Untangling Planar Graphs and Curves by Staying Positive

Santiago Aranguri* Hsien-Chih Chang† Dylan Fridman‡

October 30, 2021

Abstract

Any generic planar closed curve with n crossings can be turned into a simple closed curve by applying $O(n^{3/2})$ homotopy moves without ever increasing the number of self-crossings; this improves over the $O(n^2)$ upper bound from Steinitz [*Ency. Math. Wiss. III 1916*], and matches the best lower bound. We prove the existence of a *positive move* that decreases the depth-sum potential at every step. Using similar techniques, we show that any 2-terminal plane graph with n vertices can be reduced to a single edge between the terminals using $O(n^{3/2})$ electrical transformations, consisting of degree-1 reductions, series-parallel reductions, and ΔY-transformations; this proves a conjecture of Feo and Provan that was open for more than 30 years.

1 Introduction

1.1 Problems and History

Any generic planar closed curve γ can be transformed into another generic planar closed curve by applying the following elementary moves:

- 1→0: Remove an empty *loop*.
- 2→0: Separate two subpaths that bound an empty *bigon*.
- 3→3: Flip an empty *triangle* formed by three subpaths; equivalently, move one subpath over the opposite crossing between the other two subpaths.

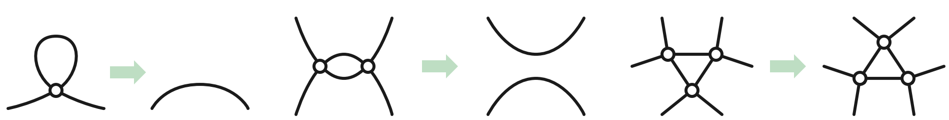

Figure 1. Homotopy moves 1→0, 2→0, and 3→3.

We call these local combinatorial changes together with their inverses the *homotopy moves*. The names of the moves are mnemonic, where the number before and after the arrow represents the local number of crossings before and after the move, respectively.

Homotopy moves are natural ways to discretize any continuous curve deformation (called a *homotopy*); such operations have been formally studied since Titus [29] but its usage can at least be traced back to Steinitz [20, 27, 28] in his proof that every 3-connected planar graph is the skeleton of a convex polytope. Naturally, homotopy moves serve as a complexity measure of how difficult it is to turn one

*Department of Mathematics, Stanford University.

†Department of Computer Science, Dartmouth College.

planar curve into another. Because every curve is homotopic to a simple circle in the plane and thus every two planar curves are homotopic, we can assume the target curve to be simple without loss of generality. We refer to the process of turning a planar curve into a simple circle as *simplification*.

A simplification process is *monotonic* if *no extra crossings are ever created* throughout. (In other words, a sequence of homotopy moves is monotonic if no 0→1 and 0→2 moves are allowed.) This is a natural and desirable property to have that fits our intuition for curve untangling—we can always untangle two strands that cross each other twice by pulling them apart.[1] A careful reading reveals that Steinitz's algorithm implemented such strategy (named the *bigon-reduction method*) by finding a bigon that is minimal in containment in the input curve which can then be emptied and removed using a sequence of monotonic homotopy moves. Let n be the number of self-crossings in the input curve. Every non-simple curve must contain a bigon, and each minimal bigon along with its two crossings can be removed using $O(n)$ moves; therefore Steinitz's algorithm simplifies any n-crossing generic planar curve using $O(n^2)$ monotonic homotopy moves. (The same $O(n^2)$ bound can also be derived from Francis' work on regular homotopy [17], if we do not require monotonicity.) A naïve $\Omega(n)$ lower bound can be observed from the fact that every homotopy move removes up to two crossings at a time. In 2015, Chang and Erickson [9] derived an $\Omega(n^{3/2})$ lower bound for both the monotonic and non-monotonic settings using a curve invariant[2] called *defect* [1, 4, 5], by demonstrating an n-crossing planar curve with defect $\Theta(n^{3/2})$. In their journal version [10] they provided a matching upper bound if the simplification process is *not* required to be monotonic. Since then, generalizations of the problem (say to higher-genus surfaces) have been studied [7, 8, 11]. But the exact complexity for the monotonic version of the planar curve simplification problem remained open.

Electrical transformations. From an algorithmic design point of view, enforcing the monotonicity requirement can be motivated by its intimate relationship to the well-studied graph operations called the *electrical transformations*, consisting of six operations in three dual pairs: *degree-1 reductions*, *series-parallel reductions*, and *ΔY transformation*, as shown in Figure 2. Electrical transformations were first introduced to analyze resistor networks [21, 25], and later on found applications as a general framework to solve combinatorial problems on planar graphs [2, 12, 13, 16, 18, 19, 22, 23].[3]

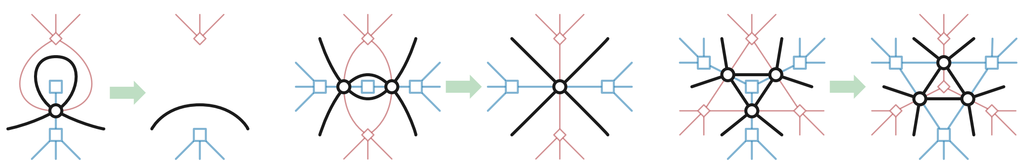

Figure 2. Electrical transformations and their corresponding medial electrical moves.

A quantitative relation between the electrical transformations and the (monotonic) homotopy moves was discovered [7, 10, 24] through the lens of the *medial construction*. Given a plane graph G, create a node for each edge in G and connect two nodes if the corresponding two edges share both a vertex and a face in G. This creates a 4-regular plane graph, which naturally decomposes into a collection of curves. From this perspective, the set of electrical transformations turns into a set of local operations on curves which looks almost identical to the set of homotopy moves. The main difference is that the series-parallel reductions turn into *2→1 move* that removes one single crossing from a bigon instead of two, while

[1]As opposed to knot diagrams, where there are examples that one has to *increase* the total number of crossings before making further progress (for example the *Culprit*).

[2]an integer-valued function on the curves that changes by a constant when applying homotopy moves; this is what computer scientists would call a *potential function*

[3]A fairly comprehensive list of citations can be found in Chang's thesis [6].

altering the number of constitute curves (in particular, medial electrical moves might transform a closed curve into a *multicurve*).

In many applications, one wants to solve some optimization problems on a planar graph between two specific vertices called *terminals*, say computing the effective resistance of a resistor network. This means that we do *not* want to remove these terminals at any time during the electrical reduction process. Such requirement can be carried out by adding *punctures* or *holes* to the corresponding faces of the medial multicurve in the plane, so that no electrical moves can be performed there. In this sense, the complexity of reducing 2-terminal plane graphs is closely tied to the number of homotopy moves required to simplify multicurves in the *annulus*. This is because the number of electrical transformations required to reduce a plane graph G is at least at many as the number of monotonic homotopy moves required to simplify the corresponding medial multicurve [10]; an extra term needs to be subtracted if we are in the 2-terminal case [7]. (See Section 4 for one particular subtlety about *terminal-leaf contractions*.) It is less clear whether any n-vertex plane graph can be reduced efficiently, if at all. When no terminals are presented, a quadratic upper bound can be extracted from the proof of Steinitz's convex polytope theorem [27, 28]. Akers [2] and Lehman [22] conjectured that any 2-terminal plane graph can be reduced to a unique edge between the two terminals, which was verified first by Epifanov [14]. Later work of Truemper [30] and Feo and Provan [15] provided more efficient methods to electrical reductions for both the 0- and 2-terminal cases. (In fact, all known algorithms for electrical reductions on plane graphs without terminals can be modified to work for the 2-terminal case.)

