Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 29, no. 3, pp. 111–142 (2025)
DOI: 10.7155/jgaa.v29i3.3006

Monotone Arc Diagrams with few Biarcs

Steven Chaplick[1] *Henry Förster*[2,3,4] *Michael Hoffmann*[5]
Michael Kaufmann[2]

[1]Maastricht University, The Netherlands
[2]Universität Tübingen, Germany
[3]School of Computation, Information and Technology, Technical University of Munich, Heilbronn, Germany
[4]John Cabot University, Rome, Italy
[5]Department of Computer Science, ETH Zürich, Switzerland

Submitted: Dec. 2024	Accepted: June 2026	Published: June 2026
Article type: Regular paper	Communicated by: S. Felsner, K. Klein	

Abstract. We show that every planar graph has a *monotone arc diagram*, also known as a *monotone topological* 2*-page book embedding*, where at most $(4n-10)/5$ (of potentially $3n-6$) edges cross the spine, and every edge crosses the spine at most once; such an edge is called a *biarc*. We can also guarantee that all edges that cross the spine cross it in the same direction (e.g., from bottom to top). For planar 3-trees we further improve the bound to $(3n-9)/4$, and for so-called Kleetopes we obtain a bound of at most $(n-8)/3$ biarcs. The bound for Kleetopes is tight, even if the drawing is not required to be monotone. A *Kleetope* is a plane triangulation that is derived from another plane triangulation T by inserting a new vertex v_f into each face f of T and then connecting v_f to the three vertices of f.

1 Introduction

Arc diagrams (Figure 1) are drawings of graphs that represent vertices as points on a horizontal line, called *spine*, and edges as *arcs*, consisting of a sequence of halfcircles centered on the spine

Special issue on Selected papers from the Thirty-second International Symposium on Graph Drawing and Network Visualization, GD 2024

Preliminary versions of this work were presented at the 36th European Workshop on Computational Geometry (EuroCG 2020) and at the 32nd International Symposium on Graph Drawing and Network Visualization (GD 2024) [4]. Steven Chaplick is partially supported by DFG grant WO 758/11-1. Henry Förster and Michael Kaufmann have been supported by DFG grant KA 812/18-2. Michael Hoffmann is supported by the Swiss National Science Foundation within the collaborative DACH project *Arrangements and Drawings* as SNSF Project 200021E-171681.

E-mail addresses: s.chaplick@maastrichtuniversity.nl (Steven Chaplick) henry.foerster@tum.de (Henry Förster) hoffmann@inf.ethz.ch (Michael Hoffmann) michael.kaufmann@uni-tuebingen.de (Michael Kaufmann)

such that the halfcircles in the sequence alternate between above and below the spine. A *proper arc* consists of one halfcircle. In *proper arc diagrams* all arcs are proper (see Figure 1a). In *plane* arc diagrams no two edges cross. Note that proper plane arc diagrams are also known as 2-*page book embeddings*. Bernhard and Kainen [1] characterized the graphs that admit proper plane arc diagrams: subhamiltonian planar graphs, i.e., subgraphs of planar graphs with a Hamiltonian cycle. In particular, non-Hamiltonian maximal planar graphs do not admit proper plane arc diagrams.

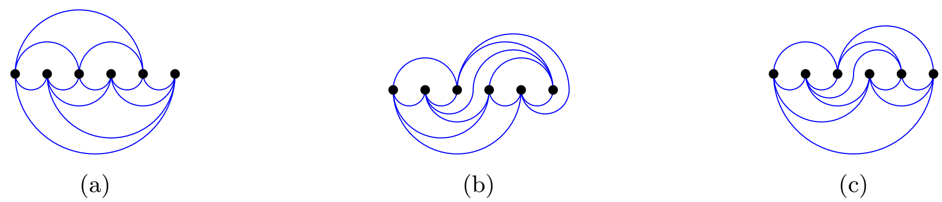


Figure 1: Arc diagrams of the octahedron: (a) proper, (b) general, and (c) monotone.

To represent all planar graphs as a plane arc diagram, it suffices to allow each edge to cross the spine once [14, 15]. The resulting arcs composed of two halfcircles are called *biarcs* (see Figure 1b), and a *biarc diagram* is an arc diagram where every arc is either a proper arc or a biarc. Additionally, all edges can be drawn as *monotone* curves w.r.t. the spine [8, 12]; such a drawing is called a *monotone arc diagram* (see Figure 1c). A monotone biarc is either *up-down* or *down-up*, depending on whether the left halfcircle is drawn above or below the spine, respectively. In the literature, monotone arc diagrams are also known as *monotone topological (*2*-page) book embeddings*. Even though this terminology suggests it, note that a *monotone topological* book embedding is not necessarily a book embedding.

In general, biarcs are needed, but *many* edges can be drawn as proper arcs. Cardinal, Hoffmann, Kusters, Tóth, and Wettstein [2, 3] gave bounds on the required number of biarcs by showing that every planar graph on $n \geq 3$ vertices admits a plane biarc diagram with at most $\lfloor (n-3)/2 \rfloor$ biarcs and how this quantity is related to the diameter of the so-called combinatorial flip graph of triangulations. However, they allow general, not necessarily monotone biarcs. When requiring biarcs to be monotone, Di Giacomo, Didimo, Liotta, and Wismath [8, 12] gave an algorithm to construct a monotone plane biarc diagram that may create close to $2n$ biarcs for an n-vertex planar graph. Cardinal, Hoffmann, Kusters, Tóth, and Wettstein [2, 3] improved this bound to at most $n-4$ biarcs. As a main result, we improve the upper bound on the number of monotone biarcs.

Theorem 1. *Every planar graph on* $n \geq 4$ *vertices admits a monotone plane biarc diagram with at most* $\lfloor (4n-13)/5 \rfloor$ *biarcs, where every biarc is down-up.*

It is an intriguing open question if there is a *monotonicity penalty*, that is, is there a graph G and a plane biarc diagram of G with k biarcs such that every monotone plane biarc diagram of G has strictly more than k biarcs? No such graph is known, even for the stronger condition that all biarcs are monotone of the same type, such as down-up.

For general plane biarc diagrams, in some cases $\lfloor (n-8)/3 \rfloor$ biarcs are required [2, 3]. The (only) graphs for which this lower bound is known to be tight belong to the class of Kleetopes. A *Kleetope* is a plane triangulation[1] that is derived from another plane triangulation T by inserting

[1] A *plane* triangulation is a triangulation associated with a combinatorial embedding. For the scope of this paper, we also consider the outer face to be fixed.

a new vertex v_f into each face f of T and then connecting v_f to the three vertices of f. One might think that Kleetopes are good candidates to exhibit a monotonicity penalty. However, we show that this is not the case, but instead the known lower bound is tight.

Theorem 2. *Every Kleetope on $n \geq 8$ vertices admits a monotone plane biarc diagram with at most $\lfloor (n-8)/3 \rfloor$ biarcs, where every biarc is down-up.*

So, to discover a monotonicity penalty we have to look beyond Kleetopes. We investigate another class of planar graphs: planar 3-trees. A *planar 3-tree* is built by starting from a (combinatorial) triangle. At each step we insert a new vertex v into a (triangular) face f of the graph built so far, and connect v to the three vertices of f. As a third result we give an improved upper bound on the number of monotone biarcs needed for planar 3-trees.

Theorem 3. *Every planar 3-tree on $n \geq 3$ vertices admits a monotone plane biarc diagram with at most $\left\lfloor \frac{3}{4}(n-3) \right\rfloor$ biarcs, where every biarc is down-up.*

Further related work. Giordano, Liotta, Mchedlidze, Symvonis, and Whitesides [13] showed that every upward planar graph admits an *upward topological book embedding* in which all edges are either proper arcs or biarcs. These embeddings are also monotone biarc diagrams that respect the orientations of the edges and use at most one spine crossing per edge. One of their directions for future work is to minimize the number of spine crossings. We believe that our approach for undirected graphs may provide some insights. Everett, Lazard, Liotta, and Wismath [9, 10] used monotone biarc diagrams to construct small universal point sets for 1-bend drawings of planar graphs, heavily using the property that all biarcs have the same *shape* (e.g., all are down-up biarcs). This result has been extended by Löffler and Tóth [16] by restricting the set of possible bend positions. They use the existence of monotone biarc diagrams with at most $n-4$ biarcs to build universal point sets of size $6n-10$ (vertices and bend points) for 1-bend drawings of planar graphs on n vertices. Using our improved upper bound on the number of biarcs in Theorem 1, the number of points required by the algorithm by Löffler and Tóth decreases by about $n/5$ (more precisely, to $\lfloor (29n-43)/5 \rfloor$). On the other hand, we remark that the best known upper bound on the size of universal point sets for 1-bend drawings of planar graphs is $4n-6$ due to Felsner, Schrezenmaier, Schröder and Steiner [11]. While both approaches make use of the same family of point sets [11], in contrast to the approach by Löffler and Tóth [16], their technique does not rely on distance to (sub)hamiltonicity with monotone defects (i.e., the number of monotone biarcs) but on distance to bipartiteness (i.e., they use bends for subdivision vertices that prevent odd cycles).

Outline. We discuss the proof of Theorem 1 in Sections 2 and 3, then in Section 4 the proof of Theorem 2, and finally, in Section 5 the proof of Theorem 3.

2 Overview of our Algorithm

To prove Theorem 1 we describe an algorithm to incrementally construct an arc diagram for a given planar graph $G = (V, E)$ on $n \geq 4$ vertices. Without loss of generality we assume that G is a combinatorial *triangulation*, that is, a maximal planar graph. As every triangulation on $n \geq 4$ vertices is 3-connected, by Whitney's Theorem [17] selecting one facial triangle as the *outer face* fixes the combinatorial embedding. This choice also determines a unique outer face for every biconnected subgraph. We may consider the outer face to be given as part of the input, as an embedding for a given abstract planar graph can be computed in linear time [5, 7]. Throughout this article we consider and work with a fixed combinatorial embedding of G (arbitrarily chosen) and—in a slight abuse of notation—we treat G as a topological *plane* graph. For a biconnected plane graph G denote the outer face (an open subset of $\mathbb{R}^2$) by $F_\circ(G)$ and denote by $C_\circ(G)$ the cycle that bounds $F_\circ(G)$. A plane graph is *internally triangulated* if it is biconnected and every inner face is a triangle. A central tool for our algorithm is the notion of a canonical ordering [6, 7].

Definition 1 (Canonical Ordering)**.** *Consider an internally triangulated plane graph G on the vertices $v_1, \ldots, v_n$, and let $V_k = \{v_j \colon 1 \le j \le k\}$. We say that the sequence $v_1, \ldots, v_n$ forms a* canonical ordering *for G if the following conditions hold for every $i \in \{3, \ldots, n\}$:*

(C1) the induced subgraph $G_i = G[V_i]$ is internally triangulated;

(C2) the edge v_1v_2 is an edge of $C_\circ(G_i)$; and

(C3) for all j with $i < j \le n$, we have $v_j \in F_\circ(G_i)$.

Every internally triangulated plane graph admits a canonical ordering, for any starting pair v_1, v_2 where v_1v_2 is an edge of $C_\circ(G)$ [6, 7]. Moreover, such an ordering can be computed by iteratively selecting v_i, for $i = n, \ldots, 3$, to be a vertex of $C_\circ(G_i) \setminus \{v_1, v_2\}$ that is not incident to a chord of $C_\circ(G_i)$. This computation can be done in $O(n)$ time [5]. In general, a triangulation may admit many canonical orderings. We will use this freedom to adapt the canonical ordering we work with to our needs. To this end, we compute a canonical ordering for G incrementally, starting with v_1, v_2, v_3, where v_1v_2 is an arbitrary edge of $C_\circ(G)$, and v_3 is the unique vertex of G such that $v_1v_2v_3$ bounds a triangular face of G and v_3 is not a vertex of $C_\circ(G)$. A canonical ordering $v_1, \ldots, v_i$ for G_i, where $3 \le i \le n$, is *extensible* if there exists a sequence $v_{i+1}, \ldots, v_n$ such that $v_1, \ldots, v_n$ is a canonical ordering for G. The next lemma formalizes the freedom mentioned above. Note that it could easily be considered as folklore (e.g., it is somewhat implicit in the initial work on canonical ordering [7]), but we still include an explicit proof of it here.

Lemma 1. *Let $v_1, \ldots, v_i$ be a canonical ordering for G_i. Then $v_1, \ldots, v_i$ is extensible if and only if $V \setminus V_i \subset F_\circ(G_i)$.*

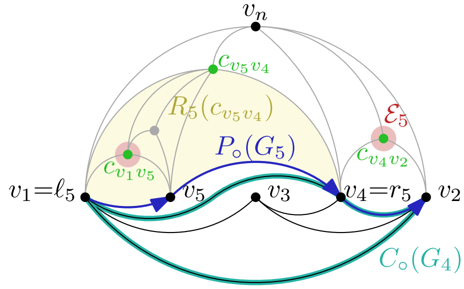


Figure 2: Overview of notation used throughout the paper.

Before we prove Lemma 1, we set up some terminology used throughout the paper; refer also to Figure 2. Consider an extensible canonical ordering $v_1, \ldots, v_i$ for G_i and some vertex $v \in V \setminus V_i$. Let $P_\circ(G_i)$ denote the path $C_\circ(G_i) - v_1v_2$ and direct it from v_1 to v_2. As G_i is an induced subgraph of the plane graph G and $v \in F_\circ(G_i)$ (by extensibility), all neighbors (in G) of v in V_i

are on $P_\circ(G_i)$. We associate a planar region $R_i(v)$ to v as follows. Let $d_i(v) = \deg_{G[V_i \cup \{v\}]}(v)$. If $d_i(v) \le 1$, then $R_i(v) = F_\circ(G_i)$; else, let $R_i(v)$ be the open bounded region bounded by the simple closed curve formed by the part of $P_\circ(G_i)$ between ℓ and r together with the edges ℓv and rv of G, where ℓ and r are the first and last, respectively, neighbor (in G) of v on $P_\circ(G_i)$. We partially order the vertices in $V \setminus V_i$ by defining $v \preceq v'$ if $R_i(v) \subseteq R_i(v')$.

A vertex $v \in V \setminus V_i$ is *eligible* (for G_i) if setting $v_{i+1} = v$ yields an extensible canonical ordering $v_1, \dots, v_{i+1}$ for G_{i+1}. Denote the set of vertices eligible for G_i by $\mathcal{E}_i$. Let $e = uw$ be an arbitrary edge of $P_\circ(G_i)$, for $i < n$. As G is a triangulation, there exists a unique vertex $c_e \in V \setminus V_i$ such that uwc_e bounds a triangular face of G; we say that c_e *covers* e. Given a canonical ordering $v_1, \dots, v_n$, vertex v_i covers exactly the edges of $P_\circ(G_{i-1})$ that are not on $P_\circ(G_i)$. Similarly, we say that v_i *covers* a vertex v of $P_\circ(G_{i-1})$ if v is not part of $P_\circ(G_i)$.

Proof: [of Lemma 1] The $\Rightarrow$ direction is a direct consequence of (C3). For the proof of the converse, let $v_1, \dots, v_i$ be a canonical ordering for G_i, for some $i \in \{3, \dots, n\}$, such that $V \setminus V_i \subset F_\circ(G_i)$. We have to show that $v_1, \dots, v_i$ is extensible.

