A Universal Slope Set for 1-Bend Planar Drawings

Patrizio Angelini[1], Michael A. Bekos[2], Giuseppe Liotta[3], and Fabrizio Montecchiani[4]

1 **Wilhelm-Schickhard-Institut für Informatik, Universität Tübingen, Tübingen, Germany**
angelini@informatik.uni-tuebingen.de
2 **Wilhelm-Schickhard-Institut für Informatik, Universität Tübingen, Tübingen, Germany**
bekos@informatik.uni-tuebingen.de
3 **Universitá degli Studi di Perugia, Perugia, Italy**
giuseppe.liotta@unipg.it
4 **Universitá degli Studi di Perugia, Perugia, Italy**
fabrizio.montecchiani@unipg.it

Abstract

We describe a set of $\Delta-1$ slopes that are universal for 1-bend planar drawings of planar graphs of maximum degree $\Delta \geq 4$; this establishes a new upper bound of $\Delta-1$ on the 1-bend planar slope number. By universal we mean that every planar graph of degree Δ has a planar drawing with at most one bend per edge and such that the slopes of the segments forming the edges belong to the given set of slopes. This improves over previous results in two ways: Firstly, the best previously known upper bound for the 1-bend planar slope number was $\frac{3}{2}(\Delta-1)$ (the known lower bound being $\frac{3}{4}(\Delta-1)$); secondly, all the known algorithms to construct 1-bend planar drawings with $O(\Delta)$ slopes use a different set of slopes for each graph and can have bad angular resolution, while our algorithm uses a universal set of slopes, which also guarantees that the minimum angle between any two edges incident to a vertex is $\frac{\pi}{(\Delta-1)}$.

1998 ACM Subject Classification G.2.1 Combinatorics, G.2.2 Graph Theory

Keywords and phrases Slope number, 1-bend drawings, planar graphs, angular resolution

Digital Object Identifier 10.4230/LIPIcs.SoCG.2017.9

1 Introduction

This paper is concerned with planar drawings of graphs such that each edge is a poly-line with few bends, each segment has one of a limited set of possible slopes, and the drawing has good angular resolution, i.e. it forms large angles between consecutive edges incident on a common vertex. Besides their theoretical interest, visualizations with these properties find applications in software engineering and information visualization (see, e.g., [12, 26, 40]). For example, planar graphs of maximum degree four (degree-4 planar graphs) are widely used in database design, where they are typically represented by orthogonal drawings, i.e. crossing-free drawings such that every edge segment is a polygonal chain of horizontal and vertical segments. Clearly, orthogonal drawings of degree-4 planar graphs are optimal both in terms of angular resolution and in terms of number of distinct slopes for the edges. Also, a classical result in the graph drawing literature is that every degree-4 planar graph, except the octahedron, admits an orthogonal drawing with at most two bends per edge [5, 34].

It is immediate to see that more than two slopes are needed in a planar drawing of a graph with vertex degree $\Delta \geq 5$. The *k-bend planar slope number* of a graph G with degree Δ is the

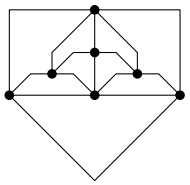


Figure 1 A 1-bend planar drawing with 4 slopes and angular resolution $\frac{\pi}{4}$ of a graph with $\Delta = 5$.

minimum number of distinct slopes that are sufficient to compute a crossing-free drawing of G with at most k bends per edge. Keszegh et al. [30] generalize the technique by Biedl and Kant [5] and prove that for any $\Delta \geq 5$, the 2-bend planar slope number of a degree-Δ planar graph is $\lceil \Delta/2 \rceil$; the construction in their proof has optimal angular resolution, that is $\frac{2\pi}{\Delta}$.

For the case of drawings with one bend per edge, Keszegh et al. [30] also show an upper bound of 2Δ and a lower bound of $\frac{3}{4}(\Delta - 1)$ on the 1-bend planar slope number, while a recent paper by Knauer and Walczak [31] improves the upper bound to $\frac{3}{2}(\Delta - 1)$. Both these papers use a similar technique: First, the graph is realized as a contact representation with T-shapes [10], which is then transformed into a planar drawing where vertices are points and edges are poly-lines with at most one bend. The set of slopes depends on the initial contact representation and may change from graph to graph; also, each slope is either very close to horizontal or very close to vertical, which gives rise to bad angular resolution. Knauer and Walczak [31] also proved that the 1-bend (outer)planar slope number of outerplanar graphs with $\Delta > 2$ is $\lceil \frac{\Delta}{2} \rceil$ and that $\Delta + 1$ slopes suffice for planar bipartite graphs.

We study the trade-off between number of slopes, angular resolution, and number of bends per edge in a planar drawing of a graph with maximum degree Δ. We improve the upper bounds in [31] on the 1-bend planar slope number, and at the same time we achieve $\Omega(\frac{1}{\Delta})$ angular resolution. More precisely, we prove the following.

▶ **Theorem 1.** *For any $\Delta \geq 4$, there exists an equispaced universal set S of $\Delta - 1$ slopes for 1-bend planar drawings of planar graphs with maximum degree Δ. That is, every such graph has a planar drawing with the following properties: (i) each edge has at most one bend; (ii) each edge segment uses one of the slopes in S; and (iii) the minimum angle between any two consecutive edge segments incident on a vertex or a bend is at least $\frac{\pi}{\Delta - 1}$.*

Theorem 1, in conjuction with [27], implies that the 1-bend planar slope number of planar graphs with $n \geq 5$ vertices and maximum degree $\Delta \geq 3$ is at most $\Delta - 1$. We prove the theorem by using an approach that is conceptually different from that of Knauer and Walczak [31]: We do not construct an intermediate representation, but we prove the existence of a *universal* set of slopes and use it to directly compute a 1-bend planar drawing of any graph with degree at most Δ. The universal set of slopes consists of $\Delta - 1$ distinct slopes such that the minimum angle between any two of them is $\frac{\pi}{(\Delta - 1)}$. An immediate consequence of the $\frac{3}{4}(\Delta - 1)$ lower bound argument in [30] is that a 1-bend planar drawing with the minimum number of slopes cannot have angular resolution larger than $\frac{4}{3}\frac{\pi}{(\Delta - 1)}$. Hence, the angular resolution of our drawings is optimal up to a multiplicative factor of at most 0.75; also, note that the angular resolution of a graph of degree Δ is at most $\frac{2\pi}{\Delta}$ even when the number of slopes and the number of bends along the edges are not bounded.

The proof of Theorem 1 is constructive and it gives rise to a linear-time algorithm assuming the real RAM model of computation. A drawing computed with this algorithm is shown in Fig. 1. The construction for triconnected planar graphs uses a variant of the shifting method of de Fraysseix, Pach and Pollack [11]; this is used as a building block for

the drawing algorithm for biconnected planar graphs, which is based on the SPQR-tree decomposition of the graph into its triconnected components (see, e.g., [12]). By using a BC-tree decomposition, our approach can be further extended to general planar graphs, details can be found in the extended version of this paper [1]. If the graph is disconnected, since we use a universal set of slopes, the distinct connected components can be drawn independently.

Related work. The results on the slope number of graphs are mainly classified into two categories based on whether the input graph is planar or not. For a (planar) graph G of maximum degree Δ, the *slope number* (*planar slope number*) is the minimum number of slopes that are sufficient to compute a straight-line (planar) drawing of G. The slope number of non-planar graphs is lower bounded by $\lceil \Delta/2 \rceil$ [41] but it can be arbitrarily large, even when $\Delta = 5$ [2]. For $\Delta = 3$ this number is 4 [35], while it is unknown for $\Delta = 4$. Upper bounds on the slope number are known for complete graphs [41] and outer 1-planar graphs [14]. Deciding whether a graph has slope number 2 is NP-complete [15, 19]. For a planar graph G of maximum degree Δ, the planar slope number of G is lower bounded by $3\Delta - 6$ and upper bounded by $O(2^{\Delta})$ [30]. Improved upper bounds are known for special subclasses of planar graphs, e.g., planar graphs with $\Delta \leq 3$ [15, 13, 28], outerplanar graphs with $\Delta \geq 4$ [32], partial 2-trees [33], planar partial 3-trees [25]. Note that determining the planar slope number of a graph is hard in the existential theory of the reals [24].