Feo-Provan conjecture. The algorithm of Feo and Provan [15] takes $O(n \cdot \mathit{depth}(\gamma))$ steps to electrically reduce any n-vertex 2-terminal planar graph G, where $\mathit{depth}(\gamma)$ is the depth of the deepest face in some medial multicurve γ of G. The correctness depends on the existence of a *positive* homotopy move such that performing the move will decrease the total sum of depths of the faces. While $\mathit{depth}(\gamma)$ can be as large as $\Omega(n)$ in the worst case, a typical planar graph (like a regular $\sqrt{n}$-by-$\sqrt{n}$ grid) has depth $O(\sqrt{n})$. It is tempting to guess that the worst-case quadratic bound is not tight; in fact, Feo and Provan themselves suspected that their algorithm/analysis can in fact be improved:

> "... there are compelling reasons to think that $O(|V|^{3/2})$ is the smallest possible order ...",

possibly referring to the earlier experimental results [16]. The same conjecture is repeated by Archdeacon, Colbourn, Gitler, and Provan [3], this time pointing to the potential improvements to Truemper's *grid-embedding method* [30]:

> "It is possible that a careful implementation and analysis of the grid-embedding schemes can lead to an $O(n\sqrt{n})$-time algorithm for the general planar case."

However, despite decades of research, the conjecture remained open until this day.

1.2 Our results

In this paper we present two main results. First, any generic n-crossing planar simple closed curve can be simplified using at most $O(n^{3/2})$ monotonic homotopy moves. Together with the worse-case $\Omega(n^{3/2})$ lower bound [9], we settle the answer to the exact complexity of monotonic planar curve simplification problem. Second, we show that any n-vertex 2-terminal plane graph can be reduced to a single edge using at most $O(n^{3/2})$ electrical transformations, and thus resolve the Feo-Provan conjecture affirmatively. Since every next electrical move can be found in $O(1)$ time using a heap data structure, the reduction process can be implemented in $O(n^{3/2})$ *time* as well.

Technical contribution. The technical heart of Feo and Provan's algorithm [15] is the *positive-move lemma*. When stated in the language of curves, it says that any non-tight closed curve γ must have a homotopy move that decreases the sum of the depths of all faces. The main obstacle in adapting the original proof to our setting is that Feo and Provan's proof crucially relies on the fact that the underlying multicurve is *closed* (that is, every constituent curve is a closed curve in the plane). In order to obtain the subquadratic bound we adapt the *useful tangle* framework from Chang and Erickson [10], by finding a tangle with lots of crossings inside compared to the number of strands. This way one can charge the number of moves performed within the tangle to the number of crossings removed in the end, and achieve a better amortized bound. (See Section 3.1.)

For this purpose we need the existence of positive moves within each useful tangle. However, because there are no canonical ways to define the depth function on the faces of a tangle, Feo and Provan's proof does not extend immediately—in fact, positive moves might not exist when the depth functions are not defined carefully. We overcome this obstacle by providing an alternative proof to the positive-move lemma, which can then be generalized to the tangle setting. To this end, we introduce a way to view the construction of multicurves from a different angle using the *contour representations*. By performing induction on the multicurves constructed based on the contours, we show that there is always a positive move within a tangle, unless the tangle is already tight. See Section 3.2 for an introduction to the contour representations, followed by a complete proof for the positive-move lemma in the tangle setting in Sections 3.3 and 3.4.

For the application of electrical reductions on 2-terminal plane graphs, we have to worry that the positive move guaranteed by the lemma in fact containing the puncture. Fortunately, with the usefulness assumption on the tangle, we can find a *backup* positive face that is not punctured. (More details can be found in Section 4.)

2 Preliminaries

A *closed curve* on surface Σ is a continuous map γ from a simple circle S^1 to Σ. An *open curve*, also referred to as a *strand*, is a continuous map γ from the interval $[0,1]$ to Σ, whose two endpoints $\gamma(0)$ and $\gamma(1)$ both lie on the boundary of the surface Σ. A *curve* is either an open or a closed curve. A *multicurve* γ on surface Σ is a collection of curves on Σ; each curve in the collection is a *constituent curve* of γ. We say a multicurve is *closed* if all its constituent curves are closed. Notice that when the surface Σ is the plane, all curves and multicurves must be closed. In this paper, all curves and multicurves are assumed to be *generic*, where each crossing must be a transverse double point. We will only consider surfaces that are subsets of the plane; in particular, we only talk about multicurves in either the *plane*, a disk, or a double-punctured plane (an *annulus*).

The image of any generic closed multicurve naturally corresponds to a 4-regular planar graph with a planar embedding (a *plane graph*), whose vertices are crossings of the multicurve and the edges are maximal subpaths between the vertices. We define the *faces* of a multicurve γ to be the components of the complement of γ as a subset of the surface. We define the *∞-face* to be the only unbounded face of γ. Define the *depth* of a face f of γ to be the minimum number of intersections between γ and any path from f to the ∞-face (counting multiplicity at the vertices). We say that two faces are *adjacent* if there is a path from one to the other that intersects γ exactly once. The *degree* of a face is the number of crossings of γ that are on the boundary of the face. An empty *loop*, a *bigon*, and a *triangle* are faces of degree 1, 2, and 3 respectively. Throughout the rest of the paper we omit the word "empty" for simplicity.

Tangle. We define a *tangle* of a planar multicurve γ to be the intersection of a closed topological disk T with γ imposing the condition that the boundary of the disk is transverse to γ and does not contain any

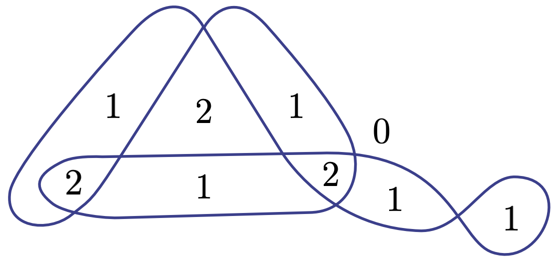

Figure 3. Multicurve with its faces labeled by their depths.

crossings (see Figure 4). A tangle T can be identified with a multicurve on a disk. The definition of depth of the faces in a tangle is more subtle and crucial to our results; we defer its definition and discussion to Section 3.2. We call a tangle *tight* if all strands are simple, and every pair of strands intersect at most once. Unlike a planar multicurve which can always be simplified in the plane, a multicurve within a tangle can at most be *tightened* (i.e., make it tight); not every crossing can be removed using only homotopy moves.

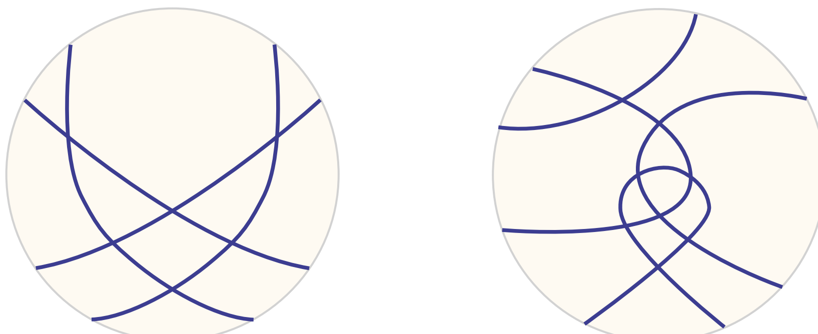

Figure 4. Left: a tight tangle. Right: a tangle that is not tight.

Positive moves. A face in the multicurve is *positive* if we can perform a homotopy move to remove or decrease its depth by one. We say that a homotopy move is *positive* if the corresponding face before the move is positive. Note that performing a positive move will decrease the total sum of the face depths, but the number of crossings might stay unchanged (for example, when the positive move performed is a 3→3 move). To state the result in full generality without worrying about whether the image of the input multicurve is connected or not, we will automatically perform a 0→0 *move* whenever an isolated closed simple curve shows up and remove it from the surface. This is necessary as otherwise an isolated circle in a face might prevent future homotopy moves to be carried out. Thus without loss of generality we can assume no isolated closed simple curves exist.