We use induction on the number $n - i$ of required extension steps from G_i to G_n. In the base case $n - i = 0 \iff i = n$ the statement is trivial, as $v_1, \dots, v_n$ is a canonical ordering for $G = G_n$. For the induction step, consider some $i \in \{n-1, \dots, 3\}$. Let $\mathcal{C}_i = \{c_e : e \in P_\circ(G_i)\}$, and observe that $\mathcal{C}_i \neq \emptyset$. Let v be a minimal element of $\mathcal{C}_i$ (w.r.t. $\preceq$). We claim that $v_{i+1} := v$ is eligible. To see this it suffices to show that $v_1, \dots, v_{i+1}$ is a canonical ordering for G_{i+1} with $V \setminus V_{i+1} \subset F_\circ(G_{i+1})$. Then the claim and the lemma follow by the induction hypothesis.

(C2) trivially holds for all permutations of V that start with v_1, v_2, where $v_1 v_2$ is an edge of $C_\circ(G)$. To prove (C1) and (C3) we use that v is a minimal element of $\mathcal{C}_i$ and our assumption $v \in V \setminus V_i \subset F_\circ(G_i)$. Note that $d_i(v) \ge 2$ (because of the edge $e \in P_\circ(G_i)$ for which $v = c_e$). Therefore, the region $R_i(v)$ is bounded by a cycle of the plane graph G through v and $P_\circ(G_i)$. We claim that $R_i(v) \cap V = \emptyset$.

Suppose to the contrary that there exists a vertex $w \in R_i(v) \cap V$. Then $w \notin V_i$ because G_i is biconnected by (C1) (so $C_\circ(G_i)$ is a cycle), $w \in F_\circ(G_i)$, and G is plane. Thus, while $w \in F_\circ(G_i)$ by assumption, w lies in a bounded face f of G_{i+1}. Then there exists an edge $xy \in P_\circ(G_i)$ on the boundary ∂f of f in G_{i+1}. But f is not a face of G because $w \in f$. So we have $z = c_{xy} \in V \setminus V_{i+1}$ which, as G is plane, implies $R_i(z) \subset R_i(v)$, in contradiction to v being a minimal element of $\mathcal{C}_i$. Therefore, there exists no such vertex w and $R_i(v) \cap V = \emptyset$, as claimed.

As G is plane, G_{i+1} is an induced subgraph, and $R_i(v) \cap V = \emptyset$, all faces of G_{i+1} in $R_i(v)$ are also bounded faces of G. Thus, (C1) holds for $v_1, \dots, v_{i+1}$ because G is internally triangulated. The additional condition $V \setminus V_{i+1} \subset F_\circ(G_{i+1})$ is implied by $F_\circ(G_{i+1}) = F_\circ(G_i) \setminus \mathrm{cl}(R_i(v_{i+1}))$ and $R_i(v) \cap V = \emptyset$, where $\mathrm{cl}(A)$ denotes the closure of A. □

The following observations are direct consequences of the definitions in this section and Lemma 1.

Corollary 1. *A vertex $v \in V \setminus V_i$ is eligible $\iff R_i(v) \cap V = \emptyset \iff R_i(v) \cap \mathcal{E}_i = \emptyset$.*

While computing a canonical ordering $v_1, \dots, v_n$, we also maintain an arc diagram, for short, *diagram* of G_i. This diagram must satisfy certain properties to be considered valid, as detailed below. In some cases we apply induction to handle a whole induced subgraph of G, for instance, within a (separating) triangle, at once. As a result, in certain steps, subgraph G_i may not correspond to a valid arc diagram, but then G_k, for some $k > i$, does.

Every vertex v_i arrives with $1 - \chi$ credits, for some constant $\chi \ge 0$.[2] For these credits we can either create biarcs (at a cost of one credit per biarc), or we place credits on edges of the outer

[2]For Theorem 1 we will set $\chi = 1/5$. But we think it is instructive to keep χ as a general constant in our arguments. For instance, this way it is easier to see the cases in which our analysis is tight.

face of the diagram for later use. The *cost* $\mathrm{cost}(D)$ of a diagram D is the sum of credits on its edges. An edge in the diagram can be one of three different types: *mountain* (proper arc above the spine), *pocket* (proper arc below the spine), or *down-up biarc*. So the diagram is determined by (1) the spine order (left-to-right) of the vertices and crossings along with (2) for every edge, its type and number of credits. The *lower envelope* of a diagram consists of all vertices and edges that are vertically visible from below, that is, there is no other vertex or edge of the diagram vertically below it. Analogously, the *upper envelope* consists of all vertices and edges that are vertically visible from above.

Definition 2 (Valid and Extensible Diagram)**.** *A diagram for $v_1, \dots, v_i$ and $i \in \{3, \dots, n\}$, is called* χ-valid *if it satisfies the following invariants:*

(I1) Every edge is either a proper arc or a down-up biarc. Every edge on the upper envelope is a proper arc.

(I2) Every mountain whose left endpoint is on $C_\circ(G_i) \setminus \{v_2\}$ carries one credit.

(I3) Every biarc carries (that is, is paid for with) one credit.

(I4) Every pocket on $P_\circ(G_i)$ carries χ credits[3].

Moreover, a diagram is called χ-extensible *if it is a χ-valid drawing and also satisfies the following additional invariant:*

(I5) Vertex v_1 is the leftmost and v_2 is the rightmost vertex on the spine. Edge v_1v_2 forms the lower envelope of $C_\circ(G_i)$. The edges of $P_\circ(G_i)$ form the upper envelope.

To prove Theorem 1 it will suffice to prove the following lemma. In its proof, we will make use of (I2) to *transform* mountains to biarcs if we require a new position on the spine for the next vertex; see, e.g., Figure 4. Since by (I3), biarcs require the same number of credits as a mountain, this operation can be done without reassigning credits. In particular, observe that by (I2), we still assign credits to specific mountains that are not anymore on $P_\circ(G_i)$, as these mountains are not yet safe from being transformed in a future iteration. In fact, we will always transform all mountains *simultaneously* that share the same left endpoint on $C_\circ(G_i)$. On the other hand, we will show that *less* than one credit suffices per inserted vertex.

Lemma 2. *Let G be a maximal plane graph on $n \geq 3$ vertices, let $v_1, \dots, v_i$ be an extensible canonical ordering for G_i, for some $3 \leq i < n$, and let D be a χ-extensible arc diagram for G_i. Then, for any $\chi \leq \frac{1}{5}$, the diagram D can be extended to a χ-extensible arc diagram D' for G with* $\mathrm{cost}(D') \leq \mathrm{cost}(D) + (n-i)(1-\chi) + \xi$*, for some $\xi \leq 2\chi$.*

Proof: [of Theorem 1 assuming Lemma 2] By assumption we have $n \geq 4$. Let $C_\circ(G) = v_1v_2v_n$, and let v_3 be the other (than v_n) vertex that forms a triangle with v_1v_2 in G. Then v_1, v_2, v_3 is an extensible canonical ordering for G_3 in G. To obtain a χ-extensible diagram D for G_3, place $v_1v_3v_2$ on the spine in this order from left to right. All three edges are drawn as pockets so that v_1v_2 is below v_1v_3 and v_3v_2. On the latter two edges we put χ credits each. It is easily verified that D is χ-extensible and $\mathrm{cost}(D) = 2\chi$. By Lemma 2 we obtain an χ-extensible diagram D' for G with $\mathrm{cost}(D') \leq 2\chi + (n-3)(1-\chi) + 2\chi = n(1-\chi) + 7\chi - 3$. Setting $\chi = 1/5$ yields $\mathrm{cost}(D') \leq \frac{4}{5}n - \frac{8}{5}$. As v_n is incident to a mountain on the outer face by (I5) which carries a credit by (I2), $\mathrm{cost}(D') - 1$ is an upper bound for the number of biarcs in D' and the theorem follows. $\square$

We can avoid the additive term ξ in Lemma 2 by dropping (I5) for D':

[3] As in the Greek word for pocket money: χαρτζιλίκι.

Lemma 3. *Let G be a maximal plane graph on $n \geq 4$ vertices, let $v_1, \ldots, v_i$ be an extensible canonical ordering for G_i, for $3 \leq i < n$, and let D be a χ-extensible arc diagram for G_i. Then, for any $\chi \leq \frac{1}{5}$, the diagram D can be extended to a χ-valid arc diagram D' for G such that (1) $\mathrm{cost}(D') \leq \mathrm{cost}(D) + (n-i)(1-\chi)$, (2) Vertex v_1 is the leftmost and v_n is the rightmost vertex on the spine. The mountain v_1v_n forms the upper envelope, and the pocket v_1v_2 along with edge v_2v_n forms the lower envelope of D', and (3) v_2v_n is not a pocket.*

3 Algorithmic Proof of Lemmas 2 and 3

We prove both Lemma 2 and Lemma 3 together by induction on n. Moreover, we provide an algorithmic proof. Since the proof is rather involved, we split the discussion into three subsections. First, in Section 3.1, we discuss the base case and a simple default insertion strategy, which unfortunately fails if all eligible vertices are *problematic*. Then, in Section 3.2, we discuss the insertion of problematic vertices in the case, where a problematic vertex can be inserted while only taking very local structural properties into account. Moreover, we discuss properties that must hold, when this is not the case. Finally, in Section 3.3, we exploit the insights of this analysis to discuss how to handle insertion in the remaining cases.

3.1 Default vertex insertion

Base Case. For Lemma 2, the base case $n = 3$ is trivial, with $D' = D$. For Lemma 3, the base case is $n = 4$ and $i = 3$. Note that the only triangulation on four vertices is K_4, and the outer face is $v_1v_2v_4$. We place v_4 as required, to the right of v_2, and draw all edges incident to v_4 as mountains. To establish (I2) it suffices to put one credit on v_1v_4 because v_3 is covered by v_4 and mountains with left endpoint v_2 are excluded in (I2). The edge of D with left endpoint v_3 is covered by v_4; thus, we can take the at least χ credits on it. The invariants (I1), (I3), and (I4) are easily checked to hold, as well as the statements in Lemma 3.

Inductive Step. The remainder of this section describes the inductive step. Unless otherwise specified, the discussion of the inductive step is identical for the proofs of Lemma 2 and Lemma 3. In order to describe a generic step of our algorithm, assume that we already have a χ-extensible arc diagram for G_{i-1}, for $i = 4, \ldots, n$. We have to select an eligible vertex $v_i \in V \setminus V_{i-1}$ and add it using at most $1 - \chi$ credits, obtaining a χ-extensible diagram for G_i. In this section we discuss some cases where a suitable vertex exists that can easily be added, using what we call a *default insertion*. Let v_i be any vertex in $\mathcal{E}_{i-1}$.

We call the sequence of (one or more) edges of $P_\circ(G_{i-1})$ between the leftmost neighbor ℓ_i of v_i and the rightmost neighbor r_i of v_i the *profile* $\mathrm{pr}(v_i)$ of v_i. By (I1) each edge on the profile is a pocket or a mountain, i.e., writing $\smile$ and $\frown$ for pocket and mountain, respectively, each profile can be described by a string over $\{\smile, \frown\}$. For a set A of characters and a positive integer k, let A^*, A^k and A^+ denote the set of all strings (including the empty string), all strings of length exactly k and all strings of length at least one, respectively, formed by characters from A. Let d_i denote the degree of v_i in G_i.

Lemma 4. *If $\mathrm{pr}(v_i) \in \{\smile, \frown\}^* \smile \frown^*$, then we can insert v_i and use ≤ 1 credit to obtain a χ-extensible arc diagram for G_i. At most $1 - \chi$ credits suffice, unless $\mathrm{pr}(v_i) = \frown\smile$.*

Proof: We place v_i into the rightmost pocket $p_\ell p_r$ it covers and draw all edges incident to v_i as proper arcs. The path $p_\ell v_i p_r$ is drawn as two pockets, all other new edges are drawn as mountains;

see Figure 3. As the pocket $p_\ell p_r$ is not on $C_\circ(G_i)$, we can take and spend the χ credits on it. If $d_i = 2$, then we place χ credits on each of the two pockets incident to v_i so as to establish (I4), for a cost of $\chi \leq 1 - \chi$. It is easily checked that the invariants are maintained, which completes the proof in this case.

(a) (b)

Figure 3: Inserting a vertex v_i into a pocket, using $1 - \chi$ credits (Lemma 4).

It remains to consider the case $d_i \geq 3$. Here, we describe how to assign credits to the edges incident to v_i. First, consider edges $v_i u$ with $u \notin \{p_\ell, p_r\}$. Note that the edge $v_i u$ is drawn as a mountain. First, assume that u lies to the right of p_r on $P_\circ(G_{i-1})$. Then v_i covers the edge e_u of $P_\circ(G_i)$ whose right endpoint is u. By the choice of $p_\ell p_r$ (as the rightmost pocket covered by v_i), the edge e_u is a mountain, which by (I2) carries one credit. As e_u is not on $C_\circ(G_i)$, we can transfer this credit to the edge $v_i u$, so as to satisfy (I2) for $v_i u$ since for two such edges uv_i and $u'v_i$ with u and u' to the right of p_r we have that $e_u \neq e_{u'}$. Second, consider the case where the vertex u lies to the left of p_ℓ on $P_\circ(G_{i-1})$. Note that the left endpoint of uv_i is not on $C_\circ(G_i)$, unless $u = \ell_i$. Therefore, it suffices to pay one credit in total and place it on $\ell_i v_i$ to establish (I2) for the resulting diagram.

As no biarc is created by the insertion of v_i, the only remaining possible sources of costs are pockets of $P_\circ(G_i)$ incident to v_i. If $\ell_i = p_\ell$, then there is such a pocket to the left of v_i, and if $r_i = p_r$, then there is such a pocket to the right of v_i on $P_\circ(G_i)$. As $d_i \geq 3$, we face at most one of these pockets. To pay the χ credits for this pocket (if it exists) to establish (I4) we can use the χ credits from the pocket $p_\ell p_r$, which is covered by v_i.

Overall, we pay at most one credit to insert v_i, which proves the first statement of the lemma. To prove the second statement, we need to argue how to save χ credits if $\mathrm{pr}(v_i) \neq \frown\smile$. If there is no pocket incident to v_i in $P_\circ(G_i)$, then we save the χ credits that we accounted for such a pocket, which completes the proof in this case. Thus, it remains to consider the two cases $\ell_i = p_\ell$ and $r_i = p_r$ only.

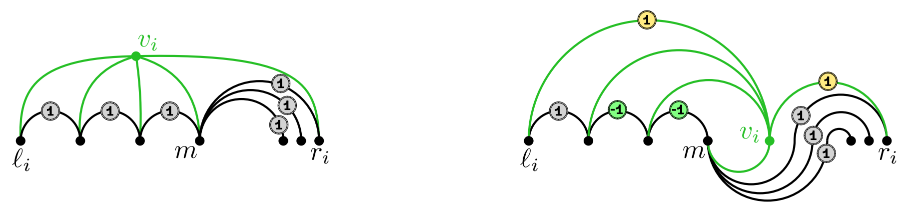
If $\ell_i = p_\ell$, then we save the one credit that we accounted for the mountain $\ell_i v_i$ in the previous analysis as $\ell_i v_i$ is actually a pocket here. So, the overall costs are zero in this case. Otherwise, we have $r_i = p_r$. If $d_i = 3$, then we have $\mathrm{pr}(v_i) = \smile\smile$ as we explicitly exclude the profile $\frown\smile$. So v_i covers two pockets and we can take the χ credits from both, whereas we spend only χ credits on the pocket $v_i p_r$. Thus, the overall costs are $1 - \chi$, which completes the proof in this case. The situation is similar in the remaining case $d_i \geq 4$ because v_i covers at least two edges of $P_\circ(G_i)$ to the left of p_ℓ. In particular, at least one edge e to the left of p_ℓ is covered by v_i such that the left endpoint of e is not on $C_\circ(G_i)$. Either e is a mountain, in which case we can take the one credit it carries, or it is a pocket, and we can take the χ credits it carries. Either way, we gain at least χ credits, for overall costs of at most $1 - \chi$. □

It is more difficult to insert v_i if it covers mountains only, at least if d_i is small. But if the degree of v_i is large, then we can actually gain credits by inserting v_i (see Figure 4).

