A Parameterized Algorithm for Vertex and Edge Connectivity of Embedded Graphs

Therese Biedl
University of Waterloo, Canada

Prosenjit Bose
School of Computer Science, Carleton University, Ottawa, Canada

Karthik Murali
School of Computer Science, Carleton University, Ottawa, Canada

Abstract

The problem of computing vertex and edge connectivity of a graph are classical problems in algorithmic graph theory. The focus of this paper is on computing these parameters for graphs drawn on the plane. A typical example of such graphs are planar graphs which can be embedded without any crossings. It has long been known that vertex and edge connectivity of planar embedded graphs can be computed in linear time. Very recently, Biedl and Murali extended the techniques from planar graphs to 1-plane graphs without ×-crossings, i.e., crossings whose endpoints induce a matching. While the tools used were novel, they were highly tailored to 1-plane graphs, and do not provide much leeway for further extension. In this paper, we develop alternate techniques that are simpler, have wider applications to near-planar graphs, and can be used to test both vertex and edge connectivity. Our technique works for all those embedded graphs where any pair of crossing edges are connected by a path that, roughly speaking, can be covered with few cells of the drawing. Important examples of such graphs include optimal 2-planar and optimal 3-planar graphs, d-map graphs, d-framed graphs, graphs with bounded crossing number, and k-plane graphs with bounded number of ×-crossings.

2012 ACM Subject Classification Theory of computation → Graph algorithms analysis

Keywords and phrases Vertex Connectivity, Edge Connectivity, 1-planar, k-planar, Linear-time

Digital Object Identifier 10.4230/LIPIcs.ESA.2024.24

Related Version *Full Version*: `https://arxiv.org/abs/2407.00586` [3]

Funding Supported by NSERC.

1 Introduction

The "connectivity" of a graph is an important measure of the resilience of the network that it models, and mainly comes in two variants. The *vertex connectivity* of a graph G, denoted $\kappa(G)$, is the size of a minimum set of vertices whose removal makes the graph disconnected. Such a set of vertices is called a *minimum vertex cut* of G. The *edge connectivity* of an unweighted graph, denoted by $\lambda(G)$, is the size of a minimum set of edges whose removal makes the graph disconnected. Such a set of edges is called a *minimum edge cut* of G. For edge-weighted graphs, there is an analogous concept of *global mincut*, which is a set of edges with minimum weight whose removal disconnects the graph. There is a rich literature on algorithms for computing vertex connectivity and global mincut of general graphs (see for example [6, 12, 13, 15, 16, 18] and the references therein).

Planar graphs. In this paper, we focus on the problem of computing vertex (edge) connectivity of graphs that come with a drawing with special properties. An example of such a class of graphs is *planar graphs*, which are graphs that can be embedded on the plane with

32nd Annual European Symposium on Algorithms (ESA 2024).
Editors: Timothy Chan, Johannes Fischer, John Iacono, and Grzegorz Herman; Article No. 24; pp. 24:1–24:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

no edge crossing another. Because of their simplicity, planar graphs have nice structural properties and fast algorithms tailored to them. For example, it is known that the vertices (edges) that constitute a minimum vertex (edge) cut of a planar graph correspond to a cycle in graph $\Lambda(G)$ (defined below). This was used by Eppstein [11] to design an algorithm for computing the vertex (edge) connectivity of a planar graph in linear time. Since our paper uses some of his techniques, we briefly review them here.

Let G be an embedded planar graph. Let $\Lambda(G)$ be an auxiliary super-graph of G obtained by inserting a *face-vertex* inside each face of G and making it adjacent to all vertices of G that bound the face. Eppstein showed that a minimal vertex cut S (i.e. no proper subset of S is a vertex cut) corresponds to a certain *separating cycle* C of length $2|S|$ in $\Lambda(G)$. Since $\kappa(G) \leq 5$ for a planar graph, a shortest separating cycle exists in a subgraph of $\Lambda(G)$ with diameter at most 10. Using breadth-first-search layering, one can cover all subgraphs of G with bounded diameter subgraphs, while maintaining the total size of subgraphs to be in $O(n)$. As bounded diameter subgraphs of planar graphs have bounded treewidth, one can then do a dynamic programming based subgraph isomorphism algorithm to find such a cycle [11]. The algorithm runs in time $2^{O(\kappa \log \kappa)}n$, where $\kappa = \kappa(G) \leq 5$.

The method to compute edge connectivity of G is very similar. As a pre-processing step, first subdivide each edge of G to get a graph $\tilde{G}$. Let T be a minimum edge cut of G. Then the set of subdivision vertices $\tilde{T}$ on the edges of T form a minimal vertex cut of G, and therefore correspond to a shortest separating cycle of $\Lambda(\tilde{G})$ (with a slightly different definition of "separating"). One can search for the modified separating cycle in $2^{O(\lambda \log \lambda)}n$ time. The running time is linear if G is simple (in which case $\lambda = \lambda(G) \leq 5$), or if G has at most a constant number of edges connecting any pair of vertices (in which case $\lambda \in O(1)$).

1-planar graphs. An extension of planar graphs are *1-planar graphs*, i.e., graphs that can be embedded on the plane with at most one crossing per edge. In [5], Biedl and Murali generalize Eppstein's techniques for computing vertex connectivity to embedded 1-plane graphs without ×-crossings, which are crossings with no edges connecting the endpoints of the crossing (apart from the crossing edges themselves). The generalization is not straight-forward because minimum vertex cuts no longer strictly correspond to separating cycles in $\Lambda(G)$ (the graph $\Lambda(G)$ is now defined on the *planarized embedding* of G, i.e., the plane graph obtained by inserting a dummy-vertex at each crossing point). Despite this, they showed that a minimum vertex cut lies in some *bounded* diameter separating subgraph of $\Lambda(G)$, and that alone is sufficient for a tree decomposition based dynamic programming algorithm to work. Unlike for planar graphs, the techniques for vertex connectivity in [5] do not immediately extend to edge connectivity, because the graph obtained by subdividing every edge always unavoidably introduces an ×-crossing. Moreover, the methods used in [5] depend heavily on the 1-planar embedding of a graph, making it difficult to extend to other classes of embedded graphs.

Our results. The main contribution of this paper is in the development of novel techniques that when combined with the framework of [5] leads to a parameterized algorithm for computing vertex and edge connectivity of general embedded graphs[1]. The running time of the algorithm is linear in the size of the embedded graph (which includes edges, vertices and crossing points) and exponential in two parameters, one of which is the vertex (or edge) connectivity, and the other is the "ribbon radius" of the embedding, that is, a measure of the closeness of crossing pairs of edges in the embedding (see the formal definition in Section 3).

[1] In some contexts, the term "embedded" may refer to graphs drawn on surfaces without crossings. However, our use of the term "embedded" refers to any drawing of a graph on the plane, including drawings that feature crossings.

As a major first step, we show that if a graph has ribbon radius μ, then all vertices of a minimum vertex cut and edges of a minimum edge cut lie in a subgraph of $\Lambda(G)$ with diameter $O(\mu\kappa)$ and $O(\mu\lambda)$ respectively (Section 3). We then make use of the framework developed in [5] to design a parameterized algorithm to obtain a minimum vertex or edge cut of the graph (Section 4). The input to our algorithm will be an embedded graph G described in terms of its *planarized embedding*, i.e., the plane embedded graph $G^\times$ obtained by inserting a dummy-vertex at each of its crossing points. The idea is to construct the auxiliary graph $\Lambda(G)$, decompose it into subgraphs of diameter $O(\mu\kappa)$ or $O(\mu\lambda)$, and then run a dynamic programming algorithm on a tree decomposition of the subgraph. Theorem 1 gives the running time of the algorithm, and will be the main result of the paper.

