

On Graph Crossing Number and Edge Planarization*

Julia Chuzhoy[†] Yury Makarychev[‡] Anastasios Sidiropoulos[§]

Abstract

Given an n -vertex graph G , a *drawing* of G in the plane is a mapping of its vertices into points of the plane, and its edges into continuous curves, connecting the images of their endpoints. A *crossing* in such a drawing is a point where two such curves intersect. In the **Minimum Crossing Number** problem, the goal is to find a drawing of G with minimum number of crossings. The value of the optimal solution, denoted by OPT , is called the graph's *crossing number*. This is a very basic problem in topological graph theory, that has received a significant amount of attention, but is still poorly understood algorithmically. The best currently known efficient algorithm produces drawings with $O(\log^2 n) \cdot (n + \text{OPT})$ crossings on bounded-degree graphs, while only a constant factor hardness of approximation is known. A closely related problem is **Minimum Planarization**, in which the goal is to remove a minimum-cardinality subset of edges from G , such that the remaining graph is planar.

Our main technical result establishes the following connection between the two problems: if we are given a solution of cost k to the **Minimum Planarization** problem on graph G , then we can efficiently find a drawing of G with at most $\text{poly}(d) \cdot k \cdot (k + \text{OPT})$ crossings, where d is the maximum degree in G . This result implies an $O(n \cdot \text{poly}(d) \cdot \log^{3/2} n)$ -approximation for **Minimum Crossing Number**, as well as improved algorithms for special cases of the problem, such as, for example, k -apex and bounded-genus graphs.

1 Introduction

A *drawing* of a graph G in the plane is a mapping, in which every vertex is mapped into a point of the plane, and every edge into a continuous curve connecting the images of its endpoints. We assume that no three curves meet at the same point (except at their endpoints), and that no curve contains an image of any vertex other than its endpoints. A *crossing* in such a drawing is a point where the drawings of two edges intersect, and the *crossing number* of a graph G , denoted by $\text{OPT}_{\text{cr}}(G)$, is the smallest integer c , such that G admits a drawing with c crossings. In the **Minimum Crossing Number** problem, given an n -vertex graph G , the goal is to find a drawing of G in the plane that minimizes the number of crossings. A closely related problem is **Minimum Planarization**, in which the goal is to find a minimum-cardinality subset E^* of edges, such that the graph $G \setminus E^*$ is planar. The optimal solution cost of the **Minimum Planarization** problem on graph G is denoted by $\text{OPT}_{\text{MP}}(G)$, and it is easy to see that $\text{OPT}_{\text{MP}}(G) \leq \text{OPT}_{\text{cr}}(G)$.

The problem of computing the crossing number of a graph was first considered by Turán [38], who posed the question of estimating the crossing number of the complete bipartite graph. Since then, the problem has been a subject of intensive study. We refer the interested reader to the expositions by Richter and Salazar [33], Pach and Tóth [32], and Matoušek [29], and the extensive bibliography maintained by Vrt'o [39]. Despite the enormous interest in the problem, and several breakthroughs over the last four decades,

*Extended abstract to appear in SODA 2011

[†]Toyota Technological Institute, Chicago, IL 60637. Email: cjulia@ttic.edu. Supported in part by NSF CAREER award CCF-0844872.

[‡]Toyota Technological Institute, Chicago, IL 60637. Email: yury@ttic.edu

[§]Toyota Technological Institute, Chicago, IL 60637. Email: tasos@ttic.edu

there is still very little understanding of even some of the most basic questions. For example, to the time of this writing, the crossing number of K_{13} remains unknown.