Lemma 5. *If* $\mathrm{pr}(v_i) \in \frown^+$ *and* $d_i \geq 5$*, then we can insert* v_i *and gain at least* $d_i - 5$ *credits to obtain a* χ*-extensible arc diagram for* G_i*.*

Proof: We push down the rightmost mountain mr_i in $\mathrm{pr}(v_i)$ and place v_i above it. By *push down* we mean that each mountain in G_{i-1} with left endpoint m (there is only one such mountain on $P_\circ(G_{i-1})$, but there may be many more underneath) is transformed into a down-up biarc; see Figure 4. The costs for these biarcs, so as to maintain (I3), are covered by the credits that each mountain whose left endpoint is on $C_\circ(G_i)$ carries according to (I2).

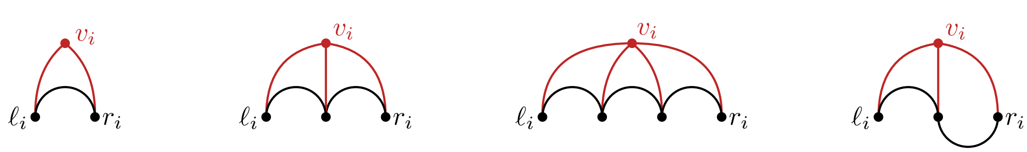
The insertion of v_i creates a new pocket and $d_i - 1$ new mountains. Out of these new edges only two mountains, namely $\ell_i v_i$ and $v_i r_i$, appear on $P_\circ(G_i)$. Therefore, two credits suffice to establish the invariants. As v_i covers $d_i - 1$ mountains, it covers $d_i - 2$ left endpoints of mountains from $P_\circ(G_{i-1})$. One of these mountains is pushed down, consuming the credit it carries. But the at least $d_i - 3$ credits on the remaining $d_i - 3$ mountains are now free to be used. Thus, the overall costs of inserting v_i as described are at most $2 - (d_i - 3) = 5 - d_i$. $\square$



(a) (b)

Figure 4: Inserting a vertex v_i into mountains, using $5 - d_i$ credits (Lemma 5): (a) Before insertion, all edges on the subpath of $P_\circ(G_{i-1})$ between ℓ_i and r_i are mountains, each of which is carrying 1 credit. (b) When inserting according to the proof of Lemma 5, the credits between the leftmost and the rightmost covered mountain can be reassigned. The credit on the leftmost mountain cannot be reassigned so to retain (I2), as ℓ_i is still located on $P_\circ(G_i)$ whereas the credit that already was on mr_i remains on the new biarc to satisfy (I3).

We denote the *type* of vertex v_i by $\mathcal{T}(d_i, \mathrm{pr}(v_i))$. An eligible vertex is called *problematic* if it is of one of the types $\mathcal{T}(2, \frown)$, $\mathcal{T}(3, \frown^2)$, $\mathcal{T}(4, \frown^3)$, or $\mathcal{T}(3, \frown\smile)$; see Figure 5. Note that these are exactly the types that cannot be handled by our default insertion strategies in Lemmas 4 and 5. Our algorithm iteratively inserts eligible vertices using at most $1 - \chi$ credits per vertex using Lemmas 4 and 5, unless all eligible vertices are problematic. This specific situation is discussed in the next section.



(a) $\mathcal{T}(2, \frown)$ (b) $\mathcal{T}(3, \frown^2)$ (c) $\mathcal{T}(4, \frown^3)$ (d) $\mathcal{T}(3, \frown\smile)$

Figure 5: The four types of problematic vertices where default insertion fails.

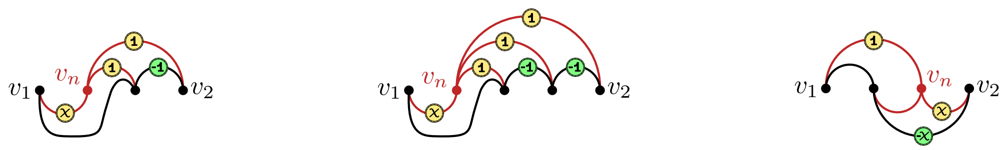
3.2 When default insertion fails

In this section, we discuss how to perform an inductive step in the proofs of Lemma 2 and Lemma 3 in the case where all eligible vertices are problematic, that is, they cannot be handled by our default insertion. Let v be an arbitrary vertex in $\mathcal{E}_{i-1}$, and let ℓ and r denote the leftmost and rightmost neighbor of v on $P_\circ(G_{i-1})$, respectively.

Special case: $v = v_n$**.** A special case arises if $v = v_n$ is the last vertex of the canonical ordering. Then $i = n$, $d_i \geq 3$, and v_n is the only vertex in $\mathcal{E}_{i-1}$. Note that since v is the last vertex to be inserted, this special case actually concludes the proof of Lemma 2 or Lemma 3. In this special scenario, the proofs of Lemma 2 and Lemma 3 differ.

To complete the proof of Lemma 2 in this case, we insert v as shown in Figure 6 and observe that the insertion costs are at most $1 + \chi$ in all cases. The extra costs of at most 2χ compared to the regular costs of $1 - \chi$ per vertex are taken care of by the $+\xi$ term in the costs bound of Lemma 2. These are the only cases where we need $\xi > 0$; that is, we actually prove the following, stronger version of Lemma 2.

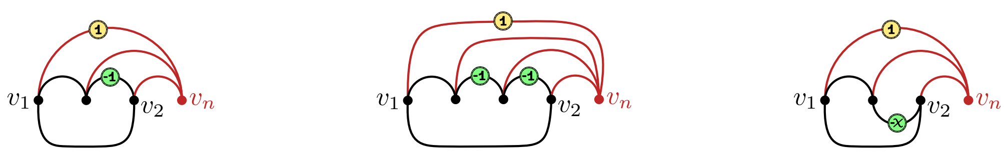
Lemma 6. *Lemma 2 holds with $\xi = 0$ or D can be extended to a χ-extensible arc diagram D' for $G \setminus \{v_n\}$ with $\mathrm{cost}(D') \leq \mathrm{cost}(D) + (n - i - 1)(1 - \chi)$ such that v_n is problematic for D'.*



(a) $\mathcal{T}(3, \frown^2)$ (b) $\mathcal{T}(4, \frown^3)$ (c) $\mathcal{T}(3, \frown\smile)$

Figure 6: Inserting a final problematic vertex v_n for a cost of $\leq 1 + \chi$.

To complete the proof of Lemma 3 for $i = n$, we insert v as shown in Figure 7.



(a) $\mathcal{T}(3, \frown^2)$ (b) $\mathcal{T}(4, \frown^3)$ (c) $\mathcal{T}(3, \frown\smile)$

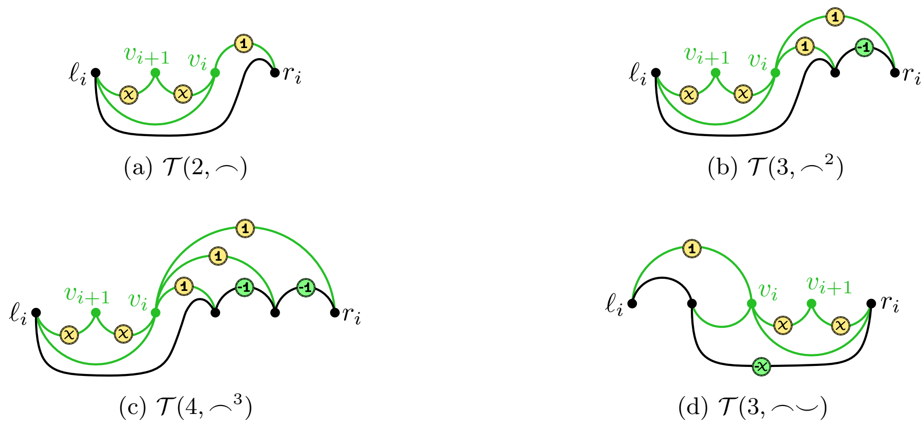
Figure 7: Inserting a final problematic vertex v_n for a cost of $\leq 1 - \chi$.

General Case: $v \neq v_n$**.** Thus, we assume in the following that $i < n$. In contrast to the special case $v = v_n$, the remaining discussion is again identical for both the proofs of Lemma 2 and Lemma 3, except for another clearly labeled special case in Section 3.3.

We begin by picking a *pivot vertex* $\mathrm{p}(v)$ as follows: If v is $\mathcal{T}(3, \frown\smile)$ we set $\mathrm{p}(v) = r$ and say that v has *right pivot type*, in the three remaining cases we set $\mathrm{p}(v) = \ell$ and say that v has *left pivot type*. Let $\mathrm{pc}(v) \in V \setminus V_i$ denote the unique vertex that covers the *pivot edge* $v\mathrm{p}(v)$.

Lemma 7. *Assume there is a vertex $v \in \mathcal{E}_{i-1}$ such that* $\mathrm{pc}(v)$ *has only one neighbor on* $P_\circ(G_{i-1})$. *Then we can set $v_i = v$ and $v_{i+1} = \mathrm{p}(v)$ and spend at most $1 + 2\chi$ credits to obtain a χ-extensible arc diagram for G_{i+1}.*

Proof: The resulting diagram is shown in Figure 8. The costs to establish are $1+\chi$ for $\mathcal{T}(3, \frown\smile)$ and $1 + 2\chi$ for the other types. Note that $1 + 2\chi \leq 2(1 - \chi)$, for $\chi \leq 1/4$. □



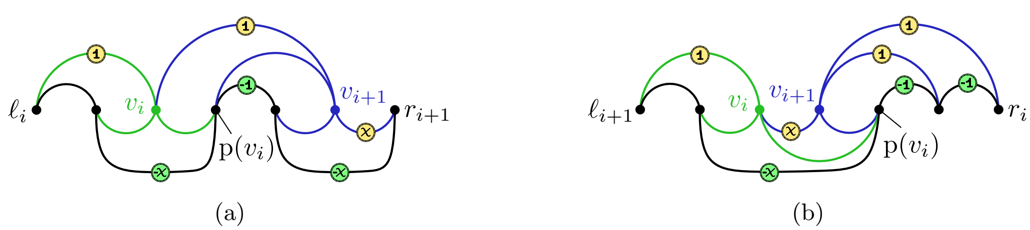
(a) $\mathcal{T}(2, \frown)$ (b) $\mathcal{T}(3, \frown^2)$

(c) $\mathcal{T}(4, \frown^3)$ (d) $\mathcal{T}(3, \frown\smile)$

Figure 8: Insertion of v_i and v_{i+1} if $v_{i+1} = \mathrm{pc}(v_i)$ has degree two in G_{i+1}.

Lemma 8. *Assume that there are $v, v' \in \mathcal{E}_{i-1}$ such that $\mathrm{pc}(v) = v'$ and at least one of v, v' has right pivot type. Then we can set $v_i = v$ and $v_{i+1} = v'$ and spend at most one credit to obtain a χ-extensible arc diagram for G_{i+1}.*

Proof: If both v and v' have right pivot type, then we use the diagram shown in Figure 9a. The costs are $1 - \chi \leq 2(1 - \chi)$, for $\chi \leq 1$. Otherwise, one of v, v' has left pivot type and the other has right pivot type, then $\mathrm{p}(v) = \mathrm{p}(v')$ and $\mathrm{pc}(v') = v$. As the roles of v and v' are symmetric, we may assume w.l.o.g. that v has right pivot type and v' has left pivot type. We use the diagram shown in Figure 9b for the case where v' is $\mathcal{T}(3, \frown^2)$; other types are handled analogously. The costs to establish the invariants are $1 \leq 2(1 - \chi)$, for $\chi \leq 1/2$. □



(a) (b)

Figure 9: Insertion of v_i and $v_{i+1} = \mathrm{pc}(v_i) \in \mathcal{E}_{i-1}$ if v_i has right pivot type.

If we can apply one of Lemmas 7 and 8, we make progress by inserting two vertices v_i and v_{i+1}. Hence, from now on, we assume that neither of Lemmas 7 and 8 can be applied. Our goal in the remainder of this section is to show that in this case we can find a vertex u that is not eligible but sufficiently close to being eligible—in a way described in the following—that we can aim to insert u next, along with some other vertices.

More specifically, the vertex u has neighbors $w_1, \dots, w_k$ on $P_\circ(G_{i-1})$, for $k \geq 2$, and each subregion X_j of $R_{i-1}(u)$ bounded by the edges uw_j and uw_{j+1} has a particularly simple structure. First of all, there exists an integer $s = s(X_j)$ such that we have $X_j \cap \mathcal{E}_{i-1} = \{c_1, \dots, c_s\}$, and every c_ℓ, for $1 \leq \ell \leq s$, is adjacent to u in G. We distinguish three types of regions, depending on whether X_j contains eligible vertices of left, right, or both pivot types.

Left-pivot region. (see Figure 10a)

- Every c_ℓ, for $1 \leq \ell \leq s$, has left pivot type.
- We have $\mathrm{pc}(c_1) = u$ and $\mathrm{pc}(c_\ell) = c_{\ell-1}$, for all $2 \leq \ell \leq s$.
- All vertices in $(V \setminus \mathcal{E}_{i-1}) \cap X_j$ lie inside the face bounded by uc_sw_{j+1}.

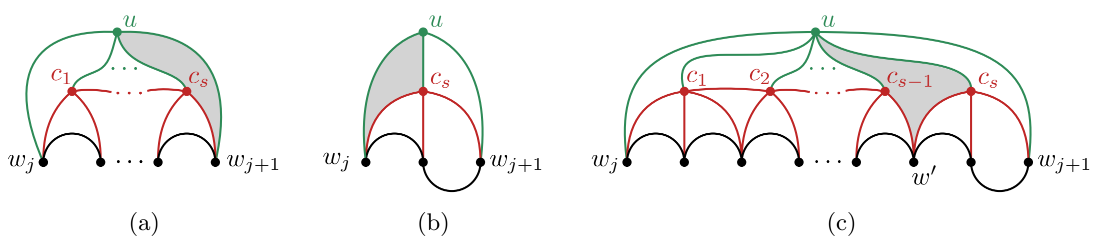


Figure 10: Structure of regions that our to-be-inserted-next vertex u spans with $P_\circ(G_{i-1})$. All eligible vertices (shown red) are adjacent to u, all other vertices lie inside the shaded region.

Right-pivot region. (see Figure 10b)

- We have $s = 1$, the vertex c_1 has right pivot type, and $\mathrm{pc}(c_1) = u$.
- All vertices in $(V \setminus \mathcal{E}_{i-1}) \cap X_j$ lie inside the face bounded by uw_jc_1.

Both-pivot region. (see Figure 10c)

- Every c_ℓ, for $1 \leq \ell \leq s-1$, has left pivot type and c_s has right pivot type.
- We have $\mathrm{pc}(c_1) = \mathrm{pc}(c_s) = u$ and $\mathrm{pc}(c_\ell) = c_{\ell-1}$, for all $2 \leq \ell \leq s-1$.
- The rightmost neighbor of c_{s-1} on $P_\circ(G_{i-1})$ is the same as the leftmost neighbor of c_s on $P_\circ(G_{i-1})$; denote this vertex by w'.
- All vertices in $(V \setminus \mathcal{E}_{i-1}) \cap X_j$ lie inside the quadrilateral $uc_{s-1}w'c_s$.