Closely related to our problem is the problem of finding *d-linear* drawings of graphs, in which all angles (formed either between consecutive segments of an edge or between edge-segments incident to the same vertex) are multiples of $2\pi/d$. For $d = 4$, an angular resolution of $2\pi/d$ implies d-linearity, while for $d > 4$ this is not always true [8]. Special types of d-linear drawings are the *orthogonal* [5, 7, 21, 39] and the *octilinear* [3, 4, 36] drawings, for which $d = 2$ and $d = 4$ holds, respectively. As already recalled, any planar graph with $\Delta \leq 4$ (except the octahedron) admits a planar orthogonal drawing with at most two bends per edge [5, 34]. Deciding whether a degree-4 planar graph has an orthogonal drawing with no bends is NP-complete [21], while it is solvable in polynomial time if one bend per edge is allowed [6]. Octilinear drawings have been mainly studied in the context of metro map visualization and map schematization [37, 38]. Nöllenburg [36] proved that deciding whether an embedded planar graph with $\Delta \leq 8$ admits a bendless planar octilinear drawing is NP-complete. Bekos et al. [3] showed that any planar graph with $\Delta \leq 5$ admits a planar octilinear drawing with at most one bend per edge and this is not always possible if $\Delta \geq 6$. Note that in our work we generalize their positive result to any Δ. Later, Bekos et al. [4] studied bounds on the total number of bends of planar octilinear drawings. Finally, trade-offs between number of bends, angular resolution, and area requirement of planar drawings of graphs with maximum degree Δ are, for example, studied in [9, 16, 17, 18, 20, 22].

Paper organization. The rest of this paper is organized as follows. Preliminaries are given in Section 2. In Section 3, we describe a drawing algorithm for triconnected planar graphs. The technique is extended to biconnected planar graphs in Sections 4. Finally, in Section 5 we discuss further implications of Theorem 1 and we list open problems.

2 Preliminaries

A graph G containing neither loops nor multiple edges is *simple*. We consider simple graphs, if not otherwise specified. The *degree* of a vertex of G is the number of its neighbors. We say

that G has *maximum degree* Δ if it contains a vertex with degree Δ but no vertex with degree larger than Δ. A graph is *connected*, if for any pair of vertices there is a path connecting them. Graph G is *k-connected*, if the removal of $k-1$ vertices leaves the graph connected. A 2-connected (3-connected) graph is also called *biconnected* (*triconnected*, respectively).

A *drawing* Γ of G maps each vertex of G to a point in the plane and each edge of G to a Jordan arc between its two endpoints. A drawing is *planar*, if no two edges cross (except at common endpoints). A planar drawing divides the plane into connected regions, called *faces*. The unbounded one is called *outer face*. A graph is *planar*, if it admits a planar drawing. A *planar embedding* of a planar graph is an equivalence class of planar drawings that combinatorially define the same set of faces and outer face.

The *slope* s of a line ℓ is the angle that a horizontal line needs to be rotated counterclockwise in order to make it overlap with ℓ. The slope of an edge-segment is the slope of the line containing the segment. Let S be a set of slopes sorted in increasing order; assume (up to a rotation) that S contains the 0 angle, which we call *horizontal slope*. A *1-bend* planar drawing Γ of graph G *on* S is a planar drawing of G in which every edge is composed of at most two straight-line segments, each of which has a slope that belongs to S. We say that S is *equispaced* if and only if the difference between any two consecutive slopes of S is $\frac{\pi}{|S|}$. For a vertex v in G, each slope $s \in S$ defines two different *rays* that emanate from v and have slope s. If s is the horizontal slope, then these rays are called *horizontal*. Otherwise, one of them is the *top* and the other one is the *bottom* ray of v. Consider a 1-bend planar drawing Γ of a graph G and a ray r_v emanating from a vertex v of G. We say that r_v is *free* if there is no edge attached to v through r_v. We also say that r_v is *incident* to face f of Γ if and only if r_v is free and the first face encountered when moving from v along r_v is f.

Let Γ be a 1-bend planar drawing of a graph and let e be an edge incident to the outer face of Γ that has a horizontal segment. A *cut at* e is a y-monotone curve that

(i) starts at any point of the horizontal segment of e,
(ii) ends at any point of a horizontal segment of an edge $e' \neq e$ incident to the outer face of Γ, and
(iii) crosses only horizontal segments of Γ.

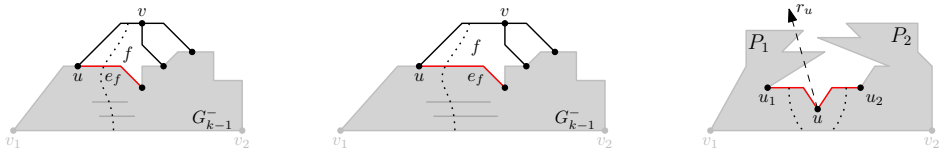
Central in our approach is the canonical order of triconnected planar graphs which can be computed in linear time [11, 29]. Let $G = (V, E)$ be a triconnected planar graph and let $\Pi = (P_0, \ldots, P_m)$ be a partition of V into paths, such that $P_0 = \{v_1, v_2\}$, $P_m = \{v_n\}$, edges (v_1, v_2) and (v_1, v_n) exist and belong to the outer face of G. For $k = 0, \ldots, m$, let G_k be the subgraph induced by $\cup_{i=0}^{k} P_i$ and denote by C_k the outer face of G_k. Π is a *canonical order* of G if for each $k = 1, \ldots, m-1$ the following hold:

(i) G_k is biconnected,
(ii) all neighbors of P_k in G_{k-1} are on C_{k-1},
(iii) $|P_k| = 1$ or the degree of each vertex of P_k is two in G_k, and
(iv) all vertices of P_k with $0 \leq k < m$ have at least one neighbor in P_j for some $j > k$.

An SPQR-tree $\mathcal{T}$ represents the decomposition of a biconnected graph G into its triconnected components and it can be computed in linear time [12, 23]. Every triconnected component of G is associated with a node μ of $\mathcal{T}$. The triconnected component itself is called the *skeleton* of μ, denoted by G_μ^{skel}. A node μ in $\mathcal{T}$ can be of four different types:

(i) μ is an *R-node*, if G_μ^{skel} is a triconnected graph,
(ii) a simple cycle of length at least three classifies μ as an *S-node*,
(iii) a bundle of at least three parallel edges classifies μ as a *P-node*,
(iv) the leaves of $\mathcal{T}$ are *Q-nodes*, whose skeleton consists of two parallel edges.

Neither two S- nor two P-nodes are adjacent in $\mathcal{T}$. A *virtual edge* in G_μ^{skel} corresponds to a



(a) (b) (c)

Figure 2 Illustrations for (a-b) Lemma 2 and (c) Lemma 3.

tree node ν that is adjacent to μ in $\mathcal{T}$, more precisely, to another virtual edge in G_ν^{skel}. If we assume that $\mathcal{T}$ is rooted at a Q-node ρ, then every skeleton (except the one of ρ) contains exactly one virtual edge, called *reference edge* and whose endpoints are the *poles* of μ, that has a counterpart in the skeleton of its parent. Every subtree $\mathcal{T}_\mu$ rooted at a node μ of $\mathcal{T}$ induces a subgraph G_μ of G called *pertinent*, that is described by $\mathcal{T}_\mu$ in the decomposition.

3 Triconnected Planar Graphs

Let G be a triconnected planar graph with $\Delta \geq 4$ and let S be a set of $\Delta - 1$ equispaced slopes containing the horizontal one. We consider the vertices of G according to a canonical order $\Pi = (P_0, \ldots, P_m)$. For $k = 0, \ldots, m$, let G_k^- be the planar graph obtained by removing edge (v_1, v_2) from G_k, and let C_k^- be the path from v_1 to v_2 obtained by removing (v_1, v_2) from C_k. We construct a 1-bend planar drawing of G_k^- on S satisfying the following invariants.

(I.1) No part of the drawing lies below vertices v_1 and v_2, which have the same y-coordinate.
(I.2) Every edge on C_k^- has a horizontal segment.
(I.3) Each vertex v on C_k^- has at least as many free top rays incident to the outer face of G_k^- as the number of its neighbors in $G \setminus G_k$.

Once a 1-bend planar drawing on S of G_m^- satisfying Invariants I.3–I.3 has been constructed, a 1-bend planar drawing on S of $G = G_m^- \cup \{(v_1, v_2)\}$ can be obtained by drawing (v_1, v_2) as a polyline composed of two straight-line segments, one attaching at the first clockwise bottom ray of v_1 and the other one at the first anti-clockwise bottom ray of v_2. Since S has at least three slopes, these two rays cross. Invariant I.3 ensures that edge (v_1, v_2) does not introduce any crossing. Next, we state two useful properties of any 1-bend planar drawing on S satisfying Invariants I.3–I.3, whose proofs can be found in [1]; see also Figs. 2a–2c).

