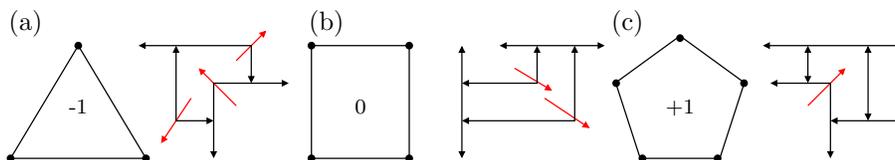


Figure 2: Faces with their demand and possible representations by grid-paths with appropriate numbers of convex and concave angles.

$A(G)$ and the pairs (v, f) , $v \in V(G)$, $f \in F(G)$, such that v is a vertex on f in G , as the edges of $A(G)$ (see Figure 4). The angle graph $A(G)$ is a plane bipartite graph with one color class consisting of vertex-vertices and the other of the face-vertices. Edges of $A(G)$ represent the angles⁴ of G . The angle graph of a 2-connected plane graph is a maximal bipartite graph, i.e., a quadrangulation. The outer face has special properties in a VCPG, therefore, we consider graphs with a fixed embedding.

The sources and sinks of the network are faces of G . A path in the network from a source to a sink has to traverse vertices of G , as the vertices will be represented as grid-paths with bends.

A directed path in the network going from a source to a sink passing through a vertex v represents a bend in the grid-path p_v representing vertex v . A path $f_1 \rightarrow v \rightarrow f_2$ represents a bend of p_v where the convex angle belongs to f_1 and the concave angle to f_2 . The difference between the network needed for VCPGs and the network defined by Tamassia for orthogonal drawings is that in the case of VCPGs the paths, resp. flow, goes through a vertex and in the case of orthogonal drawings the flow goes through an edge. In other words, the underlying graph of the network defined by Tamassia is the dual graph and not the angle graph (see Figure 3). We proceed with the formal introduction of the encoding of the bends of a VCPG.

A flow ψ is a weighted directed graph, with the network \mathcal{N} as underlying graph. The *weight* of an edge denotes the amount of flow that is routed over this edge. Flow conservation holds relative to correction terms given by the positive or negative demands. An example of a flow ψ in the angle graph of K_4 is shown on the right side of Figure 4. A face-vertex of $A(G)$ can be a source or a sink, depending on the size of the face in G (see Proposition 2 below). The vertex-vertices of $A(G)$ are neither sources nor sinks. The capacities of edges are unbounded. From a VCPG R we construct a flow ψ as follows. For each bend of p_v , such that the convex angle of this bend lies in f_1 and the concave angle lies in f_2 , add a unit of flow $f_1 \rightarrow v \rightarrow f_2$. Note that the matching between concave and convex corners at faces sharing a bend of an edge forces us to adopt a strange classification of convex and concave angles at the outer face, e.g., the outer face of a rectangle has four concave angles.

⁴The angles of a graph are the angles that are formed by any two consecutive edges that meet at a common vertex.

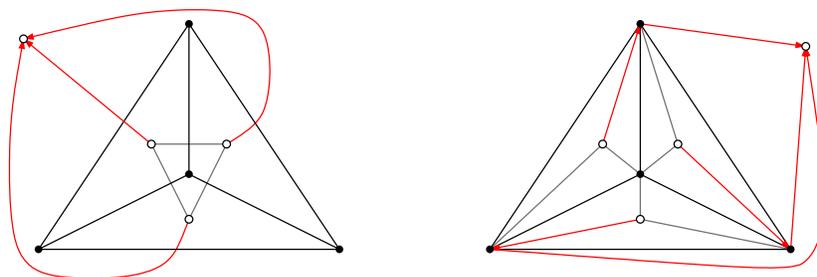


Figure 3: An example of a feasible flow for K_4 (inner faces have demand -1 , the outer face has demand 3) with the dual graph (left) or the angle graph (right) as underlying graph.

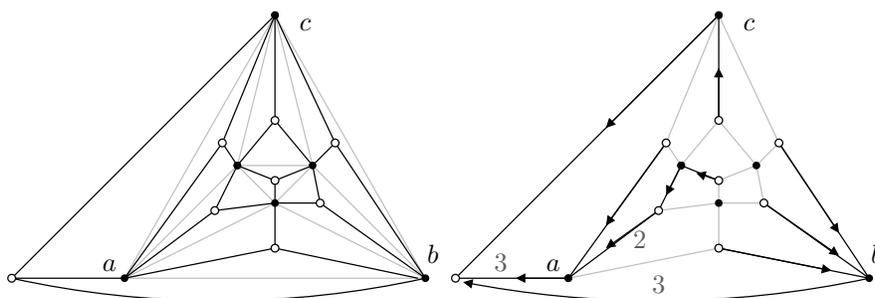


Figure 4: The angle graph (black) of the octahedron (grey) and the angle graph with the feasible flow induced by the VCPG on the left of Figure 1. In the flow, the edges that get weight zero are omitted, the edges that have weight larger than 1 are labeled with their weight.

Proposition 2 *Let ψ be the flow obtained from a VCPG of a $(2, \ell)$ -tight graph. Then the following holds. Every interior k -face f has demand $|f| - 4$ (interior 3-faces have negative demand). The outer face f_∞ has demand $(4 - 2\ell) + |f_\infty|$.*

Proof: Following the boundary of an interior region of a VCPG and adding the changes in direction, one should obtain 2π . Each edge is represented as a proper contact and therefore changes the direction with $\pi/2$. A convex angle at a vertex changes the direction with $\pi/2$ as well and a concave angle at a vertex changes the direction with $-\pi/2$. For a face f , let $c(f)$ be the number of concave angles at vertices minus the number of convex angles at vertices. For an interior face f we obtain:

$$2\pi = \pi/2 \cdot |f| - \pi/2 \cdot c(f). \tag{2}$$

The value $c(f)$ also counts the incoming minus the outgoing flow in ψ and

therefore is the demand of f . By rearranging (2), we obtain

$$c(f) = |f| - 4,$$

i.e., the desired result.

For the outer face f_∞ , the bends have to be counted differently. For the outer face, an edge implies a change in direction of $-\pi/2$. A convex angle contributes a $-\pi/2$ change of direction and a concave angle a change of $\pi/2$. A free end gives a change in direction of π . Together we obtain

$$2\pi = -\pi/2 \cdot |f_\infty| + \pi/2 \cdot c(f) + \pi\ell, \tag{3}$$

and after rearranging:

$$c(f_\infty) = (4 - 2\ell) + |f_\infty|$$

where ℓ is the number of free ends. □

A *valid flow* for the network \mathcal{N} satisfies the flow conservation law at each vertex that is not a source or a sink, i.e., the net flow is zero at each vertex $v \notin S \cup T$. At a sink, the sum of the incoming flow minus the sum of the outgoing flow is at most the demand of the sink. At a source, the sum of the incoming flow minus the sum of the outgoing flow is at least the size of the demand⁵. A valid flow ψ is *feasible* if the demand of every sink is satisfied. We next show that the demands of the sinks add up to the absolute value of the sum of the demands of the sources, hence, in a feasible flow the negative demand of each source is also satisfied.

The total value of a feasible flow is equal to the number of interior 3-faces (the number of sources), which in turn is equal to the sum of the demands of all sinks. We take the sum of all demands.

$$\begin{aligned} \sum_f c(f) &= \sum_{f \in F_{\text{int}}} (|f| - 4) + (4 - 2\ell) + |f_\infty| = \\ &= \sum_{f \in F} |f| - 4|F| + 8 - 2\ell = 2|E| - 4|F| + 8 - 2\ell = \\ &= 2|E| - 4(2 - |V| + |E|) + 8 - 2\ell = -2(|E| - (2|V| + \ell)) = 0. \end{aligned}$$

We have used Euler’s formula to replace $|F|$ and the fact that the graph is $(2, \ell)$ tight for the last equation.

The number of bends prescribed by the flow is

$$\sum_{v \in V} \psi(v),$$

where $\psi(v)$ denotes the amount of flow that goes through the vertex v . In order to relate to a VCPG it is necessary that ψ is an *integral* feasible flow for

⁵Recall that the demand of a source is negative.

the network \mathcal{N} . Note that since the demands are integral the existence of a (minimum cost) integral feasible flow is guaranteed. In the sequel, we will omit the word ‘integral’ as we only consider integral feasible flows.

A flow does not uniquely define a VCPG either. The two VCPGs in Figure 5 induce the same feasible flow, but the edges are represented in a different way. The obvious question is whether every feasible integral flow in $A(G)$ belongs to a VCPG. Unfortunately this is not the case (see Figure 6). However, using both the orientation and the feasible flow, we obtain a characterization of VCPGs. This is the subject of the following section.

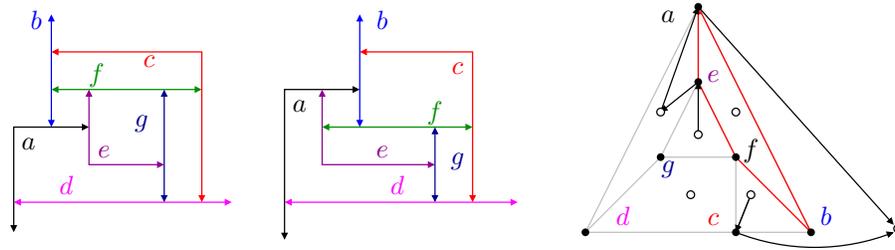


Figure 5: A feasible flow does not uniquely define a VCPG. On the left two VCPGs, and on the right the feasible flow induced by both of the VCPGs. The difference is the orientation of the cycle e, a, b, f (in red).