How to select u. In the remainder of this section we will sketch how to select a suitable vertex u such that all regions spanned by u and $P_\circ(G_{i-1})$ have the nice structure explained above. The first part of the story is easy to tell: We select u to be a minimal (w.r.t. the region containment relation $\preceq$) element of the set $\mathcal{U} := \{\mathrm{pc}(v) \colon v \in \mathcal{E}_{i-1}\} \setminus \mathcal{E}_{i-1}$. We prove that such a vertex always exists.

Lemma 9. *We have* $\mathcal{U} \neq \emptyset$.

Proof: We know that $\mathcal{E}_{i-1} \neq \emptyset$ and that all vertices in $\mathcal{E}_{i-1}$ are problematic. Assume for the sake of a contradiction that for every $v \in \mathcal{E}_{i-1}$ we have $\mathrm{pc}(v) \in \mathcal{E}_{i-1}$. Then there exists a cyclic sequence $u_0, \ldots, u_k$ of eligible vertices, for $k \geq 1$, such that $\mathrm{pc}(u_j) = u_{(j+1) \bmod k}$, for all $0 \leq j \leq k$. We may assume that all vertices $u_0, \ldots, u_k$ have left pivot type as otherwise we can apply Lemma 8.

Every edge of $P_\circ(G_{i-1})$ is covered by at most one vertex from $\mathcal{E}_{i-1}$, and conversely every vertex in $\mathcal{E}_{i-1}$ covers some subpath of at least one consecutive edge(s) of $P_\circ(G_{i-1})$. Thus, we can order the vertices in $\mathcal{E}_{i-1}$ from left to right according to the part of $P_\circ(G_{i-1})$ they cover. Without loss of generality let u_0 be the leftmost vertex among $u_0, \ldots, u_k$, and let ℓ be the leftmost neighbor of u_0 on $P_\circ(G_{i-1})$. Then by the left-to-right order the edges of $P_\circ(G_{i-1})$ covered by u_1 are to the right of the edges of $P_\circ(G_{i-1})$ covered by u_0. At the same time u_1 is adjacent to ℓ because $u_1 = \mathrm{pc}(u_0)$. It follows that $R_{i-1}(u_1) \supset R_{i-1}(u_0)$, which by Corollary 1 is in contradiction to $u_1 \in \mathcal{E}_{i-1}$. □

As there is a vertex $v \in \mathcal{E}_{i-1}$ with $u = \mathrm{pc}(v)$, we know that $u \in \mathcal{U}$ has at least one neighbor on $P_\circ(G_{i-1})$, which is $\mathrm{p}(v)$. By Lemma 7 we may assume $d_{i-1}(u) \geq 2$. Let $w_1, \ldots, w_k$ denote the sequence of neighbors of u along $P_\circ(G_{i-1})$. The edges uw_j, for $2 \leq j \leq k-1$, split $R_{i-1}(u)$ into $k-1$ subregions; let X_j denote the (open) region bounded by $w_j u w_{j+1}$ and the part of $P_\circ(G_{i-1})$ between w_j and w_{j+1}, for $1 \leq j < k$.

Lemma 10. *In every region* X_j*, for* $1 \leq j < k$*, there is at most one eligible vertex* v *of each pivot type for which* $\mathrm{pc}(v) = u$.

Proof: Let $v \in X_j \cap \mathcal{E}_{i-1}$ with $\mathrm{pc}(v) = u$. Then u is adjacent to $\mathrm{p}(v)$ in G. As u has only two neighbors on $P_\circ(G_{i-1}) \cap \partial X_j$, we have $\mathrm{p}(v) \in \{w_j, w_{j+1}\}$. So, if v has left pivot type, then $\mathrm{p}(v) = w_j$ and v is the unique vertex that covers the edge of $P_\circ(G_{i-1})$ whose left endpoint is w_j. Else v has right pivot type, $\mathrm{p}(v) = w_{j+1}$, and v is the unique vertex that covers the edge of $P_\circ(G_{i-1})$ whose right endpoint is w_{j+1}. □

Lemma 10 together with Lemma 8 implies the following.

Lemma 11. *In every region* X_j*, at most one eligible vertex has right pivot type. If there exists a vertex* $v \in X_j \cap \mathcal{E}_{i-1}$ *that has right pivot type, then* $\mathrm{pc}(v) = u$.

Proof: For every $v \in X_j$, we have $\mathrm{pc}(v) \in X_j \cup \{u\}$ by planarity. Therefore, by the choice of u as a minimal element of $\mathcal{U}$, we have $\mathrm{pc}(v) \in \mathcal{E}_{i-1} \cup \{u\}$. If v has right pivot type, then by Lemma 8 we have $\mathrm{pc}(v) \notin \mathcal{E}_{i-1}$ and, therefore, $\mathrm{pc}(v) = u$. Now the statement follows from Lemma 10. □

Lemma 12. *Let* Q *denote the set of vertices in* $X_j \cap \mathcal{E}_{i-1}$ *that have left pivot type. If* $Q \neq \emptyset$*, then the vertices in* Q *form a sequence* $x_1, \ldots, x_q$*, for some* $q \geq 0$*, such that* $x_j = \mathrm{pc}(x_{j+1})$*, for* $1 \leq j \leq q-1$*, and* $\mathrm{pc}(x_1) = u$.

Proof: For every $x \in Q$, we have $\mathrm{pc}(x) \in X_j \cup \{u\}$ by planarity. Thus, by the choice of u (as a minimal element of $\mathcal{U}$) either $\mathrm{pc}(x) = u$ or $x' = \mathrm{pc}(x) \in \mathcal{E}_{i-1}$. By Lemma 10 the former case applies to at most one vertex of Q. In the latter case we may assume that $x' \in Q$ as otherwise we can apply Lemma 8. Each $y \in \mathcal{E}_{i-1}$ covers a subpath $\sigma(y)$ of $P_\circ(G_{i-1})$ and has no other neighbors on $P_\circ(G_{i-1})$. As $x' = \mathrm{pc}(x)$, we know that x' is adjacent to $\mathrm{p}(x)$, which is the left endpoint of $\sigma(x)$; thus $\mathrm{p}(x)$ is also the right endpoint of $\sigma(x')$. Therefore, we can order the vertices in Q from left to right, according to the order of the corresponding paths $\sigma(\cdot)$ on $P_\circ(G_{i-1})$. For the leftmost vertex x_1 in this order, we must have $\mathrm{pc}(x_1) = u$. □

The following lemma asserts that each edge of $P_\circ(G_{i-1})$ on the boundary of region X_j is covered by an eligible vertex or by u.

Lemma 13. *Let $e \in P_\circ(G_{i-1}) \cap \partial X_j$, for some $1 \leq j < k$, and let $c_e \in V \setminus V_{i-1}$ denote the vertex that covers e. Then either $c_e = u$ or $c_e \in \mathcal{E}_{i-1}$.*

Proof: Assume for a contradiction that $c_e \neq u$ and $c_e \notin \mathcal{E}_{i-1}$. As $c_e \neq u$, by planarity $c_e \in X_j$ and, therefore, $R_{i-1}(c_e) \subsetneq R_{i-1}(u)$ and $c_e \preceq u$. As $c_e \notin \mathcal{E}_{i-1}$, by Corollary 1 there exists a vertex $v \in R_{i-1}(c_e) \cap \mathcal{E}_{i-1}$. Let $v' = \mathrm{pc}(v)$. By planarity $v' \in R_{i-1}(c_e) \cup \{c_e\} \subsetneq R_{i-1}(u)$. In particular, we have $v' \neq u$. We next show that $v' \in \mathcal{E}_{i-1}$. To this end, we first observe that $d_{i-1}(v') \geq 2$ by Lemma 7. Thus, by planarity $v' \preceq u$. As a result, by the choice of u (as a minimal element of $\mathcal{U}$ w.r.t. $\preceq$), we necessarily have that $v' \notin \mathcal{U}$. As $v' = \mathrm{pc}(v)$, it follows that $v' \in \mathcal{E}_{i-1}$. In particular, as $c_e \notin \mathcal{E}_{i-1}$, we conclude that $v' \neq c_e$. By Lemma 11, both of the two vertices v and v' have left pivot type. Thus, by Lemma 12 there is a sequence $x_1, \ldots, x_q$ of eligible vertices, with $x_{q-1} = v'$ and $x_q = v$, such that $x_h = \mathrm{pc}(x_{h+1})$, for all $1 \leq h \leq q-1$, and $\mathrm{pc}(x_1) = u$. In particular, we have $x_1 \notin R_{i-1}(c_e)$ because $u \notin R_{i-1}(c_e) \cup \{c_e\}$. Let $h \geq 1$ be maximal such that $x_h \notin R_{i-1}(c_e)$, and note that $1 \leq h \leq q-2$. Then $x_{h+1} \in R_{i-1}(c_e)$ and, therefore, $x_h = \mathrm{pc}(x_{h+1}) \in R_{i-1}(c_e) \cup \{c_e\}$. It follows that $x_h = c_e$, which, in particular, implies that $c_e \in \mathcal{E}_{i-1}$, a contradiction. □

We process the regions $X_1, \ldots, X_{k-1}$ together with u. Consider a region X_j so that $X_j \cap V \neq \emptyset$, and denote $E_j = P_\circ(G_{i-1}) \cap \partial X_j$. By Lemma 13 the vertices that cover one or more edges of E_j are exactly the vertices in $\mathcal{E}_{i-1} \cap X_j$. Thus, we can order these vertices from left to right, according to the edge(s) in E_j they cover. Denote this sequence by $c_1, \ldots, c_s$. By Lemma 11 the only vertex in $X_j \cap V$ that may have right pivot type is c_s. Denote $s' = s - 1$ if c_s has right pivot type, and $s' = s$, otherwise; i.e., $c_{s'}$ is the rightmost vertex of the sequence that has left pivot type. By Lemma 12 we have $c_h = \mathrm{pc}(c_{h+1})$, for $1 \leq h \leq s'-1$, and $\mathrm{pc}(c_1) = u$. It follows that the rightmost vertex w' of $P_\circ(G_{i-1})$ that is adjacent to $c_{s'}$ is the only vertex of $P_\circ(G_{i-1})$ that can be adjacent to a vertex in $(X_j \cap V) \setminus \mathcal{E}_{i-1}$. So the general situation inside X_j can be summarized as depicted in Figure 11. Note that X_j may contain only vertices of left pivot type, or only vertices of right pivot type (in fact, one such vertex only). If neither type is present, then $X_j \cap V = \emptyset$.

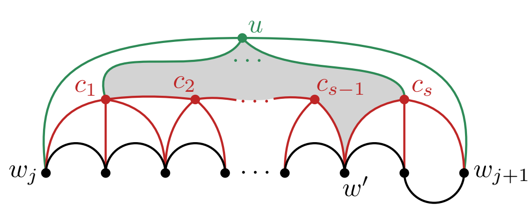


Figure 11: The structure of eligible vertices within a region X_j. All triangular faces here are empty, only the central face (shaded) may contain other vertices or edges uc_h, for $2 \leq h < s$. The left pivot vertices could be of any type $\mathcal{T}(z, \frown^{z-1})$, for any integer $z \geq 2$.

The following lemma allows us to assume that the central face in each region X_j is subdivided into empty (of vertices) triangles and at most one—not necessarily empty— triangle or quadrilateral (the latter if X_j contains eligible vertices of both pivot types).

Lemma 14. *Let X_j be a region such that there exist $v, v' \in \mathcal{E}_{i-1} \cap X_j$ with $\mathrm{pc}(v) = v'$, let v'' be the vertex that covers vv'. If $v'' \neq u$ and $\chi \leq 1/5$, there exist $v_i, \ldots, v_{i+h-1}$ with $h \geq 3$ such that a χ-valid diagram for G_{i+h-1} can be obtained by spending at most $(1-\chi)h$ credits.*

Proof: By Lemma 11 both v and v' have left pivot type. In particular, if $c_s \neq c_{s'}$, this implies that we have $c_s \notin \{v, v'\}$ (see also Figure 11). By planarity and as $v'' \neq u$, we have $v'' \in X_j$. If v'' is not adjacent to w', then v'' is eligible after adding v and v' and we can set $v_i = v$, $v_{i+1} = v'$, and $v_{i+2} = v''$ and use the diagram for G_{i+2} shown in Figure 12a, for a cost of $2 + 2\chi \leq 3 - 3\chi$, for $\chi \leq 1/5$. The figure shows the drawing where both v and v' are $\mathcal{T}(2, \frown)$; it easily extends to the types $\mathcal{T}(3, \frown^2)$ and $\mathcal{T}(4, \frown^3)$ because more mountains to the right of v can be paid for by the corresponding mountains whose left endpoint is covered by v and for more mountains to the left of v' their left endpoint is covered by v'.

Otherwise, v'' is adjacent to w'. We claim that we may assume $v = c_{s'}$ and $v' = c_{s'-1}$. To see this let $\tilde{v} \neq v''$ be the vertex that covers $c_{s'-1}c_{s'}$ and observe that $\tilde{v}$ is enclosed by a cycle formed by $vv''w'$ and the part of $P_\circ(G_{i-1})$ between the right neighbor of v and w'. In particular, we have $\tilde{v} \neq u$ and so $c_{s'-1}, c_{s'}, \tilde{v}$ satisfy the conditions of the lemma, as claimed. We set $v_i = v$ and $v_{i+1} = v'$, and use the diagram shown in Figure 12b. If v'' is eligible in G_{i+1}, that is, the triangle $vv''w'$ is empty of vertices, then we set $v_{i+2} = v''$ and have a diagram for G_{i+2} for a cost of $2 + \chi \leq 3 - 3\chi$, for $\chi \leq 1/4$.

Otherwise, that is, if we have that v'' is adjacent to w' but v'' is not eligible in G_{i+1}, by Lemma 3 we inductively obtain a χ-valid diagram D for the subgraph of G induced by taking $v''vw'$ as an outer triangle together with all vertices inside, with $v''v$ as a starting edge and w' as a last vertex. Then we plug D into the triangle $v''vw'$ as shown in Figure 12b. All mountains of D with left endpoint v'' carry a credit by (I2) for D. Thus, the resulting diagram is χ-extensible. For the costs we have to account for the fact that w' is considered to contribute $1 - \chi$ credits to D, whereas we had already accounted for w' in the diagram for G_{i-1}. On the other hand, the edge $v''w'$ is paid for as a part of D. Thus, the additional costs to handle v, v', v'' are $(1 - \chi) + 1 + \chi = 2 \leq 3 - 3\chi$, for $\chi \leq 1/3$. □

(a) (b)

Figure 12: Two vertices v, v' that have left pivot type and $v'' \neq u$ covers the edge vv'.

3.3 Processing regions

To complete the proof of Lemmas 2 and 3 it remains to insert u along with the set $\mathcal{V}_u := V \cap R_{i-1}(u)$ of all vertices inside $X_1, \ldots, X_{k-1}$, at a cost of $1 - \chi$ credits per vertex. We process these regions from right to left in two phases: In Phase 1, we select a suitable collection $X_j, \ldots, X_{k-1}$ of regions, for some $j \in \{1, \ldots, k-1\}$, so that we can insert u together with all the vertices inside these regions. Then in Phase 2, we process the remaining regions, assuming that u is already placed on the spine, somewhere to the right. To achieve this we do a case analysis, depending on the four types of regions: left, right, both pivot, or empty. In this section, we show that in all cases $u \cup \mathcal{V}_u$ can be inserted as required, which then concludes the proofs of Lemma 2 and Lemma 3 and therefore also the proof of Theorem 1.