▶ **Lemma 2.** *Let Γ_k be a 1-bend planar drawing on S of G_k^- satisfying Invariants I.3–I.3. Let (u, v) be an edge of C_k^- such that u precedes v along path C_k^- and let $\sigma > 0$. There exists a 1-bend planar drawing Γ_k' on S of G_k^-, satisfying Invariants I.3–I.3, in which the horizontal distance between any two consecutive vertices along C_k^- is the same as in Γ_k, except for u and v, whose horizontal distance is increased by σ.*

▶ **Lemma 3.** *Let Γ_k be a 1-bend planar drawing on S of G_k^- satisfying Invariants I.3–I.3. Let u be any vertex of C_k^-, and let r_u be any free top ray of u that is incident to the outer face of G_k^- in Γ_k. Then, it is possible to construct a 1-bend planar drawing Γ_k' on S of G_k^-, satisfying Invariants I.3–I.3, in which r_u does not cross any edge of Γ_k'.*

We now describe our algorithm. First, we draw $P_0 = \{v_1, v_2\}$ and $P_1 = \{v_3, \ldots, v_j\}$ such that $v_1, v_3, \ldots, v_j, v_2$ lie along a horizontal line, in this order (edge (v_1, v_2) is not considered). Invariants I.3 and I.3 hold. Invariant I.3 follows since S contains $\Delta - 2$ top rays and all vertices drawn so far (including v_1 and v_2) have at most $\Delta - 2$ neighbors later in the canonical

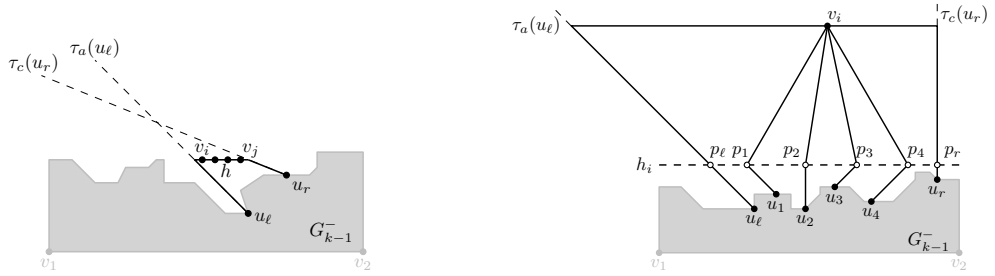


Figure 3 Illustration of the cases of: (a) a chain, (b) a singleton of degree δ_i in G_k.

order. We now show how to add path P_k, for some $k > 1$, to a drawing Γ_{k-1} satisfying Invariants I.3–I.3, so that the resulting drawing Γ_k of G_k^- is a 1-bend planar drawing on S satisfying Invariants I.3–I.3. We distinguish the cases in which P_k is a chain or a singleton.

Suppose first that P_k is a chain, say $\{v_i, v_{i+1}, \ldots, v_j\}$; refer to Fig. 3a. Let u_ℓ and u_r be the neighbors of v_i and v_j in C^-_{k-1}, respectively. By Invariant I.3, each of u_ℓ and u_r has at least one free top ray that is incident to the outer face of Γ_{k-1}; among them, we denote by $\tau_a(u_\ell)$ the first one in anti-clockwise order for u_ℓ, and by $\tau_c(u_r)$ the first one in clockwise order for u_r. By Lemma 3, we can assume that $\tau_a(u_\ell)$ and $\tau_c(u_r)$ do not cross any edge in Γ_{k-1}. This implies that there exists a horizontal line-segment h whose left and right endpoints are on $\tau_a(u_\ell)$ and $\tau_c(u_r)$, respectively, that does not cross any edge of Γ_{k-1}. We place all the vertices $v_i, v_{i+1}, \ldots, v_j$ of P_k on interior points of h, in this left-to-right order. Then, we draw edge (u_ℓ, v_i) with a segment along h and the other one along $\tau_a(u_\ell)$; we draw edge (u_r, v_j) with a segment along h and the other one along $\tau_c(u_r)$, and we draw every edge (v_q, v_{q+1}), with $q = i, \ldots, j-1$, with a unique segment along h.

By construction, Γ_k is a planar drawing on S. All the vertices of P_k lie above u_ℓ and u_r, since $\tau_a(u_\ell)$ and $\tau_c(u_r)$ are top rays of u_ℓ and u_r, respectively. Hence, these vertices and their incident edges lie above v_1 and v_2, and thus Invariant I.3 is satisfied by Γ_k. Invariant I.3 is satisfied since every edge that is drawn at this step has a segment along h, which is horizontal. Invariant I.3 is satisfied since we attached edges (u_ℓ, v_i) and (u_r, v_j) at vertices u_ℓ and u_r using the first anti-clockwise free top ray of u_ℓ and the first clockwise free top ray of u_r among those incident to the outer face, respectively. Thus, we reduced only by one the number of free top rays incident to the outer face for u_ℓ and u_r. For the other vertices of P_k, the invariant is satisfied since their $\Delta - 2$ top rays are free and incident to the outer face. This concludes our description for the case in which P_k is a chain.

Suppose now that P_k is a singleton $\{v_i\}$ of degree $\delta_i \leq \Delta$ in G_k^-. This also includes the case in which $P_k = P_m$ is the last path of Π. If $\delta_i = 2$, then v_i is placed as in the case of a chain. So, we may assume that $\delta_i \geq 3$. Let $u_\ell, u_1, u_2, \ldots, u_{\delta_i - 2}, u_r$ be the neighbors of v_i as they appear along C^-_{k-1}; see Fig. 3b. By Invariant I.3, each neighbor of v_i in C^-_{k-1} has at least one free top ray incident to the outer face of Γ_{k-1}; let $\tau_a(u_\ell)$ be the first of them in anti-clockwise order for u_ℓ and $\tau_c(u_r)$ be the first of them in clockwise order for u_r. Also, for each vertex u_q, with $q = 1, \ldots, \delta_i - 2$, let $\tau(u_q)$ be any of these rays. By Lemma 3, we can assume that these rays do not cross any edge in Γ_{k-1}.

Consider a horizontal line h_i above all vertices of Γ_{k-1}. Rays $\tau_a(u_\ell)$, $\tau(u_1), \ldots, \tau(u_{\delta_i-2})$, $\tau_c(u_r)$ cross h_i; however, the corresponding intersection points p_ℓ, $p_1, \ldots, p_{\delta_i-2}$, p_r may not appear in this left-to-right order along h_i; see Fig. 4a. To guarantee this property, we perform a sequence of stretchings of Γ_{k-1} by repediately applying Lemma 2. First, if p_ℓ

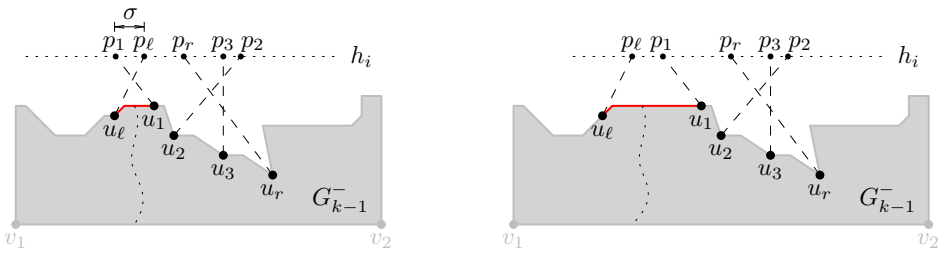


Figure 4 (a) Intersection points $p_\ell, p_1, \ldots, p_{\delta_i-2}, p_r$ appear in a wrong order along h_i. (b) Applying Lemma 2 to make p_ℓ be the leftmost intersection point.

is not the leftmost of these intersection points, let σ be the distance between p_ℓ and the leftmost intersection point. We apply Lemma 2 on any edge between u_ℓ and u_1 along C^-_{k-1} to stretch Γ_{k-1} so that all the vertices in the path of C^-_k from u_1 to v_2 are moved to the right by a quantity σ' slightly larger than σ. This implies that p_ℓ is not moved, while all the other intersection points are moved to the right by a quantity σ', and thus they all lie to the right of p_ℓ in the new drawing; see Fig. 4b. Analogously, we can move p_1 to the left of every other intersection point, except for p_ℓ, by applying Lemma 2 on any edge between u_1 and u_2 along C^-_{k-1}. Repeating this argument allows us to assume that in Γ_{k-1} all the intersection points appear in the correct left-to-right order along h_i.

We now describe the placement of v_i. Let $\beta_1(v_i), \ldots, \beta_{\delta_i-2}(v_i)$ be any set of $\delta_i - 2$ consecutive bottom rays of v_i; since S contains $\Delta - 1$ slopes and $\delta_i \leq \Delta$, v_i has enough bottom rays. If we place v_i above h_i, rays $\beta_1(v_i), \beta_2(v_i), \ldots, \beta_{\delta_i-2}(v_i)$ intersect h_i in this left-to-right order. Let $\rho_1, \ldots, \rho_{\delta_i-2}$ be the corresponding intersection points. The goal is to place v_i so that each ρ_q, with $q = 1, \ldots, \delta_i - 2$, coincides with p_q. To do so, we consider the line λ_1 through p_1 with the same slope as $\beta_1(v_i)$. Note that placing v_i on λ_1 above h_i results in ρ_1 to coincide with p_1. Also note that, while moving v_i upwards along λ_1, the distance $d(\rho_q, \rho_{q+1})$ between any two consecutive points ρ_q and ρ_{q+1}, with $q = 1, \ldots, \delta_i - 3$, increases.