3.2 Realizable Pairs

Starting with a VCPG, a 2-orientation α and a feasible flow ψ can be obtained, as described in the previous sections. In this section we identify a necessary property of a pair, (α, ψ) that comes from a VCPG. Not every pair (α, ψ) on G induces a VCPG of G . We call a pair (α, ψ) *realizable* when it does. We will prove that the necessary property is also sufficient, hence, realizable pairs are in bijection to VCPGs. Our proof method is algorithmic, it shows how one can construct a VCPG (the geometric setting) from a realizable pair (the combinatorial setting).

3.2.1 A Property of Realizable Pairs

We will deduce a property of realizable pairs and show that this property is necessary and sufficient for a pair, a 2-orientation α and a feasible flow ψ , to be realizable. The property depends on α and on ψ . First we define the property for a vertex v (see Figure 7 (c)). Let $A[N_{A(G)}[v]]$ denote the angle graph induced by the closed neighborhood of a vertex v , i.e. induced by v and all its neighbors in $A(G)$. Let n_1, n_2 be the neighbors of v along the outgoing edges of v in α . If v has outdegree 0, then n_1, n_2 are its neighbors on the outer face. If v has outdegree 1, then n_1 is its neighbor along the outgoing edge. The vertex n_2 is the neighbor of v on the outer face, chosen such that the units of

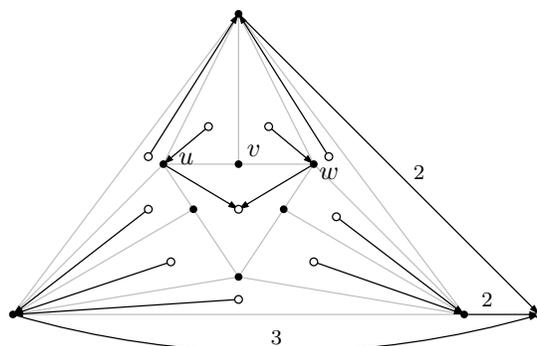


Figure 6: A feasible flow ψ that does not belong to a VCPG, i.e. there exists no 2-orientation α such that the pair (α, ψ) is realizable.

flow are equally distributed on the clockwise and counterclockwise side of the path n_1, v, n_2 . Informally, a unit of flow through a vertex v represents a bend of the grid-path of v . Following the grid-path from n_1 to n_2 , looking left and right, the bends are met at the same time. This implies that the flow through a vertex must be laminar, i.e., non-crossing.

Definition 2 (Realizability Condition) *The pair (α, ψ) satisfies the realizability condition at vertex v if and only if, given $A[N_{A(G)}[v]]$ and the flow in this subgraph (see Figure 7 (b)):*

- *there is a decomposition of the flow into non-crossing paths, and,*
- *every path of such a decomposition crosses the path n_1, v, n_2 .*

When the pair (α, ψ) satisfies the realizability condition at each vertex we say that the pair is realizable.

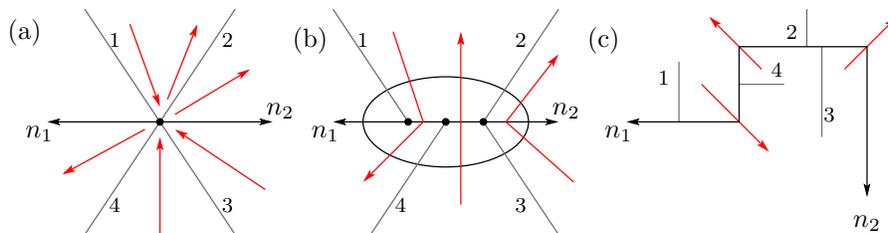


Figure 7: (a) The vertex v in G and the local flow. (b) The expansion of v : The flow through v is decomposed into disjoint paths and each of these paths cuts the path consisting of the outgoing edges of v . (c) A close look at the grid-path of a vertex v in a VCPG.

Let us look at the feasible flow ψ from Figure 6 in the context of the realizability condition. The graph is $(2,0)$ -tight and therefore all vertices have outdegree 2 in a 2-orientation that comes from a VCPG. A 2-orientation α such that (α, ψ) is a realizable pair, is such that $u \rightarrow v$ and $w \rightarrow v$. Then v has outdegree at most 1. This shows that there is no 2-orientation α such that (α, ψ) is realizable, and hence, due to Lemma 2, ψ does not come from a VCPG.

We now show that a pair that comes from a VCPG satisfies the realizability condition.

Lemma 2 *A pair (α, ψ) that comes from a VCPG is realizable.*

Proof: First note that a VCPG of G describes an embedding of G . If there is a grid-path with one free end, then before proceeding, we reduce all unnecessary bends, i.e., if a grid-path has bends between its last neighbor and its free end, these bends are removed. The 2-orientation α that is induced by the VCPG has an edge $u \rightarrow v$ if and only if the grid-path of u ends on the grid-path of v . Consider the grid-path that represents a vertex v . If this path has no bends, the realizability condition is satisfied at this vertex. Suppose the path has k bends. Draw an arrow from the face containing a convex corner to the face in which the associated concave corner lies. Now the set of arrows represents the flow $\psi(v)$. This flow is non-crossing through v and every unit of flow is cut by the the grid-path of v . When these arrows are introduced for all bends of all grid-paths, the flow given by these arrows satisfies the demand of each face. Contract the strictly interior steps of the grid-path to a vertex. Every unit of the non-crossing flow through v is now cut by the outermost two segments of the grid-path, which correspond to the outgoing edges of v , or to the outgoing edge and the location of the last incoming edge before the free end of the grid-path. Hence, the realizability condition is satisfied at each vertex. Therefore, the pair (α, ψ) obtained from the VCPG is realizable. \square

3.3 Realizable Pairs Are in Bijection with VCPGs

Now we are ready for the main result.

Theorem 1 *The realizable pairs are in bijection with VCPGs.*

The remainder of this section is dedicated to showing that given a realizable pair (α, ψ) , there exists a VCPG, with the same vertices on the outer face as in the chosen embedding of G (this embedding is implied by the pair (α, ψ) since ψ lives on the angle graph), and such that:

- (a) The grid-path of u ends on the grid-path of v if and only if the edge uv is oriented from u to v in α ; and
- (b) The grid-path of v has precisely $\psi(v)$ bends.

We will show how to construct a VCPG given a realizable pair. Note that an embedding can be derived from $A(G)$ (in which the flow ψ is defined). Consider

a realizable pair (α, ψ) . The construction consists of four steps, which we first outline here.

Step 1: First, we expand the vertices that have k units of flow going through them, to a path⁶ of length k . We obtain a bipartite graph.

Step 2: We introduce help-edges and vertices in the bipartite graph to construct a quadrangulation (see Figure 8(b)). The orientation α is extended to a 2-orientation of the quadrangulation.

Step 3: We then find a segment contact representation of the quadrangulation. It has been shown that the 2-orientations of maximal bipartite planar graphs are in bijection with separating decompositions of this graph (e.g. [5]). In turn, a separating decomposition induces a segment contact representation (cf. [11, 6, 8]). Hence we can construct a segment contact representation where the representation of the edges is in bijection with the given 2-orientation. An example is shown in Figure 8(c).

Step 4: Last, we will show that the extra edges that have been introduced to make a quadrangulation of the bipartite graph can be deleted in order to obtain a VCPG of G (see Figure 8(d)).

Let us describe the steps in more detail.

Step 1. Let (α, ψ) be a realizable pair for G . We expand all vertices with non-zero flow. The plane graph we obtain is denoted by \tilde{G} . For every vertex v for which $\psi(v) \neq 0$, expanding v consists of the following steps (see Figure 7):

1. Expand v to a circle. We will call this the *bag* of v .
2. Inside the circle, add a path with $\psi(v) + 1$ vertices between the two outgoing edges of v . If v has outdegree 1 or 0, then v is on the outer face. The path is added between the edge on the outer face and the outgoing edge, or between the two edges on the outer face, respectively. The edges of the path are called *path-edges*, the inner vertices of the path by *path-vertices*, and all the vertices that belong to a bag are called *bag-vertices*.
3. Each incoming edge of v , is now connected to a path-vertex. This is done such that the flow between two faces only crosses a single edge and this is an edge of the path.

After all the expansions have been done we obtain a graph where all faces have even length. Each face gets $||f| - 4| + 2k$ extra vertices due to the expansion step, where k is the amount of flow crossing through the face. The resulting faces in \tilde{G} have size $|f| + ||f| - 4| + 2k$, for $|f| = 3$ this is $4 + 2k$ and for $|f| > 3$ this is $2|f| - 4 + 2k$, both are even. So, all faces in \tilde{G} have even length and therefore \tilde{G} is a bipartite graph.

Step 2. Now we add help-edges to extend \tilde{G} to a quadrangulation. We denote the quadrangulation by G_Q . We will also orient the new edges to obtain a 2-orientation of G_Q . In order to explain how the help-edges are added, we need the following lemma.

⁶The length of a path counts the number of edges.