Using the insights on the type and structure of eligible vertices within the regions covered by our selected "minimally not eligible" vertex $u \in \mathcal{U}$ that we have developed in Section 3.2 we can now describe how to handle the generic case. It consists of processing u along with all regions $X_1, \ldots, X_{k-1}$ covered by u, thereby adding $\nu := |R_{i-1}(u) \cap V| + 1$ vertices to the diagram. So our main goal is to extend the given χ-extensible arc diagram for G_{i-1} to an χ-extensible (for Lemma 2) or at least χ-valid (for Lemma 3) arc diagram for $G_{i-1+\nu}$.

We can classify the regions covered by u into four different types. Each region is either *empty*, *left pivot*, *right pivot*, or *both pivot*—depending on whether it contains no vertices of G, or at least one eligible vertex of left, right, or both pivot types, respectively. An empty region has a unique edge of $P_\circ(G_{i-1})$ on its boundary; depending on whether this edge is a pocket or mountain we call the corresponding region an *empty pocket* or an *empty mountain*, respectively.

We proceed in several steps. As a general rule, we process $X_{k-1}, \ldots, X_1$ in this order from right to left. When processing X_j we assume that X_j is not empty and that u and all edges and vertices inside or on the boundary of X_h, for all $h > j$, are placed already; specifically, the edge uw_{j+1}, which is shared between X_{j+1} and X_j, is drawn already, and it is already paid for.

(I6) The region X_j is not empty. If uw_{j+1} is a mountain, then it carries $1 - \chi$ credits, and if uw_{j+1} is a pocket, then it carries 2χ credits.

As an initialization we process some regions $X_{j+1}, \ldots, X_{k-1}$ so as to establish (I6) for X_j. Note that there exists a region X_j, with $1 \leq j < k$, that is nonempty because $u \notin \mathcal{E}_{i-1}$. Moreover, in the following procedure, if all regions $X_{h'}$ with $h' \in \{j-1, \ldots, j'+1\}$ are empty for some j', we process $X_j, X_{j-1}, \ldots, X_{j'+1}$ together so that the next region $X_{j'}$ to be processed is non-empty again.

3.3.1 Initialization: Placing u and selecting $\mathbf{X_j}$

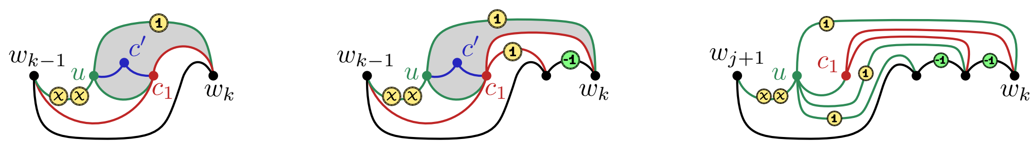
Special case in the proof of Lemma 3: $u = v_n$. In this case, the placement of u is determined, as u must be the rightmost vertex on the spine. As $w_k = v_2$, we have to ensure that the edge uw_k is not drawn as a pocket. Let X_j be the rightmost region that is not empty. We place u as the rightmost vertex on the spine and draw all edges $uw_k, \ldots, uw_{j+1}$ as mountains and put $1 - \chi$ credits, paid by the new vertex u, on uw_{j+1} so as to establish (I6) for X_j.

General cases. In all other cases (again identical for the proofs of Lemma 2 and Lemma 3), we have to place u somewhere between v_1 and v_2 on the spine. To this end, we will select a region X_j, for some $1 \leq j < k$, and place u as a part of processing X_j. We start with the rightmost region X_{k-1} and work our way from there to the left. We may suppose without loss of generality that X_{k-1} is not an empty mountain. To see this, suppose that X_{k-1} is an empty mountain. Then we continue as if X_{k-2} was the rightmost region. Once all regions are processed, we add the edge uw_k to the diagram as a mountain. The costs can be paid for by a mountain in X_{k-1} whose left endpoint is covered by u.

If X_{k-1} is an empty pocket, then we place u into this pocket. Let X_j be the rightmost region that is not empty. We pay χ credits for the pocket uw_k, which can be paid for using the χ credits on the pocket of X_{k-1}. Then to establish (I6) we have to pay $1-\chi$ credits for uw_{j+1}, which is exactly what the new vertex u provides. So it remains to consider the case $X_{k-1} \cap V \neq \emptyset$ only. We distinguish three subcases according to the type of X_{k-1}. In all of them, we consider the plane graph $G' = G[V_{i-1} \cup X_{k-1} \cup \{u\}]$.

$\mathbf{X_{k-1}}$ is a left pivot region. Then, by Lemma 14 we may assume that at most one face of G' may contain other vertices of G, namely the triangle $\Delta = uc_s w_k$ (shaded in figures).

If $s=1$, then we insert c_1 by pushing down the leftmost mountain it covers and place u into the pocket to the left of c_1, see Figure 13a. The figure shows the case that c_1 is $\mathcal{T}(2, \frown)$; if c_1 covers more mountains to the right (i.e., $\mathrm{pr}(c_1) = \frown^i$ for $i \geq 2$), then the additional mountain(s) at c_1 can be paid for using the credit(s) on the mountains that are covered, see Figure 13b.



(a) (b) (c)

Figure 13: Only one vertex in X_{k-1} and it has left pivot type.

If $\Delta \cap V = \emptyset$, then the costs are $1 + 2\chi \leq 2(1-\chi)$, for $\chi \leq 1/4$, which the two new vertices u and c_1 can pay. This suffices to establish (I6) for X_{k-2} if X_{k-2} is nonempty. If X_{k-2} is empty and $\Delta \cap V = \emptyset$, then we use the following diagram instead. Let j be minimal such that all of $X_{j+1}, \ldots, X_{k-2}$ are empty. We place u into the unique edge of X_{j+1} on $P_\circ(G_{i-1})$, pushing it down if it is a mountain, and add all edges $uw_{j+1}, \ldots, uw_{k-1}$ as proper arcs. Then we push down all mountains with left endpoint u and place c_1 to the right of u. Finally, add uw_k as a mountain; see Figure 13c. Since by assumption c_1 has left pivot type, u covers only mountains. Thus, the costs for the biarcs at u can be paid for by the mountains covered by u, if $j+1 < k-2$, plus by the at least one mountain covered by c_1. So to establish (I6) for X_j we only have to pay 2χ credits for the pocket to the left of u and one credit for uw_k. This amounts to $1 + 2\chi \leq 2(1-\chi)$, for $\chi \leq 1/4$, which the two new vertices u and c_1 can pay. This approach also works in case $j = 0$, we even paid χ credits too much for uw_1.

Else we have $\Delta \cap V \neq \emptyset$, and using Lemma 3 we inductively obtain a χ-valid diagram D for the subgraph of G induced by taking Δ as an outer triangle together with all vertices inside, with uc_1 as a starting edge and w_k as a last vertex. Then we plug D into Δ; to this end, note that we place c_1 and u as shown in Figure 13. If the edge $c_1 w_k$ is drawn as a biarc in D, then we push down all mountains with left endpoint c_1 (if any exist) to make room. This is where we need the credits on these mountains if $d_{i-1}(c_1) > 2$. All mountains of D with left endpoint u carry a credit by (I2) for D. As for the costs, let c' be the vertex that covers uc_1 in D. We pay 2χ credits to initialize the pockets incident to c' in D. We also have to account for the fact that w_k is considered to contribute $1-\chi$ credits to D, whereas we had already accounted for w_k in G_{i-1}. In return the edge uw_k is paid for as a part of D. Finally, we have to place 2χ credits on $w_{k-1}u$ to establish (I6) for X_{k-2} if X_{k-2} is nonempty. Otherwise, let j be minimal such that all of $X_{j+1}, \ldots, X_{k-2}$ are empty and add all edges $uw_{j+1}, \ldots, uw_{k-1}$ as mountains. This costs one credit, for the leftmost mountain uw_{j+1}. So in any case the costs are at most $2\chi + (1-\chi) + 1 = 2 + \chi \leq 3(1-\chi)$, for $\chi \leq 1/4$, which the three new vertices u, c_1, c' can pay. Either we have established (I6) for some X_j or, if $j = 0$, that is, if X_{k-1} is the only nonempty region, then this step is complete.

It remains to consider the case $s > 1$. If $s \geq 3$, then for each vertex c_h, with $h \neq s-1$, we push down the leftmost mountain it covers. Then we place first c_{s-1} and then u into the pocket to the left of c_s; see Figure 14a. If $s = 2$, then we push down the leftmost mountain covered by c_1 and place first u and then c_2 into the pocket to the left of c_1; see Figure 14b. By (I2) the costs for the biarcs created can be paid for by using the credits on the mountains that are pushed down. If there are any mountains with left endpoint c_s other than $c_s w_k$, they can be paid for using the credits on the mountains covered by c_s to the right. We pay at most one credit for the edge uw_{k-1}.

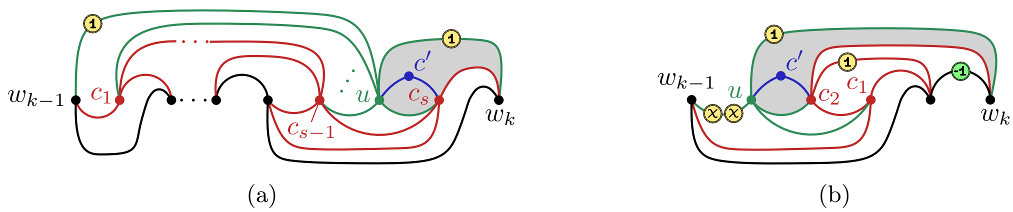


Figure 14: All $s \geq 2$ vertices have left pivot type in X_{k-1}.

If $\Delta \cap V = \emptyset$, then we also pay one credit for uw_k, for overall costs of at most $2 \leq 3(1-\chi) \leq (s+1)(1-\chi)$, for $\chi \leq 1/3$. Else we have $\Delta \cap V \neq \emptyset$, and using Lemma 3 we inductively obtain a χ-valid diagram D for the subgraph of G induced by taking Δ as an outer triangle together with all vertices inside, with uc_s as a starting edge and w_k as a last vertex. Then we plug D into Δ. Regarding the costs we argue as above in the case $s = 1$ to bound them by $2\chi + (1-\chi) + 1 = 2 + \chi \leq 4(1-\chi) \leq (s+2)(1-\chi)$, for $\chi \leq 2/5$.

Note that in all cases above we accounted for a cost of one credit for the edge uw_{k-1} (even though for $s = 2$ we would have to pay 2χ credits only). Therefore, for any sequence $X_{k-2}, \ldots, X_{j+1}$ of empty regions, we can afford to add the edges $uw_{k-2}, \ldots, uw_{j+2}$ and put one credit on uw_{j+2} so as to establish (I6) for X_j if it is nonempty, or even complete this step in case $j = 0$ (that is, if X_{k-1} is the only nonempty region).

$\mathbf{X_{k-1}}$ is a right pivot region. If $k = 2$ or if X_{k-2} is nonempty, then we push down the mountain covered by c_1 and then place u into the pocket to the left of c_1, see Figure 15a. If $\Delta \cap V = \emptyset$, then the costs to either finish this step (if $k = 2$) or establish (I6) for X_{k-2} are $1 + \chi \leq 2(1-\chi)$, for $\chi \leq 1/3$, which the two new vertices u and c_1 can pay. Otherwise, using Lemma 6 we inductively obtain a χ-extensible diagram D for the subgraph G_Δ of G induced by taking Δ as an outer triangle together with all vertices inside, with $w_{k-1}c_1$ as a starting edge and u as a last vertex. Then we plug D into Δ and add the edge uw_k. Let c' be the vertex that covers $w_{k-1}c_1$ in D. The costs are 2χ credits to initialize the two pockets incident to c' in D, one credit for the mountain uw_k, and possibly an additional χ credits if the edge $w_{k-1}u$ is a pocket in D. To compensate we may take the χ credits on the pocket covered by c_1. Thus, by Lemma 6 the costs to add c_1 and c' are at most $1+2\chi \leq 2(1-\chi)$, for $\chi \leq 1/4$, or there exists an appropriate χ-extensible diagram D' for $G_\Delta \setminus \{u\}$ for which u is problematic. In the latter case, we just plug D' into Δ. Then, we distinguish two cases.

If u has left pivot type in G_Δ, we push down the leftmost arc covered by u in D' to place u there. In this case we pay 2χ credits to initialize the pockets incident to c'. Then we put 2χ credits on the pocket $w_{k-1}u$ and two credits on mountains with left endpoint u. There could be three or four mountains with left endpoint u, but any but the first and the last can be paid using the credit on a corresponding mountain covered by the insertion of u. Finally, we can take the χ credits on the pocket covered by c_1. So the costs to add c_1, c', u and establish (I6) for X_{k-2} are at most $4\chi + 2 - \chi = 2 + 3\chi$. This is too much by χ because in order to be upper bounded by $3(1-\chi)$ we would need $\chi \leq 1/6$. However, recall that either $k = 2$ or X_{k-2} is nonempty by assumption. If $k = 2$, then there is no need to place 2χ credits on $w_{k-1}u$ and we can take the missing χ credits from there. Otherwise, we undo the insertion of u but keep the drawing $G_\Delta \setminus \{u\}$. Next, we pretend that X_{k-2} is the rightmost region and process it accordingly, as described in this section. Doing so also places u somewhere to the left of w_{k-1}. Finally, in order to incorporate X_{k-1} we add the missing edges to u as mountains and put one credit on each of them. The credits for those mountains that cover $G_\Delta \setminus \{u\}$ can be taken from the mountains of $G_\Delta \setminus \{u\}$ that are covered by u. We need to pay one credit for the mountain uw_k only. In addition, to insert c_1 and c' we pay 2χ credits to initialize the pockets incident to c', but we can take the χ credits from the pocket covered by c_1. Thus, these costs are $1 + 2\chi - \chi = 1 + \chi$, and we can afford to pay χ more credits, to cover the missing χ credits in case that we end up in this very same case when processing X_{k-2}. So in total we account for $1 + 2\chi \leq 2(1-\chi)$ credits to insert c_1 and c', for $\chi \leq 1/4$.

Otherwise, the vertex u has right pivot type in G_Δ and we just place it into the pocket of D' it covers. We pay 2χ credits to initialize the pockets incident to c', which can be paid by the χ credits each from the pockets covered by c_1 and u. Then we put $1-\chi$ credits on the mountain $w_{k-1}u$ and one credit on uw_k. So the costs to add c_1, c', u and establish (I6) for X_{k-2} are at most $2 - \chi \leq 3(1-\chi)$, for $\chi \leq 1/4$.

Figure 15: Exactly one eligible vertex in X_{k-1} and it has right pivot type.

It remains to consider the case that X_{k-2} is empty. Select j to be minimal such that all of $X_{j+1}, \ldots, X_{k-2}$ are empty. If $\Delta \cap V = \emptyset$, then we place c_1 into the pocket it covers and then place u into the pocket to the right of c_1, see Figure 15b. We place one credit on the mountain uw_{j+1} and χ credits on the pocket uw_k. As we can take $1 + \chi$ credits from the edges covered by c_1, the costs are zero to establish (I6) for X_j, or end this step if $j = 0$. Otherwise, we have $\Delta \cap V \neq \emptyset$ and proceed exactly as described above for the case that X_{k-2} is nonempty—except that we also include all of the regions $X_{j+1}, \ldots, X_{k-2}$ in the induction. To formally obtain a triangulation, we add virtual edges $c_1w_{j+1}, \ldots, c_1w_{k-2}$, which we immediately remove from the resulting drawing again. The analysis remains unchanged.