We move v_i upwards along λ_1 so that $d(\rho_q, \rho_{q+1}) > d(p_1, p_{\delta_i-2})$, for each $q = 1, \ldots, \delta_i - 3$. This implies that all points $p_2, \ldots, p_{\delta_i-2}$ lie strictly between ρ_1 and ρ_2. Then, we apply Lemma 2 on any edge between u_1 and u_2 along C^-_{k-1} to stretch Γ_{k-1} so that all the vertices in the path of C^-_k from u_2 to v_2 are moved to the right by a quantity $d(p_2, \rho_2)$. In this way, u_1 is not moved and so p_1 still coincides with ρ_1; also, p_2 is moved to the right to coincide with ρ_2; finally, since $d(\rho_2, \rho_3) > d(p_1, p_{\delta_i-2}) > d(p_2, p_{\delta_i-2})$, all points $p_3, \ldots, p_{\delta_i-2}$ lie strictly between ρ_2 and ρ_3. By repeating this transformation for all points $p_3, \ldots, p_{\delta_i-2}$, if any, we guarantee that each ρ_q, with $q = 1, \ldots, \delta_i - 2$, coincides with p_q. We draw each edge (v_i, u_q), with $q = 1, \ldots, \delta_i - 2$, with a segment along $\tau(u_q)$ and the other one along $\beta_q(v_i)$.

It remains to draw edges (v_i, u_ℓ) and (v_i, u_r), which by Invariant I.3 must have a horizontal segment. After possibly applying Lemma 2 on any edge between u_ℓ and u_1 along C^-_{k-1} to stretch Γ_{k-1}, we can guarantee that $\tau_a(u_\ell)$ crosses the horizontal line through v_i to the left of v_i. Similarly, we can guarantee that $\tau_c(u_r)$ crosses the horizontal line through v_i to the right of v_i by applying Lemma 2 on any edge between u_{δ_i-2} and u_r. We draw (v_i, u_ℓ) with one segment along $\tau_a(u_\ell)$ and one along the horizontal line through v_i, and we draw (v_i, u_r) with one segment along $\tau_c(u_r)$ and one along the horizontal line through v_i. A drawing produced by this algorithm is illustrated in Fig. 3b.

The fact that Γ_k is a 1-bend planar drawing on S satisfying Invariant I.3–I.3 can be shown as for the case in which P_k is a chain. In particular, for Invariants I.3 and I.3, note that vertices $u_1, \ldots, u_{\delta_i-2}$ do not have neighbors in $G \setminus G_k$ and do not belong to C_k^-. Thus, they do not need to have any free top ray incident to the outer face of G_k^- and the edges connecting them to v_i do not need to have a horizontal segment. This concludes our description for the case in which P_k is a singleton, and yields the following theorem (a discussion about the time complexity can be found in [1]).

▶ **Theorem 4.** *For any $\Delta \geq 4$, there exists a equispaced universal set S of $\Delta - 1$ slopes for 1-bend planar drawings of triconnected planar graphs with maximum degree Δ. Also, for any such graph on n vertices, a 1-bend planar drawing on S can be computed in $O(n)$ time.*

4 Biconnected Planar Graphs

In this section we describe how to extend Theorem 4 to biconnected planar graphs, using the SPQR-tree data structure described in Section 2.

The idea is to traverse the SPQR-tree of the input biconnected planar graph G bottom-up and to construct for each visited node a drawing of its pertinent graph (except for its two poles) inside a rectangle, which we call *chip*. Besides being a 1-bend planar drawing on S, this drawing must have an additional property, namely that it is possible to increase its width while changing neither its height nor the slope of any edge-segment. We call this property *horizontal stretchability*. We give a formal definition of this drawing and describe how to compute it for each type of node of the SPQR-tree.

Let $\mathcal{T}$ be the SPQR-tree of G rooted at an arbitrary Q-node ρ. Let μ be a node of $\mathcal{T}$ with poles s_μ and t_μ. Let G_μ be the pertinent graph of μ. Let $\overline{G}_\mu$ be the graph obtained from G_μ as follows. First, remove edge (s_μ, t_μ), if it exists; then, subdivide each edge incident to s_μ (to t_μ) with a dummy vertex, which is a *pin of* s_μ (is a *pin of* t_μ); finally, remove s_μ and t_μ, and their incident edges. Note that, if μ is a Q-node other than the root ρ, then $\overline{G}_\mu$ is the empty graph. We denote by $\delta(s_\mu, \mu)$ and $\delta(t_\mu, \mu)$ the degree of s_μ and t_μ in G_μ, respectively; note that the number of pins of s_μ (of t_μ) is $\delta(s_\mu, \mu) - 1$ (is $\delta(t_\mu, \mu) - 1$), if edge (s_μ, t_μ) exists in G, otherwise it is $\delta(s_\mu, \mu)$ (it is $\delta(t_\mu, \mu)$).

We construct a 1-bend planar drawing of $\overline{G}_\mu$ on S inside an axis-aligned rectangle, called the *chip* of μ and denoted by $C(\mu)$, so that the following properties are satisfied (see Fig. 5a):

(P.1) All the pins of s_μ lie on the left side of $C(\mu)$ and all the pins of t_μ lie on its right side;
(P.2) for each pin, the unique edge incident to it is horizontal; and
(P.3) there exist pins on the bottom-left and on the bottom-right corners of $C(\mu)$.

We call *horizontally-stretchable* (or *stretchable*, for short) a drawing of $\overline{G}_\mu$ satisfying Properties P.4-P.4. Note that a stretchable drawing Γ remains stretchable after any uniform scaling, any translation, and any horizontal or vertical flip, since the horizontal slope is in S and the slopes are equispaced. On the other hand, it is generally not possible to perform any non-uniform scaling of Γ (in particular, a horizontal or a vertical scaling) without altering the slopes of some segments. However, we can simulate a horizontal scaling up of Γ by elongating the horizontal segments incident to all the pins lying on the same vertical side of the chip, thus obtaining a new stretchable drawing inside a new chip with the same height and a larger width. Conversely, a horizontal scaling down cannot always be simulated in this way.

Before giving the details of the algorithm, we describe a subroutine that we will often use to add the poles of a node μ to a stretchable drawing of $\overline{G}_\mu$ and draw the edges incident to them. The proof of the next lemma can be found in [1]; see also Fig. 5b.

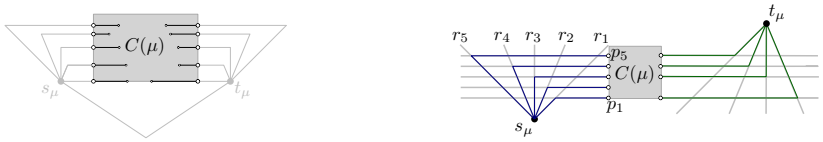


Figure 5 Illustrations (a) of a pin $C(\mu)$ and (b) for Lemma 5.

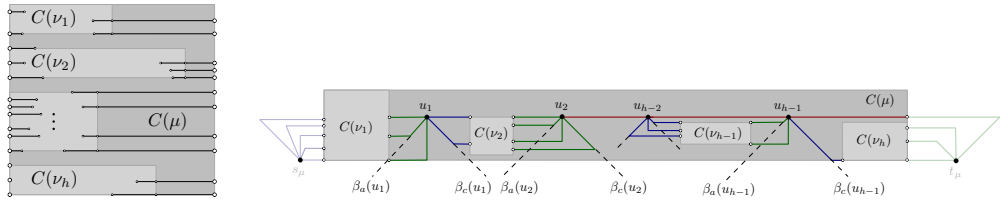


Figure 6 Illustrations for the cases in which μ is: (a) a P-node and (b) an S-node.