▶ **Theorem 1.** *If G is an embedded graph with ribbon radius μ, one can compute $\kappa(G)$ and $\lambda(G)$ in time $2^{O(\mu\kappa)}|V(G^\times)|$ and $2^{O(\mu\lambda)}|V(G^\times)|$ respectively.*

In consequence, for graphs G with bounded ribbon radius and bounded connectivity, we can obtain a minimum vertex or minimum edge cut in $O(|V(G^\times)|)$ time. Several well-known graph classes such as graphs with bounded crossing number, d-map graphs, d-framed graphs, optimal 2-planar and optimal 3-planar graphs etc. have bounded ribbon radius (Section 5). This gives us an $O(n)$-time algorithm for testing both vertex and edge connectivity for these graph classes. We also briefly discuss how to compute the ribbon radius in Section 6, and conclude in Section 7.

2 Preliminaries and Background

(For basic definitions involving graphs, see [10].) A graph $G = (V, E)$ is said to be *connected* if for any pair of vertices $u, v \in V(G)$, there is a path in G that connects u and v; otherwise G is *disconnected*. We always assume that all graphs are connected, unless specified otherwise. A set of vertices $S \subseteq V(G)$ is a *vertex cut* if $G \setminus S$ is disconnected. A vertex cut of minimum size is called a *minimum vertex cut* of G, and its size is the *vertex connectivity* of G, denoted by $\kappa(G)$. Likewise, a set of edges T is called an *edge cut* if $G \setminus T$ is disconnected. For an unweighted graph G, an edge cut of minimum size is called a *minimum edge cut* of G, and its size is the *edge connectivity* of G, denoted by $\lambda(G)$. The minimum degree $\delta(G)$ is an upper bound for both $\kappa(G)$ and $\lambda(G)$; in fact $\kappa(G) \le \lambda(G) \le \delta(G)$. For a vertex cut S or an edge cut T, the connected components of $G \setminus S$ and $G \setminus T$ are called the *flaps* of S and T respectively. A set of vertices S *separates* two sets A and B in G if no flap of $G \setminus S$ contains vertices of both A and B. A set S is a *minimal vertex cut* if no subset of S is a vertex cut. Likewise, a set T is a *minimal edge cut* if no subset of T is an edge cut. If S is a minimal vertex cut, then each vertex of S has a neighbour in each flap of S. Similarly, if T is a minimal edge cut, each edge of T is incident to vertices from all flaps of T; in particular, this means that a minimal edge cut has exactly two flaps.

Embedded Graphs. We follow the conventions of [19] for a *drawing* of a graph: vertices are mapped to distinct points in the plane and an edge e becomes a homeomorphic mapping from $[0, 1]$ into the plane where $e(0)$ and $e(1)$ are the endpoints of the edge and $e(0, 1)$ does not contain any vertices. A *crossing* of two edges e_1, e_2 is a common point $c = e_1(s) = e_2(t)$ that is interior to both edges, i.e., $s, t \in (0, 1)$. We require that no third edge uses c, and that e_1 and e_2 *cross transversally* at c, i.e., in a sufficiently small disk around c there are no other crossings, and the curves alternate in the order around c. (We otherwise need no restrictions on the drawing; in particular, a pair of edges can cross repeatedly and edges with a common endpoint can cross each other.) The *endpoints* of a crossing c are the endpoints of e_1 and e_2; a pair of endpoints is said to be *consecutive* if one belongs to e_1 and the other to e_2. A drawing D of a graph G is *planar* if it has no crossing points. The maximal connected regions of $\mathbb{R}^2 \setminus D$ are called *faces* of the drawing. They can be described abstractly via a *planar rotation system*, which specifies a clockwise order of edges incident with each vertex. This defines the *face boundary walks*, which are the maximal closed walks where each next edge comes after the previous edge in the rotation system at the common endpoint. For a connected graph, the faces are in correspondence with these face boundaries, while for a disconnected graph the faces correspond to sets of face boundaries.

The *planarization* $G^\times$ of a drawing of an arbitrary graph G is obtained by inserting a *dummy-vertex* at each crossing point, making the resulting drawing planar. In this paper, we will use the terms dummy-vertex and crossing points synonymously. We describe a drawing of a graph G abstractly by giving the graph $G^\times$ with a planar rotation system. If e is an edge, then $e^\times$ denotes the *planarization of* e, i.e., the walk in $G^\times$ whose end vertices are endpoints of the edge and the internal vertices (if any) are the crossing points on e, in order. The edges of $e^\times$ are called the *edge-segments* of e, while e is the *parent-edge* of its edge-segments. For each dummy-vertex d on e, the sub-walk of $e^\times$ connecting d to one of the endpoints of e is called a *part-edge*.

Consider a walk W in $G^\times$ that contains a dummy-vertex z as an interior node; say the incident edge-segments are e_1 and e_2. We say that W *reverses at* z if $e_1 = e_2$, *goes straight at* z if $e_1 \neq e_2$ but they have the same parent-edge, and *makes a turn* otherwise. We call W a *G-respecting walk* if and only if it begins and ends at vertices of G and goes straight at all interior crossing points, so it corresponds to a walk in G along the parent-edges. W is said to be a *G-respecting path* if it visits every vertex of G at most once. We now discuss two tools used in [5] for computing vertex connectivity of 1-plane graphs. We slightly simplify and generalize them here so that we may use them for general embedded graphs.

Radial Planarization. The term *radial planarization* refers to the process of planarizing an embedded graph G to obtain $G^\times$, and then inserting a face-vertex inside each face of $G^\times$ that is adjacent to all vertices of $G^\times$ bounding the face. The subgraph of $\Lambda(G)$ induced by all edges incident to face-vertices is called the *radial graph* of G, denoted by $R(G)$. Note that $R(G)$ is a simple bipartite graph (A, B), where $A = V(G^\times)$ and B is the set of all face-vertices. The *face-distance* between two vertices $u, v \in \Lambda(G)$, denoted by $d_F(u, v)$, is the number of face-vertices on a shortest path between u and v in $R(G)$. (This is approximately half the number of vertices on a shortest path in $R(G)$.) For any subset of vertices S in $\Lambda(G)$, we write $d_F(u, S) = \min_{v \in S} d_F(u, v)$. We call a face-vertex f in $\Lambda(G)$ *adjacent* to an edge of $G^\times$ if f and the endpoints of e form a face of $\Lambda(G)$.

Co-Separating Triple. In [5], the authors defined "co-separating triple" as a vertex-partition (A, X, B) of $\Lambda(G)$ for which X separates A and B in both $\Lambda(G)$ and G. We generalize their definition, specifically Condition 3, from 1-plane graphs to arbitrary embedded graphs.