$\mathbf{X_{k-1}}$ is a both pivot region. By Lemma 11 there is exactly one vertex c_s of right pivot type in $X_j \cap \mathcal{E}_{i-1}$ and we have $\mathrm{pc}(c_s) = u$. All vertices in $(X_j \cap V) \setminus \mathcal{E}_{i-1}$ (if any exist) are in the open quadrilateral $\square = c_{s-1}w'c_su$.

If $\square \cap V = \emptyset$, then for each c_h, with $1 \leq h \leq s$, we push down the leftmost mountain covered by c_h and place c_h there. Then we place u into the pocket to the left of c_s; see Figure 16a, which shows the case $s = 2$. The costs are $2 - \chi \leq 3(1 - \chi)$, for $\chi \leq 1/2$, which can be paid by the new vertices u, c_{s-1}, c_s. As $w_{k-1}u$ is a mountain that carries one credit, we can add more mountains from u to the left in case there are empty regions there and then move the credit to the leftmost such mountain. So we either establish (I6) for some X_j, with $1 \leq j \leq k - 2$, or we finish this step if X_{k-1} is the only nonempty region.

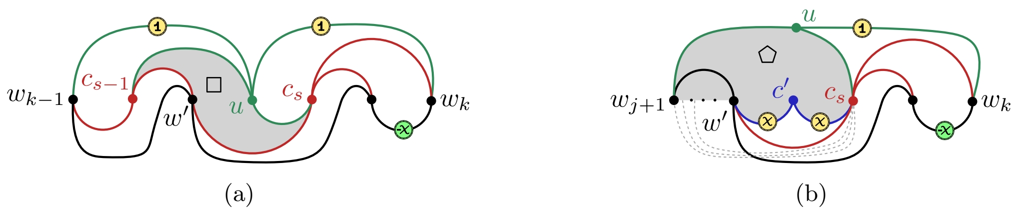


Figure 16: There are eligible vertices of both pivot types in X_{k-1}.

Otherwise, let j be minimal such that all of $X_{j+1}, \ldots, X_{k-2}$ are empty. Note that we may have $j = k - 2$ if X_{k-2} is nonempty or $j = 0$ if X_{k-1} is the only nonempty region. Let c' be the vertex inside $\square$ that forms a triangle with $w'c_s$ in G, and note that w' is the only neighbor of c' on $P_\circ(G_{i-1})$. We place c_s by pushing down the mountain it covers. Let $\pentagon$ be the open region bounded by the path $w'c'c_suw_{j+1}$ together with the part of $P_\circ(G_{i-1})$ between w_{j+1} and w' in G, and let $G_{\pentagon}$ be the graph obtained by adding the virtual edges $w_{j+1}c_s, \ldots, w_{k-1}c_s$ (which are not in G) to the subgraph of G induced by the cycle $\partial\pentagon$ together with all vertices inside; see Figure 16b. Observe that $G_{\pentagon}$ is a triangulation; in particular the inserted virtual edges triangulate the outer face of the graph induced by $\partial\pentagon$ and the vertices contained within. Thus, using Lemma 2 we inductively obtain a χ-extensible diagram D for $G_{\pentagon}$, with $w_{j+1}c_s$ as a starting edge, the profile (w_{j+1}, w', c', c_s) shown in Figure 16b, and u as a last vertex. Then we remove the virtual edges from D, plug the resulting diagram into $\pentagon$, add the edge uw_k as a mountain, and place one credit on it. We also pay 2χ credits to initialize the two pockets incident to c' in D and another 2χ credits for Lemma 2. But we can take χ credits from the pocket covered by c_s. So the costs to add c_s and c' are at most $1 + 3\chi \leq 2(1 - \chi)$, for $\chi \leq 1/5$, to either establish (I6) for X_j or finish this step.

3.3.2 Processing the remaining regions

If the initialization described in the previous section does not complete processing of u and its regions already, then it establishes (I6) for some region X_j, with $1 \leq j \leq k-1$. Denote the current working diagram (for $G[V_{i-1} \cup \bigcup_{h=j+1}^{k-1}(X_h \cap V) \cup \{u\}]$) by Γ. As X_j is nonempty by (I6), the region X_j is either left, right, or both pivot. These three different cases are discussed below. In all cases the edge uw_j is drawn as a mountain and we place one credit on it. Therefore, any number of empty regions $X_h, \ldots, X_{j-1}$, for $1 \leq h \leq j$, are easy to handle: Just add the edges $uw_h, \ldots, uw_{j-1}$ and move the credit from uw_j to uw_h, to establish (I6) for X_{h-1} or finish this step if $h = 1$.

$\mathbf{X_j}$ is a left pivot region. If $s = 1$ and $\Delta \cap V = \emptyset$, then we place c_1 by pushing down the leftmost mountain it covers. We pay one credit for the mountain w_ju, but we can take the credits on $w_{j+1}u$; see Figure 17a. So by (I6) we pay at most $1 - 2\chi$ credits, which the new vertex c_1 is happy to supply. If $s = 1$ and $\Delta \cap V \neq \emptyset$, then we distinguish two cases.

(a) (b)

Figure 17: Exactly one eligible vertex in X_j, for $j < k$, and it has left pivot type. Also observe that the solutions for subproblems reinserted into the gray shaded regions are actually rotated by π.

If the edge $w_{j+1}u$ is a mountain, then we push down the leftmost mountain covered by c_1. Using Lemma 6 we inductively obtain a χ-extensible diagram D for the subgraph G_Δ of G induced by taking Δ as an outer triangle together with all vertices inside, with uc_1 as a starting edge and w_{j+1} as a last vertex. Note that D appears upside down compared to Γ. But this 180° rotation is fine because down-up biarcs remain down-up biarcs when turned upside down. Let c' be the vertex that covers uc_1 in D.

The statement of Lemma 6 specifies two options. First we consider the case that a χ-valid diagram D for G_Δ is obtained. In order to plug D into Δ in Γ, we push down all mountains with left endpoint u or w_{j+1} in D and all mountains with left endpoint w_{j+1} in Γ; see Figure 17b. By (I2) for D and Γ these biarcs can be paid for using the corresponding mountain credits. We pay 2χ to initialize the two pockets incident to c' in D and $1-\chi$ for w_{j+1}, which is part of Γ already. Further, as we pay for uw_{j+1} as a part of D, we get a refund for the $1-\chi$ credits that according to (I6) are placed on uw_{j+1}. We also need to place one credit on the mountain w_ju. So the costs to add c_1 and c' are $2\chi + (1-\chi) - (1-\chi) + 1 = 1 + 2\chi \le 2(1-\chi)$, for $\chi \le 1/4$.

The other option in Lemma 6 is that we obtain a χ-extensible diagram D' for $G_\Delta \setminus \{w_{j+1}\}$ such that w_{j+1} is problematic for D'. Then we complete D' to a χ-valid diagram D for G_Δ as follows. As w_{j+1} has degree at least three in G_Δ, there are three possibilities. If w_{j+1} is $\mathcal{T}(3, \frown\smile)$ in D', then we just insert w_{j+1} into the pocket it covers, for a cost of one (for the incident mountain). Then we proceed exactly as described above for the first option of Lemma 6, noting that we can save the χ credits from the pocket incident to w_{j+1} in D, so that in fact the costs for inserting w_{j+1} into D' are $1-\chi$, as they should be. Otherwise, the vertex w_{j+1} is $\mathcal{T}(z, \frown^{z-1})$ in D', for $z \in \{3, 4\}$. We complete D' to D by placing w_{j+1} as the rightmost vertex on the spine and drawing the edges incident to w_{j+1} as mountains, in a similar fashion as for Lemma 3. Next, we plug D into Δ, making room by pushing down all mountains with left endpoint u in D and all mountains with left endpoint w_{j+1} in Γ; see Figure 18a. As for the costs, we pay 2χ credits to initialize the two pockets incident to c' in D and χ credits for the biarc uw_{j+1}, which already carries $1-\chi$ credits by (I6). We also pay one credit each for the mountains w_jc_1 and w_ju, and we can take one credit from one of the mountains covered by w_{j+1} in D that is not incident to u. So the costs to insert c_1 and c' are at most $2\chi + \chi + 2 - 1 = 1 + 3\chi \le 2(1-\chi)$, for $\chi \le 1/5$.

Otherwise, the edge $w_{j+1}u$ is a pocket. Using Lemma 6 we inductively obtain a χ-extensible diagram D for the subgraph G_Δ of G induced by taking Δ as an outer triangle together with all vertices inside, with $w_{j+1}u$ as a starting edge and c_1 as a last vertex; see Figure 18b. Let c' be the vertex that covers $w_{j+1}u$ in D. Then we plug D into Δ. The 2χ credits to initialize the two pockets incident to c' in D can be paid for by the credits that are on $w_{j+1}u$ by (I6). Then we need to pay one credit each for the mountains w_jc_1 and w_ju. But in return we can take the at least $1+\chi$ credits on the upper envelope of D because its edges are not incident to the outer face anymore after adding the edge uw_j. So by Lemma 6 the costs to add c' are $2-(1+\chi) = 1-\chi$,

(a) (b)

Figure 18: Exactly one eligible vertex in X_j, for $j < k$, and it has left pivot type.

or there exists an appropriate χ-extensible diagram D' for $G_\Delta \setminus \{c_1\}$ for which c_1 is problematic. In the latter case, we just plug D' into Γ and then push down $w_j w_{j+1}$ to place c_1 and draw all its edges to D' as mountains. None of these edges remain on the outer face after adding $w_j u$. So the costs are one for $w_j u$ and $1 \leq 2(1-\chi)$, for $\chi \leq 1/2$, for the two new vertices c_1 and c'.

It remains to consider the case $s > 1$. We place each vertex c_h, with $1 \leq h < s$, by pushing down the leftmost mountain it covers, and handle c_s in exactly the same way as in the case $s = 1$ described above. See Figure 19 for an example of the case where $\Delta \cap V \neq \emptyset$ and $w_{j+1}u$ is a mountain. The edge $w_j c_1$ is always a pocket, the only mountain that remains on the outer face is $w_j u$, and no additional biarc is created. Therefore, the same bounds on the costs as for $s = 1$ also hold for $s > 1$, and in fact decrease by $(s-1)(1-\chi)$, due to the larger number of vertices added.

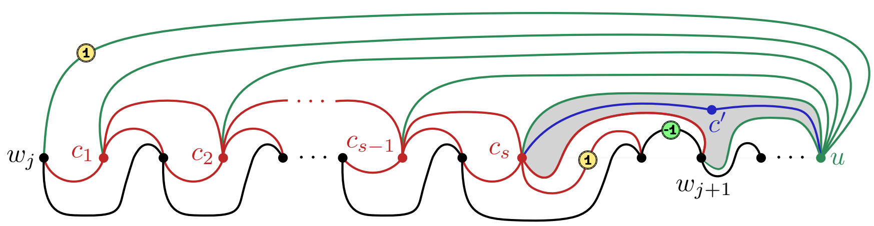


Figure 19: All ≥ 2 eligible vertices in X_j, for $j < k$, have left pivot type.

$\mathbf{X_j}$ is a right pivot region. By Lemma 11 there is exactly one vertex c_1 in $X_j \cap \mathcal{E}_{i-1}$ and $\text{pc}(c_1) = u$. Let Δ denote the open triangle $w_k c_1 u$. We push down the leftmost mountain covered by c_1, see Figure 20. If $\Delta \cap V = \emptyset$, then the costs are at most $1 - 3\chi$, which the new vertex c_1 can pay. Otherwise, we have $\Delta \cap V \neq \emptyset$ and using Lemma 3 we inductively obtain a χ-extensible diagram D for the subgraph of G induced by taking Δ as an outer triangle together with all vertices inside, with $w_j c_1$ as a starting edge and u as a last vertex. Then we plug D into Δ. Let c' be the vertex that covers $w_j c_1$ in D. The at least 2χ credits that are on $u w_{j+1}$ by (I6) can pay the initialization of the two pockets incident to c'. We can also take the χ credits from the pocket covered by c_1. We pay $1 - \chi$ credits for u in D. So the total costs to insert c_1 and c' are at most $1 - 2\chi < 2(1-\chi)$.

$\mathbf{X_j}$ is a both pivot region. By Lemma 11 there is exactly one vertex c_s of right pivot type in $X_j \cap \mathcal{E}_{i-1}$ and we have $\text{pc}(c_s) = u$. All vertices in $(X_j \cap V) \setminus \mathcal{E}_{i-1}$ are in the open quadrilateral $\square = c_{s-1} w' c_s u$. We argue in the same way as above in Section 3.3.1, except that now u is placed to the right of w_k and so we use Lemma 3 for the induction; see Figure 21.

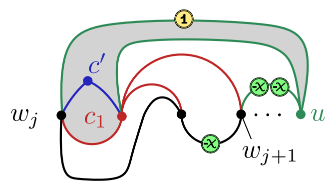


Figure 20: All eligible vertices in X_j, for $j < k$, have right pivot type.

(a) (b)

Figure 21: There are eligible vertices of both pivot types in X_j, for $j < k$.

4 Triangulations with many degree-three vertices

In this section we investigate the family of Kleetopes. They are probably the most well-known class of non-Hamiltonian planar graphs, and they have also been used by Cardinal, Hoffmann, Kusters, Tóth, and Wettstein [2, 3] to obtain their lower bound for the required number of biarcs in (not necessarily monotone) arc diagrams. In fact, we will show that we can tightly match the known lower bound algorithmically. Our result relies on the following theorem which holds for an even larger family of graphs, with many degree-three vertices.

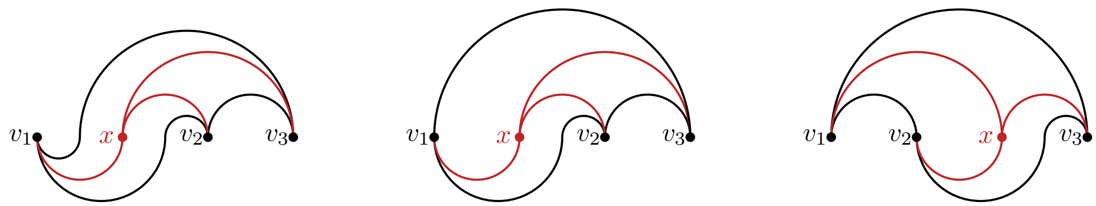
Theorem 4. *Let G be a triangulation with $n \geq 6$ vertices, and let d denote the number of degree-three vertices in G. Then G admits a monotone plane arc diagram with at most $n - d - 4$ biarcs, where every biarc is down-up.*

Proof: Let G be as in the statement, and let T denote the triangulation that results from removing all degree-three vertices from G. We will proceed in two steps. As a first step, we will prove below that for a triangulation on $n' \geq 4$ vertices, we can compute a monotone plane arc diagram with at most $n' - 4$ biarcs. To apply this algorithm to T we need $n' = n - d \geq 4$, that is, $d \leq n - 4$. The degree-three vertices form an independent set in G, and each degree-three vertex in G corresponds to a face of T. As T has $2(n - d) - 4$ faces, we have $d \leq 2(n - d) - 4$, which implies $3d \leq 2n - 4$. If $d \geq n-3$, we obtain $3(n-3) \leq 3d \leq 2n-4$, which implies $n \leq 5$. As we assume in the statement of the theorem that $n \geq 6$, we can thus conclude $n - d \geq 4$, as required.