▶ **Lemma 5.** *Let $u \in \{s_\mu, t_\mu\}$ be a pole of a node $\mu \in \mathcal{T}$ and let $u_1, \ldots, u_q$ be q neighbors of u in $\overline{G}_\mu$. Consider a stretchable drawing of $\overline{G}_\mu$ inside a chip $C(\mu)$, whose pins $p_1, \ldots, p_q$ correspond to $u_1, \ldots, u_q$. Suppose that there exists a set of q consecutive free rays of u and that the elongation of the edge incident to each pin $p_1, \ldots, p_q$ intersects all these rays. Then, it is possible to draw edges $(u, u_1), \ldots, (u, u_q)$ with two straight-line segments whose slopes are in S, without introducing any crossing between two edges incident to u or between an edge incident to u and an edge of $\overline{G}_\mu$.*

We now describe the algorithm. At each step of the bottom-up traversal of $\mathcal{T}$, we consider a node $\mu \in \mathcal{T}$ with children $\nu_1, \ldots, \nu_h$, and we construct a stretchable drawing of $\overline{G}_\mu$ inside a chip $C(\mu)$ starting from the stretchable drawings of $\overline{G}_{\nu_1}, \ldots, \overline{G}_{\nu_h}$ inside chips $C(\nu_1), \ldots, C(\nu_h)$ that have been already constructed. We distinguish the different cases in which μ is a Q-, a P-, an S-, or an R-node.

Suppose that μ is a Q-node. If μ is not the root ρ of $\mathcal{T}$, we do not do anything, since $\overline{G}_\mu$ is the empty graph; the edge (s_μ, t_μ) of G corresponding to μ will be drawn when visiting either the parent ξ of μ, if ξ is not a P-node, or the parent of ξ.

Otherwise, $\mu = \rho$ and hence it has only one child ν_1. Since $\overline{G}_\mu$ coincides with $\overline{G}_{\nu_1}$, the stretchable drawing of $\overline{G}_{\nu_1}$ is also a stretchable drawing of $\overline{G}_\mu$. Vertices s_μ and t_μ, and their incident edges, will be added at the end of the traversal of $\mathcal{T}$.

Suppose that μ is a P-node; refer to Fig. 6a. We consider a chip $C(\mu)$ for μ whose height is larger than the sum of the heights of chips $C(\nu_1), \ldots, C(\nu_h)$ and whose width is larger than the one of any of $C(\nu_1), \ldots, C(\nu_h)$. Then, we place chips $C(\nu_1), \ldots, C(\nu_h)$ inside $C(\mu)$ so that no two chips overlap, their left sides lie along the left side of $C(\mu)$, and the bottom side of $C(\nu_h)$ lies along the bottom side of $C(\mu)$. Finally, we elongate the edges incident to all the pins on the right side of $C(\nu_1), \ldots, C(\nu_h)$ till reaching the right side of $C(\mu)$. The resulting drawing is stretchable since the drawings of $\overline{G}_{\nu_1}, \ldots, \overline{G}_{\nu_h}$ are. In particular, Property P.4 holds for $C(\mu)$ since it holds for $C(\nu_h)$.

Suppose that μ is an S-node; refer to Fig. 6b. Let $u_1, \ldots, u_{h-1}$ be the internal vertices of the path of virtual edges between s_μ and t_μ obtained by removing the virtual edge (s_μ, t_μ) from the skeleton of μ. We construct a stretchable drawing of $\overline{G}_\mu$ as follows.

First, we place vertices $u_1, \ldots, u_{h-1}$ in this order along a horizontal line l_μ. For $i = 1, \ldots, h-1$, let $\beta_a(u_i)$ and $\beta_c(u_i)$ be the first bottom rays of u_i in anti-clockwise and in clockwise order, respectively. To place each chip $C(\nu_i)$, with $i = 2, \ldots, h-1$, we first flip it vertically, so that it has pins on its top-left and top-right corners, by Property P.4. Then, after possibly scaling it down uniformly, we place it in such a way that its left side is to the right of u_{i-1}, its right side is to the left of u_i, it does not cross $\beta_c(u_{i-1})$ and $\beta_a(u_i)$, and either its top side lies on line l_μ (if edge $(u_{i-1}, u_i) \notin G$; see $C(\nu_2)$ in Fig. 6b), or it lies slightly below it (otherwise; see $C(\nu_{h-1})$ in Fig. 6b).

Then, we place $C(\nu_1)$ and $C(\nu_h)$, after possibly scaling them up uniformly, in such a way that:

- **(i)** Chip $C(\nu_1)$ lies to the left of u_1 and does not cross $\beta_a(u_1)$. Also, if $(s_\mu, u_1) \in G$, then $C(\nu_1)$ lies entirely below l_μ; otherwise, as in Fig. 6b, the topmost pin on its right side has the same y-coordinate as u_1.
- **(ii)** Chip $C(\nu_h)$ lies to the right of u_h and does not cross $\beta_c(u_h)$. Also, if $(u_h, t_\mu) \in G$, as in Fig. 6b, then $C(\nu_h)$ lies entirely below u_h; otherwise, the topmost pin on its left side has the same y-coordinate as u_h.
- **(iii)** The bottom sides of $C(\nu_1)$ and of $C(\nu_h)$ have the same y-coordinate, which is smaller than the one of the bottom side of any other chip $C(\nu_2), \ldots, C(\nu_{h-1})$.

We now draw all the edges incident to each vertex u_i, with $i = 1, \ldots, h-1$. If edge $(u_{i-1}, u_i) \in G$, then it can be drawn as a horizontal segment, by construction. Otherwise, u_i can be connected with a horizontal segment to its neighbor in $\overline{G}_{\nu_i}$ corresponding to the topmost pin on the right side of $C(\nu_i)$. In both cases, one of these edges is attached at a horizontal ray of u_i. Analogously, one of the edges connecting u_i to its neighbors in $\overline{G}_{\nu_{i+1}} \cup \{u_{i+1}\}$ is attached at the other horizontal ray of u_i. Thus, it is possible to draw the remaining $\delta(u_i, \nu_i) + \delta(u_i, \nu_{i+1}) - 2 \leq \Delta - 2$ edges incident to u_i by attaching them at the $\Delta - 2$ bottom rays of u_i, by applying Lemma 5. In fact, since $C(\nu_i)$ and $C(\nu_{i+1})$ lie to the left and to the right of u_i, respectively, and do not cross $\beta_c(u_i)$ and $\beta_a(u_i)$, the elongations of the edges incident to the pins of u_i in $C(\nu_i)$ and in $C(\nu_{i+1})$ corresponding to these edges intersect all the bottom rays of u_i, hence satisfying the preconditions to apply the lemma.

Finally, we construct chip $C(\mu)$ as the smallest rectangle enclosing all the current drawing. Note that the left side of $C(\mu)$ contains the left side of $C(\nu_1)$, while the right side of $C(\mu)$ contains the right side of $C(\nu_h)$. Thus, all the pins of s_μ, possibly except for the one corresponding to edge (s_μ, u_1), lie on the left side of $C(\mu)$. Also, if (s_μ, u_1) exists, we can add the corresponding pin since, by construction, $C(\nu_1)$ lies entirely below u_1. The same discussion applies for the pins of t_μ. This proves that the constructed drawing satisfies properties P.4 and P.4. To see that it also satisfies P.4, note that the bottom side of $C(\mu)$ contains the bottom sides of $C(\nu_1)$ and of $C(\nu_h)$, by construction, which have a pin on both corners, by Property P.4. Thus, the constructed drawing of $\overline{G}_\mu$ is stretchable.

Suppose that μ is an R-node. We compute a stretchable drawing of $\overline{G}_\mu$ as follows. First, we compute a 1-bend planar drawing on S of the whole pertinent G_μ of μ, including its poles s_μ and t_μ; then, we remove the poles of μ and their incident edges, we define chip $C(\mu)$, and we place the pins on its two vertical sides so to satisfy Properties P.4–P.4.

Since the skeleton G_μ^{skel} of μ is triconnected, we use the algorithm described in Section 3 as a main tool for drawing G_μ, with suitable modifications to take into account the fact

that each virtual edge (u, v) of G_μ^{skel} actually corresponds to a whole subgraph, namely the pertinent graph G_ν of the child ν of μ with poles $s_\nu = u$ and $t_\nu = v$. Thus, when the virtual edge (u, v) is considered, we have to add the stretchable drawing of $\overline{G}_\nu$ inside a chip $C(\nu)$; this enforces additional requirements for our drawing algorithm.

The first obvious requirement is that (u, v) will occupy $\delta(u, \nu)$ consecutive rays of u and $\delta(v, \nu)$ consecutive rays of v, and not just a single ray for each of them, as in the triconnected case. However, reserving the correct amount of rays of u and v is not always sufficient to add $C(\nu)$ and to draw the edges between u, v, and vertices in $\overline{G}_\nu$. In fact, we need to ensure that there exists a placement for $C(\nu)$ such that the elongations of the edges incident to its pins intersect all the reserved rays of the poles u and v of ν, hence satisfying the preconditions to apply Lemma 5. In a high-level description, for the virtual edges that would be drawn with a horizontal segment in the triconnected case (all the edges of a chain, and the first and last edges of a singleton), this can be done by using a construction similar to the one of the case in which μ is an S-node. For the edges that do not have any horizontal segment (the internal edges of a singleton), instead, we need a more complicated construction.