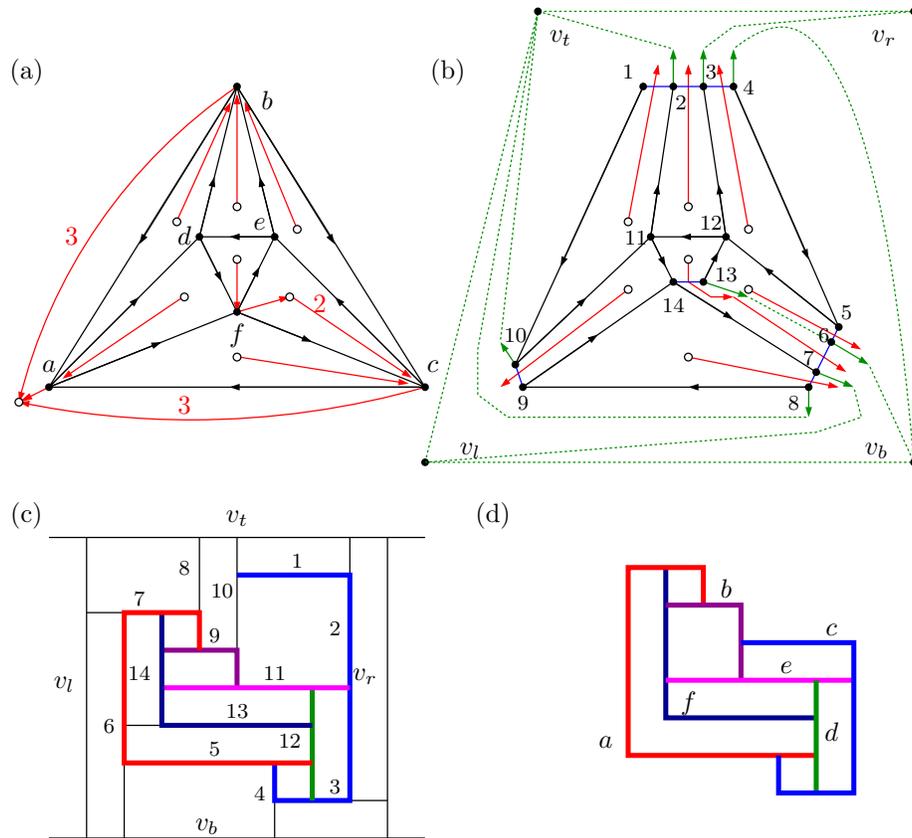


Figure 8: From a realizable pair to a VCPG: (a) a plane (2,0)-tight graph with a realizable pair (ψ in red); (b) expanding the vertices according to the flow (in blue) and extending the bipartite graph to a quadrangulation (in green); (c) a segment contact representation of the quadrangulation with the segments that belong to the original graph highlighted; (d) a VCPG.

Lemma 3 *Every interior face \tilde{f} of \tilde{G} has $(|\tilde{f}| - 4)/2$ units of incoming flow.*

Proof: Let $\psi^+(f)$ (respectively $\psi^-(f)$) denote the incoming (respectively outgoing) flow in face f . Let f be the face corresponding to \tilde{f} in the original graph G .

As ψ is a feasible flow, the demand $c(f)$ of f is the difference between incoming and outgoing flow:

$$\psi^+(f) - \psi^-(f) = c(f) = |f| - 4 .$$

Now we use that the size of the extended face \tilde{f} is the size of f plus the incoming and the outgoing flow.

$$|\tilde{f}| = |f| + \psi^+(f) + \psi^-(f) = |f| + \psi^+(f) + \psi^+(f) - |f| + 4 = 2\psi^+(f) + 4$$

Hence we find

$$\psi^+(f) = \frac{|\tilde{f}| - 4}{2} . \tag{4}$$

□

Using $(|\tilde{f}| - 4)/2$ help-edges, we can quadrangulate \tilde{f} (see Figure 9 and the proof of Lemma 4). The help-edges should be added in such a way that every bag (vertex expansion) gets as many help-edges as the value of the flow going into \tilde{f} through this vertex. Informally, a concave corner arises from two segments that both end in one point. In the segment contact representations that we will use, there are only proper contacts or free ends. Each help-edge represents a segment of a concave corner that proceeds into the face, and, this part will later be removed. Such a help-edge will be oriented as outgoing from the bag.

Later, we will construct a segment contact representation of the quadrangulation. For this, it is necessary that every interior vertex has outdegree precisely 2. Therefore, the help-edges must be added in such a way that this is possible for all vertices. Each interior bag-vertex should finally get assigned two outgoing arcs (which are not edges in the original graph). The vertices on the boundary of the bag, i.e., the vertices incident to the outgoing edges of the vertex in the original graph, need one additional outgoing arc. To begin with, the help-edges are added along the flow from a vertex into a face. This gives the required number of new edges leaving a bag (a bag with k vertices and 2 outgoing edges in α has $k - 1$ internal vertices and flow value $k - 1$, hence, adding one out-edge for each unit of flow is the right amount).

Lemma 4 *Each inner face \tilde{f} of \tilde{G} can be quadrangulated in such a way that each bag through which k units of flow enter \tilde{f} gets k new outgoing arcs. The outer face of \tilde{G} can be quadrangulated using four help-vertices (v_t, v_r, v_b, v_l) , in such a way that each bag through which k units of flow enter the outer face gets k new outgoing arcs.*

An example of quadrangulating an interior face is depicted in Figure 9. The flow ψ is given by the red arrows. First, half-arcs are added, the green solid arcs. Then these half-arcs are subsequently connected in such a way that they close one 4-face (green dashed lines). The quadrangulation of the outer face is based on the same idea.

Proof of Lemma 4: Add an outgoing half-arc into \tilde{f} from each vertex that comes clockwise after a unit of incoming flow, see Figure 9. This ensures that every bag gets the correct number of new outgoing arcs. Now we will show that extending these half-arcs allows us to quadrangulate \tilde{f} . There are $(|\tilde{f}| - 4)/2$ vertices that get a half-arc, thus $(|\tilde{f}| + 4)/2$ without a half-arc. Hence, there exists a vertex with a half-arc which is followed by two consecutive vertices u, v without a half-arc. We will extend this half-arc and show that afterwards there is again a half-arc which is followed by two consecutive vertices without a half-arc. Consider the half-arc clockwise before u, v and connect it to the vertex after u, v . We have constructed a 4-face and completed one half-arc. The other face has $|\tilde{f}| - 2$ vertices and $(|\tilde{f}| - 2 + 4)/2$ half-arcs. Therefore, there must again be a vertex with a half-arc that is followed by two vertices without a half-arc and the step can be repeated on this face. In every step the size of the face is reduced by two and one half-arc is extended, there are $(|\tilde{f}| - 4)/2$ steps. Since

$$|\tilde{f}| - 2 \cdot (|\tilde{f}| - 4)/2 = 4$$

it follows that the result is a quadrangulation of \tilde{f} . The way the half-arcs are added ensures that that each bag through which k units of flow enter \tilde{f} gets k new outgoing arcs.



Figure 9: Adding help-edges in a face. The flow ψ is depicted by red arcs. The green half-arcs together with the dashed extensions represent the help-edges.

To take care of the outer face we add a quadrilateral around the graph and quadrangulate the new inner face between the graph and the quadrangle.

We distinguish four cases (see Figure 10), a detailed description of the process in each of the cases is given below.

- (a) There exists a vertex s in the original graph G which has no outgoing arcs under α .
- (b) There exist two vertices s, t in the original graph G which both have precisely one outgoing arc under α .
- (c) There exists precisely one vertex s which has precisely one outgoing arc under α .

(d) All vertices have outdegree 2 under α .

We add a quadrilateral around the graph and construct an inner face, of size $2k + 4$ such that the amount of incoming flow is k (see Figure 10). Then we can use the same method as for the interior faces.

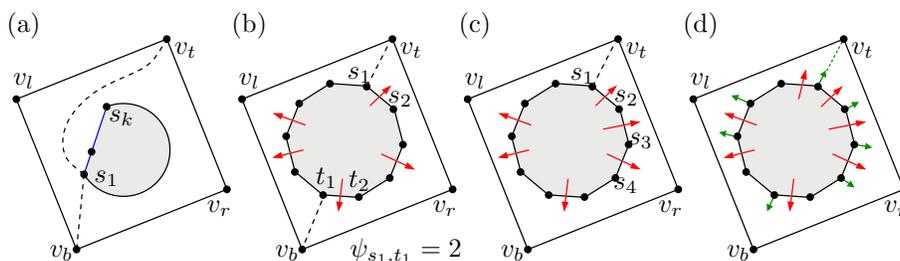


Figure 10: Adding a quadrilateral around the graph.

Recall that we consider $(2, \ell)$ -tight graphs. Hence the graph has $2|V| - \ell$ edges and there are precisely ℓ free ends in the outer face of the representation.

(a) Note that $\ell = 2$. Add a quadrangle around the graph with vertices v_l, v_t, v_r, v_b in clockwise order. If s is expanded, we label the vertices in the expansion in counterclockwise order, following the boundary of \tilde{f}_∞ , by s_1, \dots, s_k . If s is not expanded then we label $s = s_1$. Add the arcs (s_1, v_t) and (s_1, v_b) . Now the bounded face f^* containing s_1, v_t, v_r, v_b on its boundary has the following properties: $|f^*| = |\tilde{f}_\infty| + 4$, it has $\frac{1}{2}|\tilde{f}_\infty|$ incoming flow. The same method as for an inner face can be used: first, half-arcs are added and then these are extended to arcs by consecutively closing 4-faces.