In the first step, we obtain a monotone plane arc diagram for T with at most $n' - 4 = n - d - 4$ biarcs, where every biarc is down-up and such that every triangle t in the diagram crosses the spine. A triangle *crosses* the spine if its interior intersects the spine in at least one line segment (that is not just a single point). In the second step, we place all degree-three vertices of $G \setminus T$ in the drawing, each vertex on a spine segment of the triangle in T that contains it in G and connect it to each of the three vertices of the triangle by a proper arc[4]. This is always possible: Let $T = v_1v_2v_3$ such that v_1 precedes v_2, which again precedes v_3. First, assume that v_1v_3 is a down-up biarc and

[4]As the degree-3 vertices of G necessarily form an independent set, the three vertices of the triangle belong to T.

that w.l.o.g. the intersection of v_1v_3 with the spine precedes v_2 (the other case is symmetric by rotation by π). Then v_1v_2 is necessarily a pocket or a down-up biarc. Even if v_1v_2 is a down-up biarc and v_2v_3 is a mountain, there is a suitable space for x, namely between the two intersection of v_1v_3 and v_1v_2 with the spine; see Figure 22a. If v_1v_2 is a pocket or v_2v_3 is not mountain, the same position remains viable. Second, assume w.l.o.g. that v_1v_3 is a mountain (again the other case is symmetric by rotation by π). Then, at least one of v_1v_2 or v_2v_3 is a pocket or a down-up biarc. If v_1v_2 is a down-up biarc or a pocket, we place x directly after v_1; see Figure 22b. Observe that this works independently of the shape of v_2v_3. Similarly, if v_2v_3 is a down-up biarc or a pocket, we place x directly after v_2; see Figure 22c. Again, this works independently of the shape of v_1v_2. In either case, there no biarcs are created in the second step. Thus, it suffices to argue how to



(a) (b) (c)

Figure 22: Insertion of a degree-3 vertex x into a triangular face $v_1v_2v_3$ that intersects the spine.

obtain a diagram for T fulfilling Invariants (I1)–(I3) and (I5) in the first step. In fact, our drawing algorithm for T also maintains the following:

(I6) Every vertex on $C_\circ(G_i)$ is the left endpoint of at most one mountain, which necessarily occurs on $P_\circ(G_i)$.

We choose any canonical ordering $w_1, \dots, w_k$ for T. Then we start off by drawing the edge w_1w_2 as a pocket, into which we insert w_3 and draw the edge w_1w_3 as a pocket and the edge w_3w_2 as a mountain, onto which we place one credit. It is easily verified that this diagram satisfies (I1)–(I3), (I5) and (I6) and that each triangle in the diagram crosses the spine. Then we insert the vertices $w_4, \dots, w_k$ one by one while maintaining an arc diagram for $T_i = T[\{v_1, \dots, v_i\}]$ that satisfies (I1)–(I3), (I5) and (I6), for $i = 4, \dots, k$, along with the property that all triangles cross the spine. Note that for each biarc, both incident triangles cross the spine. Hence, pushing down a mountain maintains the spine crossing property. When inserting a new vertex v_i we distinguish two cases.

(a) (b)

Figure 23: Insert vertices so as to make every triangle cross the spine.

If v_i covers at least one pocket, then we place it into the rightmost pocket it covers; see Figure 23a. All edges to vertices to the left of v_i are drawn as mountains. Only the mountain to the leftmost neighbor requires a credit so as to establish (I2). We push down every mountain covered by v_i to the left of v_i, transforming it into a biarc. This transformation is paid for by the credit that is on the mountain by (I2) and sufficient since according to (I6) there is no other mountain with the same endpoint that must be pushed down. In this way, we ensure that all triangles to the left of v_i cross the spine. The edge to the immediate neighbor of v_i to the right is drawn as a pocket. If there are further neighbors of v_i to the right, then we draw the edge to the rightmost neighbor as a mountain and all other edges in between as biarcs. The costs for each such mountain or biarc are paid using the credit on the mountain underneath whose left endpoint is covered by v_i (all underneath edges are mountains because we insert v_i into the rightmost pocket it covers). In this way, we ensure that all triangles to the right of v_i cross the spine. Overall, the insertion of v_i costs one credit in this case.

Otherwise, all edges covered by v_i are mountains; see Figure 23b. We push down the leftmost such mountain to place v_i there. Again, according to (I6) there is no other mountain with the same endpoint that must be pushed down to allow this operation. The edge to the immediate neighbor of v_i to the left is drawn as a pocket. The edge to the rightmost neighbor of v_i is drawn as a mountain, on which we place one credit. All other edges (which are to vertices in between that are to the right of v_i) are drawn as biarcs. To pay for such a biarc we use the credit on the mountain underneath whose left endpoint is covered by v_i. In this way, we ensure that all triangles incident to v_i cross the spine. Overall, the insertion of v_i costs one credit in this case.

It is easily checked that the Invariants (I1)–(I3), (I5) and (I6) are maintained by the algorithm described above. Inserting each of $v_3, \ldots, v_k$ costs one credit, which is $k-2$ credits in total. Furthermore, we can (1) take the credit spent to insert v_k because this mountain remains a mountain in the final diagram and (2) observe that the edge on the outer face that is incident to v_2 is a mountain in the diagram for T_j, for all $3 \leq j \leq k$. In particular, as v_k has degree at least three in T, its insertion covers the mountain on the outer face of the diagram for T_{k-1}, and so we can also take back the credit on this mountain. Therefore, no more than $k-4$ credits are spent in total. As by (I3) every biarc in the diagram corresponds to a credit, the theorem follows. □

Theorem 2. *Every Kleetope on $n \geq 8$ vertices admits a monotone plane biarc diagram with at most $\lfloor (n-8)/3 \rfloor$ biarcs, where every biarc is down-up.*

Proof: Let G be a Kleetope on n vertices, and let d denote the number of degree-three vertices in G. By Theorem 4 the graph G admits a monotone plane arc diagram with at most $n-d-4$ biarcs, where every biarc is down-up. Removing the degree-three vertices from G we obtain a triangulation T on $n-d$ vertices, which by Euler's formula has $2(n-d)-4$ triangular faces. As G is a Kleetope, it is obtained by inserting a vertex into each of these faces, that is, we have $n = (n-d)+2(n-d)-4$ and thus $d = (2n-4)/3$. So there are at most $n-d-4 = (n-8)/3$ biarcs in the diagram. □

5 Planar 3-Trees

For 3-trees it is natural to follow their recursive construction sequence and build a corresponding diagram incrementally. A planar 3-tree G is built by starting from a (combinatorial) triangle. At each step we insert a new vertex v into a (triangular) face f of the graph built so far, and connect v to the three vertices of f. Every planar 3-tree G on at least four vertices is 3-connected. So its combinatorial embedding is unique, and for each triangle of the abstract graph we know whether it is facial or separating. In the former case, there is exactly one vertex of G that is adjacent to all vertices of the triangle, in the latter case there are exactly two such vertices. In particular, we can pick any facial triangle to be the starting triangle of our construction sequence for G and to become the outer face of our diagram.

Let $v_1, \dots, v_n$ be such a construction sequence for G. For $i \in \{3, \dots, n\}$, let $V_i = \{v_1, \dots, v_i\}$ and $G_i = G[V_i]$. Each vertex v_i, for $i \in \{4, \dots, n\}$, is inserted into a face $\mathrm{F}(v_i) = uvw$ of G_{i-1}, creating three *child faces* uvv_i, vwv_i and wuv_i of uvw in G_i. We also say that v_i is the *face vertex* $\mathrm{v}(uvw)$ of face uvw. We call a face f of G_i *active* if it has a face vertex in $V \setminus V_i$; otherwise, it is *inactive*. The *grand-degree* $\mathrm{gd}(f)$ is the maximum number of active child faces of f in all of $G_3, \dots, G_n$. Observe that by construction $\mathrm{gd}(f) \in \{0, \dots, 3\}$ and that f is active for some G_i if and only if $\mathrm{gd}(f) > 0$. Similarly, a vertex is a gd-i vertex, for $i \in \{0, 1, 2, 3\}$, if it is the face vertex of a face f with $\mathrm{gd}(f) = i$. For a construction sequence we define its dual *face tree* $\mathcal{T}$ on the faces of all G_i such that the root of $\mathcal{T}$ is $v_1v_2v_3$, and each active face uvw has three children: the faces uvz, vwz, and wuz, where $z = \mathrm{v}(uvw)$. Note that the leaves of $\mathcal{T}$ are inactive for all G_i. Let us first observe that no biarcs are needed if all faces have small grand-degree. To this end, also recall that G admits a plane proper arc diagram if and only if it is subhamiltonian and planar [1].

Theorem 5. *Let G be a planar 3-tree that has a construction sequence $v_1, \dots, v_n$ such that for each face f in its dual tree $\mathrm{gd}(f) \leq 2$. Then G is subhamiltonian planar; in particular, the graph G admits a plane proper arc diagram.*

Proof: We start by drawing the face $v_1v_2v_3$ as a *drop*, that is, a face where the two short edges are proper arcs on different sides of the spine; see Figure 24. Then we iteratively insert the vertices v_i, for $i = 4, \dots, n$, such that every face that corresponds to an internal vertex of the dual tree $\mathcal{T}$ is a drop in the diagram D_i for G_i. This can be achieved because by assumption at least one of the three faces of D_i created by inserting v_i is a leaf of $\mathcal{T}$, which need not be realized as a drop. But we can always realize the two other faces as drops, as shown in Figure 24. In this way, we obtain a diagram for G without any biarc. Since such a drawing exists only for subhamiltonian planar graphs [1], we can conclude that G is such a graph. □

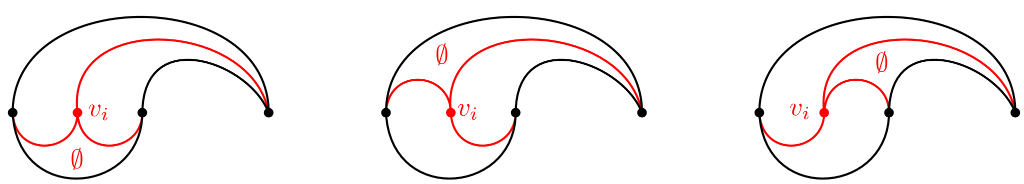


Figure 24: Insert a vertex v_i into a drop such that any chosen two of the faces created are drops.

As $\mathcal{T}$ is a tree, we can relate the number of internal vertices to the number of leaves.

Lemma 15. *Let f_d denote the number of faces in $\mathcal{T}$ with grand-degree exactly d, and let n_{inact} denote the number of face vertices that create inactive faces only. Then $n_{\mathrm{inact}} \geq 2f_3 + f_2$.*

Proof: Consider the rooted tree $\mathcal{T}'$ obtained by removing all leaves of $\mathcal{T}$, and observe that the grand-degree in $\mathcal{T}$ corresponds to the vertex degree in $\mathcal{T}'$. □

We are now ready to describe our drawing algorithm for general planar 3-trees.

Theorem 3. *Every planar 3-tree on $n \geq 3$ vertices admits a monotone plane biarc diagram with at most $\left\lfloor \frac{3}{4}(n-3) \right\rfloor$ biarcs, where every biarc is down-up.*

Proof: Our algorithm is iterative and draws G in the sequence prescribed by $\mathcal{T}$. Namely, at each step of our algorithm, we select an arbitrary already drawn face uvw and insert its face vertex $\mathrm{v}(uvw)$, possibly together with the face vertex of a child face. We will consider faces of a particular shape mostly. Consider a face $f = uvw$ such that u, v, w appear in this order along the spine and uw forms the upper envelope of f. (There is a symmetric configuration, obtained by a rotation by an angle of π where uw forms the lower envelope of f.) We say that f is *ottifant-shaped*[5] if it contains a region bounded by a down-up biarc between u and w, a down-up biarc between u and v and a mountain between v and w; see Figure 25a. Note the word "contains" in the definition of ottifant-shaped, which allows the actual face to be larger. For instance, the top boundary could be a mountain, but we treat it as if it was a biarc for the purposes of drawing edges; that is, we only connect to u from below the spine.

To control the number of biarcs drawn we maintain a charge $\mathrm{ch}(v)$ for each vertex v and a charge $\mathrm{ch}(f)$ for each face f. Whenever we draw a biarc, we place one unit of charge on it. This charge is taken from adjacent vertices and/or incident faces. We require additional flexibility from the edge vw of an ottifant-shaped face $f = uvw$, which we call the *belly* of f. To accommodate the insertion of a new vertex, we will sometimes transform the belly of a face into a biarc. Every mountain can be drawn as a biarc for only one credit. But in some cases redrawing a mountain as a biarc requires other adjacent mountains to be redrawn as a biarc as well. In our scenario, we allow a single adjacent edge to be redrawn. Having an extra reserve of half a credit on f turns out sufficient to cover these additional costs, as shown in the analysis below. Formally, a mountain vw with left endpoint v where vw is the belly of a face f is called *transformable* if either (1) vw is the unique mountain with left endpoint v or (2) there is exactly one other mountain vw' below vw and $\mathrm{ch}(f) = 1/2$. In the latter case, $\mathrm{ch}(f)$ will be used together with the charge of the vertex causing vw to cover redrawing both mountains vw and vw' as biarcs. Observe that a transformable belly vw can be redrawn as a down-up biarc (potentially together with its covered edge vw') for at most $3/2$ units of charge. In the description of the algorithm, we will explicitly state if two edges must be redrawn to push down (as in Section 3) the belly of an active face. In all other cases, it will be easy to verify that the belly can be redrawn as a biarc later.

More specifically, we maintain the following invariants:

(O1) Each internal active face is ottifant-shaped.

(O2) If the belly of an active face f is a mountain, it is transformable. If transforming the belly of f requires two edges to be redrawn, then $\mathrm{ch}(f) = 1/2$.

(O3) Every biarc has one unit of charge.

Note that symmetric invariants hold for ottifant-shaped faces where uw forms the lower envelope of f – that is, bellies are pockets that can be transformed by a "push-up" operation close to its right endpoint. While we have to maintain the invariant also for such faces throughout our algorithm, in the following, we focus the description of the case where uw forms the upper envelope. The symmetric operations are obtained by a rotation by π.

Charging the vertices. Before we describe the drawing algorithm, we first discuss how to assign charges to vertices. The total amount of charge is $\frac{3}{4}(n-3)$. Then (O3) implies the statement of the theorem. When inserting a vertex into the drawing, its charge is discharged to incident biarcs or faces.

[5]An *ottifant* is a cartoon abstraction of an elephant designed and popularized by the artist Otto Waalkes. Use of the term *ottifant* with kind permission of Ottifant Productions GmbH.

No charge is used for the first three vertices v_1, v_2, v_3 of the construction sequence. As a first step, we assign to each of the remaining vertices a charge of $\frac{3}{4}$. Then the total amount of charge is $\frac{3}{4}(n-3)$, as stated above. In a second step, we redistribute these charges while keeping the total amount of charge unchanged.