▶ **Definition 2** (Co-separating triple). *Let G be an embedded graph and Λ be an auxiliary graph derived from $\Lambda(G)$. A partition of the vertices of Λ into three sets (A, X, B) is called a* co-separating triple of (Λ, G) *if it satisfies the following conditions:*

(C1) *Each of A, X, B contains at least one vertex of G.*

(C2) *There is no edge of Λ with one endpoint in A and the other endpoint in B.*

(C3) *For every edge $e \in E(G)$, either both endpoints of G are in X, or all vertices of $e^\times \cap \Lambda$ belong entirely to $A \cup X$ or all vertices of $e^\times \cap \Lambda$ belong entirely to $B \cup X$.*

If (A, X, B) is a co-separating triple of $(\Lambda(G), G)$, then Condition 2 implies that X separates A and B in $\Lambda(G)$. Due to Conditions 1 and 3 the same holds for the corresponding sets in G

▶ **Observation 3.** *If (A, X, B) is a co-separating triple of $(\Lambda(G), G)$, then $X \cap V(G)$ separates $A \cap V(G)$ and $B \cap V(G)$ in G.*

Proof. By Condition 1, $A \cap V(G)$ and $B \cap V(G)$ are non-empty. Consider any pair of vertices $a \in A \cap V(G)$ and $b \in B \cap V(G)$. By Condition 3, any path of G that connects a and b must intersect a vertex of $X \cap V(G)$. Therefore, $X \cap V(G)$ separates $A \cap V(G)$ and $B \cap V(G)$. ◀

The following definition of *kernel* is new and does not appear in [5], even though the authors use the concept implicitly.

▶ **Definition 4** (Kernel of a co-separating triple)**.** *Let (A, X, B) be a co-separating triple of (Λ, G). Then a* kernel *of (A, X, B) is a set of vertices K such that $X \cap V(G) \subseteq K$ and K contains at least one vertex of $A \cap V(G)$ and at least one vertex of $B \cap V(G)$. The maximum distance in Λ between any two vertices of a kernel is called the* kernel diameter*.*

3 Ribbon Radius and Minimal Vertex Cuts

Our algorithm to compute vertex and edge connectivity of an embedded graph uses a parameter that we call the ribbon radius. For any crossing point c of edges $\{e_1, e_2\}$, a *ribbon of c* is a path in G that begins with e_1 and ends with e_2. Informally, the ribbon radius measures how far away from c we have to go in $\Lambda(G)$ if we want to include the planarization of a ribbon of c. Before defining this formally, we need the notion of a *ball of face-radius r*. Let S be a non-empty set of vertices of $G^\times$. Let $\Lambda(S, r)$ be the set of all vertices of $\Lambda(G)$ that have face-distance at most r from at least one vertex of S. Let $\mathcal{B}(S, r)$ be the restriction of $\Lambda(S, r)$ to $G^\times$, i.e., it is obtained from $\Lambda(S, r)$ by deleting all face-vertices. We use both notations often for a single vertex $S = \{c\}$ and then write $\Lambda(c, r)$ and $\mathcal{B}(c, r)$ for simplicity.

For any crossing c, define $\mu(c)$ to be the radius of a smallest ball centered at c that contains a *planarized* ribbon at c. Put differently, the ball $\mathcal{B}(c, \mu(c))$ must contain the crossing point c, all four part-edges of c, and for some path in G that connects two consecutive endpoints of c, all planarizations of all edges of the path. For a vertex v of $G^\times$ that is not a crossing, define $\mu(v) := 0$. The *ribbon radius of an embedded graph G*, denoted by $\mu(G)$, is defined as $\mu(G) = 1 + \max_{v \in V(G^\times)} \mu(v)$. [2] One can easily show that the ribbon-radius upper-bounds the face-distance between vertices of a planarized edge $e^\times$.

▶ **Observation 5.** *Let $e \in E(G)$ and $e^\times$ be its planarization. For any pair of vertices d_1, d_2 in $e^\times$, we have $d_F(d_1, d_2) \leq \mu(G)$. If d_1 is a dummy-vertex, then $d_F(d_1, d_2) \leq \mu(d_1)$.*

Proof. If d_1 is a dummy-vertex, then by definition of $\mu(d_1)$, ball $\mathcal{B}(d_1, \mu(d_1))$ contains the entire planarized edge $e^\times$, so $d_F(d_1, d_2) \leq \mu(d_1) < \mu(G)$. Similarly $d_F(d_1, d_2) \leq \mu(G)$ if d_2 is a dummy-vertex, so assume that d_1, d_2 are the endpoints of e. If e is an uncrossed edge, then $d_F(d_1, d_2) = 1$ (we can connect them in the radial graph via a face incident to e) and $\mu(G) \geq 1$ and the result holds. If e is crossed then let x be the first dummy-vertex in the direction d_1 to d_2. Then $d_F(x, d_1) = 1$ and $d_F(x, d_2) \leq \mu(x) \leq \mu(G) - 1$. Therefore, $d_F(d_1, d_2) \leq \mu(G)$. ◀

[2] The "1+" in the definition may seem unusual, but will be needed later for Claim 9.

For the proof of our main result (as well as for Section 5), we also need to define the boundaries of a ball. Formally, for a vertex set S in $G^\times$ and an integer r, the *boundary* $\mathcal{Z}(S,r)$ of $\mathcal{B}(S,r)$ is the set of all edges $e=(u,v)$ such that of the two face-vertices f,f' that are adjacent to e, exactly one belongs to $\Lambda(S,r)$. Thus (say) $d_F(f,S)\le r$ while $d_F(f',S)>r$. Since we could get from S to f' via u,v (which in turn can be reached via f), and since the face-distance only counts the number of face-vertices in a path, this implies that $d_F(f,S)=r=d_F(u,S)=d_F(v,S)$ while $d_F(f',S)=r+1$.

We now give the main theoretical ingredient to prove Theorem 1. Since minimum edge cuts of a graph can be expressed via minimal vertex cuts (see also Section 4), we state the lemma only for minimal vertex cuts.

▶ **Lemma 6.** *Let G be an embedded graph with ribbon radius $\mu:=\mu(G)$. For any minimal vertex cut S of G, there is a co-separating triple (A,X,B) of $(\Lambda(G),G)$ such that $X\cap V(G)=S$ and a kernel of (A,X,B) with diameter at most $|S|(4\mu+1)$.*

Proof. Let $S=\{s_1,s_2,\dots,s_p\}$ be the given minimal vertex cut of G and $\{\phi_i\}_{i=1,2,\dots}$ be the set of at least two flaps of S. We first label all vertices of $\Lambda(G)$ with A, X or B; this will lead us to a co-separating triple. There are three types of vertices in $\Lambda(G)$, namely vertices of G, dummy-vertices and face-vertices. Vertices of G are labelled as follows. Each vertex of G in S is labelled X, each vertex of G in flap ϕ_1 with A, and each vertex of G in a flap other than ϕ_1 with B. Next, dummy-vertices are labelled as follows. For each dummy-vertex c, if all the endpoints of the crossing at c are labelled the same then give c the same label, otherwise label it X. Lastly, face-vertices are labelled as follows. For each face-vertex f, if all vertices of $V(G^\times)$ on the corresponding face of $G^\times$ are labelled the same then give f the same label, otherwise label it X.

These labels give a partition of the vertices of $\Lambda(G)$ into sets A,X,B. We will show that this forms a co-separating triple. Condition 1 holds since S is non-empty and defines at least two non-empty flaps. Condition 3 is also easy to show:

▶ **Observation 7.** *Consider any edge $e\in E(G)$ and its planarization $e^\times$. Then either all vertices of $e^\times$ belong to $A\cup X$, or they all belong to $B\cup X$.*

Proof. If one endpoint of e is in S, then this endpoint and *all* crossing points along e are labelled X and the claim holds. If neither endpoint of e is in S, then they both belong to the same flap, so they are labelled the same, say A. Therefore any crossing point along e can only be labelled A or X. ◀

Condition 3 implies Condition 2 for any edge of $G^\times$. For any other edge of $\Lambda(G)$, at least one endpoint is a face-vertex, and this can be labelled with A or B only if the other endpoint is labelled the same way. So Condition 2 also holds. So we have shown that (A,X,B) is a co-separating triple and $X\cap V(S)=S$ clearly holds by our labelling scheme.