The algorithm is again based on considering the vertices of $H = G_\mu$ according to a canonical order $\Pi = (P_0, \ldots, P_m)$ of H, in which $v_1 = s_\mu$ and $v_2 = t_\mu$, and on constructing a 1-bend planar drawing of H_k^- on S satisfying a modified version of Invariants I.3–I.3.

(M.1) No part of the drawing lies below vertices v_1 and v_2, which have the same y-coordinate.

(M.2) For every virtual edge (w, z) on C_k^-, if (w, z) belongs to H then it has a horizontal segment; also, the edge-segments corresponding to edges incident to the pins of the chip of the child of μ corresponding to (w, z) are horizontal.

(M.3) Each vertex v on C_k^- has at least as many free top rays incident to the outer face of H_k^- as the number of its neighbors in H that have not been drawn yet.

We note that Invariant M.4 is identical to Invariant I.3, while Invariant M.4 is the natural extension of Invariant I.3 to take into account our previous observation. Finally, Invariant M.4 corresponds to Invariant I.3, as it ensures that we can still apply Lemma 2 and Lemma 3.

At the first step, we draw $P_0 = \{v_1, v_2\}$ and $P_1 = \{v_3, \ldots, v_j\}$. Consider the path of virtual edges $(v_1, v_3), (v_3, v_4), \ldots, (v_j, v_2)$. Let $\nu_{1,3}, \nu_{3,4}, \ldots, \nu_{j,2}$ be the corresponding children of μ, and let $C(\nu_{1,3}), C(\nu_{3,4}), \ldots, C(\nu_{j,2})$ be their chips. We consider this path as the skeleton of an S-node with poles v_1 and v_2, and we we apply the same algorithm as in the case in which μ is an S-node to draw the subgraph composed of $v_3, \ldots, v_j$ and of chips $C(\nu_{1,3}), C(\nu_{3,4}), \ldots, C(\nu_{j,2})$ inside a larger chip, denoted by $C(v_1, v_2)$. Note that, by construction, $C(v_1, v_2)$ has pins on its bottom-left and on its bottom-right corners. We then place v_1 and v_2 with the same y-coordinate as the bottom side of $C(v_1, v_2)$, with v_1 to the left and v_2 to the right of $C(v_1, v_2)$. We draw one of the edges incident to v_1 horizontal, and the remaining $\delta(v_1, \nu_{1,3}) - 1$ by applying Lemma 5, and the same for v_2. Invariants M.4 and M.4 are satisfied by construction. For Invariant M.4, note that $v_3, \ldots, v_j$ have all their $\Delta - 2$ top rays free, by construction, and at least two of their neighbors have already been drawn. Also, v_1 and v_2 have consumed only $\delta(v_1, \nu_{1,3}) - 1$ and $\delta(v_2, \nu_{j,2}) - 1$ top rays, respectively. Since edge (v_1, v_2) does not belong to H_k^- (but belongs to H), v_1 and v_2 satisfy Invariant M.4.

We now describe how to add path P_k, for some $k > 1$, to the current drawing Γ_{k-1} in the two cases in which P_k is a chain or a singleton.

Suppose that P_k is a chain $\{v_i, v_{i+1}, \ldots, v_j\}$; let u_ℓ and u_r be the neighbors of v_i and v_j in C_k^-. Let $\nu_\ell, \nu_i, \ldots, \nu_{j-1}, \nu_r$ be the children of μ corresponding to virtual edges $(u_\ell, v_i), (v_i, v_{i+1}), \ldots, (v_{j-1}, v_j), (v_j, v_r)$, and $C(\nu_\ell), C(\nu_i), \ldots, C(\nu_{j-1}), C(\nu_r)$ be their chips.

We define rays $\tau_a(u_\ell)$ and $\tau_c(u_r)$, and the horizontal segment h between them, as in the triconnected case. Due to Lemma 3, we can assume that $\tau_a(u_\ell)$ and the $\delta(u_\ell, \nu_\ell) - 1$ top

rays of u_ℓ following it in anti-clockwise order do not cross any edge of Γ_{k-1}, and the same for $\tau_c(u_r)$ and the $\delta(u_r, \nu_r) - 1$ top rays of u_r following it in clockwise order. Note that, by Invariant M.4, all these rays are free. As in the step in which we considered P_0 and P_1 of Π, we use the algorithm for the case in which μ is an S-node to construct a drawing of the subgraph composed of $v_i, \dots, v_j$ and of chips $C(\nu_\ell), C(\nu_i), \dots, C(\nu_{j-1}), C(\nu_r)$ inside a larger chip $C(u_\ell, u_r)$, which has pins on its bottom-left and on its bottom-right corners. We then place $C(u_\ell, u_r)$ so that its bottom side lies on h and it does not cross $\tau_a(u_\ell)$ and $\tau_c(u_r)$, after possibly scaling it down uniformly. Finally, we draw the $\delta(u_\ell, \nu_\ell)$ edges between u_ℓ and its neighbors in $\overline{G}_{\nu_\ell} \cup \{v_i\}$, and the $\delta(u_r, \nu_r)$ edges between u_r and its neighbors in $\overline{G}_{\nu_r} \cup \{v_j\}$, by applying Lemma 5, whose preconditions are satisfied. The fact that the constructed drawing satisfies the three invariants can be proved as in the previous case.

Suppose that P_k is a singleton $\{v_i\}$ of degree $\delta_i \leq \Delta$ in H_k^-. As in the triconnected case, we shall assume that $\delta_i \geq 3$. Let $u_\ell, u_1, \dots, u_{\delta_i-2}, u_r$ be the neighbors of v_i as they appear along C_{k-1}^-, let $\nu_\ell, \nu_1, \dots, \nu_{\delta_i-2}, \nu_r$ be the children of μ corresponding to the virtual edges connecting v_i with these vertices, and let $C(\nu_\ell), C(\nu_1), \dots, C(\nu_{\delta_i-2}), C(\nu_r)$ be their chips.

For each $q = 1, \dots, \delta_i - 2$, we select any set T_q of consecutive $\delta(u_q, \nu_q)$ free top rays of u_q incident to the outer face and a set B_q of consecutive $\delta(v_i, \nu_q)$ bottom rays of v_i; see Fig. 7b. Sets $B_1, \dots, B_{\delta_i-2}$ are selected so that the rays in B_q precede those in B_{q+1} in anti-clockwise order. Since $\delta(v_i, \nu_\ell) + \delta(v_i, \nu_r) \geq 2$, we have that v_i has enough bottom rays for sets $B_1, \dots, B_{\delta_i-2}$. Sets T_ℓ and T_r contain the first $\delta(u_\ell, \nu_\ell)$ free top rays of u_ℓ in anti-clockwise order and of the first $\delta(u_r, \nu_r)$ free top rays of u_r in clockwise order, respectively.

We then select a horizontal line h_i lying above every vertex in Γ_{k-1}. As in the algorithm described in Section 3, after possibly applying $O(\Delta)$ times Lemma 2, we can assume that all the rays in sets $T_\ell, T_1, \dots, T_{\delta_i-2}, T_r$ intersect h_i in the correct order. Namely, when moving along h_i from left to right, we encounter all the intersections with the rays in T_ℓ, then all those with the rays in T_1, and so on. This property is already guaranteed for the rays in $B_1, \dots, B_{\delta_i-2}$. This defines two total left-to-right orders $\mathcal{O}_T$ and $\mathcal{O}_B$ of the intersection points of $T_\ell, T_1, \dots, T_{\delta_i-2}, T_r$ and of $B_1, \dots, B_{\delta_i-2}$ along h_i, respectively. For simplicity, we extend these orders to the rays in $T_\ell, T_1, \dots, T_{\delta_i-2}, T_r$ and in $B_1, \dots, B_{\delta_i-2}$, respectively.

Our goal is to merge the two sets of intersection points, while respecting $\mathcal{O}_T$ and $\mathcal{O}_B$, in such a way that the following condition holds for each $q = 1, \dots, \delta_i - 2$. If edge (v_i, u_q) belongs to H, then the first intersection point of T_q in $\mathcal{O}_T$ coincides with the first intersection point of B_q in $\mathcal{O}_B$, and the second intersection point of T_q in $\mathcal{O}_T$ is to the right of the last intersection point of B_q in $\mathcal{O}_B$; see T_1 and B_1 in Fig. 7b. Otherwise, $(v_i, u_q) \notin H$ and the first intersection point of T_q in $\mathcal{O}_T$ is to the right of the last intersection point of B_q in $\mathcal{O}_B$; see T_3 and B_3 in Fig. 7b. In both cases, all the intersection points of T_q and B_q are to the left of all the intersection points of T_{q+1} and B_{q+1}.