3 Monotonic Homotopy Moves

In this section we prove that it takes $O(n^{3/2})$ monotonic moves to simplify an n-crossing planar multicurve. First we review the *useful tangle* framework by Chang and Erickson [10] and reduce the problem to finding a positive move for both planar curves and tangles. Then we introduce the contour representation and its equivalent definition and structures in Section 3.2. We prove the positive-move lemma, first in the plane and then within a tangle with some extra assumptions, using contour representations in Section 3.3. Finally, in Section 3.4, we remove the extra assumptions.

3.1 Untangling Curves through Useful Tangles

A *useful tangle* of γ [10] is a tangle such that the number of crossings of γ inside T is at least quadratic in the number of strands. In other words, let m be the number of crossings in tangle T; then $m \geq s^2$, where σ is the boundary of T and $s := |\sigma \cap \gamma|/2$ is the number of strands in T. Notice that the boundary of a useful tangle σ can intersect an n-vertex multicurve γ in at most $O(\sqrt{n})$ places.

Lemma 3.1 (Chang and Erickson [10, Lemma 4.4]). *Let γ be an arbitrary planar multicurve with n vertices.*[4] *There is always a useful tangle of depth at most $O(\sqrt{n})$.*

Tightening all the strands inside a useful tangle T will reduce the number of crossings in γ by at least $m/2$ as any tight tangle has at most $\binom{s}{2}$ crossings. One can charge the homotopy moves spent on tightening the tangle to the removal of crossings. Therefore, to obtain an $O(n^{3/2})$ upper bound, it is sufficient to bound the number of homotopy moves by $O(\sqrt{n})$ times the number of crossings in the tangle.

Let T be a tangle and γ' be the corresponding multicurve inside T. Fix an arbitrary *basepoint* p_0 near the boundary of the tangle, disjoint from γ'. Define *depth*(p, γ') to be the minimum number of times a path from p to p_0 crosses γ'. Any two points within the same face of γ' have the same depth, and therefore the depth of a face *depth*(f) is well-defined. The depth of the tangle T, *depth*(T), is defined to be the maximum depth over all faces of γ'.

Lemma 3.2. *Any non-tight m-vertex tangle T can be tightened using at most $O(m \cdot \mathit{depth}(T))$ monotonic homotopy moves.*

We postpone the proof to Lemma 3.2 and use it first to prove the following theorem.

Theorem 3.3. *Every n-vertex multicurve γ can be simplified in $O(n^{3/2})$ monotonic moves.*

Proof: Let T be a (nontrivial) useful tangle of γ of depth $O(\sqrt{n})$ guaranteed by Lemma 3.1. Define s to be the number of strands of T and m to be the number of crossings in T. By Lemma 3.2, tightening T takes $O(m\sqrt{n})$ moves. Since T is useful, $m \geq s^2$. Also, every tight tangle has at most $\frac{s(s-1)}{2}$ crossings. Hence, tightening the tangle removes at least $m - \frac{s(s-1)}{2} \geq \frac{m}{2}$ crossings in at most $O(m\sqrt{n})$ moves. So each crossing takes amortized $O(\sqrt{n})$ moves to remove. If γ is a non-simple multicurve, we can iteratively find a useful tangle and tighten it until γ is simple. The process terminates because at each iteration we remove at least one crossing. As the process took amortized $O(\sqrt{n})$ moves to remove each intersection, multicurve γ can be simplified in at most $O(n\sqrt{n})$ moves in total. □

The rest of the section is devoted to a proof to Lemma 3.2.

3.2 Contour representation

Let T be a tangle with the corresponding multicurve γ in T. We also fix an arbitrary basepoint p_0 at the boundary of T and define the depth of the faces accordingly. For each integer j, define the region F_j to be the union of points inside tangle T whose depth is at least j. Each region might consist of several components, where the closures of two of them possibly share vertices (but not edges). The boundary of

[4] Chang and Erickson [10, Lemma 4.4] only stated the lemma for single closed curves; however, the proof directly generalizes to multicurves as well.

each component is either a simple closed curve or a boundary-to-boundary curve in T; we refer to the former as a *closed contour* and the latter as an *open contour*. Denote the region bounded by a contour κ *not* containing the basepoint p_0 as D_κ. The collection of contours for γ forms a nested family; contour κ is the ancestor of contour κ' if region D_κ properly contains $D_{\kappa'}$.

We use a *contour tree* $\mathcal{T}$ to represent this containment relationship. There might be multiple top-level contours not contained by anyone else even when the input multicurve is closed and connected. So we add an auxiliary node as the root of $\mathcal{T}$, whose children are these inclusion-wise maximal contours. More explicitly, each node (except for the root) corresponds to a contour κ and its children are precisely all the contours $\kappa_1, \ldots, \kappa_s$ immediately contained inside κ. The uniqueness of the contour tree depends on the choice of the basepoint p_0. We call κ the *parent contour* and $\kappa_1, \ldots, \kappa_s$ the *children contours*. We say that two or more contours are *siblings* if they have the same parent. We define the depth of a contour κ to be the depth of the face within D_κ with smallest depth.

Smoothings and shortcuts. *Smoothing* is the operation to replace an arbitrarily small neighborhood of a vertex v in a multicurve γ with two simple disjoint paths. The resulting object is still a multicurve. There are two possible smoothings at each vertex, based on how one reconnects the four endpoints at the boundary of the small neighborhood. The collection of contours of γ can be obtained by performing smoothings on every vertex based on the depths of the four incident faces by merging the two with minimal identical depths as illustrated in Figure 5.

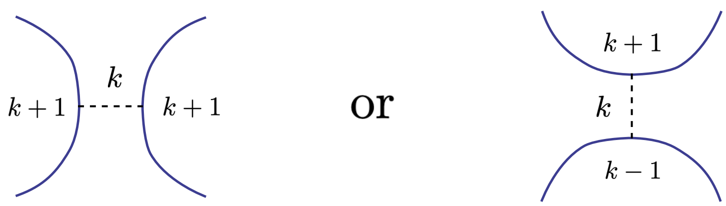

Figure 5. Smoothing vertices based on the depths of the four incident faces.

Note that two different multicurves can have the same collection of contours, since crossings are lost in the smoothing process. To keep all the information, we also draw segments joining two sibling contours or between parent-children contours, representing the crossings. Such segments are called *shortcuts*.

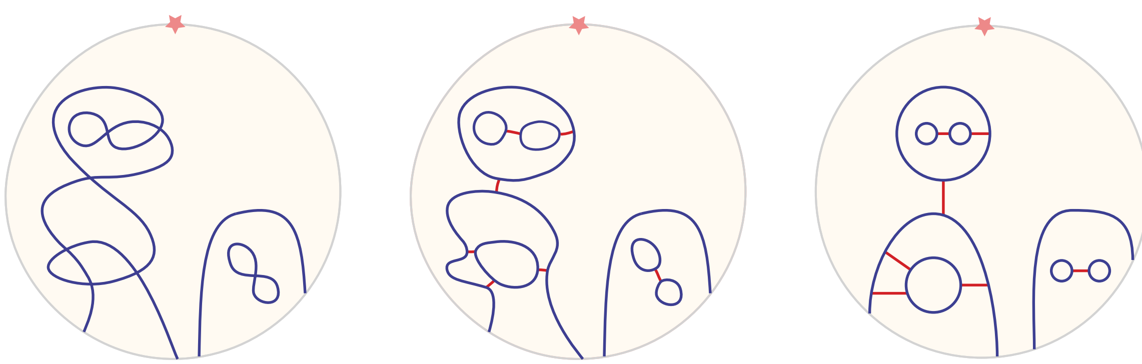

Figure 6. The star indicates the basepoint. Left: a tangle multicurve γ. Middle: contour representation of γ. Right: simplified contour representation of γ.