Perhaps even more surprisingly, the **Minimum Crossing Number** problem remains poorly understood algorithmically. In their seminal paper, Leighton and Rao [27], combining their algorithm for balanced separators with the framework of Bhatt and Leighton [4], gave the first non-trivial algorithm for the problem. Their algorithm computes a drawing with at most $O(\log^4 n) \cdot (n + \text{OPT}_{\text{cr}}(G))$ crossings, when the degree of the input graph is bounded. This algorithm was later improved to $O(\log^3 n) \cdot (n + \text{OPT}_{\text{cr}}(G))$ by Even et al. [12], and the new approximation algorithm for the **Balanced Cut** problem by Arora, Rao and Vazirani [3] improves it further to $O(\log^2 n) \cdot (n + \text{OPT}_{\text{cr}}(G))$, thus implying an $O(n \cdot \log^2 n)$ -approximation for **Minimum Crossing Number** on bounded-degree graphs. Their result can also be shown to give an $O(n \cdot \log^2 n \cdot \text{poly}(d_{\text{max}}))$ -approximation for general graphs with maximum degree d_{max} . We remark that in the worst case, the crossing number of a graph can be as large as $\Omega(n^4)$, e.g. for the complete graph.

On the negative side, computing the crossing number of a graph was shown to be NP-complete by Garey and Johnson [13], and it remains NP-complete even on cubic graphs [18]. Combining the reduction of [13] with the inapproximability result for **Minimum Linear Arrangement** [2], we get that there is no PTAS for the **Minimum Crossing Number** problem unless problems in NP have randomized subexponential time algorithms. Interestingly, even for the very restricted special case, where there is an edge e in G , such that $G \setminus e$ is planar, the **Minimum Crossing Number** problem still remains NP-hard [7]. However, an $O(d_{\text{max}})$ -approximation algorithm is known for this special case, where d_{max} is the maximum degree in G [21]. Therefore, while the current techniques cannot exclude the existence of a constant factor approximation for **Minimum Crossing Number**, the state of the art gives just an $O(n \cdot \text{poly}(d_{\text{max}}) \cdot \log^2 n)$ -approximation algorithm.

In this paper, we provide new technical tools that we hope will lead to a better understanding of the **Minimum Crossing Number** problem. We also obtain improved approximation algorithms for special cases where the optimal solution for the **Minimum Planarization** problem is small or can be approximated efficiently.

1.1 Our Results

Our main technical result establishes the following connection between the **Minimum Crossing Number** and the **Minimum Planarization** problems:

Theorem 1 *Let $G = (V, E)$ be any n -vertex graph with maximum degree d_{max} , and suppose we are given a subset $E^* \subseteq E$ of edges, $|E^*| = k$, such that $H = G \setminus E^*$ is planar. Then we can efficiently find a drawing of G with at most $O(d_{\text{max}}^3 \cdot k \cdot (\text{OPT}_{\text{cr}}(G) + k))$ crossings.*

Remark 1 *Note that there always exists a subset E^* of edges of size $\text{OPT}_{\text{MP}}(G) \leq \text{OPT}_{\text{cr}}(G)$, such that $H = G \setminus E^*$ is planar. However, in Theorem 1, we do not assume that E^* is the optimal solution to the **Minimum Planarization** problem on G , and we allow k to be greater than $\text{OPT}_{\text{cr}}(G)$.*

A direct consequence of Theorem 1 is that an α -approximation algorithm for **Minimum Planarization** would immediately give an algorithm for drawing any graph G with $O(\alpha^2 \cdot d_{\text{max}}^3 \cdot \text{OPT}_{\text{cr}}^2(G))$ crossings. We note that while this connection between **Minimum Planarization** and **Minimum Crossing Number** looks natural, it is possible that in the optimal solution φ to the **Minimum Crossing Number** problem on G , the induced drawing of the planar subgraph $H = G \setminus E^*$ is not planar, that is, the edges of H may have to cross each other (see Figure 1 for an example).

Theorem 1 immediately implies a slightly improved algorithm for **Minimum Crossing Number**. In particular, while we are not aware of any approximation algorithms for the **Minimum Planarization** problem, the following is an easy consequence of the **Planar Separator** theorem of Lipton and Tarjan [28]:

Theorem 2 *There is an efficient $O(\sqrt{n \log n} \cdot d_{\max})$ -approximation algorithm for Minimum Planarization.*

The next corollary then follows from combining Theorems 1 and 2, and using the algorithm of [12].