(b) Note that $\ell = 2$. Add a quadrangle around the graph with vertices v_l, v_t, v_r, v_b in clockwise order. If s and t are expanded, we label the expansion vertices in the respective bags such that s_1 and t_1 have no outgoing arc under α and they are end vertices of the expansion path. If s respectively t is not expanded, we label $s = s_1$ respectively $t = t_1$. Add the arc (s_1, v_t) and let f^* be the inner face that is bounded by the quadrangle. Let ψ_{t_1, s_1} denote the incoming flow to f^* between t_1 and s_1 clockwise around \tilde{f}_∞ . Assign the label q to the vertex at distance $2\psi_{t_1, s_1} + 3$ from t_1 walking counterclockwise around f^* . Add the arc (t_1, q) . Now we have obtained two faces f_u, f_d , for which the incoming flow $\psi^+(f_u), \psi^+(f_d)$ is equal to $|f_u|/2 - 2, |f_d|/2 - 2$. The same method as for an inner face can be used: first, half-arcs are added and then these are extended to arcs by consecutively closing 4-faces.

(c) Note that $\ell = 1$. Add a quadrangle around the graph with vertices v_l, v_t, v_r, v_b in clockwise order. The vertex s has outdegree precisely 1 under α . Label the vertex in the bag of s that has no outgoing edge and is an end vertex of the extension path s_1 or if s is not expanded we label $s = s_1$. Add the arc (s_1, v_t) . Let f^* be the inner face that is bounded by the quadrangle. The incoming flow of f^* is of size $\frac{1}{2}|\tilde{f}_\infty| + 1$ and the size is given by $|f^*| = |\tilde{f}_\infty| + 6$.

The same method as for an inner face can be used: first, half-arcs are added and then these are extended to arcs by consecutively closing 4-faces.

(d) Note that $\ell = 0$. Add a quadrangle around the graph with vertices v_l, v_t, v_r, v_b in clockwise order and let f^* be the inner face that is bounded by the quadrangle. We will use one unit of flow to connect \tilde{f}_∞ to the quadrangle. First add a half-arc into f^* from each vertex clockwise after a unit of incoming flow. Choose any half-arc and connect it to v_t . We obtain a face f with $\frac{1}{2}|\tilde{f}_\infty| + 1$ incoming flow and $|f| = \tilde{f}_\infty + 6$. The same method as for an inner face can be used: first, half-arcs are added and then these are extended to arcs by consecutively closing 4-faces.

In the resulting graph each inner face \tilde{f} of \tilde{G} is quadrangulated and the outer face of \tilde{G} is quadrangulated using four help-vertices (v_t, v_r, v_b, v_l) . Due to the introduction of the half-arcs according to the incoming flow in a face, we have ensured that each bag through which k units of flow enter the face gets k new outgoing arcs. \square

To obtain a 2-orientation of the quadrangulation G_Q , the edges that are strictly inside the bags, and the four boundary edges need to be oriented. The orientation of the original edges is inherited from α , the help-edges are already oriented. Each bag b_v contains $|b_v| - 1 = \psi(v)$ edges which are not yet oriented.

Lemma 5 *The outdegree of each bag b_v in G_Q is precisely $|b_v| + 1$ and the outdegree of each vertex is at most 2.*

Proof: The bag b_v of a vertex v has $\psi(v) + 1$ vertices. According to the flow, $\psi(v)$ outgoing arcs are added at the introduction of the help-edges.

If a bag b_v comes from a vertex v which has outdegree 2, the bag has outdegree $2 + \psi(v) = |b_v| + 1$. If a bag b_v comes from a vertex v which has outdegree 1 in α , then the quadrangulation step has assigned another outgoing arc to this bag. Therefore the bag has outdegree $|b_v| + 1$. If a bag b_v comes from a vertex v which has outdegree 0 in α , then the quadrangulation step has assigned two outgoing arcs to this bag. Therefore the bag has outdegree $|b_v| + 1$.

Suppose one of the bag-vertices has outdegree 3 or more. At most one of the arcs is an edge of the original graph, as otherwise the vertex would not have been expanded and no new outgoing arc is added. When an outgoing arc is added to a vertex, the vertex is located clockwise after a path-edge, i.e., an edge that was added during the expansion of the vertex, and this path-edge is crossed by a unit of flow routed into the adjacent face for which the vertex is clockwise after the edge. If a vertex has two added arcs, it must be incident to two faces, one on each side of the path and such that the vertex is clockwise next of the path-edge with respect to this face. Hence, this vertex is incident to two path-edges and is not an end vertex of the path. Therefore, it cannot have an outgoing arc that is an edge of the original graph. It follows that every vertex has outdegree at most 2. \square

Orient $v_l v_t, v_r v_t, v_l v_b$ and $v_r v_b$ towards v_t and v_b , respectively. The vertices v_t and v_b are the two *poles* of the 2-orientation. The orientation can be completed in a greedy way.

Lemma 6 *The path-edges can be oriented such that the resulting orientation is a 2-orientation of G_Q .*

Proof: Let u_1, u_2, \dots, u_s be the vertices of a bag in the ordering given by the path. Think of the path as being horizontal. We know that u_1 and u_s have outgoing edges that come from the 2-orientation α or an outgoing edge that connects to the surrounding quadrilateral. In addition, a unit of flow that goes through the edge u_i, u_{i+1} into the “upper face” implies an outgoing edge at u_i . We orient such an edge as $u_{i+1} \rightarrow u_i$ so that the flow through the edge contributes a single outgoing edge at each of the endpoints. A unit of flow that goes through the edge u_j, u_{j+1} into the “lower face” implies an outgoing edge at u_{j+1} . We orient such an edge as $u_j \rightarrow u_{j+1}$ so that again the flow through the edge contributes a single outgoing edge at each of the endpoints. Taking into account the special outgoing edges at u_1 and u_s we have a total of $2s$ outgoing edges from the bag. Also, every vertex in the bag has outdegree at most 2. Therefore, every vertex of the path now has outdegree 2. \square

Step 3. From G and the realizable pair (α, ψ) we constructed a quadrangulation G_Q with a 2-orientation $\hat{\alpha}$ (v_t and v_b are the only two vertices with outdegree 0 instead of outdegree 2). We construct a segment contact representation, the vertices of the two color classes of G_Q become horizontal and vertical segments and the edges are proper contacts between the segments satisfying $\hat{\alpha}$.

Step 4. It remains to show that this segment representation of G_Q can be transformed to a VCPG of G that realizes (α, ψ) . The segment contact representation of G_Q contains a paths p_v for each vertex that reflects the behaviour prescribed by the pair (α, ψ) . If the bag b_v of v contains the path u_1, u_2, \dots, u_s , then path p_v starts and ends at the contact points of the segments representing u_1 and u_s with the segments representing their special outneighbors. The bends of p_v are at the contact points of segments of pairs u_i, u_{i+1} of adjacent vertices in the bag.

The collection of these paths p_v for $v \in V$ does not immediately induce the VCPG. In the example in Figure 11 (a), the path of 1, belonging to vertex b does not end on the part of 10 that will belong to vertex a . To be precise, the path of 1 ends on a part of 10 that belongs to a help-edge. In general it might be that a grid-path that is supposed to end on another grid-path ends on a part that belongs to a help-edge. The following lemma shows that we can change the order of segments ending on different sides of a segment. It follows that the segment contact representation can be changed such that all contacts are as we want them to be.

Lemma 7 *Given a vertical segment s in a segment contact representation, suppose a horizontal segment s_ℓ ends on the left of s and s_r ends on the right of s such that the endpoint of s_ℓ on s is below the endpoint of s_r on s . Then there is an equivalent segment contact representation, i.e., with the same corresponding 2-orientation, where the endpoint of s_ℓ on s is above the endpoint of s_r on s .*

Proof: Consider a cutline consisting of an interval on s and two horizontal rays. The first ray starts just below s_ℓ and points to the left. The second ray starts just above s_r and points to the right. The base-points of both rays on s are chosen such that the rays only intersect vertical segments. The interval on s is the interval between the two base-points. Now cut the graph along the cutline and shift the half containing s_ℓ up until s_ℓ is above s_r . Vertical segments that are intersected by the rays now have disconnected parts, connect these parts vertically through the slab opened by the shift. This results in a new, equivalent, segment contact representation where s_ℓ is higher than s_r . Figure 11 shows an example where $s = 10$, $s_\ell = 9$ and $s_r = 1$. \square

Clearly, there is a symmetric construction that allows to change the order of two segments touching a horizontal segment one from above and the other from below. We call this operation a *shift operation*.

Let s be a vertical segment and let $s_{\ell_1}, \dots, s_{\ell_a}$ and s_{r_1}, \dots, s_{r_b} be the segments touching s from the left resp. from the right, where each of the two sets have increasing order of y -coordinates. Let a *merge* of the segments be a permutation of the $a + b$ segments where the subpermutation of the segments touching from the left is $s_{\ell_1}, \dots, s_{\ell_a}$ and the order of the segments touching from the right is s_{r_1}, \dots, s_{r_b} . We can realize any merge of the $a + b$ segments touching s as the order of their y coordinates. A procedure that achieves this is working upwards on s . Lemma 7 allows to shift the needed next segment from the left or the right chain of segments below all the still untouched segments. Again there is a symmetric statement for horizontal segments.