The hard part in proving Lemma 6 is in finding the kernel, or more precisely, showing that $\Lambda(S,\mu(G))$ satisfies all its required conditions. We first need to show that this subgraph contains all vertices labelled X.

▷ Claim 8. For any vertex $v\in V(G^\times)$ that is labelled X, there exists a vertex $s\in S$ in $\mathcal{B}(v,\mu(v))$.

Proof. This clearly holds if v itself is in S. Otherwise v is a crossing point of edges $\{e,e'\}$. Now there are two sub-cases. If some endpoint of the crossing belongs to S, we can choose s to be this endpoint; by Observation 5, $d_F(v,s)\le\mu(v)$. Now assume that all the endpoints of e,e' are all labelled A or B. Since adjacent vertices cannot belong to different flaps, the endpoints of e are labelled the same, say A. Since v is labelled X, the endpoints of e' must be labelled B. Now consider a ribbon of v for which the planarization is in $\mathcal{B}(v, \mu(v))$ (this must exist by definition of $\mu(v)$). Walking along the ribbon, we begin at a vertex labelled A (so in flap ϕ_1) and end at a vertex labelled B (so in a flap other than ϕ_1). Hence there must be a vertex of S in-between; set s to be this vertex. ◁

Recall that we defined the boundary of a ball earlier; we will now study the subgraph $\mathcal{Z}$ formed by the edges in $\mathcal{Z}(S, \mu(G))$.

▷ Claim 9. For every edge $e = (u, v)$ of $\mathcal{Z}$, either both u, v are labelled A, or both are labelled B.

Proof. Assume first for contradiction that v is labelled X. Then some vertex of S is in $\mathcal{B}(v, \mu(v))$ (Claim 8), or symmetrically $v \in \mathcal{B}(S, \mu(v))$. But by $\mu(v) \leq \mu - 1$, then all faces incident to v (including the two incident to e) have face-distance at most μ from S, so e is not in $\mathcal{Z}$. Similarly one argues that u cannot by labelled X, and by 3 therefore they are both labelled A or both labelled B. ◁

▷ Claim 10. There is a unique face f_S of $\mathcal{Z}$ such that all vertices of S are strictly inside f_S.

Proof. Since every vertex of $\mathcal{Z}$ is incident to an edge in $\mathcal{Z}$, it is labelled A or B (Claim 9). So all vertices of S must be strictly inside some face of $\mathcal{Z}$. Now assume for contradiction that there are two vertices s_1, s_2 that are in distinct faces of $\mathcal{Z}$, hence separated by a cycle Z within $\mathcal{Z}$. Vertices in cycle Z are all labelled the same (propagate Claim 9 around Z); up to symmetry, say A. Since S is a minimal vertex cut, vertex s_i (for $i = 1, 2$) has in graph G a neighbor v_i in flap ϕ_2. Since ϕ_2 is a connected component of $G \setminus S$, there exists a path P in G from v_1 to v_2 within flap ϕ_2. This implies that all vertices on the planarized path $P^\times$ are labelled X or B. Notice that $P^\times$ connects v_1 and v_2, which lie on either side of the cycle Z. However, $P^\times$ cannot use a vertex of Z (because those are labelled A), or cross an edge of Z (since $G^\times$ is planar). This is a contradiction. ◁

We are now ready to show show that $\Lambda(S, \mu)$ (for $\mu := \mu(G)$) is indeed the desired kernel. All vertices of G that are labelled X belong to $\mathcal{B}(S, \mu) \subset \Lambda(S, \mu)$ by Claim 8. To show the second required property of kernels, consider an arbitrary vertex $s \in S$, which by minimality of S has (in G) a neighbor in each flap; in particular it has a neighbor $a \in \phi_1$ (hence labelled A) and a neighbor $b \in \phi_2$ (hence labelled B). From Observation 5 we know that $d_F(a, s) \leq \mu(G)$, so $a \in \mathcal{B}(S, \mu) \subset \Lambda(S, \mu)$ and likewise $b \in \Lambda(S, \mu)$. Therefore, $\Lambda(S, \mu)$ is a kernel, and with arguments similar to what was used in [5] one can show that its diameter is small enough.

▷ Claim 11. The distance between any two vertices u, v of $\Lambda(S, \mu)$ is at most $|S|(4\mu + 1)$.

Proof. For every vertex z in $\Lambda(S, \mu)$, we write $S(z)$ for some vertex s in S such that $d_F(s, z) \leq \mu$ (this exists by definition of $\Lambda(S, \mu)$). Also write $\pi_S(z)$ for a path from z to $S(z)$ that achieves this face-distance; note that $\pi_S(z)$ lies within $\Lambda(S, \mu)$ and contains at most 2μ edges.

Since $\Lambda(G)$ is connected, and u and v are both inside face f_S, we can find a (u, v)-path P within the part of $\Lambda(G)$ that lies within f_S, i.e., within $\Lambda(S, \mu)$. Enumerate the vertices of P as a sequence $u = w_1, w_2, \ldots, w_{|P|} = v$ and expand this into a walk W by adding detours to the corresponding vertices in S, i.e., $W = w_1 \rightsquigarrow S(w_1) \rightsquigarrow w_1, w_2 \rightsquigarrow S(w_2) \rightsquigarrow w_2 \ldots S(w_{|P}) \rightsquigarrow w_{|P|}$, using path $\pi_S(w_i)$ for each detour from w_i to $S(w_i)$ or vice versa. For as long as W contains some vertex s of S twice, prune the entire sub-walk between the two occurrences of s. We therefore end with a walk

$$W^* = w_1 \rightsquigarrow S(w_1) \rightsquigarrow w_{i_1}, w_{i_1+1} \rightsquigarrow S(w_{i_1+1}) \rightsquigarrow w_{i_2} \cdots \rightsquigarrow S(w_{|P|}) \rightsquigarrow w_{|P|}$$

that contains every vertex of S at most once. It consists of at most $2|S|$ paths $\pi_S(z)$ for some $z \in \Lambda(S, \mu)$ (and each of those paths has length at most 2μ), plus at most $|S| - 1$ edges connecting w_{i_j} to $w_{i_{j+1}}$ for some index j. Therefore $|W^*| \leq |S|4\mu + |S| - 1 \leq |S|(4\mu + 1)$ as required. ◁

This completes the proof of Lemma 6. ◀

4 Finding Minimum Vertex and Edge Cuts

Lemma 6 implies that we can compute the vertex connectivity of an embedded graph by finding a co-separating triple (A, X, B) of $(\Lambda(G), G)$ with $|X \cap V(G)|$ minimized since that gives us a minimum vertex cut of G. Our task can be simplified by using that the desired co-separating triples have low kernel diameter. Taking the approach of [5], we break down $\Lambda(G)$ into low-diameter subgraphs such that the kernel resides completely in some such subgraph. Since $\Lambda(G)$ is planar, and planar graphs with small diameter have small treewidth, we then use their low-width tree decompositions to search for a co-separating triple that can be expanded into a co-separating triple of $(\Lambda(G), G)$.

Set-Up. The input to our algorithm is an embedded graph G (given via a rotation system of $G^\times$) with known ribbon radius $\mu(G)$. We first compute $\Lambda(G)$ and perform a Breadth-First Search (BFS) on it, thereby partitioning $V(\Lambda(G))$ into layers $V_1, V_2, \ldots, V_d$, where vertices of layer V_i are at distance exactly $(i-1)$ from the source vertex of the BFS. Let s be a user-given parameter; we wish to test whether there exists a minimal vertex cut of size s. (We will later test all values of s until we succeed.) Set $w := s(4\mu + 1)$; this corresponds to the bound on the kernel diameter by Theorem 6.