Corollary 1 *There is an efficient algorithm, that, given any n -vertex graph G with maximum degree d_{\max} , finds a drawing of G with at most $O(n \log n \cdot d_{\max}^5) \text{OPT}_{\text{cr}}^2(G)$ crossings. Moreover, there is an efficient $O(n \cdot \text{poly}(d_{\max}) \cdot \log^{3/2} n)$ -approximation algorithm for Minimum Crossing Number.*

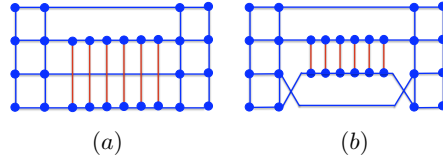


Figure 1: (a) Graph G . Red edges belong to E^* , blue edges to the planar sub-graph $H = G \setminus E^*$. Any drawing of G in which the edges of H do not cross each other has at least 6 crossings. (b) An optimal drawing of G , with 2 crossings.

Theorem 1 also implies improved algorithms for several special cases of the problem, that are discussed below.

Nearly-Planar and Apex Graphs. We say that a graph G is k -nearly planar, if it can be decomposed into a planar graph H , and a collection of at most k additional edges. For the cases where the decomposition is given, or where k is constant, Theorem 1 immediately gives an efficient $O(d_{\max}^3 \cdot k^2)$ -approximation algorithm for Minimum Crossing Number. It is worth noting that although this graph family might seem restricted, there has been a significant amount of work on the crossing number of 1-nearly planar graphs. Cabello and Mohar [7] proved that computing the crossing number remains NP-hard even for this special case, while Hliněný and Salazar [21] gave an $O(d_{\max})$ -approximation. Riskin [34] gave a simple efficient procedure for computing the crossing number when the planar sub-graph H is 3-connected, and Mohar [31] showed that Riskin’s technique cannot be extended to arbitrary 3-connected planar graphs. Gutwenger et al. [17] gave a linear-time algorithm for the case where every crossing is required to be between e and an edge of G .

A graph G is a k -apex graph iff there are k vertices v_1, \dots, v_k , whose removal makes it planar. Chimani et al. [8] obtained an $O(d_{\max}^2)$ -approximation for Minimum Crossing Number on 1-apex graphs. Theorem 1 immediately implies an $O(d_{\max}^5 \cdot k^2)$ -approximation for k -apex graphs, where either k is constant, or the k apices are explicitly given.

Bounded Genus Graphs. Recall that the genus of a graph G is the minimum integer g such that G can be drawn on an orientable surface of genus g with no crossings.

Börözký et al. [5] proved that the crossing number of a bounded-degree graph of bounded genus is $O(n)$. Djidjev and Venkatesan [9] show that $\text{OPT}_{\text{MP}}(G) \leq O(\sqrt{g \cdot n} \cdot d_{\max})$ for any genus- g graph. Moreover, if the embedding of G into a genus- g surface is given, a planarizing set of this size can be found in time $O(n + g)$. If no such embedding is given, they show how to efficiently compute a planarizing set of size $O(\sqrt{d_{\max} \cdot g \cdot n} \cdot \log g)$.

Hliněný and Chimani [19], building on the work of Gitler et al. [14] and Hliněný and Salazar [20] gave an algorithm for approximating Minimum Crossing Number on graphs that can be drawn “densely enough”¹ in an orientable surface of genus g , with an approximation guarantee of $2^{O(g)} d_{\max}^2$. Despite the rather technical conditions on the input, this is the largest family of graphs for which a constant-factor approximation for the crossing number is known. We prove the following easy consequence of Theorem 1 and the result of [19]:

¹More precisely, the density requirement is that the nonseparating dual edge-width of the drawing is $2^{\Omega(g)}$.