The *contour-shortcut multigraph* $\mathcal{S}$ is defined so that the nodes are the contours and edges are shortcuts between the contours. Given a contour-shortcut multigraph $\mathcal{S}$, one can reverse the smoothing process by collapsing the shortcuts into a point, transforming $\mathcal{S}$ back into a multicurve. Each shortcut then corresponds exactly to a single crossing in γ. The contours together with the shortcuts inherit a canonical planar embedding from the multicurve γ. We will refer to the contour-shortcut multigraph of γ together with its embedding as the *contour representation* of γ.

We emphasize that there might be multiple shortcuts between a pair of contours, and each of them is recorded in $\mathcal{S}$. Also, not every pair of parent-child contours in the contour tree $\mathcal{T}$ has shortcuts between them. (See Figure 6 for an example.)

Structure of contour representations. The benefit of treating a multicurve as adding shortcuts to a collection of contours is that we are able to consider intermediate multicurve objects as we perform induction on the contour tree $\mathcal{T}$. Formally, fix a contour tree $\mathcal{T}$ of a multicurve γ with respect to a fixed basepoint. For each node (except the root) with associated contour κ of $\mathcal{T}$, we construct the *multicurve associated with* κ, denoted as γ_κ, whose contour tree is exactly the subtree of $\mathcal{T}$ rooted at κ (after adding an auxiliary root). The vertices in γ_κ correspond to shortcuts that are in the region D_κ bounded by κ.

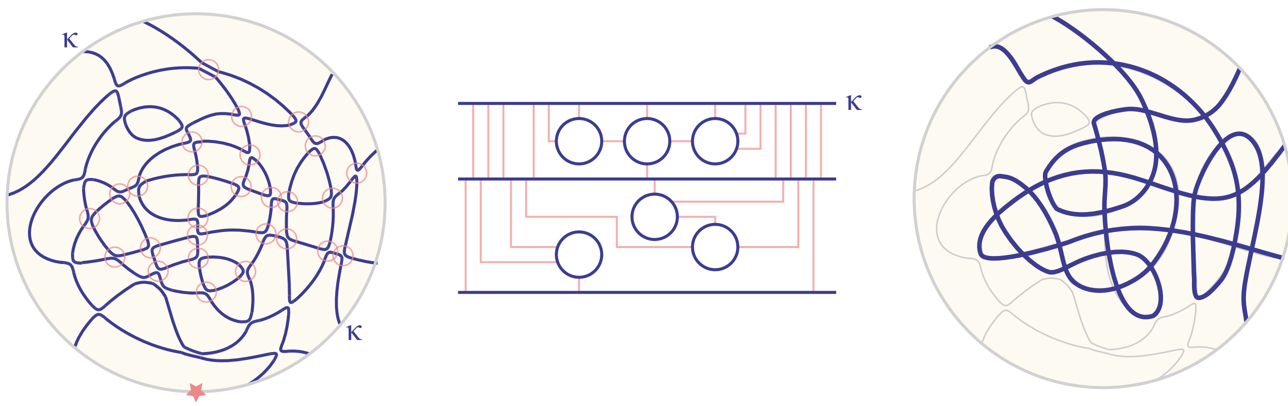

Figure 7. Left: contour representation at the subtree rooted at κ. Middle: contour-shortcut multigraph of γ_κ. Right: multicurve associated with contour κ.

Denote the subgraph of $\mathcal{S}$ induced on the children of the contour κ in $\mathcal{T}$ as $\mathcal{S}(\kappa)$. The shortcuts presented in $\mathcal{S}(\kappa)$ are precisely the sibling-sibling shortcuts between children of κ.

Fix a child κ' of κ. We define a *region* of κ' as a maximal subset of $D_\kappa \setminus D_{\kappa'}$ in which every pair of points has a path connecting them that does not cross a shortcut between κ and κ' (see Figure 8 for an example). This definition has an exception: whenever we have that κ and κ' are open contours and there is at least one shortcut between κ and κ', there are going to be at exactly two regions of κ' having exactly one shortcut on the boundary. In this case, we will replace those two regions by a single region that is the union of the two. We will call this single region the *last region*. Fix a region of κ' and two shortcuts between κ' and κ. We say that the region is *bounded* by the two shortcuts if the two shortcuts are on the boundary of the region. Note that there can be two different regions bounded by the same two shortcuts. We say κ' is *outermost* if it has at most one non-empty region and it has at most one sibling-sibling shortcut. Note that if κ' is the only contour in $\mathcal{S}(\kappa)$, it will be trivially outermost.

Lemma 3.4. *Let κ be an arbitrary contour.*

(a) The subgraph $\mathcal{S}(\kappa)$ forms a forest.

(b) There is no path of shortcuts connecting one open contour to another open contour in $\mathcal{S}(\kappa)$.

(c) If κ is closed with at least two children, then $\mathcal{S}(\kappa)$ has at least two outermost contours.

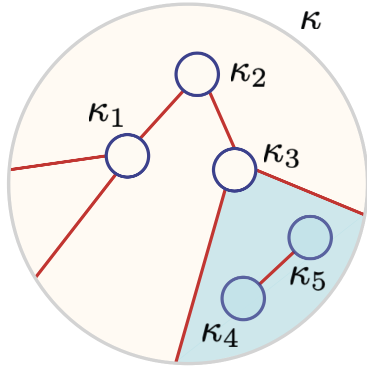

Figure 8. A region of κ_3 is colored blue. The only outermost components in $\mathcal{S}(\kappa)$ are κ_1, κ_4, and κ_5.

(d) If κ is open with at least one closed children, then $\mathcal{S}(\kappa)$ has at least one closed outermost contour.

Proof: We will prove items (a)–(d) in order.

(a) If there is a cycle in $\mathcal{S}(\kappa)$, consider the subset of contours $\kappa_1, \ldots, \kappa_s$ together with the shortcuts connecting them into a cycle Γ. A closed curve can be found in the image of the union of such contours and shortcuts. Therefore, by definition, no faces within the closed curve should have the same depth as the faces in between κ and its children. This is a contradiction to the way we resolve the shortcuts in Γ into vertices (the two faces incident to a shortcut should have the same depth).

(b) Again, if there is a path Π connecting two open contours in $\mathcal{S}(\kappa)$, then we can draw a boundary-to-boundary curve along the contours on path Π and the shortcuts, and the faces on the side of the boundary-to-boundary curve that do not contain the basepoint must have larger depths, which is a contradiction to how the shortcuts were constructed.

(c) All contours we consider here are closed since they are children of a closed contour. Let κ_1 be any leaf contour in $\mathcal{S}(\kappa)$. (By *leaf contour*, we mean a contour that is a leaf in its component in $\mathcal{S}(\kappa)$, which is a tree by (a).) Note that a leaf contour that is not outermost will have at least two non-empty regions. At least one region of κ_1 is non-empty since $\mathcal{S}(\kappa)$ has at least two contours.

Suppose that exactly one of the regions of κ_1 is non-empty. Then, κ_1 is outermost. Now, pick a leaf contour κ_2 in the non-empty region of κ_1. If κ_2 is not outermost, we run the following inductive process.

- Inductive hypothesis: κ_n is a leaf contour in $\mathcal{S}(\kappa)$ fulfilling (a) all $\{\kappa_1, \kappa_2, \ldots, \kappa_{n-1}\}$ lie in the same region of κ_n and (b) at least two regions of κ_n are non-empty.
- Inductive step: We pick κ_{n+1} to be any leaf contour in $\mathcal{S}(\kappa)$ inside the non-empty region R_n of κ_n that does not contain $\{\kappa_1, \kappa_2, \ldots, \kappa_{n-1}\}$. If only one region of κ_{n+1} is non-empty, then κ_{n+1} is outermost. If instead at least two regions of κ_{n+1} are non-empty, we go to the next step of the induction. For doing so, we have to show that $\{\kappa_1, \kappa_2, \ldots, \kappa_n\}$ lie in the same region of κ_{n+1}. Note that the intersection between the boundary of the region R_n and the boundary of D_κ forms a segment. Among the shortcuts between κ_{n+1} and κ, let s be the shortcut that is closest to one of the endpoints of the segment and s' the one that is closest to the other endpoint. (Note there are at least two shortcuts between κ_{n+1} and κ since κ_{n+1} is not outermost.) Now, note that $\{\kappa_1, \kappa_2, \ldots, \kappa_n\}$ lie in a region bounded by s and s', as desired.