The idea is to focus on searching for a co-separating triple (A, X, B) with minimum $|X \cap V(G)|$ in the subgraphs of $\Lambda(G)$ induced by $V_i, V_{i+1}, \ldots, V_{i+w-1}$, for all $1 \leq i \leq d-w+1$. To omit specifying the range for i every time, we define $V_i = \emptyset$ for all $i < 1$ and $i > d$. Define Λ_i to be the subgraph $\Lambda[V_{i-1} \cup \cdots \cup V_{i+w}]$ of $\Lambda(G)$ induced by the vertices $V_{i-1} \cup \cdots \cup V_{i+w}$, with two added edge-sets U_{i-1} and L_{i+w}. Here U_{i-1} and L_{i+w} are edges within V_{i-1} and V_{i+w}, respectively, and represent the components of $\Lambda(G)$ of all other layers. More precisely, there is a one-to-one correspondence between the components of $(V_{i-1}, U_{i-1}) \cup (V_{i+w}, L_{i+w})$ and the components of $\Lambda[V_0 \cup \cdots \cup V_{i-1}] \cup \Lambda[V_{i+w-1} \cup \cdots \cup V_d]$ (see Figure 1 for an illustration and [5] for more details). Lemmas 12 and 13 (see [3] for a proof) show that we can restrict our search for co-separating triples (A, X, B) of $(\Lambda(G), G)$ to co-separating-triples (A_i, X, B_i) of (Λ_i, G) in each Λ_i.

▶ **Lemma 12.** *If there exists a co-separating triple (A, X, B) of $(\Lambda(G), G)$ with kernel diameter at most w, then there exists an index i and a co-separating triple (A_i, X, B_i) of (Λ_i, G) for which $X \subseteq \Lambda[V_i \cup \cdots \cup V_{i+w-1}]$.*

▶ **Lemma 13.** *If (A_i, X, B_i) is a co-separating triple of (Λ_i, G) where $X \subseteq V_i \cup \cdots \cup V_{i+w-1}$, then there exist sets A and B such that (A, X, B) is a co-separating triple of $(\Lambda(G), G)$.*

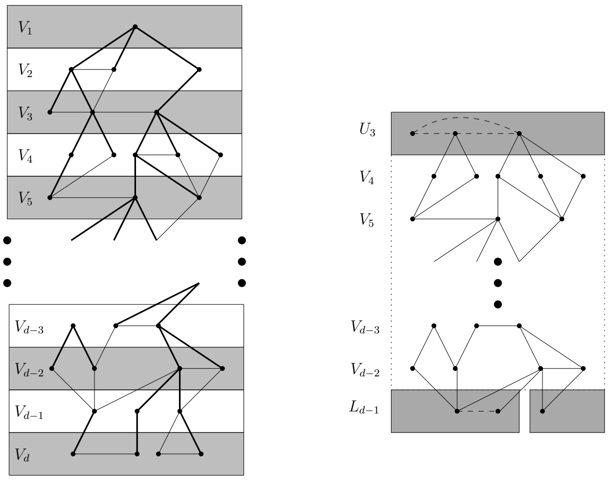

Figure 1 Constructing Λ_i for $i = 4$.

A Tree Decomposition Based Algorithm. The broad idea of our algorithm is to iterate through $s = 1, 2, \ldots,$ where for each iteration, we test each Λ_i, in the order $i = 1, 2, \ldots, d$, for a co-separating triple (A_i, X, B_i) where $|X \cap V(G)| = s$. If we fail to find such a co-separating triple for all Λ_i for the current value of s we try $s + 1$. Thus, the problem now reduces to designing an algorithm that can test whether Λ_i has a co-separating-triple (A_i, X, B_i) with $|X \cap V(G)| = s$.

We know from [5] that Λ_i is planar and has radius at most $w + 2$, and hence we can find a tree decomposition of Λ_i of width $O(w)$ in $O(w|V(\Lambda_i)|)$ time [5, 11]. (We assume that the reader is familiar with treewidth and tree decompositions; see [10] for a reference). It was shown in [4] that conditions 1 and 2 can be phrased as a formula Φ in so-called monadic second-order logic (MSOL). We can argue (see [3]) that Condition 3 need not be tested if we add edges to the graph (the new graph has asymptotically the same treewidth). By Courcelle's theorem [8], we can therefore test in $O(f(w, |\Phi|)\, n)$ time (for some function $f(\cdot)$ that only depends on the treewidth and the size of formula) whether Λ_i has a co-separating triple. Since the formula from [4] has constant size, we hence can determine $\kappa(G)$ in linear time. Unfortunately, function $f(\cdot)$ could be large, e.g. a tower of exponentials. So rather than relying on Courcelle's theorem, we have also developed a direct dynamic programming based algorithm that has running time $2^{O(w)}|V(G^\times)|$. The idea of this is quite straightforward ("consider all possible assignment of labels "A_i", "B_i" or "X" to the vertices of a bag and retain only those assignments that are compatible with valid assignments at the children"), but the details are intricate; see [3]. Since $w = O(\mu s)$, the total running time is $\sum_{s=1}^{\kappa} 2^{O(\mu s)}|V(G^\times)| \subseteq 2^{O(\mu\kappa)}|V(G^\times)|$. This establishes Theorem 1 for vertex connectivity.

Modifications for Edge Connectivity. For computing the edge connectivity of G, we first construct a graph $\tilde{G}$ by subdividing every edge of G exactly once. In terms of the drawing of G, we place the subdivision vertices close to an endpoint of the edge, i.e., before the first dummy-vertex on the edge. The subdivision vertices are stored in a separate list $D(\tilde{G})$. Observe that a set of edges T is an edge cut of G if and only if the set $\tilde{T}$ of the corresponding subdivision-vertices separates two vertices of G. Therefore, T is a minimum edge cut of G if and only if $\tilde{T}$ is a minimum subset of $D(\tilde{G})$ that separates two vertices of G. By Lemma 6, if $\tilde{T}$ is a minimal vertex cut of $\tilde{G}$, then there is a co-separating triple of $(\Lambda(\tilde{G}), \tilde{G})$ such that $X \cap V(\tilde{G}) = \tilde{T}$. Conversely, any co-separating triple of $(\Lambda(\tilde{G}), \tilde{G})$ will give an edge cut of G, as the following observation shows.

▶ **Observation 14.** *If (A, X, B) is a co-separating triple of $(\Lambda(\tilde{G}), \tilde{G})$ such that $X \cap V(\tilde{G}) \subseteq D(\tilde{G})$, then the edges of G with subdivision vertices in $X \cap V(\tilde{G})$ form an edge cut of G.*

Proof. By Observation 3, we know that $X \cap V(\tilde{G})$ separates $A \cap V(\tilde{G})$ and $B \cap V(\tilde{G})$. So it suffices to show that $A \cap V(G)$ and $B \cap V(G)$ are both non-empty. Suppose that A contains a subdivision vertex $\tilde{e}$ of some edge $e = (u, v) \in E(G)$; without loss of generality, say it is placed close to u. This implies that $(u, \tilde{e})$ is an edge of $\tilde{G}$, and therefore, of $\Lambda(\tilde{G})$. By Condition 2, this implies that $u \in A \cup X$. However, since X contains only subdivision vertices, we must have $u \in A$. So A always contains a vertex of G, as does B. ◀

Therefore, one needs only few minor changes to the algorithm for computing vertex connectivity: We let the input graph be $\tilde{G}$ instead of G, and try to find a co-separating triple such that $X \cap V(\tilde{G}) \subseteq D(\tilde{G})$; this condition can easily be enforced via dynamic programming. This establishes Theorem 1 for edge-connectivity.