Theorem 3 *Let G be any graph embedded in an orientable surface of genus $g \geq 1$. Then we can efficiently find a drawing of G into the plane, with at most $2^{O(g)} \cdot d_{\max}^{O(1)} \cdot \text{OPT}_{\text{cr}}^2(G)$ crossings. Moreover, for any $g \geq 1$, there is an efficient $\tilde{O}(2^{O(g)} \cdot \sqrt{n})$ -approximation for Minimum Crossing Number on bounded degree graphs embedded into a genus- g surface.*

We notice that when g is a constant, a drawing of a genus- g graph on a genus- g surface can be found in linear time [30, 23].

1.2 Our Techniques

We now provide an informal overview of the proof of Theorem 1. We will use the words “drawing” and “embedding” interchangeably. We say that a drawing ψ of the planar graph $H = G \setminus E^*$ is *planar* iff ψ contains no crossings. Let φ be the optimal drawing of G , and let φ_H be the induced drawing of H . For simplicity, let us first assume that the graph H is 3-vertex connected. Then we can efficiently find a planar drawing ψ of H , which by Whitney’s Theorem [40] is guaranteed to be unique. Notice however that the two drawings φ_H and ψ of H are not necessarily identical, and in particular φ_H may be non-planar.

We now add the edges $e \in E^*$ to the drawing ψ of H . The algorithm for adding the edges is very simple. For each edge $e \in E^*$, we choose the drawing c_e that minimizes the number of crossings between c_e and the images of the edges of H in ψ . This task reduces to finding the shortest path in the graph dual to H . We can ensure that the drawings of any pair e, e' of edges in E^* cross at most once, by performing an un-crossing step, which does not increase the number of other crossings. Let ψ' denote this new drawing of the whole graph. The total number of crossings between pairs of edges that both belong to E^* is then bounded by k^2 , and it only remains to bound the number of crossings between the edges of E^* and the edges of H . In order to complete the analysis, it is enough, therefore, to show, that for every edge $e \in E^*$, there is a drawing of e in ψ , that has at most $\text{poly}(d_{\max})\text{OPT}_{\text{cr}}(G)$ crossings with the edges of H . Since our algorithm finds the best possible drawing for each edge e , the bound on the total number of crossings will follow.

One of our main ideas is the notion of *routing edges along paths*. Consider the optimal drawing φ of G , and let $e = (u, v)$ be some edge in E^* , that is mapped into some curve γ_e in φ . We show that we can find a path P_e in the graph H , whose endpoints are u and v , such that, instead of drawing the edge e along γ_e , we can draw it along a different curve γ'_e , that “follows” the drawing of the path P_e . That is, we draw γ'_e very close to the drawing of P_e , in parallel to it. Moreover, we show that this re-routing of the edge e along P_e does not increase the number of crossings in which it participates by much. Consider now the drawing of P_e in the planar embedding ψ of the graph H . We can again draw the edge e along the embedding of the same path P_e in ψ . Let γ''_e be the resulting curve. Since the embeddings φ_H and ψ are different, it is possible that γ''_e participates in more crossings than γ'_e . However, we show that the number of such new crossings can be bounded by the number of vertices and edges in P_e , whose local embeddings are different in φ and ψ . We then bound this number, in turn, by $\text{poly}(d_{\max})\text{OPT}_{\text{cr}}(G)$.