This process must stop since there are finitely many contours and the κ_i we built are all different. The process stopping means we get an outermost contour. Thus, in the case that exactly one of the regions of κ_1 is non-empty, we get the two outermost contours.

Now suppose instead that at least two of the regions of κ_1 are non-empty. Pick two non-empty regions of κ_1 and run the following process on each. We pick a leaf contour κ_2 in $\mathcal{S}(\kappa)$ from a non-empty region of κ_1. If κ_2 is not outermost, we run the previous inductive process. It is easy to see that the outermost contour coming from the induction is inside the region we started with. Since we ran the process in two different regions of κ_1, there are two different outermost contours, as desired.

(d) Fix a closed leaf contour κ_1 in $\mathcal{S}(\kappa)$. If κ_1 is outermost, we are done. So we assume κ_1 has two non-empty regions. Consider a non-empty region R of κ_1 that is not the last region. As R is non-empty, it contains a leaf contour, κ_2 in $\mathcal{S}(\kappa)$. As R is not the last region, κ_2 is closed (since open contours can only lie in the last region). If κ_2 is outermost, we are done. If it is not, we can run the inductive process from (c). This is valid since all the contours κ_i considered in the induction will be inside R which means they will be closed. This inductive process gives us a closed outermost contour, as desired. □

3.3 Existence of positive moves

Given a pair of parent-child contours κ and κ', we say that two shortcuts between κ and κ' are *consecutive* if they, together with κ' and κ, form the boundary of some region of κ'.

We call a face *exposed* if it is adjacent to the ∞-face, and *protected* if it is not exposed. Notice that all faces in a contour κ that have the same depth as κ in γ become exposed in γ_κ. Also, if there are two contours κ and κ' in $\mathcal{T}$ where κ is the parent of κ', then any protected face in $\gamma_{\kappa'}$ must be protected in γ_κ as well. Now, assume we have an exposed face in $\gamma_{\kappa'}$ of degree 2 (an *exposed bigon*). If there are no shortcuts connecting the exposed bigon to κ or its sibling contours, then it becomes a protected bigon in γ_κ. Adding a single shortcut from the exposed bigon to κ turns the bigon into a protected positive triangle in γ_κ. Adding two consecutive shortcuts from the exposed bigon to κ then *transfers* the exposed bigon from $\gamma_{\kappa'}$ to γ_κ *assuming the condition that* $\mathcal{S}(\kappa)$ *has exactly one contour*. As we will see in Proposition 3.3(c), this condition can be weakened to that of κ' being an outermost contour if we allow ourselves to lose an exposed bigon. We will rely on this transfer operation extensively in our proofs.

Let κ and κ' be a pair of parent-child contours.

(a) Every protected positive face of $\gamma_{\kappa'}$ must also be a protected positive face of γ_κ.

(b) If $\mathcal{S}(\kappa)$ has exactly one contour, then any exposed bigon in $\gamma_{\kappa'}$ either becomes a protected positive face or transfers to γ_κ.

(c) If κ' is an outermost contour, then any exposed bigon in $\gamma_{\kappa'}$ except possibly one either becomes a protected positive face or transfers to γ_κ.

Proof: Items (a) and (b) follow from the discussion above. Let us look at item (c). If there is any exposed bigon that does not become a protected positive face and does not transfer to γ_κ, then that means that there are at least two shortcuts from the exposed bigon to κ and those two shortcuts bound a non-empty region. Hence, by the definition of outermost, all other regions are empty. It is easy to see that for any other exposed bigon, it either becomes a protected positive face or transfers to γ_κ. □

The previous proposition motivates the definition we gave for an outermost contour as follows. Suppose that κ and κ' are a parent-child pair as before, that κ' has an exposed bigon, and that there are two consecutive shortcuts connected from the exposed bigon to κ. Then the exposed bigon cannot be transferred from κ' to κ if the regions bounded by the two shortcuts are "blocked" (i.e., are non-empty). So, we need most regions bounded by the shortcuts between κ' and κ to be empty.

Now we are ready to prove the existence of positive moves. First, we present the simpler case when the contour representation of the multicurve only consists of closed contours. We need some notation to state the lemma. We say that a multicurve γ *fulfills* (α, β) if it has at least α protected positive faces and

β exposed bigons. Further, we say that γ *fulfills* (L) if it has one protected loop, and γ *fulfills* (S) if it is simple.

Lemma 3.5. *Every closed multicurve γ fulfills at least one of the following:* (S), (L), $(2,0)$, $(1,2)$, $(0,4)$.

Proof: We prove the lemma by induction on the height of the contour tree $\mathcal{T}$ of γ. Let κ be an arbitrary contour in $\mathcal{T}$. We show γ_κ satisfies the statement, assuming the same holds for the multicurves associated with all the descendants of κ. If γ_κ is simple then it fulfills (S). Otherwise, we separate into the case when $\mathcal{S}(\kappa)$ has exactly one contour and when $\mathcal{S}(\kappa)$ has at least two outermost contours based on Lemma 3.4(c).

Suppose $\mathcal{S}(\kappa)$ has exactly one contour, call it κ'.

- Suppose $\gamma_{\kappa'}$ fulfills (L) or $(2,0)$, then so does γ_κ since the faces will remain protected.
- Suppose $\gamma_{\kappa'}$ fulfills (S). We further divide into subcases based on σ, the number of shortcuts between κ and κ'. Note that $\sigma = 0$ cannot happen because we assume in Section 2 that there are no isolated simple closed curves; $\sigma = 1$ implies that γ_κ fulfills (L); $\sigma \in \{2,3\}$ implies that γ_κ fulfills $(1,2)$; $\sigma \geq 4$ implies that γ_κ fulfills $(0,4)$.
- Suppose $\gamma_{\kappa'}$ fulfills $(1,2)$. By Proposition 3.3(b), either one of the two exposed bigons becomes protected so that γ_κ fulfills $(2,0)$ or both exposed bigons transfer to γ_κ so that γ_κ fulfills $(1,2)$.
- Suppose $\gamma_{\kappa'}$ fulfills $(0,4)$. By Proposition 3.3(b), there are three cases. (a) γ_κ fulfills $(2,0)$ by protecting two faces, (b) γ_κ fulfills $(1,2)$ by protecting one face and transferring two exposed bigons, or (c) γ_κ fulfills $(0,4)$ by transferring all four exposed bigons.

Now suppose that $\mathcal{S}(\kappa)$ has at least two outermost contours, call two of them κ_1 and κ_2. If either γ_{κ_1} or γ_{κ_2} fulfills $(2,0)$ or (L) then so does γ_κ since the faces will remain protected by Proposition 3.3(a). Otherwise, we show that if each γ_{κ_i} fulfills (S), $(1,2)$, or $(0,4)$, then γ_κ fulfills $(2,0)$, $(1,2)$ or $(0,4)$. It suffices to show that for each $i \in \{1,2\}$, either γ_{κ_i} has a protected positive face or two exposed bigons are created/transferred to γ_κ.