5 Applications to Near-Planar Graphs

The name "near-planar graphs" is an informal term for graphs that are close to being planar; these have received a lot of attention in recent years (see [9, 14] for surveys in this area). An important class of near-planar graphs is the class of *k-planar graphs*, which are graphs that can be embedded on the plane with each edge crossed at most k times, with no two edge-curves sharing more than one point. A k-planar graph that achieves the maximum possible number of edges is called an *optimal k-planar graph*.

▷ Claim 15 (Corollary of Theorem 1). Let G be a simple k-plane graph for some constant k. If G has ribbon radius μ, then one can compute both $\kappa(G)$ and $\lambda(G)$ in time $2^{O(\mu)}n$.

Proof. A simple k-plane graph has at most $4.108\sqrt{k}n$ edges [17]. This implies that $\kappa(G) \leq \lambda(G) \leq \delta(G) \leq 8.216\sqrt{k}$. Since each edge of G is crossed at most k times, $|V(G^\times)| \in O(nk^{3/2})$. Substituting these bounds into Theorem 1, we get that both vertex and edge connectivity can be computed in time $2^{O(\mu\sqrt{k})}nk^{3/2}$, and k is a constant. ◁

We assumed simplicity for this claim because the presence of parallel edges affects the edge connectivity. However, we can can permit parallel edges for as long as the edge connectivity remains constant.

We now give examples of several classes of k-plane graphs that have small ribbon radius. Recall that an *$\times$-crossing* of an embedded graph is a crossing where the only edges between the endpoints are the two edges that cross. From [5], we know how to compute vertex connectivity for 1-plane graphs without $\times$-crossings in linear time. We show the same for k-plane graphs by combining the Claim 16 with Claim 15.

▷ Claim 16. Let G be a k-plane graph. Let c be a crossing that is not a $\times$-crossing. Then $\mu(c) \leq \lfloor \frac{3k+1}{2} \rfloor$.

Proof. Let $\{e, e'\} = \{(u, v), (w, x)\}$ be the two edges that cross at c, and assume (up to renaming) that $(v, w) = \hat{e}$ is an edge in G that connects two consecutive endpoints of c. This gives a ribbon $e, \hat{e}, e'$, and we argue now that its planarization lies within $\mathcal{B}(c, \lfloor \frac{3k+1}{2} \rfloor)$. In the planarized graph $G^\times$, we can walk from c to any of the four endpoints of e, e' along at most k edge-segments. There is also a closed walk W containing c and $\hat{e}^\times$ as follows: walk from c to v along at most k edge-segments of e, then walk from v to w along at most $k{+}1$ edge-segments of $\hat{e}$, and then walk from w to c along at most k edge-segments of e'. By using the face-vertices incident to these edge-segments, and the shorter of the two connections from c via W, we have $d_F(c,z) \le \max\{k, \lfloor\frac{3k+1}{2}\rfloor\} = \lfloor\frac{3k+1}{2}\rfloor$ for all vertices z of the planarized ribbon as desired. ◁

The authors of [5] posed as an open problem whether the vertex connectivity of a 1-plane graph can still be computed in linear time if there is a bounded number of ×-crossings. We answer this in the affirmative and generalize it to all k-plane graphs. We state the result more broadly for any graph class where most crossings have small ribbon radius.

▶ **Lemma 17.** *Let G be a connected embedded graph, and let α, γ be two constants such that there are at most γ crossing points c with $\mu(c) > \alpha$. Then $\mu(G) \le \gamma + \alpha + 1$.*

Proof. It suffices to show that $\mu(c) \le \gamma + \alpha$ for any crossing point c. This holds trivially if $\mu(c) \le \alpha$, so we may assume that c is one of the (at most γ) "special" crossings. For $r = 0, 1, 2, \dots, \gamma$, consider the boundary $\mathcal{Z}(c,r)$ of the ball $\mathcal{B}(c,r)$ centered at c. As argued earlier, any edge in $\mathcal{Z}(c,r)$ connects two vertices with face-distance r from c, therefore $\mathcal{Z}(c,0), \dots, \mathcal{Z}(c,\gamma)$ form γ+1 vertex-disjoint graphs. By the pigeon-hole principle, there must exist some $r_c \le \gamma$ such that $\mathcal{Z}(c, r_c)$ contains no special crossing points, or in other words $\mu(z) \le \alpha$ for all $z \in \mathcal{Z}(c, r_c)$. Since c is a special crossing point, $r_c > 0$. Our goal is to find a planarized ribbon within $\mathcal{B}(c, r_c + \alpha)$, i.e., a G-respecting walk that begins and ends with $e^\times$ and $(e')^\times$, where e and e' are the two edges that cross at c.

▷ Claim 18. $e^\times$ belongs to $\mathcal{B}(c, r_c + \alpha)$.

Proof. Suppose that $e^\times$ is not a subgraph of $\mathcal{B}(c, r_c + \alpha)$. Since c is within $\mathcal{B}(c, r_c)$, the path $e^\times$ then must leave ball $\mathcal{B}(c, r_c + \alpha)$ somewhere and hence contains a dummy-vertex $z \in \mathcal{Z}(c, r_c)$. By choice of r_c we have $\mu(z) \le \alpha$, so there exists some planarized ribbon of z within $\mathcal{B}(z, \alpha)$. This planarized ribbon contains all of $e^\times$ since e crosses some other edge at z (Figure 2a). So $e^\times \subset \mathcal{B}(z,\alpha) \subseteq \mathcal{B}(z, r_c + \alpha)$. ◁

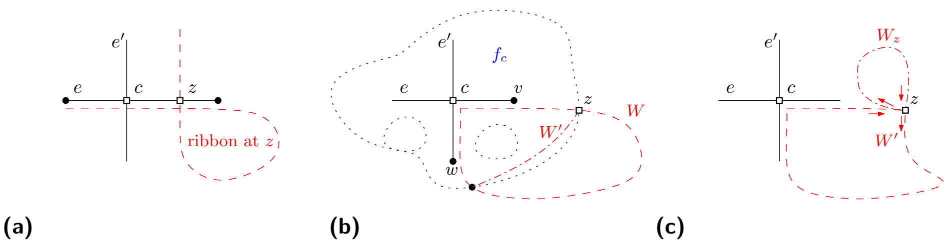

Figure 2 (a) $e^\times$ is in $\mathcal{B}(c, r_c + \alpha)$. (b) With a shortcut along $\mathcal{Z}(c, r_c)$ the walk stays within $\mathcal{B}(c, r_c)$. (c) Inserting a ribbon-walk W_z at z removes the turn at z.

Likewise the planarization $(e')^\times$ is in $\mathcal{B}(c, r_c + \alpha)$, so it remains to find a G-respecting walk between consecutive endpoints of c in $\mathcal{B}(c, r_c+\alpha)$. For this we analyze a related concept. Let us call a walk in $G^\times$ a *ribbon walk* if it: (a) begins and ends at a crossing point; (b) its first and last edge-segments have different parent-edges; (c) it makes no turns but is allowed to reverse. (The definitions of "turn" and "reverse" can be found in Section 2.)