Specifically, we will see that no charge is needed when inserting a gd-0 vertex. Therefore, for each gd-0 vertex v we redistribute the charge $\frac{3}{4}$ of v as follows: (1) a charge of $\frac{1}{4}$ goes to the *parent* of v (i.e., the vertex $\mathrm{v}(f)$ of the parent f of $\mathrm{F}(v)$ in $\mathcal{T}$)—if it exists—and (2) a charge of $\frac{1}{2}$ goes to the so-called *preferred ancestor* $p(v)$ of v, as defined next. Preferred ancestors are determined by selecting an arbitrary surjective map p from the set of gd-0 vertices to the set of gd-2 and gd-3 vertices. According to Lemma 15 there exists such a map such that every gd-2 vertex is selected at least once and every gd-3 vertex is selected at least twice as a preferred ancestor. As a result, we obtain the following vertex charges:

(CH1) $\mathrm{ch}(v_1), \mathrm{ch}(v_2), \mathrm{ch}(v_3) \geq 0$,

(CH2) $\mathrm{ch}(v) = 0$ if $\mathrm{gd}(v) = 0$,

(CH3) $\mathrm{ch}(v) \geq \frac{3}{4} + (\mathrm{gd}(v) - 1) \cdot \frac{1}{2} \geq \frac{5}{4}$ if $\mathrm{gd}(v) \in \{2, 3\}$,

(CH4) $\mathrm{ch}(v) \geq \frac{3}{4}$ if $\mathrm{gd}(v) = 1$, and

(CH5) if $\mathrm{gd}(v) = 1$ and v is the parent of a gd-0 vertex, then $\mathrm{ch}(v) \geq 1$.

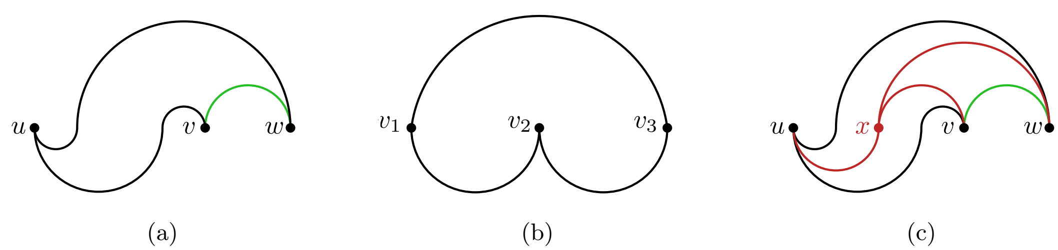


Figure 25: (a) An ottifant-shaped face uvw, where the long edge is on the top page (green edges are transformable). (b) Drawing of the initial face $v_1v_2v_3$. (c) Insertion of a gd-0 vertex $x = \mathrm{v}(uvw)$.

Initialization. We put $v_1v_2v_3$ on the spine in this order and draw the edges v_1v_2 and v_2v_3 as pockets and v_1v_3 as a mountain; see Figure 25b. The Invariants (O1)–(O3) hold.

Iterative step. We select an arbitrary active face $f = uvw$, which is ottifant-shaped by (O1), and insert its face vertex $x := \mathrm{v}(f)$ into f. Assume w.l.o.g. (up to rotation by an angle of π) that uw forms the top boundary of f. We make a case distinction based on $\mathrm{gd}(f)$.

Case 1: $\mathrm{gd}(f) = 0$. We insert x inside f between u and v on the spine, draw the edge ux as a pocket and xv and xw as mountains; see Figure 25c. Since all child faces of f are inactive, (O1) and (O2) trivially hold. Moreover, No biarcs are created, so (O3) holds.

Case 2: $\mathrm{gd}(f) \geq 2$. We insert x as in Case 1, except that xv is drawn as a biarc rather than as a mountain; see Figure 26a. All created child faces are ottifant-shaped (O1) and all bellies are transformable (O2) because they do not cover another mountain with same left endpoint (or another pocket with same right endpoint for face uxv, respectively). We created one biarc, to which we assign one unit of charge. So to establish (O3) it suffices to have $\mathrm{ch}(x) \geq 1$, which we can assume by (CH3).

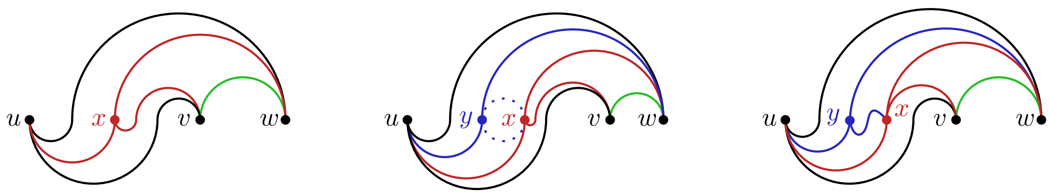


Figure 26: Insertion of a gd-2 vertex x (a) default case; (b) together with a gd-1 child y; (c) together with a gd-2 or gd-3 child y.

Case 3: $\mathrm{gd}(f) = 1$. Then only one of the three child faces of f is active. If uvx is the active child face, then we use the same drawing as for a gd-0 vertex (see Figure 25c) and all invariants hold. However, if one of the other child faces is active, then we cannot use this drawing because xw is not transformable and xvw is not ottifant-shaped.

So we also consider the face vertex y of the unique child face f' of x and insert both x and y into the drawing together. We consider two subcases, according to f'.

Case 3A: $f' = uxw$. If $\mathrm{gd}(f') = 0$, then we can once again use the drawing for a gd-0 vertex (see Figure 25c) because f' is ottifant-shaped and none of its child faces are active.

If $\mathrm{gd}(f') = 1$, then we add first x as described for a gd-2 vertex above (see Figure 26a). Then we add y into f' and draw all incident edges as proper arcs; the edge yx can be drawn either as a mountain (if uxy is the active child face of f') or as a pocket (otherwise); see Figure 26b. In either case, invariants (O1)–(O2) hold. We added one biarc (xv), to which we assign one unit of charge. Hence, to establish (O3) it suffices to have $\mathrm{ch}(x) + \mathrm{ch}(y) \geq 1$, which we can assume by (CH4).

Otherwise, we have $\mathrm{gd}(f') \geq 2$. We first add x as described above for a gd-0 vertex and then y as a gd-2 vertex; see Figure 26c. Invariant (O1) holds. To establish (O2) we have to make the bellies xw and uy of yxw and uyx, respectively, transformable. To this end, we assign $1/2$ units of charge to yxw so that both xv and xw can be redrawn as biarcs for $3/2$ units of charge, as required. Moreover, we observe that uy can be transformed into a biarc for 1 unit of charge if necessary as there is no other edge that must be transformed in this scenario. We also added a biarc, namely, yx. To establish (O3) it suffices to have $\mathrm{ch}(x) + \mathrm{ch}(y) \geq \frac{3}{2}$, which we may assume by (CH4).

Case 3B: $f' = xvw$. We consider several subcases according to $\mathrm{gd}(f')$. If $\mathrm{gd}(f') = 0$, we first insert x as described above for a gd-2 vertex and then y as a gd-0 vertex; see Figure 27a. Invariants (O1)–(O2) hold trivially. We used one biarc (xv) , to which we assign 1 unit of charge. Hence, to establish (O3), we observe that $\mathrm{ch}(x) = 1$ by (CH5), because x is the parent of y and y is a gd-0 vertex.

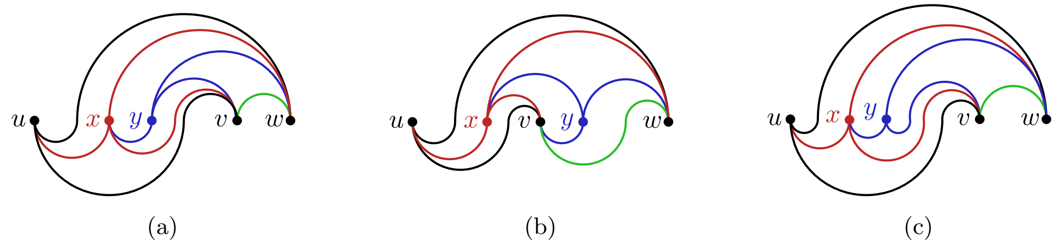


Figure 27: Insertion of a gd-1 vertex x together with its (a) gd-0-child y; (b) gd-1 child y; (c) gd-2 or gd-3 child y.

We use the same drawing if $\mathrm{gd}(f') = 1$ and the (only) active child face of f' is xvy or xyw. If xvy is active, then we require $\mathrm{ch}(x) + \mathrm{ch}(y) \geq 1$ to establish (O3), which we can assume by (CH4). If xyw is active, then we assign $1/2$ units of charge to xyw to make yw transformable and establish (O2). Then we need $\mathrm{ch}(x)+\mathrm{ch}(y) \geq \frac{3}{2}$ to establish (O3), which we can assume by (CH4).

If $\mathrm{gd}(f') = 1$, then it remains to consider the case that the (only) active child face of f' is yvw. We transform vw into a biarc, then insert x between u and v, and finally insert y between v and w on the spine inside f. All edges incident to x and y are drawn as proper arcs; see Figure 27b. The only active (grand)child face of f is yvw, and (O1)–(O2) hold. We have spent $3/2$ units of charge to transform vw, and we did not create any biarc by inserting new edges. Thus, $\mathrm{ch}(x) + \mathrm{ch}(y) = \frac{3}{2}$ suffices to establish (O3), which we can assume due to (CH4).

If $\mathrm{gd}(f') \geq 2$, then we first insert x between u and v and then y between x and v on the spine inside f. Then we draw xv and yv as biarcs and the remaining edges as proper arcs such that xy is a pocket; see Figure 27c. Invariants (O1)–(O2) hold. We created two biarcs (xv and yv) and assign one unit of charge to each of them. To establish (O3), it suffices to have $\mathrm{ch}(x) + \mathrm{ch}(y) \geq 2$, which can be assumed by (CH3) and (CH4).

It follows that (O1)–(O3) hold after each step . □

6 Conclusions

We proved the first upper bound of the form $c \cdot n$, with $c = 1 - \chi < 1$, for the number of monotone biarcs in arc diagrams of planar graphs. In our analysis, only some cases require $\chi \leq 1/5$, indicating a possibility to further refine the analysis to achieve an even better bound. It remains open whether there exists a "monotonicity penalty" in this problem, but we ruled out the probably most prominent class of non-Hamiltonian maximal planar graphs, the Kleetopes, as candidates to exhibit such a phenomenon. It would be very interesting to close the gap between upper and lower bounds, both in the monotone and in the general settings. Finally, we also ask how the problem of minimizing the number of biarcs in (monotone) arc diagrams can be solved algorithmically for restricted classes of planar graphs such as the planar graphs of bounded treewidth. Perhaps our approach in Section 5 can be improved to obtain optimal results.

Acknowledgements

This work started at the workshop on *Graph and Network Visualization* (GNV 2017) in Heiligkreuztal, Germany. We thank Stefan Felsner and Stephen Kobourov for useful discussions.

References

[1] Frank Bernhart and Paul C. Kainen. The book thickness of a graph. *J. Combin. Theory Ser. B*, 27:320–331, 1979. doi:10.1016/0095-8956(79)90021-2.

[2] Jean Cardinal, Michael Hoffmann, Vincent Kusters, Csaba D. Tóth, and Manuel Wettstein. Arc diagrams, flip distances, and Hamiltonian triangulations. In *Proc. 32nd Int. Sympos. Theoret. Aspects Comput. Sci. (STACS 2015)*, volume 30 of *LIPIcs*, pages 197–210, 2015. doi:10.4230/LIPIcs.STACS.2015.197.

[3] Jean Cardinal, Michael Hoffmann, Vincent Kusters, Csaba D. Tóth, and Manuel Wettstein. Arc diagrams, flip distances, and Hamiltonian triangulations. *Comput. Geom. Theory Appl.*, 68:206–225, 2018. doi:10.1016/j.comgeo.2017.06.001.

[4] Steven Chaplick, Henry Förster, Michael Hoffmann, and Michael Kaufmann. Monotone arc diagrams with few biarcs. In *Proc. 32nd Int. Sympos. Graph Drawing Network Visualization (GD 2024)*, volume 320 of *LIPIcs*, pages 11:1–11:16, 2024. doi:10.4230/LIPIcs.GD.2024.11.

[5] Marek Chrobak and Thomas H. Payne. A linear-time algorithm for drawing a planar graph on a grid. *Inform. Process. Lett.*, 54:241–246, 1995. doi:10.1016/0020-0190(95)00020-D.

[6] Hubert de Fraysseix, János Pach, and Richard Pollack. Small sets supporting Fary embeddings of planar graphs. In *Proc. 20th Annu. ACM Sympos. Theory Comput. (STOC 1988)*, pages 426–433, 1988. doi:10.1145/62212.62254.

[7] 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.

[8] Emilio Di Giacomo, Walter Didimo, Giuseppe Liotta, and Stephen K. Wismath. Curve-constrained drawings of planar graphs. *Comput. Geom. Theory Appl.*, 30(1):1–23, 2005. doi:10.1016/j.comgeo.2004.04.002.

[9] Hazel Everett, Sylvain Lazard, Giuseppe Liotta, and Stephen K. Wismath. Universal sets of n points for 1-bend drawings of planar graphs with n vertices. In *Proc. 15th Int. Sympos. Graph Drawing (GD 2007)*, volume 4875 of *LNCS*, pages 345–351, 2007. doi:10.1007/978-3-540-77537-9_34.

[10] Hazel Everett, Sylvain Lazard, Giuseppe Liotta, and Stephen K. Wismath. Universal sets of n points for one-bend drawings of planar graphs with n vertices. *Discrete Comput. Geom.*, 43(2):272–288, 2010. doi:10.1007/s00454-009-9149-3.

[11] Stefan Felsner, Hendrik Schrezenmaier, Felix Schröder, and Raphael Steiner. Linear size universal point sets for classes of planar graphs. In *Proc. 39th Int. Sympos. Computational Geometry (SoCG 2023)*, volume 258 of *LIPIcs*, pages 31:1–31:16, 2023. doi:10.4230/LIPICS.SOCG.2023.31.

[12] Emilio Di Giacomo, Walter Didimo, Giuseppe Liotta, and Stephen K. Wismath. Drawing planar graphs on a curve. In *Proc. 13th Int. Workshop Graph-Theoret. Concepts Comput. Sci. (WG 2003)*, volume 2880 of *LNCS*, pages 192–204, 2003. doi:10.1007/978-3-540-39890-5_17.

[13] Francesco Giordano, Giuseppe Liotta, Tamara Mchedlidze, Antonios Symvonis, and Sue Whitesides. Computing upward topological book embeddings of upward planar digraphs. *J. Discrete Algorithms*, 30:45–69, 2015. doi:10.1016/j.jda.2014.11.006.

[14] Michael Kaufmann and Roland Wiese. Embedding vertices at points: Few bends suffice for planar graphs. In *Proc. 6th Int. Sympos. Graph Drawing (GD 1998)*, volume 1547 of *LNCS*, pages 462–463, 1998. doi:10.1007/3-540-37623-2_47.

[15] Michael Kaufmann and Roland Wiese. Embedding vertices at points: Few bends suffice for planar graphs. *J. Graph Algorithms Appl.*, 6(1):115–129, 2002. doi:10.7155/jgaa.00046.

[16] Maarten Löffler and Csaba D. Tóth. Linear-size universal point sets for one-bend drawings. In *Proc. 23rd Int. Sympos. Graph Drawing Network Visualization (GD 2015)*, volume 9411 of *LNCS*, pages 423–429, 2015. doi:10.1007/978-3-319-27261-0_35.

[17] Hassler Whitney. Congruent graphs and the connectivity of graphs. *Amer. J. Math.*, 54(1):150–168, 1932. URL: https://doi.org/10.2307/2371086.