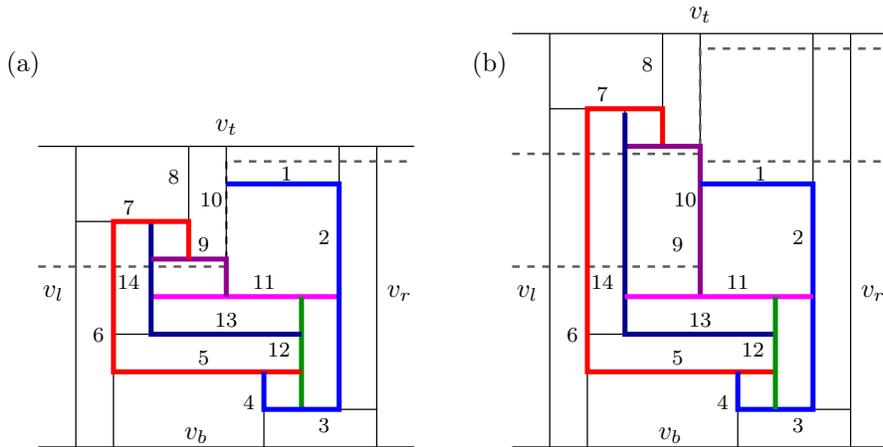


Figure 11: Cutting open, shifting and extending shows that neighbors on each side can be moved independently: (a) a segment contact representation where the highlighted part of 1 does not end on the highlighted part of 10, and a cutting line (dashed); (b) the segment contact representation is cut and the top is pulled upwards extending the vertical segments that are cut.

Theorem 2 *The segment representation of G_Q obtained from a realizable pair (α, ψ) induces a VCPG of G .*

Proof: We use three operations *identification*, *shift operations* and *deletion* to transform the segment representation of G_Q into a VCPG of G .

Identification. For each vertex v we will identify a set of parts of segments that form a grid-path p_v . Basically the identified set represents the path edges and the original outgoing edges of v . This grid-path will represent the vertex in the VCPG. We call the parts of the segments that are selected the *highlighted parts* (see Figure 11). For a vertex v that is not expanded, we take the whole segment as p_v . For each bag b_v , select the segments representing the bag-vertices u_1, \dots, u_s . For each bag-vertex u_i , use the part of the segment between u_{i-1} and u_{i+1} for p_v . For the end vertices of the bag, u_1 (respectively u_s) take the part of the segment between u_2 (respectively u_{s-1}) and the outgoing neighbor of u_1 (respectively u_s) that is not in the bag nor reached via a help-edge. Together these pieces form the grid-path p_v associated with v . When $\ell = 1, 2$ we have added arcs while quadrangulating the outer face that do not correspond to the flow nor to original edges. These edges represent the free ends of grid-paths. We cut such segments right after the last neighbor ending on this segment to get the free end of p_v .

Shift operations. It may occur that for an arc of G , say $v \rightarrow u$, the endpoint from v on u appears on a non-highlighted part of a segment that contributes to the path p_u of color u . This may occur because of the added help-edges. This is to be repaired with a shift operation as described in Lemma 7. We have to show that such a shift operation can indeed be used for the repair.

Let u_i be the vertex of the bag b_u corresponding to the segment s_i on which p_v ends. Let u_{i-1} and u_{i+1} be the neighbors of u_i in the bag b_u . For $i = 1$, we also let u_0 be the special outneighbor of u_1 , i.e., the outneighbor that is not introduced because of the flow unit associated with u_1, u_2 . Similarly, u_{s+1} is the special outneighbor of u_s .

In the following, it is crucial that when walking clockwise around s_i , we see the contacts with other segments in the same order as we see around u_i in G_Q the edges to neighbors.

If $u_{i-1} \leftarrow u_i \rightarrow u_{i+1}$, then the complete segment s_i belongs to p_u . Hence, the end-point of p_v on s_i is already on p_u .

Our second case is $u_{i-1} \leftarrow u_i \leftarrow u_{i+1}$. For ease of reference, we assume that s_i is vertical with the segment of u_{i+1} to its right. Figure 12(a) shows a generic situation. The figure shows G_Q and the flow associated to u_i, u_{i+1} in the upper sketch, and the representation with segment contacts in the lower sketch. With a and b we have included two possible neighbors of u that are attached to u_i . The path p_b already meets p_u , a contact of p_a with p_u can be created with the shift operation. It is crucial that on the right side of s_i above u_{i+1} there can be no contact with p_v . This is because the flow associated to u_i, u_{i+1} and the help-edge leaving u_i that is associated to this flow point into the same face of G . The case $u_{i-1} \rightarrow u_i \rightarrow u_{i+1}$ is the same as the previous, just the numbering of the vertices in the bag is reversed.

We are left with the case $u_{i-1} \rightarrow u_i \leftarrow u_{i+1}$. Again we assume that s_i is vertical with the segment of u_{i+1} to its right. The segment of u_{i-1} is to the left of s_i because around u_i in G_Q vertices u_{i-1} and u_{i+1} are separated by the outgoing edges. As above on the right side of s_i above u_{i+1} and on the left side of s_i below u_{i-1} , there can be no contact with a path p_v . Figure 12(b) shows a generic situation. With a and b we have included two possible neighbors of u that are attached to u_i . Taking the segments of a and b as the reference segments for a shift operation we transform the representation such that p_a and p_b both end on p_u .

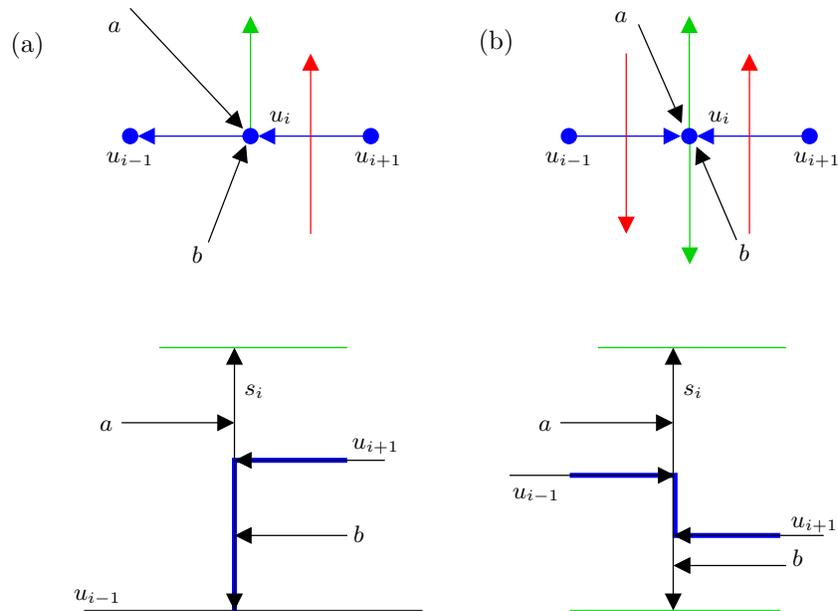


Figure 12: Using shift operations, the missing contacts from p_a and p_b to p_u can be established.

Deletion. After the shift operations, all edges $v \rightarrow u$ of G are represented by an endpoint of p_v on p_u . We delete all parts of the segments that are not used by paths p_v .

Conclusion. It follows from the three steps that each edge $v \rightarrow u$ of G is represented by a contact of the corresponding paths p_v and p_u . Each vertex v is represented by a path consisting of intervals on $\psi(v) + 1$ segments, hence, it is a grid-path with $\psi(v)$ bends. The result is a VCPG of G that agrees with α and ψ . \square

With the four steps we have obtained a VCPG from a realizable pair. This completes the proof of Theorem 1.

4 Bounding the Number of Bends

When a planar graph has a 2-orientation, it easily follows that it has a VCPG. An example is shown in Figure 13.

Lemma 8 *Let G be a planar graph, not necessarily simple. If G has a 2-orientation, i.e., an orientation in which every vertex has outdegree at most 2, then G admits a VCPG.*

Proof: Consider an embedding of G and a 2-orientation α of G . Subdivide each loop twice. If a pair of vertices is connected by multiple edges, all but one of the multiple edges are subdivided. The result is a simple plane graph, which admits a straight-line drawing. The idea is to represent a vertex v by a path p_v that contains v and extends in both directions close to the outgoing edges of v . An end of p_v is a contact on the path p_u of an outneighbor u . Figure 13 illustrates the construction of a VCPG of G . \square

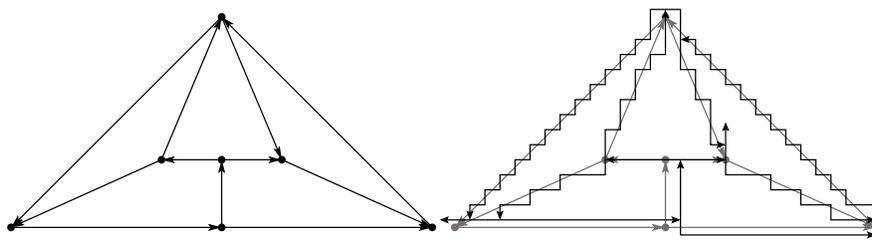


Figure 13: A VCPG drawn by approximating a straight-line drawing of a graph and using a 2-orientation.

Lemma 1 shows that every planar $(2, \ell)$ -tight graph has a 2-orientation. It follows that every planar $(2, \ell)$ -tight graph admits a VCPG.