We now explain the notion of local embeddings in more detail. Given two drawings φ_H and ψ of the graph H , we say that a vertex $v \in V(H)$ is *irregular* iff the ordering of its adjacent edges, as their images enter v , is different in the two drawings. In other words, the local drawing around the vertex v is different in φ_H and ψ (see Figure 2(a)). We say that an edge $e = (u, v) \in E(H)$ is *irregular* iff both of its endpoints are not irregular, but their orientations are different. That is, the orderings of the edges adjacent to each one of the two endpoints are the same in both φ_H and ψ , but say, for vertex v , both orderings are clock-wise, while for vertex u , one is clock-wise and the other is counter-clock-wise (see Figure 2(b)). In a way, the number of irregular edges and vertices measures the difference between the two drawings. We show that, on the one hand, if H is 3-vertex connected, and ψ is a planar embedding of H , then the number of irregular vertices and edges is bounded by roughly the number of crossings in φ_H , which is in turn bounded by $\text{OPT}_{\text{cr}}(G)$. On the other hand, we show that for each edge $e \in E^*$, the number of new crossings incurred by the curve γ''_e is bounded by the total number of irregular edges and vertices on the path P_e , thus obtaining the desired

bound.

Assume now that H is not 3-vertex connected. In this case, it is easy to see that the number of irregular vertices and edges cannot be bounded by the number of crossings in φ_H anymore. In fact, it is possible that both ψ and φ_H are planar drawings of H , so the number of crossings in φ_H is 0, while the number of irregular vertices may be large (see Figure 3 for an example). However, if the original graph G was 3-vertex connected, then for any 2-vertex cut (u, v) in H , there is an edge $e \in E^*$ connecting the resulting two components of $H \setminus \{u, v\}$. We use this fact to find a specific planar drawing ψ' of H , that is “close” to φ_H , in the sense that, if we define the irregular edges and vertices with respect to the embeddings φ_H, ψ' of H , then we can bound their number by the number of crossings in φ_H .

Finally, if G is not 3-vertex connected, then we first decompose it into 3-vertex connected components, and then apply the above algorithm to each one of the components separately. In the end, we put all the resulting drawings together, while only losing a small additional factor in the number of crossings.

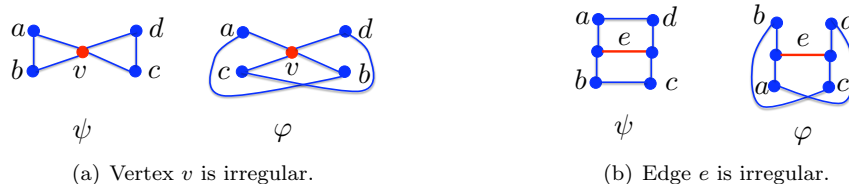


Figure 2: Irregular vertices and edges.

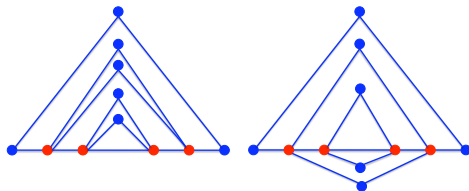


Figure 3: Example of planar drawings φ_H and ψ of graph H . Irregular vertices are shown in red.

1.3 Other Related work

Although it is impossible to summarize here the vast body of work on Minimum Crossing Number, we give a brief overview of some of the highlights, and related results.

Exact algorithms. Grohe [15], answering a question of Downey and Fellows [10], proved that the crossing number is fixed-parameter tractable. In particular, for any fixed number of crossings his algorithm computes an optimal drawing in $O(n^2)$ time. Building upon the breakthrough result of Mohar [30] for embedding graphs into a surface of bounded genus, Kawarabayashi and Reed [24] gave an improved fixed-parameter algorithm with running time $O(n)$.

Bounds on the crossing number of special graphs. Ajtai et al. [1], and independently Leighton [26], settling a conjecture of Erdős and Guy [11], proved that every graph with $m \geq 4n$ edges has crossing number $\Omega(m^3/n^2)$. Börözky et al. [5] proved that the crossing number of a bounded-degree graph of bounded genus is $O(n)$. This bound has been extended to all families of bounded-degree graphs that exclude a fixed minor by Wood and Telle [41]. Spencer and Tóth [35] gave bounds on the expected value of the crossing number of a random graph.

Organization Most of this paper is dedicated to proving Theorem 1. We start in Section 2 with prelimi-

naries, where we introduce some notation and basic tools. We then prove Theorem 1 in Section 3. We prove Theorem 2 and Corollary 1 in Section 4. The proof of Theorem 3 appears in Section D of the Appendix.