To obtain this goal, we perform a procedure analogous to the one described in Section 3 to make points $p_1, \dots, p_{\delta_i-2}$ coincide with points $\rho_1, \dots, \rho_{\delta_i-2}$. Namely, we consider a line λ_1, whose slope is the one of the first ray in B_1, that starts at the first intersection point of T_1 in $\mathcal{O}_T$, if edge (v_1, u_1) belongs to H, or at any point between the last intersection point of T_1 and the first intersection point of T_2 in $\mathcal{O}_T$, otherwise. Then, we place v_i along λ_1, far enough from h_i so that the distance between any two consecutive intersection points in $\mathcal{O}_B$ is larger than the distance between the first and the last intersection points in $\mathcal{O}_T$; see Fig. 7a. Finally, we apply Lemma 2 at most $\delta_i - 3$ times to move the intersection points of sets $T_2, \dots, T_{\delta_i-2}$, one by one, in the their correct positions; see Fig. 7b.

Once the required ordering of intersection points along h_i has been obtained, we consider another horizontal line h_i' lying above h_i and close enough to it so that its intersections with

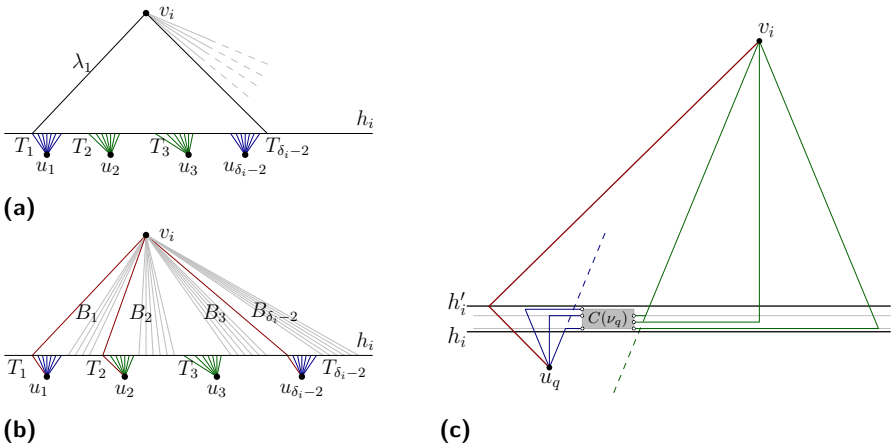


Figure 7 Ilustrations for placing singleton v_i in the case of an R-node.

the rays in $T_\ell, T_1, \ldots, T_{\delta_i-2}, T_r$ and $B_1, \ldots, B_{\delta_i-2}$ appear along it in the same order as along h_i. We place each chip $C(\nu_q)$, with $q = 1, \ldots, \delta_i - 2$, after possibly scaling it down uniformly, in the interior of the region delimited by these two lines, by the last ray in T_q, and by a ray in B_q (either the second or the first, depending on whether $(v_i, u_q) \in H$ or not); see Fig. 7c.

We draw the edges incident to v_i and u_q, for each $q = 1, \ldots, \delta_i - 2$, as follows. If (v_i, u_q) belongs to H, we draw it with one segment along the first ray in T_q and one along the first ray in B_q (red edge in Fig. 7c). For the other edges we apply Lemma 5 twice, whose preconditions are satisfied due to the placement of $C(\nu_q)$ (blue and green edges in Fig. 7c).

We conclude by drawing the edges connecting v_i, u_ℓ, and vertices in $\overline{G}_{\nu_\ell}$; the edges connecting v_i, u_r and vertices in $\overline{G}_{\nu_r}$ are drawn symmetrically. First, after possibly applying Lemma 2, we assume that the last ray of T_ℓ intersects the horizontal line through v_i to the left of v_i, at a point p_i. After possibly scaling $C(\nu_\ell)$ down uniformly, we place it so that its left side is to the right of p_i, its right side is to the left of v_i, it does not cross the first top ray of v_i in clockwise order, and its bottom side is horizontal and lies either above the horizontal line through v_i, if $(u_\ell, v_i) \in H$, or along it, otherwise. Then, we draw (u_ℓ, v_i), if it belongs to H, with one segment along the last ray of T_ℓ and the other one along the horizontal line through v_i. Otherwise, $(u_\ell, v_i) \notin H$ and we can draw one of the edges incident to v_i with a horizontal segment. We finally apply Lemma 5 twice, to draw the edges from u_ℓ to its neighbors in $\overline{G}_{\nu_\ell}$, and from v_i to its other neighbors in $\overline{G}_{\nu_\ell}$. The fact that the constructed drawing satisfies Invariants M.4–M.4 can be proved as in the triconnected case.

Once the last path P_m of Π has been added, we have a drawing Γ_μ of $H = G_\mu$ satisfying Invariants M.4–M.4. We construct chip $C(\mu)$ as the smallest axis-aligned rectangle enclosing Γ_μ. By Invariant M.4, vertices v_1 and v_2 lie on the bottom side of $C(\mu)$. Also, by Invariant M.4, all the edges incident to v_1 or to v_2 have a horizontal segment. Thus, it is possible to obtain a drawing of $\overline{G}_\mu$ inside $C(\mu)$ by removing v_1 and v_2 (and their incident edges) from Γ_μ, by elongating the horizontal segments incident to them till reaching the vertical sides of $C(\mu)$, and by placing pins at their ends. The fact that this drawing satisfies Properties P.4–P.4 follows from the observation that v_1 and v_2 were on the bottom side of $C(\mu)$. This concludes the case in which μ is an R-node.

Once we have visited the root ρ of $\mathcal{T}$, we have a stretchable drawing of $\overline{G}_\rho$ inside a chip $C(\rho)$, which we extend to a drawing of G as follows. Refer to Fig. 5a. We place s_ρ and t_ρ at the same y-coordinate as the bottom side of $C(\rho)$, one to its left and one to its right, so

that $C(\rho)$ does not cross any of the rays of s_ρ and of t_ρ. Then, we draw edge (s_ρ, t_ρ) with one segment along the first bottom ray in clockwise order of s_ρ and the other one along the first bottom ray in anti-clockwise order of t_ρ. Also, we draw the edges connecting s_ρ and t_ρ to the vertices corresponding to the lowest pins on the two vertical sides of $C(\rho)$ as horizontal segments. Finally, we draw all the remaining edges incident to s_ρ and t_ρ by applying Lemma 5 twice. The following theorem summarizes the discussion in this section (a discussion about the time complexity can be found in [1]).

▶ **Theorem 6.** *For any $\Delta \geq 4$, there exists a equispaced universal set S of $\Delta - 1$ slopes for 1-bend planar drawings of biconnected planar graphs with maximum degree Δ. Also, for any such graph on n vertices, a 1-bend planar drawing on S can be computed in $O(n)$ time.*

5 Conclusions and Open Problems

In this paper, we improved the upper bound on the 1-bend planar slope number from $\frac{3}{2}(\Delta-1)$ to $\Delta - 1$, for $\Delta \geq 4$. We mention two side-results of our work. Since the angular resolution of our drawings is at least $\frac{\pi}{\Delta-1}$, at the cost of increased drawing area our main result also improves the best-known upper bound of $\frac{\pi}{4\Delta}$ on the angular resolution of 1-bend poly-line planar drawings by Duncan and Kobourov [17]. For $\Delta = 4$, it also guarantees that planar graphs with maximum degree 4 admit 1-bend planar drawings on a set of slopes $\{0, \frac{\pi}{3}, \frac{2\pi}{3}\}$, while previously it was known that such graphs can be embedded with one bend per edge on a set of slopes $\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$ [3] and with two bends per edge on a set of slopes $\{0, \pi\}$ [5].

Our work raises several open problems.

- **(i)** Reduce the gap between the $\frac{3}{4}(\Delta - 1)$ lower bound and the $\Delta - 1$ upper bound.
- **(ii)** Our algorithm may produce drawings with super-polynomial area. Is this unavoidable for 1-bend planar drawings with few slopes and good angular resolution?
- **(iii)** Study the straight-line case (e.g., when $\Delta = 4$). Note that stretching might be difficult in this setting.
- **(iv)** We proved that a set of $\Delta - 1$ equispaced slopes is universal for 1-bend planar drawings. Is *every* set of $\Delta - 1$ slopes universal? Note that for $\Delta \leq 4$ a positive answer descends from our work and from a result by Dujmovic et al. [15], who proved that any planar graph that can be drawn on a particular set of three slopes can also be drawn on any set of three slopes. If the answer to this question is negative for $\Delta > 4$, what is the minimum value $s(\Delta)$ such that every set of $s(\Delta)$ slopes is universal?