Using this construction, there is no control over the number of bends of a grid-path. A realizable pair precisely gives the number of bends of each vertex. Adding costs to the vertices in the flow network, it is possible to request few bends for certain vertices, and allow for more bends at others. The characterization also gives a certificate that the total number of bends is minimized. If all the costs are set to 1, and there is a realizable pair that attains a minimum flow, then this is a certificate. Even if this is not the case, the flow value gives a lower bound on the number of bends needed in any representation. We do not know an example where the minimum flow value deviates from the minimum number of bends in a representation, i.e., it might be that there is always a realizable pair (α, ψ) where ψ is a minimum cost flow.

In the following sections, we will show some results on the number bends for a vertex, we will try to locally minimize the number of bends for particular graph classes.

The following proposition is an application of the VCPG of a $(2, \ell)$ -sparse graph G .

Lemma 9 *Let $\ell = 0, 1$ or 2 . Every planar $(2, \ell)$ -sparse graph is the subgraph of a planar $(2, \ell)$ -tight graph, and, of a planar $(2, 0)$ -tight graph.*

Proof: Let G be a $(2, \ell)$ -sparse graph. Let B be the bipartite graph consisting of two copies of each vertex of G in one color class, and the edges of G in the second color class. The edges of B are the incidences between the edges and vertices in G . From the sparsity condition, it follows that B has a matching in which all the edge-vertices are matched:

$$\forall A \subseteq E: |A| \leq 2|V \cap A| - \ell \leq 2|V \cap A| = |N_B(A)|.$$

Hence, the graph G has a 2-orientation and by Lemma 8 G admits a VCPG. Let R be a VCPG of G . Choose ℓ special free ends, which we do not consider in the following.

Within a face f that contains a free end of the grid-path p_v which represents v , the following step is taken:

- If the f is not a 3-face, p_v is extended such that it ends on the grid-path of a non-neighbor of v .
- If the f is a 3-face, we add a new segment perpendicular to p_v at the free end. This segment is extended on both sides to make contacts with the other two vertices of the 3-face.

This takes care of all but ℓ of the free ends, and the resulting graph must be $(2, \ell)$ -tight. To obtain a $(2, 0)$ -tight supergraph, the ℓ remaining free ends are removed with the same technique. The graph that is represented by the resulting VCPG is a tight graph that contains G as a subgraph. \square

Lemma 9 is part of the PhD thesis of the first author. In a recent paper Alam et al. [2] prove a slightly stronger result. They show that for $\ell \in \{0, 1, 2, 3\}$ there is a $(2, \ell)$ -tight graph that has G as a spanning subgraph.

4.1 B_1 -VCPGs

In a B_1 -VCPG, each vertex is represented by a grid-path with at most one bend. Kobourov et al. have shown that every $(2, 3)$ -tight graph admits a strict B_1 -VCPG, i.e., every vertex has precisely one bend [12]. To obtain a strict B_1 -VCPG, the grid-paths of the two vertices on the outer face, get a bend in the outer face, i.e., both the convex and the concave angle of these bends are in the outer face. The representation obtained by the algorithm of Kobourov et al. has the special property that every interior face has precisely one convex angle of a vertex. The authors call this property ‘*proper*’ and they show that plane Laman graphs are precisely the graphs that admit a proper strict B_1 -VCPG. For readers familiar with this work, the *angular tree* can be seen as a flow in the angle graph (see Figure 14). The 2-orientation obtained from splitting along the angular tree, together with the flow given by the angular tree, is a realizable pair. Kobourov et al. asked whether all planar $(2, 2)$ -tight graphs admit a strict B_1 -VCPG.

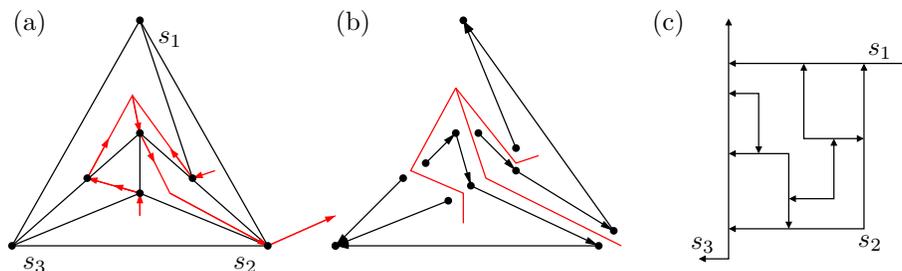


Figure 14: An angular tree, splitting to obtain a 2-orientation and a proper, strict B_1 -VCPG.

In this section we will answer this question by showing a graph that admits B_1 -VCPG but no strict B_1 -VCPG and that not all planar (2,2)-tight graphs admit a B_1 -VCPG. We will also show that every (2,2)-tight planar graph that has an edge e such that the removal of e leaves a (2,3)-tight graph, has a strict B_1 -VCPG (not a proper strict B_1 -VCPG).

There are graphs that are not (2,3)-sparse but admit a B_1 -VCPG, for example K_4 . In this section, we give some insight into the class of graphs that admit a B_1 -VCPG. As mentioned before, it is known that plane (2,3)-tight graphs are precisely the graphs that admit a proper, strict B_1 -VCPG [12]. It follows that planar (2,3)-sparse graphs admit a strict B_1 -VCPG. The complete graph on four vertices, K_4 , shows that there are graphs that have strict B_1 -VCPG but no proper strict B_1 -VCPG. We will show that there are graphs that have a B_1 -VCPG but no strict B_1 -VCPG, i.e., there exist graphs that can be represented when segments are allowed. We will also show that not all planar (2,2)-tight graphs admit a B_1 -VCPG.

Lemma 10 *There are graphs admitting a B_1 -VCPG but no strict B_1 -VCPG.*

Proof: The graph G in Figure 15 admits a B_1 -VCPG but no strict B_1 -VCPG. Every embedding of the graph G contains a subgraph embedded as the subgraph H shown in the right part of the figure. We now concentrate on a B_1 -VCPG representation of H . To satisfy the demands of all 3-faces, the six boundary vertices of H in total have six bends. If x had a bend, then there would be a unit of flow traversing each of the two 4-faces incident to x . This implies that a boundary vertex would get one additional unit of flow. Hence, at least one of the boundary vertices will have two bends. Therefore, x has to be represented as a segment in a B_1 -VCPG of this graph. \square

Lemma 11 *Not every planar (2,2)-tight graph admits a B_1 -VCPG.*

Proof: Suppose a graph has a B_1 -VCPG, then, by Theorem 1, there must be a realizable pair (α, ψ) such that $0 \leq \psi(v) \leq 1$ for all vertices v . Consider the graph from Figure 16. Suppose that there is a plane embedding and a flow

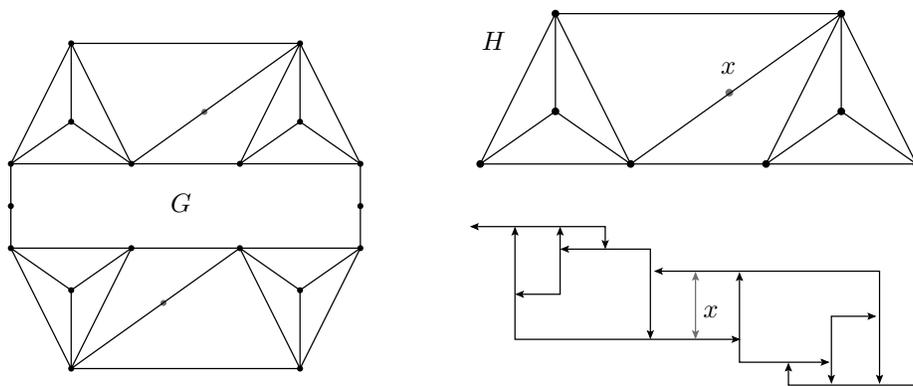


Figure 15: Every embedding of the graph G contains a subgraph embedded as H . In a B_1 -VCPG corresponding to the embedding of H vertex x must be represented by a segment.

ψ which satisfies $0 \leq \psi(v) \leq 1$ for all vertices v . Each embedding has two K_4 subgraphs whose faces are bounded by triangles sharing only vertex m (the grey-colored K_4 subgraphs). These K_4 subgraphs both have demand -3 . So, there must be 6 units of flow going out of the two K_4 subgraphs. However, there are only 5 vertices bounding the two K_4 subgraphs. We conclude that there is no feasible flow such that there is at most one unit of flow going through each vertex. \square

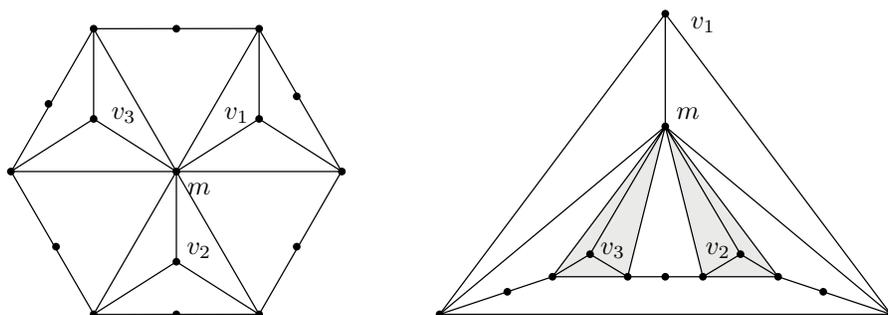


Figure 16: Two embeddings of a planar $(2, 2)$ -tight graph that is not B_1 -VCPG.