- Suppose γ_{κ_i} fulfills (S). By definition of outermost contour, κ_i is a leaf. So it has at most one shortcut to its sibling contours. Thus, there needs to be at least three shortcuts between κ_i and κ for γ_κ to not protect a face. Adding at least three such shortcuts assures us that κ_i has at least three regions. At least two are empty since κ_i is an outermost contour. So at least two exposed bigons will be transferred to γ_κ if γ_κ does not protect a face.
- Suppose γ_{κ_i} fulfills $(1,2)$. Then γ_{κ_i} contains a protected positive face by Proposition 3.3(a).
- Suppose γ_{κ_i} fulfills $(0,4)$. Because κ_i has at most one shortcut to the siblings, γ_{κ_i} has at least three exposed bigons that have to be transferred to γ_κ for them to not be protected. So by Proposition 3.3(c) there are at least two exposed bigons that will either be protected or transferred.

In either of the cases given by Lemma 3.4(c), the statement of the Lemma 3.5 is satisfied. □

Corollary 3.6. *Every non-simple closed multicurve γ has a positive face.*

Proof: The corollary holds in all cases of Lemma 3.5, because any exposed bigon in γ is positive. □

Now we can extend the positive-move lemma to arbitrary tangles.

Lemma 3.7. *Let γ be a multicurve in a tangle with contour tree $\mathcal{T}$. If γ_{κ_0} has an exposed bigon for some contour κ_0 with depth smaller than the minimum depth of a closed contour of $\mathcal{T}$, then γ has an exposed bigon or a protected positive face.*

Proof: We prove by induction on the height of the contour tree $\mathcal{T}$. Let κ' be an arbitrary contour in $\mathcal{T}$ with parent κ such that κ_0 is a descendant of κ'. If $\gamma_{\kappa'}$ has a protected positive face, so does γ_κ. If $\gamma_{\kappa'}$ has an exposed bigon, it must be transferred to γ_κ since κ' has no closed contours as siblings to block the transfer operation. □

Lemma 3.8. *Let γ be a multicurve in a tangle with contour tree $\mathcal{T}$. If there are closed contours in $\mathcal{T}$, γ has a positive move.*

Proof: Let κ be the parent of a closed contour with minimal depth. We first show that γ_κ has a protected positive face or an exposed bigon. Note that κ must be open. Thus, Lemma 3.4(d) gives us a closed outermost contour κ' in $\mathcal{S}(\kappa)$. Applying Lemma 3.5 to $\gamma_{\kappa'}$ gives us that it fulfills one of (S), (L), $(2,0)$, $(1,2)$, or $(0,4)$;

- Suppose $\gamma_{\kappa'}$ fulfills (L), $(2,0)$, or $(1,2)$, then γ_κ will have a protected positive face since the faces will remain protected by Proposition 3.3(a).
- Suppose $\gamma_{\kappa'}$ fulfills (S). As κ' has at most one shortcut to other siblings, we need at least three shortcuts to κ to prevent the face from becoming a protected positive face, which forces γ_κ to have an exposed bigon.
- Suppose $\gamma_{\kappa'}$ fulfills $(0,4)$. By Proposition 3.3(c), there are two cases: either γ_κ protects a positive face or γ_κ transfers all but one of the exposed bigons.

If γ_κ has a protected positive face, then so does γ by repeatedly applying Proposition 3.3(a). Else, γ satisfies Lemma 3.7 with κ and therefore also has a positive face. □

3.4 Getting the first exposed bigon

What is left is to argue that in a non-tight tangle without closed contours, some open contour is always incident to an exposed bigon so the induction can be kicked off. For this purpose, we will introduce yet another alternative view to the contour representation for tangles consisting of only open contours. Choose an arbitrary basepoint p_0 and fix the depths of the faces; let d be the depth of the tangle. Prepare d horizontal lines at various heights called *levels*. For each j from 1 to d, draw every open contour of depth j on the j-th horizontal line as a line segment from left to right, sorted based on their appearance as we traverse the boundary of the tangle disk in counterclockwise order starting from the basepoint. (Because the contours are disjoint, without loss of generality one can take the first encounter point.) *We do not distinguish between a contour κ and its associated horizontal segment.* Between two adjacent levels j and $j+1$, we align the segments so that the child contours of depth $j+1$ have their segments lying below the segment corresponding to the parent contour at depth j. Now the shortcuts can be drawn as vertical *ladders* between two contours at neighboring levels. We call this the *ladder representation* of the tangle multicurve.

Every tangle strand can now be viewed as a path on the ladder representation between two horizontal segment endpoints. One important observation is that any such path must go monotonically from left to right, start at a left segment endpoint and keep following the segment to the right, turn at every ladder encountered to the corresponding neighboring segment until some right segment endpoint is reached.

The multicurve γ_κ associated with a contour κ corresponds to the contour representation given by all the segments at and below segment κ. A face in the ladder representation naturally corresponds to a bounded face in the tangle not incident to the boundary. Whenever we look at a segment κ, it is convenient to work with the multicurve associated with κ. For instance, when we talk about a face bounded by two consecutive ladders on segment κ' at level $j+1$ to some segment κ at level j such that

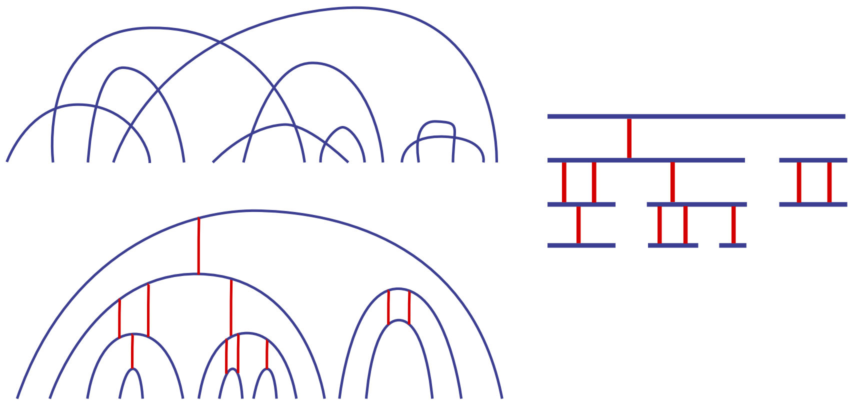

Figure 9. Top left: a tangle without closed contours which was deformed so that the endpoints of all strands lie in a line. Bottom left: contour representation of the tangle. Right: ladder representation of the tangle.

there are no ladders to the deeper levels in between, we will refer to the face as an *exposed bigon* (in the multicurve γ_κ) even when there might be other ladders in between leading upwards to level $j-1$.

Now let us prove the existence of an exposed bigon in a non-tight tangle.

Lemma 3.9. *Let γ be a non-tight tangle whose contour tree has no closed contours. Then there is an exposed bigon in γ_κ for some contour κ.*

Proof: Any non-tight tangle multicurve γ must contain either an empty loop (which is impossible because there are no closed contours) or a bigon formed by two strands α and β. Consider the two strands as paths in the ladder representation, which must meet at two different ladders. The observation above implies that neither path can be monotonically walking upwards or downwards because of the way a strand traces the ladder representation; in fact, one of the paths must turn at two ladders of maximal levels in consecutive and the other path must turn at two consecutive ladders of minimal levels. The face in between the two minimal level consecutive ladders, say between depth-j contour κ and depth-$(j+1)$ contour κ', corresponds to an exposed bigon in γ_κ because no ladder connecting to deeper levels are presented. This proves the lemma. □

Corollary 3.10. *Any multicurve γ in a non-tight tangle has a positive face.*

Proof: Let $\mathcal{T}$ be the contour tree of γ. If $\mathcal{T}$ has a closed contour, then the result follows from Lemma 3.8. If it has no closed contours, then Lemma 3.9 gives us an exposed bigon in some γ_κ and the result follows from Lemma 3.7. □

With Corollary 3.10 and the fact that the sum of the face depths of an m-vertex tangle T will not exceed $O(m \cdot \mathit{depth}(T))$, this concludes the proof of Lemma 3.2.

4 Electrical Transformations

In this section we resolve affirmatively the Feo-Provan conjecture, which says that every n-vertex 2-terminal plane graph can be electrically reduced in $O(n^{3/2})$ steps. The argument is presented completely from the medial multicurve perspective, and we will prove the equivalent statement that any n-crossing *connected* multicurve in the annulus can be simplified using $O(n^{3/2})$ medial electrical moves.