▷ Claim 19. There is a ribbon walk Q at c within $\mathcal{B}(c, r_c + \alpha)$.

Proof. Let v, w be two consecutive endpoints of c. We first construct an arbitrary ribbon walk W at c: walk along $e^\times$ to v, then along any G-respecting path from v to w (this exists since G is connected), and then along $(e')^\times$ back to c. We will convert W into another walk W' at c (not necessarily a ribbon-walk) that strictly stays within $\mathcal{B}(c, r_c)$.

The boundary $\mathcal{Z}(c, r_c)$ of $\mathcal{B}(c, r_c)$ induces a plane subgraph of $G^\times$. Since $r_c > 0$, vertex c is strictly inside one face f_c of this plane subgraph. Assume that W goes strictly outside face f_c. This means that some cycle Z (within the boundary of f_c) contains c strictly inside and some other vertex of W strictly outside. Therefore W reaches a vertex z on Z, and then goes strictly outside Z (Figure 2b). But to return to c, walk W must return to cycle Z at some point later. We can therefore re-route W by walking along cycle Z, hence stay on the boundary of f_c throughout. We repeat at all places where W goes outside f_c to get the new walk W'.

Walk W' begins and ends with the same edge-segments as W, but it may have turns at dummy-vertices $z \in \mathcal{Z}(c, r_c)$ due to the short-cutting of W (Figure 2c). However, we know that $\mu(z) \leq \alpha$, by which we can find a ribbon-walk W_z at z that stays within $\mathcal{B}(z, \alpha)$. By merging W_z at the occurrence of z in W' suitably, we can remove the turn at z without adding any new turns. Repeating at all turns of W', we obtain a ribbon-walk Q at z that lies within $\mathcal{B}(c, r_c + \alpha)$. ◁

We can now turn Q into a planarized ribbon at c as follows. If Q reverses at some dummy-vertex, then we simply omit the twice-visited edge-segment e_1 from Q and retain a ribbon-walk. Since Q now does not turn and goes straight at dummy vertices, it must begin and end with edge-segments of e and e'. To Q, we attach the part-edges of e and e' that are possibly not visited; we argued in Claim 18 these are also within $\mathcal{B}(c, r_c + \alpha)$. This gives a G-respecting walk within $\mathcal{B}(c, r_c + \alpha)$ that begins and ends with the planarizations of e and e', hence a planarized ribbon as desired. ◀

The following corollaries follow easily from Lemma 17 and Claim 16.

► **Corollary 20.** *If a graph G can be embedded with at most q crossings, then $\mu(G) \leq q + 1$.*

► **Corollary 21.** *For a k-plane graph G with at most γ crossings that are $\times$-crossings, $\mu(G) \leq \gamma + \lfloor \frac{3k+5}{2} \rfloor$.*

Let the *skeleton* of an embedded graph G, denoted $\mathrm{sk}(G)$, be the plane subgraph of G induced by the set of all uncrossed edges. (This implies that the crossed edges are in the interior of the faces of $\mathrm{sk}(G)$.) An embedded graph G is a *d-framed graph* if $\mathrm{sk}(G)$ is simple, biconnected, spans all vertices and all its faces have *degree* at most d (i.e., the face boundary has at most d edges) [2]. Examples of graphs that are d-framed include optimal k-plane graphs for $k \in \{1, 2, 3\}$ [1, 20]. We can redraw the edges within each face of $\mathrm{sk}(G)$ so that pairs of edges cross at most once, by locally rearranging whenever two edges cross twice and hence reducing the total number of crossings. The resulting drawing then has $O(d^2)$ crossings in each face and therefore small ribbon radius:

▷ Claim 22. If G is an embedded graph such that $\mathrm{sk}(G)$ contains at most q crossings in each of its faces, then $\mu(G) \leq q + 1$.

Proof. For every face F of $\mathrm{sk}(G)$ and every crossing c in F, one can find a ribbon at c by using the planarization of the edges e, e' that cross at c, and connect consecutive endpoints of c via edges on the boundary of F. It is intuitive that the entire subgraph G_F of the edges of $G^\times$ that are on or inside F belongs to $\mathcal{B}(c, q)$. In particular therefore this ribbon is in $\mathcal{B}(c, q)$ and $\mu(c) \leq q$ and $\mu(G) \leq q + 1$. We formalize this intuition below.

Consider as before the boundaries $\mathcal{Z}(c, r)$ of balls $\mathcal{B}(c, r)$ for $r = 0, 1, 2, \ldots$, and let $r_c \leq q$ be such that $\mathcal{Z}(c, r)$ contains no crossing point that lies inside F. Since c is inside F we have $r_c > 0$, and there exists a face f_c of $\mathcal{Z}(c, r_c)$ that contains c strictly inside. If all vertices of

G_F are on or inside face f_c then they all belongs to $\mathcal{B}(c, r_c)$ and we are done. So assume for contradiction that a vertex v of G_F is strictly outside f_c, hence separated from c by a cycle Z within the boundary of f_c. Since both c and v are on or inside F, cycle Z must go strictly inside F to separate them, so it contains an edge e that is strictly inside F. If e ends at a dummy-vertex, then Z (hence $\mathcal{Z}(c, r_c)$) includes a crossing-point inside F, contradicting the choice of r_c. If both ends of e are vertices of G, then e is a chord of F that lies inside F; this contradicts that F is a face of skeleton sk(G). ◁

A *map graph* is the intersection graph of a map of *nations*, where a nation is a closed disc homeomorph on the sphere, and the interiors of any two nations are disjoint [7]. A *d-map graph* has a map where at most d nations intersect at a point.

▷ Claim 23. If G is a d-map graph, then it has an embedding with $\mu(G) \le \frac{3}{8}d^2$.

Proof. It is folklore that any d-map graph G can be drawn by using a planar subgraph G_0 and then replacing some faces of G_0 of degree at most d of G_0 by a complete graph (see the Appendix in [3]). We can draw each such complete graph inside the face of G_0 such that every pair of edges cross at most once. Then every edge is crossed by at most $\lceil \frac{d-2}{2} \rceil \lfloor \frac{d-2}{2} \rfloor$ other edges. Therefore a d-map graph G is k-planar for $k \le \frac{1}{4}(d-2)(d-1) = \frac{1}{4}(d^2 - 3d + 2) \le \frac{1}{4}(d^2 - 10)$. Also for every crossing c the four endpoints induce a complete graph, so c is not an $\times$-crossing. By Claim 22, $\mu(c) \le \frac{3k+1}{2} \le \frac{3}{8}d^2 - 1$. ◁

In summary, the following classes of embedded graphs have small ribbon radius: k-plane graphs with a constant number of $\times$-crossings (Corollary 21), optimal k-planar graphs for $k \in \{1, 2, 3\}$ (Claim 22), graphs with a constant number of crossings (Corollary 20), d-map graphs (Claim 23), d-framed graphs, and more generally, graphs with a constant number of crossings in each face of its skeleton (Claim 22). Since each of these classes of graphs are also k-plane for some constant k, by Corollary 15, one can compute their vertex connectivity and edge connectivity in $O(n)$ time.

6 Computing The Ribbon Radius

Our main Theorem 1 (and all applications derived from it) assumed that we know the ribbon radius $\mu(G)$ of the given embedded graph G. The definition of ribbon radius leads to a straight-forward polynomial-time algorithm to computing $\mu(G)$, but here we discuss how to organize this algorithm to be somewhat more efficient. As usual we assume that G is given in form of the planarization $G^\times$ with a fixed rotation system.