Acknowledgements. This work started at the 19^{th} *Korean Workshop on Computational Geometry*. We wish to thank the organizers and the participants for creating a pleasant and stimulating atmosphere and in particular Fabian Lipp and Boris Klemz for useful discussions.

References

1 Patrizio Angelini, Michael A. Bekos, Giuseppe Liotta, and Fabrizio Montecchiani. Universal slope sets for 1-bend planar drawings. *CoRR*, 1703.04283, 2017.

2 János Barát, Jirí Matoušek, and David R. Wood. Bounded-degree graphs have arbitrarily large geometric thickness. *Electr. J. Comb.*, 13(1), 2006.

3 Michael A. Bekos, Martin Gronemann, Michael Kaufmann, and Robert Krug. Planar octilinear drawings with one bend per edge. *J. Graph Algorithms Appl.*, 19(2):657–680, 2015. doi:10.7155/jgaa.00369.

4 Michael A. Bekos, Michael Kaufmann, and Robert Krug. On the total number of bends for planar octilinear drawings. In *LATIN*, volume 9644 of *LNCS*, pages 152–163. Springer, 2016. doi:10.1007/978-3-662-49529-2_12.

5 Therese C. Biedl and Goos Kant. A better heuristic for orthogonal graph drawings. *Comput. Geom.*, 9(3):159–180, 1998. doi:10.1016/S0925-7721(97)00026-6.

6 Thomas Bläsius, Marcus Krug, Ignaz Rutter, and Dorothea Wagner. Orthogonal graph drawing with flexibility constraints. *Algorithmica*, 68(4):859–885, 2014. doi:10.1007/s00453-012-9705-8.

7 Thomas Bläsius, Sebastian Lehmann, and Ignaz Rutter. Orthogonal graph drawing with inflexible edges. *Comput. Geom.*, 55:26–40, 2016.

8 Hans L. Bodlaender and Gerard Tel. A note on rectilinearity and angular resolution. *J. Graph Algorithms Appl.*, 8:89–94, 2004.

9 Nicolas Bonichon, Bertrand Le Saëc, and Mohamed Mosbah. Optimal area algorithm for planar polyline drawings. In *WG*, volume 2573 of *LNCS*, pages 35–46. Springer, 2002.

10 Hubert de Fraysseix, Patrice Ossona de Mendez, and Pierre Rosenstiehl. On triangle contact graphs. *Combinatorics, Probability & Computing*, 3:233–246, 1994. doi:10.1017/S0963548300001139.

11 Hubert de Fraysseix, János Pach, and Richard Pollack. How to draw a planar graph on a grid. *Combinatorica*, 10(1):41–51, 1990. doi:10.1007/BF02122694.

12 Giuseppe Di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice-Hall, 1999.

13 Emilio Di Giacomo, Giuseppe Liotta, and Fabrizio Montecchiani. The planar slope number of subcubic graphs. In *LATIN*, volume 8392 of *LNCS*, pages 132–143. Springer, 2014. doi:10.1007/978-3-642-54423-1_12.

14 Emilio Di Giacomo, Giuseppe Liotta, and Fabrizio Montecchiani. Drawing outer 1-planar graphs with few slopes. *J. Graph Algorithms Appl.*, 19(2):707–741, 2015. doi:10.7155/jgaa.00376.

15 Vida Dujmović, David Eppstein, Matthew Suderman, and David R. Wood. Drawings of planar graphs with few slopes and segments. *Comput. Geom.*, 38(3):194–212, 2007.

16 Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Drawing trees with perfect angular resolution and polynomial area. *Discrete & Computational Geometry*, 49(2):157–182, 2013.

17 Christian A. Duncan and Stephen G. Kobourov. Polar coordinate drawing of planar graphs with good angular resolution. *J. Graph Algorithms Appl.*, 7(4):311–333, 2003.

18 Stephane Durocher and Debajyoti Mondal. Trade-offs in planar polyline drawings. In *GD*, volume 8871 of *LNCS*, pages 306–318. Springer, 2014.

19 Michael Formann, Torben Hagerup, James Haralambides, Michael Kaufmann, Frank Thomson Leighton, Antonios Symvonis, Emo Welzl, and Gerhard J. Woeginger. Drawing graphs in the plane with high resolution. *SIAM J. Comput.*, 22(5):1035–1052, 1993. doi:10.1137/0222063.

20 Ashim Garg and Roberto Tamassia. Planar drawings and angular resolution: Algorithms and bounds (extended abstract). In *ESA*, volume 855 of *LNCS*, pages 12–23. Springer, 1994.

21 Ashim Garg and Roberto Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.*, 31(2):601–625, 2001. doi:10.1137/S0097539794277123.

22 Carsten Gutwenger and Petra Mutzel. Planar polyline drawings with good angular resolution. In *GD*, volume 1547 of *LNCS*, pages 167–182. Springer, 1998.

23 Carsten Gutwenger and Petra Mutzel. A linear time implementation of SPQR-trees. In *GD*, volume 1984 of *LNCS*, pages 77–90. Springer, 2000. doi:10.1007/3-540-44541-2_8.

24 Udo Hoffmann. On the complexity of the planar slope number problem. *J. Graph Algorithms Appl.*, 21(2):183–193, 2017.

25 Vít Jelínek, Eva Jelínková, Jan Kratochvíl, Bernard Lidický, Marek Tesar, and Tomás Vyskocil. The planar slope number of planar partial 3-trees of bounded degree. *Graphs and Comb.*, 29(4):981–1005, 2013. doi:10.1007/s00373-012-1157-z.

26 Michael Jünger and Petra Mutzel, editors. *Graph Drawing Software*. Springer, 2004.

27 Goos Kant. Drawing planar graphs using the lmc-ordering (extended abstract). In *FOCS*, pages 101–110. IEEE Computer Society, 1992. doi:10.1109/SFCS.1992.267814.

28 Goos Kant. Hexagonal grid drawings. In *WG*, volume 657 of *LNCS*, pages 263–276. Springer, 1992. doi:10.1007/3-540-56402-0_53.

29 Goos Kant. Drawing planar graphs using the canonical ordering. *Algorithmica*, 16(1):4–32, 1996. doi:10.1007/BF02086606.

30 Balázs Keszegh, János Pach, and Dömötör Pálvölgyi. Drawing planar graphs of bounded degree with few slopes. *SIAM J. Discrete Math.*, 27(2):1171–1183, 2013. doi:10.1137/100815001.

31 Kolja Knauer and Bartosz Walczak. Graph drawings with one bend and few slopes. In *LATIN*, volume 9644 of *LNCS*, pages 549–561. Springer, 2016. doi:10.1007/978-3-662-49529-2_41.

32 Kolja B. Knauer, Piotr Micek, and Bartosz Walczak. Outerplanar graph drawings with few slopes. *Comput. Geom.*, 47(5):614–624, 2014. doi:10.1016/j.comgeo.2014.01.003.

33 William Lenhart, Giuseppe Liotta, Debajyoti Mondal, and Rahnuma Islam Nishat. Planar and plane slope number of partial 2-trees. In *GD*, volume 8242 of *LNCS*, pages 412–423. Springer, 2013. doi:10.1007/978-3-319-03841-4_36.

34 Yanpei Liu, Aurora Morgana, and Bruno Simeone. A linear algorithm for 2-bend embeddings of planar graphs in the two-dimensional grid. *Discrete Applied Mathematics*, 81(1-3):69–91, 1998. doi:10.1016/S0166-218X(97)00076-0.

35 Padmini Mukkamala and Dömötör Pálvölgyi. Drawing cubic graphs with the four basic slopes. In *GD*, volume 7034 of *LNCS*, pages 254–265. Springer, 2011. doi:10.1007/978-3-642-25878-7_25.

36 Martin Nöllenburg. Automated drawings of metro maps. Technical Report 2005-25, Fakultät für Informatik, Universität Karlsruhe, 2005.

37 Martin Nöllenburg and Alexander Wolff. Drawing and labeling high-quality metro maps by mixed-integer programming. *IEEE Trans. Vis. Comput. Graph.*, 17(5):626–641, 2011. doi:10.1109/TVCG.2010.81.

38 Jonathan M. Stott, Peter Rodgers, Juan Carlos Martinez-Ovando, and Stephen G. Walker. Automatic metro map layout using multicriteria optimization. *IEEE Trans. Vis. Comput. Graph.*, 17(1):101–114, 2011. doi:10.1109/TVCG.2010.24.

39 Roberto 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.

40 Roberto Tamassia, editor. *Handbook on Graph Drawing and Visualization*. CRC Press, 2013.

41 Greg A. Wade and Jiang-Hsing Chu. Drawability of complete graphs using a minimal slope set. *Comput. J.*, 37(2):139–142, 1994. doi:10.1093/comjnl/37.2.139.