Terminal-leaf contraction. One subtlety about curve simplification in the annulus is that *not all curves can be completely simplified*. Such issue has been observed since the early work and was thoroughly discussed [7, 18]. In fact, a quadratic lower bound exists if we restrict ourselves to the setting where all the medial electrical moves have to perform on actual empty faces [7, 11]. Fortunately, since our problem is motivated by solving combinatorial problems on general planar graphs, we will allow one extra operation.

- *Terminal-leaf contraction* [15, 18]: Contract an edge incident to a degree-1 vertex that is a terminal. The neighbor on the other side of the edge becomes the new terminal.

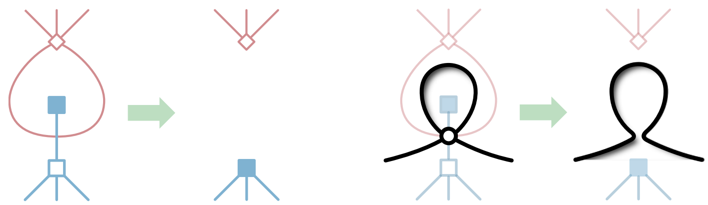

Figure 10. Terminal-leaf reduction.

From the curve perspective, terminal-leaf contraction is equivalent to performing a smoothing at an otherwise empty loop containing a single hole/puncture of the annulus. Using the terminal-leaf contraction one can again prove that every 2-terminal plane graph can be reduced to a single edge between the terminals. All the existing algorithms that perform electrical reductions on 2-terminal plane graphs always include terminal-leaf contractions as one of the allowed operations. On the flip side, an $\Omega(n^{3/2})$ lower bound still stands even when the terminal-leaf contractions are presented [7].

4.1 Backup exposed bigon

When there are no terminals, the whole argument for the case of monotonic homotopy moves applies here without much change. The main difference is that every 2→0 move needs to be replaced by a 2→1 move. The most important observation here is that both 2→0 and 2→1 moves decrease the sum of face depths, and the depth of the faces around the empty bigon do not change after a 2→1 move. Although 2→1 move changes the number of constituent curves, the argument from Section 3 works for arbitrary multicurves just fine.

When terminals are presented, the situation becomes more interesting as the positive face guaranteed by Corollary 3.6 and Corollary 3.10 can now be pinned by a puncture. In case that the first exposed bigon is pinned, we show that there is a second exposed bigon by utilizing the fact that the tangle is *useful* with lots of excess crossings. To see the reason why usefulness is necessary, consider the following instructional example in Figure 11.

The tangle given by the ladder representation in the example has s strands and $\sum_{1\le i<s} 2i = s(s-1)$ shortcuts (and thus not useful). If we consider the intermediate multicurve associate with each level j, there is always an exposed bigon in the middle, forming a "tower" from bottom to top and preventing the existence of positive faces (except at the top). Now let us put a puncture in the topmost exposed bigon. Suddenly all the "former" exposed bigons in the earlier levels are now rendered useless for the purpose of the induction proof for Lemma 3.5, because if we base the induction on any of these exposed bigons, we will eventually find the face with the puncture and no moves can be performed there. Formally, we say two exposed bigons (at two different levels $i < j$ of the ladder representation and thus in two different associated multicurve γ_{κ_i} and γ_{κ_j}) are *related* if the exposed bigon at level i was transferred iteratively from the exposed bigon at level j.

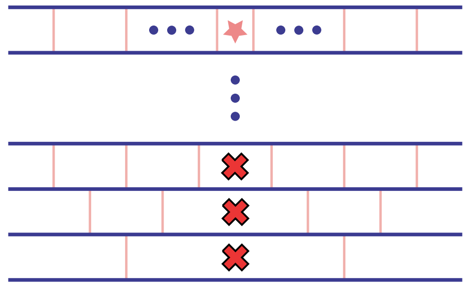

Figure 11. Ladder graph of a tangle where the puncture (drawn as a star) lying in the topmost exposed bigon kills off all the $s-1$ "former" exposed bigons below (marked with red crosses).

Fortunately, with some more extra shortcuts we can safely find a second exposed bigon that does not lead to the punctured face, for the open contour case.

Lemma 4.1. *Let γ be the multicurve of a useful tangle (that is, an s-strand tangle with at least s^2 crossings), whose contour representation has no closed contours. Then there is at least one exposed bigon not related to the punctured face.*

Proof: First we assume that there is exactly one contour at every level of the ladder representation. By assumption there are at least $s^2 > s(s-1)$ crossings in γ. This implies that there are two consecutive levels $j-1$ and j, such that level-j has at least *three* more ladders going up than level-$(j-1)$. Therefore in γ_{κ_j} we have at least *two* exposed bigons. Because the "related" relationship forms a forest, one of the two exposed bigons must be not related to the punctured face, and thus the lemma is proved.

Now we deal with the case when there is more than one contour appears in a level. Assume for contradiction that there are at most one exposed bigon. Then at every contour the number of ladders going up must be at most two plus the sum of the ladders going down to the children contours, so the total number of all ladders/crossings is again at most $s(s-1)$. (The top level has at most $2(s-1)$ ladders. Otherwise, by pigeonhole, there is a level with at least 3 more ladders than the previous level.) This contradicts to the fact that the tangle is useful. □

Closed contours. In the presence of closed contours, there are some extra complications that can be resolved by a careful analysis. We sketch a proof below and postpone the complete proof to the full version of this paper.

If there are two closed contours not containing each other, then we are essentially done by first finding an outermost child contour from each, apply Lemma 3.5 and Proposition 3.3 to either obtain a positive move or transfer an exposed bigon. The puncture can lie in one of the two closed contours and the other move/bigon will be available. In case when all closed contours are nesting, take the maximal closed contour κ' containing everyone else. The contour tree at κ' must be a path and all contours in κ' must be outermost by Lemma 3.4(d). Let the parent contour of κ' be κ, and let κ'' be the only sibling (open) contour κ' connects to (if exists). Consider the shallowest closed descendant contour κ^* of κ *not* containing the puncture.

- First assume κ^* exists. Apply Lemma 3.5 on κ^*. If there are any positive moves, they do not contain the puncture and thus we are done. If there are four exposed bigons, by Proposition 3.3(c) we have at least one transferred out without containing the puncture. If κ^* is simple, having at most three shortcuts to the parent induces a positive move; having at least four shortcuts transfers at least two exposed bigons. Either case we are fine.

- If there is no such contour κ^*, the puncture must lie in the deepest contour κ^*. By assumption the multicurve is connected, κ^* has at least one shortcut to its parent κ^{**}. If there is exactly one shortcut to κ^{**} and κ^{**} is closed, we apply terminal-leaf contraction. If there are at least two shortcuts to κ^{**} and κ^{**} is closed, κ^{**} has the only child and has two exposed bigons, which we can transfer out by Proposition 3.3(b) to the maximal closed contour κ', then transfers to κ by Proposition 3.3(c). If $\kappa^* = \kappa'$ and there are no shortcut to κ (and one shortcut to κ''), we apply terminal-leaf contraction. If $\kappa^* = \kappa'$ and there are at least two shortcuts to κ, there is an exposed bigon on κ.

  The only problematic case is when there is exact one shortcut to κ'' and one to κ. This means locally there are only two crossings. In the context of getting a second exposed bigon, such local configuration can only show up in a face of the ladder representation. If we requires a useful tangle to have more than $s(s-1)+2$ crossings then we are guaranteed to get a second exposed bigon unrelated to the puncture.

4.2 Proof of Feo-Provan conjecture

The proof of the following lemma is similar to Lemma 3.7 together with Lemma 3.8.