A Laman-plus-one graph G is a $(2, 2)$ -tight graph such that there exists an edge e in G for which $G - e$ is $(2, 3)$ -tight, i.e. a Laman graph [10]. In the following, we extend the result of Kobourov et al. from planar $(2, 3)$ -tight graphs to planar Laman-plus-one graphs. There are two bends of vertices that are completely in the outer face (the bends of s_1 and s_3 in Figure 14(c)). The idea is to use these bends for the additional edge.

Theorem 3 *Every planar Laman-plus-one graph has a strict B_1 -VCPG. This representation has the following three properties:*

1. *Precisely one face has two convex corners.*
2. *The outer face has no convex corner.*
3. *All other faces have one convex corner.*

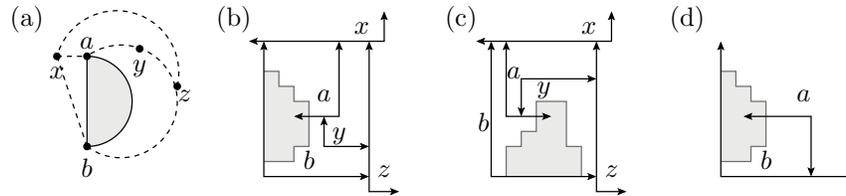


Figure 17: The extension of G and the addition of e .

Proof: Let G^+ be a planar Laman-plus-one graph and $e = ab$ an edge such that $G = G^+ - e$ is a Laman graph. Consider an embedding of G^+ such that e is incident to the outer face and b comes before a in clockwise order around the boundary (as in Figure 17(a)). We use the induced drawing of G and extend this graph to a graph with a triangular outer face (see Figure 17(a)). Three new vertices x, y and z are added to G .

1. Add x and the edges xa and xb .
2. Add y and the edges ya and yb in such a way that the outer face now consists of a, x, b and y .
3. Subdivide the edge by , call the new vertex z and add the edge xz such that the outer face now consists of x, b and z .

The addition of x and y are Henneberg type 1 steps and the addition of z is a Henneberg type 2 step. Therefore, the graph G' , obtained from G with the extension, is a Laman graph. We construct a strict B_1 -VCPG of G' according to the method of Kobourov, Ueckerdt and Verbeek [12], in such a way that x and z are the “special vertices”. Hence, in this VCPG every face has precisely one convex corner and x and z have their bend in the outer face.

Consider the three interior faces that are incident to at least one of x, y and z . All three have one convex corner, which is not the corner of x or z . We will show that a, b and y have their convex corner in these faces. Consider the representation of G that arises by removing x, y and z . There are precisely three free ends, two at b and one at a . This follows from the fact that x and z together have three free ends and xz must use the fourth of their ends, therefore, the edges of a and b are oriented such that $b \rightarrow z, b \rightarrow x$ and $a \rightarrow x$. The boundary of G is a closed cycle and in the representation it has three free ends into the outer face. We claim that all other vertices on this boundary have their concave angle in the outer face.

To prove the claim, we do some counting. Let s be the number of edges, c^+ be the number of convex corners and c^- be the number of concave corners on the outer face of G . Since the total change in direction around the outer face is 2π and the three free ends contribute π each we get: $2 = s - c^- + c^+$. Between two contacts (edges), we may see a straight segment, a convex corner or a concave corner. Hence, with t being the number of straight segments: $s = t + c^+ + c^-$. Substituting this in the previous formula, we obtain $2 = t + 2c^+$. We know that a ends on x and y ends on z , since x and z contribute a horizontal respectively vertical part, a and y have a vertical respectively horizontal part. Since y must also end on a both y and a must have a horizontal and a vertical part outside of G . Hence, a contributes no corner to the boundary of G , i.e., $t > 0$. Since c^+ is a non-negative integer, we conclude that $c^+ = 0$. This was the claim.

From the claim, it follows that a , b and y must have their convex corners in the three new faces. Therefore, the rest of the graph connects to only one side of one leg of b , see Figure 17(b) and (c). The two cases are similar and therefore we only consider the first one, where the rest of the graph lives on the vertical part of the grid-path of b . The convex corner of b only has one option. Since y only contributes to two faces and one of them is taken by b , there is no choice for the convex corner of y . Finally, the convex corner of a must be in $axyz$. The corners must be precisely as in Figure 17(b).

Removing x , y and z from the representation leaves a strict B_1 -VCPG in which the leg of the free end of a has no vertices ending on it. Moreover, the legs of a can be extended such that no vertex but one of the free ends of b is to the right of this leg of a , since a has its corner in the outer face of G in the representation. Hence, we can extend this leg in such a way that it ends on b . The result is a strict B_1 -VCPG of G^+ . Figure 17(d) shows the result. It follows from the construction of Kobourov et al. that all faces except for the two incident to the bend of a , have precisely one convex corner. The face incident to the convex corner of a has also a convex corner from b and the outer face has no convex corners. \square

The graph in Figure 15 with vertex x deleted is not a Laman-plus-one graph. However, it has a strict B_1 -VCPG. It would be interesting to characterize of the class of graphs that have a strict B_1 -VCPG.

4.2 B_2 -VCPGs

For simple (2,2)-tight graphs, we show that for every 2-orientation α , there exists a flow ψ such that $\psi(v) \leq 2$ for all vertices v and the pair (α, ψ) is realizable.

Theorem 4 *Every planar (2,2)-tight graph admits a B_2 -VCPG.*

Proof: Let G be a planar (2,2)-tight graph. We fix an embedding of G . For the proof, we construct an appropriate realizable pair (α, ψ) .

The underlying graph of the flow ψ is the angle graph $A(G)$ of G . We construct ψ in two steps. First we define an auxiliary flow ψ^* on the dual G^*

of G such that ψ^* satisfies all demands of ψ . In the second step, we route the auxiliary flow on the edges of $A(G)$.

The demand of the faces is given by $c(f) = |f| - 4$ and for the outer face it is $c(f_\infty) = |f_\infty|$ (Proposition 2 with $\ell = 2$). We claim that there is a flow ψ^* satisfying all demands such that the flow value in each dual edge is at most one. This is shown by checking the size of all cuts. Let H be a subgraph induced by a subset F_H of bounded faces of G . Let b be the number of boundary edges of H . We have to show that $b \geq |\sum_{f \in F_H} c(f)|$.

Let v_H , e_H and f_H^* be the number of vertices, edges and bounded faces of H , respectively. Using Euler's formula ($v_H - e_H + f_H^* = 1$) and the upper bound on the edges in H , we obtain the following.

$$\begin{aligned} \sum_{f \in F_H} (4 - |f|) &= 4f_H^* - 2e_H + b & (5) \\ &= 4e_H - 4v_H + 4 - 2e_H + b \quad (\text{Euler's Formula}) \\ &= 2e_H - 4v_H + 4 + b \\ &\leq 4v_H - 2\ell - 4v_H + 4 + b \quad (\text{since } e_H \leq 2v_H - \ell) \\ &= b + 4 - 2\ell = b \quad (\text{since } \ell = 2) \end{aligned}$$

The vertices V_H of H are partitioned into boundary vertices B_H and inner vertices I_H . We have $|E[V \setminus I_H]| \leq 2|V| - 2|I_H| - \ell$ and hence, $e_H - b = |\overline{E[V \setminus I_H]}| \geq (2|V| - \ell) - (2|V| - 2|I_H| - \ell) = 2|I_H| = 2(v_H - b)$. Using $e_H \leq 2v_H - b$ in Euler's formula, we obtain $f_H^* \geq v_H - b + 1$.

Therefore, we obtain the following:

$$\begin{aligned} \sum_{f \in F_H} (|f| - 4) &= 2e_H - b - 4f_H^* & (6) \\ &= (2v_H + 2f_H^* - 2) - b - 4f_H^* \quad (\text{Euler's Formula}) \\ &= 2v_H - 2f_H^* - 2 - b \\ &\leq 2v_H - 2v_H + 2b + 2 - 2 - b \quad (\text{since } f_H^* \geq v_H - b + 1) \\ &= b \end{aligned}$$

Putting this together we obtain:

$$\left| \sum_{f \in H} c(f) \right| = \left| \sum_{f \in H} (|f| - 4) \right| \leq b.$$

It follows from the max-flow min-cut theorem, that there is a flow in the dual graph that satisfies all demands and respects the capacity bound of 1 in each edge. Since all constraints are integral we can assume that the flow is integral, i.e., a 0,1-flow. From Lemma 1, we know that there is a 2-orientation α of G such that the only vertices with outdegree less than 2 are incident to the outer face. Using α , we construct the flow ψ in the angle graph as follows: If there is a flow from f_1 to f_2 crossing edge $u \rightarrow v$, then we add $f_1 \rightarrow u$ and $u \rightarrow f_2$ to ψ .

Since the flow in the dual graph is edge-disjoint and each vertex has at most two outgoing edges we have $\psi(v) \leq 2$ for all vertices v . At each vertex, the flow cuts off the outgoing edge. Hence, the realizability condition is satisfied at each vertex. We conclude that the pair (α, ψ) is realizable. \square

4.3 B_k -VCPGs for $k > 2$

For (2,2)-tight graphs, using a flow in the dual gives a tight bound. We have seen before that there are planar (2,2)-tight graphs which do not admit a B_1 -VCPG (Lemma 11). On the other hand, using the dual graph, we could show that two bends per vertex are sufficient. Unfortunately this is not the case for (2,1)- and (2,0)-tight graphs.