▶ **Proposition 24.** *Let G be an embedded graph with n vertices and q crossings. Then we can compute the ribbon radius $\mu(G)$ in time $O(q(n+q)\log\mu(G))$.*

Proof. We first compute the radial planarization $\Lambda(G)$ and the radial graph $R(G)$; this can be done in $O(n+q)$ time since we are given $G^\times$ and it is a planar graph with $n+q$ vertices.

The main idea is now to do a galloping search for the correct value of μ. Thus we will first test whether $\mu(G) \le \mu{+}1$ (or equivalently, $\mu(v) \le \mu$ for all dummy-vertices v) for $\mu = 1, 2, 4, 8, \dots$. Once we succeed (say at μ') we do a binary search for the correct μ in the interval $[\mu'/2, \mu']$. Since $\mu' \in O(\mu(G))$, we will need $O(\log\mu(G))$ rounds in total. Each round consists of testing of testing whether $\mu(v) \le \mu$ for each of the q dummy-vertices v. It therefore suffices to show how to test in $O(n+q)$ time whether $\mu(v) \le \mu$ for a given dummy-vertex v and value μ.

To do this test, we first perform a breadth-first search starting at v in $R(G)$ and cut it off once we reached all vertices of distance 2μ from v; this determines exactly the vertices of $\Lambda(v, \mu)$ whence we can obtain $\mathcal{B}(v, \mu)$. Test whether all edge-segments of the two edges that cross at v are in $\mathcal{B}(v, \mu)$; if not then $\mu(v) > \mu$ by Observation 5 and we can stop and return "no". Otherwise, we next modify $\mathcal{B}(v, \mu)$ so that all walks in the resulting graph $\mathcal{B}'(v, \mu)$ are G-respecting, i.e., go straight at every crossing. To do so, keep the vertices of $\mathcal{B}(v, \mu)$ and insert all those edges of G (not $G^{\times}$) for which the entire planarization lies in $\mathcal{B}(v, \mu)$. Then $\mathcal{B}(v, \mu)$ contains a ribbon if and only if two consecutive endpoints of crossing v are in the same connected component of $\mathcal{B}'(v, \mu)$, which can easily be tested in time $O(|\mathcal{B}'(v, \mu)|) = O(|\mathcal{B}(v, \mu)|) = O(n + q)$ time. ◀

Unfortunately we can have $q \in \Omega(n)$, even in 1-planar graphs, so the run-time can be $\Omega(n^2)$ even if μ is treated as a constant. This leads us to the following open problem.

▶ **Open problem.** *For a given embedded graph G, can we compute its ribbon radius in $o(n^2 \log(\mu(G)))$ time, and ideally $O(n)$ time? This is open even for 1-planar graphs.*

7 Outlook

In this paper, we explored the ribbon radius and showed that embedded graphs with small ribbon radius share the property with planar graphs that all vertices of a minimum vertex cut lie in a bounded diameter subgraph of $\Lambda(G)$. This property enabled a linear-time algorithm to test vertex and edge connectivity for many classes of near-planar graphs. Nonetheless, there exist k-plane graphs that are not amenable to our methods. In [3], we give a method to construct k-plane graphs with large ribbon radius, large connectivity, and large pairwise face distance between vertices (edges) of every minimum vertex (edge) cut. We also discuss the limitations of our method to graphs embeddable on surfaces of higher genus.

As for open problems, one wonders what properties and algorithms of planar graphs can be generalized to embedded graphs with small ribbon radius. Of special interest are weighted versions: Can the global mincut of a graph that is embedded with small ribbon radius be computed as fast as for planar graphs?

References

1 Michael A. Bekos, Michael Kaufmann, and Chrysanthi N. Raftopoulou. On optimal 2- and 3-planar graphs. In *33rd International Symposium on Computational Geometry, SoCG 2017, July 4-7, 2017, Brisbane, Australia*, volume 77 of *LIPIcs*, pages 16:1–16:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.SoCG.2017.16.

2 Michael A. Bekos, Giordano Da Lozzo, Svenja M. Griesbach, Martin Gronemann, Fabrizio Montecchiani, and Chrysanthi N. Raftopoulou. Book embeddings of k-framed graphs and k-map graphs. *Discret. Math.*, 347(1):113690, 2024. doi:10.1016/j.disc.2023.113690.

3 Therese Biedl, Prosenjit Bose, and Karthik Murali. A parameterized algorithm for vertex and edge connectivity of embedded graphs, 2024. arXiv:2407.00586.

4 Therese Biedl and Karthik Murali. On computing the vertex connectivity of 1-plane graphs. *CoRR*, abs/2212.06782, 2022. doi:10.48550/arXiv.2212.06782.

5 Therese Biedl and Karthik Murali. On computing the vertex connectivity of 1-plane graphs. In *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany*, volume 261 of *LIPIcs*, pages 23:1–23:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.ICALP.2023.23.

6 Li Chen, Rasmus Kyng, Yang P. Liu, Richard Peng, Maximilian Probst Gutenberg, and Sushant Sachdeva. Maximum flow and minimum-cost flow in almost-linear time. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 612–623. IEEE, 2022. doi:10.1109/FOCS54457.2022.00064.

7 Zhi-Zhong Chen, Michelangelo Grigni, and Christos H Papadimitriou. Map graphs. *Journal of the ACM (JACM)*, 49(2):127–138, 2002. doi:10.1145/506147.506148.

8 Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

9 Walter Didimo, Giuseppe Liotta, and Fabrizio Montecchiani. A survey on graph drawing beyond planarity. *ACM Computing Surveys (CSUR)*, 52(1):1–37, 2019. doi:10.1145/3301281.

10 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 5th edition, 2012.

11 David Eppstein. Subgraph isomorphism in planar graphs and related problems. *J. Graph Algorithms Appl.*, 3(3):1–27, 1999. doi:10.7155/jgaa.00014.

12 Mohsen Ghaffari, Krzysztof Nowicki, and Mikkel Thorup. Faster algorithms for edge connectivity via random 2-out contractions. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1260–1279. SIAM, 2020. doi:10.1137/1.9781611975994.77.

13 Monika Henzinger, Satish Rao, and Di Wang. Local flow partitioning for faster edge connectivity. *SIAM J. Comput.*, 49(1):1–36, 2020. doi:10.1137/18M1180335.

14 Seok-Hee Hong and Takeshi Tokuyama, editors. *Beyond Planar Graphs, Communications of NII Shonan Meetings*. Springer, 2020. doi:10.1007/978-981-15-6533-5.

15 Jason Li. Deterministic mincut in almost-linear time. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 384–395, 2021. doi:10.1145/3406325.3451114.

16 Jason Li, Danupon Nanongkai, Debmalya Panigrahi, Thatchaphol Saranurak, and Sorrachai Yingchareonthawornchai. Vertex connectivity in poly-logarithmic max-flows. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 317–329, 2021. doi:10.1145/3406325.3451088.

17 János Pach and Géza Tóth. Graphs drawn with few crossings per edge. *Combinatorica*, 17(3):427–439, 1997. doi:10.1007/BF01215922.

18 Thatchaphol Saranurak and Sorrachai Yingchareonthawornchai. Deterministic small vertex connectivity in almost linear time. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 789–800. IEEE, 2022. doi:10.1109/FOCS54457.2022.00080.

19 Marcus Schaefer. The graph crossing number and its variants: A survey. *Electronic Journal of Combinatorics*, 1000, 2013. doi:10.37236/2713.

20 Von H Schumacher. Zur Struktur 1-planarer Graphen. *Mathematische Nachrichten*, 125(1):291–300, 1986.