Lemma 4.2. *Let γ be a multicurve in a tangle, possibly containing a puncture. If some γ_{κ_0} has an exposed bigon unrelated to the puncture, then γ admits a positive move.*

Theorem 4.3. *Any connected annular n-crossing multicurve can be simplified using $O(n^{3/2})$ electrical moves and terminal-leaf contractions.*

Proof: Similar to Lemma 3.2 and Theorem 3.3, we will tighten each useful tangle T using at most $O(m \cdot depth(T))$ electrical moves and charge the number of moves spent to the number of crossings removed, and therefore each crossing takes amortized $O(\sqrt{n})$ moves to remove. When there are no useful tangles remain, we simplify the multicurve (now with $O(\sqrt{n})$ depth) using the original algorithm by Feo and Provan, which takes $O(n^{3/2})$ moves.

To argue that we can tighten each useful tangle using at most $O(m \cdot depth(T))$ electrical moves, we use Lemma 4.1 to find an exposed bigon not related to the punctured face. Plug in this exposed bigon into Lemma 4.2 gives us a positive face; and now the number of moves is bounded by the total sum of face depths, which is $O(m \cdot depth(T))$. □

Corollary 4.4. *Any connected n-vertex 2-terminal plane graph can be reduced using $O(n^{3/2})$ electrical transformations and terminal-leaf contractions.*

4.3 Concluding remark

In this section we resolve the Feo-Provan conjecture that every 2-terminal plane graph can be electrically reduced using $O(n^{3/2})$ electrical transformations. Feo and Provan never specify the order to resolve the positive moves when they stated the conjecture. Song [26] demonstrated that if you always take the shallowest positive move there are examples where it takes $\Omega(n^2)$ moves. Gitler, in his thesis [18], conjectured that the Feo-Provan algorithm uses at most $O(n^{3/2})$ moves if we always choose a positive move at the deepest level.[5] We take the liberty in interpreting the original conjecture and perform the positive moves *with respect to* the current useful tangle. The stronger claim by Gitler remains unsolved.

[5]"We have proposed that by imposing in the Feo-Provan algorithm that reductions should always be done at the highest contour possible and until it admits no other reduction could provide a [*sic*] $O(n\sqrt{n})$ version of the algorithm."

References

[1] Francesca Aicardi. Tree-like curves. In Vladimir I. Arnold, editor, *Singularities and bifurcations*, volume 21 of *Advances in soviet mathematics*, pages 1–31. Amer. Math. Soc., 1994.

[2] Sheldon B. Akers. The use of Wye-Delta transformations in network simplification. *Operations Research*, 8(3):311–323, June 1960.

[3] Dan Archdeacon, Charles J. Colbourn, Isidoro Gitler, and J. Scott Provan. Four-terminal reducibility and projective-planar wye-delta-wye-reducible graphs. *Journal of Graph Theory*, 33(2):83–93, 2000.

[4] Vladimir I. Arnold. Plane curves, their invariants, perestroikas and classifications. In Vladimir I. Arnold, editor, *Singularities and bifurcations*, volume 21 of *Adv. Soviet math.*, pages 33–91. Amer. Math. Soc., 1994.

[5] Vladimir I. Arnold. *Topological invariants of plane curves and caustics*, volume 5 of *University lecture series*. Amer. Math. Soc., 1994.

[6] Hsien-Chih Chang. *Tightening curves and graphs on surfaces*. Ph.D. Dissertation, University of Illinois at Urbana-Champaign, 2018.

[7] Hsien-Chih Chang, Marcos Cossarini, and Jeff Erickson. Lower bounds for electrical reduction on surfaces. In *35th International Symposium on Computational Geometry (SoCG 2019)*, volume 129 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 25:1–25:16, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[8] Hsien-Chih Chang and Arnaud de Mesmay. Tightening curves on surfaces monotonically with applications. February 2020.

[9] Hsien-Chih Chang and Jeff Erickson. Electrical reduction, homotopy moves, and defect. October 2015.

[10] Hsien-Chih Chang and Jeff Erickson. Untangling planar curves. *Discrete & Computational Geometry*, 58(4):889–920, December 2017.

[11] Hsien-Chih Chang, Jeff Erickson, David Letscher, Arnaud de Mesmay, Saul Schleimer, Eric Sedgwick, Dylan Thurston, and Stephan Tillmann. Tightening curves on surfaces via local moves. In *Proceedings of the 2018 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 121–135. January 2018.

[12] Charles J. Colbourn, J. Scott Provan, and Dirk Vertigan. A new approach to solving three combinatorial enumeration problems on planar graphs. *Discrete Applied Mathematics*, 60(1-3):119–129, June 1995.

[13] E. B. Curtis, D. Ingerman, and J. A. Morrow. Circular planar graphs and resistor networks. *Linear Algebra and its Applications*, 283(1):115–150, November 1998.

[14] G. V. Epifanov. Reduction of a plane graph to an edge by a star-triangle transformation. *Doklady Akademii Nauk*, 166:19–22, 1966.

[15] Thomas A. Feo and J. Scott Provan. Delta-Wye Transformations and the Efficient Reduction of Two-Terminal Planar Graphs. *Operations Research*, 41(3):572–582, June 1993.

[16] Thomas Aurelio Feo. *I. A Lagrangian Relaxation Method for Testing the Infeasibility of Certain VLSI Routing Problems II. Efficient Reduction of Planar Networks for Solving Certain Combinatorial Problems*. Ph.D. Dissertation, University of California, Berkeley, December 1985.

[17] George K. Francis. The folded ribbon theorem. A contribution to the study of immersed circles. *Transactions of the American Mathematical Society*, 141:271–303, 1969.

[18] Isidoro Gitler. *Delta-Wye-Delta transformations: algorithms and applications*. PhD thesis, University of Waterloo, 1991.

[19] Gramoz Goranci, Monika Henzinger, and Pan Peng. Improved guarantees for vertex sparsification in planar graphs. In Kirk Pruhs and Christian Sohler, editors, *25th Annual European Symposium on Algorithms (ESA 2017)*, volume 87 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 44:1–44:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[20] Branko Grünbaum. *Convex polytopes*. Number XVI in Monographs in pure and applied mathematics. John Wiley & Sons, 1967.

[21] Arthur Edwin Kennelly. Equivalence of triangles and three-pointed stars in conducting networks. *Electrical World and Engineer*, 34(12):413–414, 1899.

[22] Alfred Lehman. Wye-Delta transformation in probabilistic networks. *Journal of the Society for Industrial and Applied Mathematics*, 11(3):773–805, 1963.

[23] Hiroyuki Nakahara and Hiromitsu Takahashi. An algorithm for the solution of a linear system by Δ-Y transformations. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, E79-A(7):1079–1088, July 1996.

[24] S. D. Noble and D. J. A. Welsh. Knot graphs. *Journal of Graph Theory*, 34(1):100–111, 2000.

[25] Alexander Russell. The method of duality. In *A treatise on the theory of alternating currents*, chapter XVII, pages 380–399. Cambridge Univ. Press, 1904.

[26] Xiaohuan Song. *Implementation issues for Feo and Provan's delta-wye-delta reduction algorithm*. PhD Thesis, University of Victoria, 2001.

[27] Ernst Steinitz. Polyeder und Raumeinteilungen. *Enzyklopädie der mathematischen Wissenschaften mit Einschluss ihrer Anwendungen*, III.AB(12):1–139, 1916.

[28] Ernst Steinitz and Hans Rademacher. *Vorlesungen über die theorie der polyeder: unter einschluß der elemente der topologie*, volume 41 of *Grundlehren der mathematischen wissenschaften*. Springer-Verlag, 1934.

[29] Charles J. Titus. The combinatorial topology of analytic functions of the boundary of a disk. *Acta Mathematica*, 106(1–2):45–64, 1961.

[30] K. Truemper. On the delta-wye reduction for planar graphs. *Journal of Graph Theory*, 13(2):141–148, 1989.