Theorem 5 *Every simple planar (2,1)-tight graph admits a B_4 -VCPG.*

Proof: We claim that there is a flow in the dual graph such that each edge has capacity 2. Similarly to the proof of Theorem 4, we obtain that a simple planar (2,1)-tight graph admits a B_4 -VCPG.

Again, let H be a subgraph induced by a subset F_H of bounded faces of G and let b be the number of boundary edges of H . From Equations (5) and (6) we obtain:

$$\left| \sum_{f \in F_H} c(f) \right| = \left| \sum_{f \in F_H} (|f| - 4) \right| \leq b + 4 - 2\ell = b + 2.$$

There are no loops, therefore, $b > 1$. It follows that the demands can be satisfied using b edges with capacity 2. \square

We have only been able to show that the lower bound is 2. An example is given by the octahedron with an interior edge removed. We conjecture that every simple planar (2,1)-tight graph admits a B_2 -VCPG.

Theorem 6 *Every simple planar (2,0)-tight graph admits a B_6 -VCPG.*

Proof: Let G be a simple planar (2,0)-tight graph. Again, let H be a subgraph induced by a subset F_H of bounded faces of G and let b be the number of boundary edges of H . From Equations (5) and (6) we obtain:

$$\left| \sum_{f \in F_H} c(f) \right| = \left| \sum_{f \in F_H} (4 - |f|) \right| \leq 4 + b. \quad (7)$$

As there are no loops, we can satisfy all the demands in the dual graph by using every edge at most thrice.

As in the proof of Theorem 4, this induces a realizable pair, in which every vertex has at most 6 units of flow. This yields a B_6 -VCPG. \square

4.4 Obtaining Better Bounds

In the previous section, we have used an auxiliary flow ψ^* in the dual graph to obtain a feasible flow ψ in the angle graph. This way, we are certain that there exists a 2-orientation such that the pair is realizable and, hence, is in bijection to a VCPG. Even stronger, the choice of the 2-orientation does not matter. Unfortunately, it neither minimizes the total number of bends nor does it minimize the number of bends per vertex when a 2-orientation is given. We

illustrate this with the following example. In Figure 18, two VCPGs of the octahedron are shown, the left of which does not relate to a feasible flow in the dual graph. Any feasible flow in the dual graph will lead to a VCPG with at least 12 bends in total, while 8 is the minimum. Any feasible flow in the dual graph together with the chosen orientation, ensures that c has no bends incident to the outer face in the VCPG. To close the outer cycle at least 7 bends are needed, which will be divided over a and b , hence, there will be a vertex with 4 bends while 3 is the minimum.

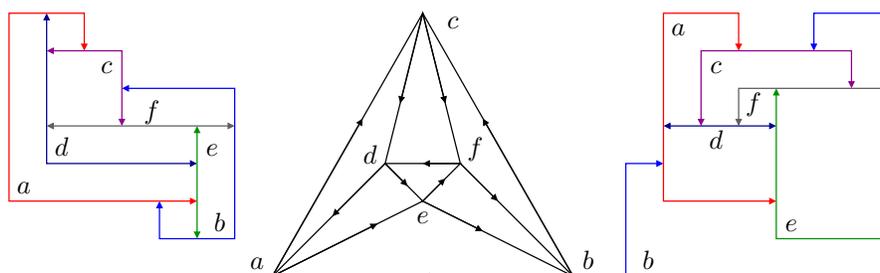


Figure 18: Two VCPGs of the octahedron that induce the same 2-orientation. The VCPG on the left does not relate to a flow in the dual graph as the rightmost bend on the bottom (of the grid-path of vertex b) cannot be transformed into a flow in the dual graph.

It has been fruitful to use the dual graph for bounding the number of bends locally at each vertex, but what if we want to globally minimize the number of bends. Ideally, there would always exist a 2-orientation such that the minimum cost feasible flow is realizable. However, for the graph in Figure 19, a minimum cost flow is given (the value of the flow is equal to the number of 3-faces). For this flow, there does not exist a 2-orientation such that the pair is realizable. Suppose there is such a 2-orientation. The realizability condition requires the orientations $a \rightarrow c$ and $b \rightarrow c$. Then c has only one outgoing edge, but the flow requires the free ends to be in the outer face, contradiction. Note that rerouting a unit of flow that uses a or b to go through c leaves a flow for which there does exist a 2-orientation such that the pair is realizable.

5 Conclusion

We have shown that there exist planar $(2, 2)$ -tight graphs that do not admit a B_1 -VCPG. However, the only type of $(2, 2)$ -tight planar graph that we found not to have a B_1 -VCPG has at least one vertex which is the intersection of “many” critical subsets. Do all planar $(2, 2)$ -tight graphs that have no such vertex admit a B_1 -VCPG?

We have also obtained bounds for simple $(2, 1)$ -tight and $(2, 0)$ -tight planar graphs. However, we believe that these bounds are not tight. A lower bound of

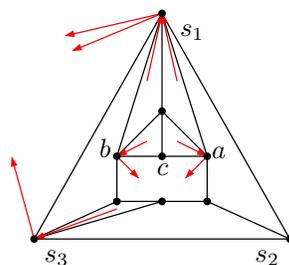


Figure 19: A minimum feasible flow for which there is no 2-orientation such that the pair is realizable.

three bends for simple, planar $(2,0)$ -tight graphs is given by the octahedron.

Conjecture 1 *Every simple planar $(2,0)$ -tight graph admits a B_3 -VCPG.*

Conjecture 2 *Every simple planar $(2,1)$ -tight graph admits a B_2 -VCPG.*

The bounds that we have shown do not depend on a chosen 2-orientation (i.e. the bounds hold for every 2-orientation). It would be interesting to find a sufficient condition on a flow such that, when satisfied, there exists a 2-orientation such that the pair is realizable. For $(2,3)$ -tight graphs, the algorithm of Kobourov, Ueckerdt and Verbeek, takes a particular flow and a particular network as an input. Is there a way to construct a realizable pair simultaneously for all $(2,0)$ -sparse graphs?

References

- [1] N. Aerts and S. Felsner. Vertex Contact Graphs of Paths on a Grid. In D. Kratsch and I. Todinca, editors, *Proc. WG*, volume 8747 of *Lecture Notes in Computer Science*, pages 56–68. 2014. doi:10.1007/978-3-319-12340-0_5.
- [2] M. J. Alam, D. Eppstein, M. Kaufmann, S. G. Kobourov, S. Pupyrev, A. Schulz, and T. Ueckerdt. Contact representations of sparse planar graphs. *CoRR*, abs/1501.00318, 2015. URL: <http://arxiv.org/abs/1501.00318>.
- [3] A. Asinowski, E. Cohen, M. C. Golumbic, V. Limouzy, M. Lipshteyn, and M. Stern. Vertex Intersection Graphs of Paths on a Grid. *J. Graph Alg. and Appl.*, 16(2):129–150, 2012. doi:10.7155/jgaa.00253.
- [4] S. Chaplick and T. Ueckerdt. Planar graphs as VPG-graphs. *J. Graph Alg. and Appl.*, 17(4):475–494, 2013. doi:10.7155/jgaa.00300.
- [5] H. de Fraysseix and P. O. de Mendez. On topological aspects of orientations. *Discr. Math.*, 229(1-3):57–72, 2001. doi:10.1016/S0012-365X(00)00201-6.
- [6] H. de Fraysseix, P. O. de Mendez, and J. Pach. A left-first search algorithm for planar graphs. *Discr. and Comput. Geom.*, 13:459–468, 1995. doi:10.1007/BF02574056.
- [7] H. de Fraysseix, P. O. de Mendez, and P. Rosenstiehl. On triangle contact graphs. *Comb., Probab. and Comput.*, 3(02):233–246, 1994. doi:10.1017/S0963548300001139.
- [8] S. Felsner. Rectangle and square representations of planar graphs. In J. Pach, editor, *Thirty Essays on Geometric Graph Theory*, pages 213–248. Springer, 2013. doi:10.1007/978-1-4614-0110-0_12.
- [9] U. Fößmeier, G. Kant, and M. Kaufmann. 2-visibility drawings of planar graphs. In *Proc. Graph Drawing*, volume 1190 of *Lec. Notes Comp. Sci.*, pages 155–168. Springer, 1996. doi:10.1007/3-540-62495-3_45.
- [10] R. Haas, D. Orden, G. Rote, F. Santos, B. Servatius, H. Servatius, D. L. Souvaine, I. Streinu, and W. Whiteley. Planar minimally rigid graphs and pseudo-triangulations. *Comp. Geom.: Theory and Appl.*, 31(1-2):31–61, 2005. doi:10.1016/j.comgeo.2004.07.003.
- [11] I. B.-A. Hartman, I. Newman, and R. Ziv. On grid intersection graphs. *Discr. Math.*, 87(1):41–52, 1991. doi:10.1016/0012-365X(91)90069-E.
- [12] S. G. Kobourov, T. Ueckerdt, and K. Verbeek. Combinatorial and geometric properties of planar Laman graphs. In *Proc. ACM-SIAM Symp. Discr. Algo.*, pages 1668–1678, 2013. doi:10.1137/1.9781611973105.120.
- [13] M. W. Schäffter. Drawing graphs on rectangular grids. *Discr. Appl. Math.*, 63(1):75–89, 1995. doi:10.1016/0166-218X(94)00020-E.
- [14] R. Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM J. Comput.*, 16(3):421–444, 1987. doi:10.1137/0216030.