2 Preliminaries

In this section we provide some basic definitions and tools used in the proof of Theorem 1. In order to avoid confusion, throughout the paper, we denote the input graph by $\mathbf{G} = (V, E)$, with $|V| = n$, and maximum degree d_{\max} . We also denote $\mathbf{H} = \mathbf{G} \setminus E^*$, the planar sub-graph of \mathbf{G} (where E^* is the set of edges from the statement of Theorem 1), and by φ the optimal drawing of \mathbf{G} with $\text{OPT}_{\text{cr}}(\mathbf{G})$ crossings. When stating definitions or results for general arbitrary graphs, we will be denoting them by G and H , to distinguish them from the specific graphs \mathbf{G} and \mathbf{H} .

We use the words “drawing” and “embedding” interchangeably. Given any graph G , a drawing φ of G , and any sub-graph H of G , we denote by φ_H the drawing of H induced by φ , and by $\text{cr}_{\varphi}(G)$ the number of crossings in the drawing φ of G . For any pair $E_1, E_2 \subseteq E(G)$ of subsets of edges, we denote by $\text{cr}_{\varphi}(E_1, E_2)$ the number of crossings in φ in which images of edges of E_1 and edges of E_2 intersect, and by $\text{cr}_{\varphi}(E_1)$ the number of crossings in φ between pairs of edges that both belong to E_1 . Finally, for any curve γ , we denote by $\text{cr}_{\varphi}(\gamma, E_1)$ the number of crossings between γ and the images of the edges of E_1 , and $\text{cr}_{\varphi}(\gamma, H)$ denotes $\text{cr}_{\varphi}(\gamma, E(H))$. We will omit the subscript φ when clear from context. If G is a planar graph, and φ is a drawing of G that contains no crossings, then we say that φ is a *planar* drawing of G .

For the sake of brevity, we write $P : u \rightsquigarrow v$ to denote that a path P connects vertices u and v . Similarly, if we have a drawing of a graph, we write $\gamma : u \rightsquigarrow v$ to denote that a curve γ connects the images of vertices u and v (curve γ may not be a part of the current drawing). In order to avoid confusion, when a curve γ is a part of a drawing φ of some graph G , we write $\gamma \in \varphi$. We denote by $\Gamma(\varphi)$ the set of all curves that can be added to the drawing φ of G . In other words, these are all curves that do not contain images of vertices of G (except as their endpoints), and do not contain any crossing points of φ . Finally, for a graph $G = (V, E)$, and subsets $V' \subseteq V$, $E' \subseteq E$ of its vertices and edges respectively, we denote by $G \setminus V'$, $G \setminus E'$ the sub-graphs of G induced by $V \setminus V'$, and $E \setminus E'$, respectively.

Definition: For any graph $G = (V, E)$, a subset $V' \subseteq V$ of vertices is called a c -separator, iff $|V'| = c$, and the graph $G \setminus V'$ is not connected. We say that G is c -connected iff it does not contain any c' -separators, for any $c' < c$.

We will be using the following two well-known results:

Theorem 4 (Whitney [40]) *Every 3-connected planar graph has a unique planar embedding.*

Theorem 5 (Hopcroft-Tarjan [22]) *For any graph G , there is an efficient algorithm to determine whether G is planar, and if so, to find a planar drawing of G .*

Irregular Vertices and Edges Given any pair φ, ψ of drawings of a graph G , we measure the distance between them in terms of *irregular edges* and *irregular vertices*:

Definition: We say that a vertex x of G is irregular iff its degree is greater than 2, and the circular ordering of the edges incident on it, as their images enter x , is different in φ and ψ (ignoring the orientation). Otherwise we say that v is regular. We denote the set of irregular vertices by $\text{IRG}_V(\varphi, \psi)$. (See Figure 2(a)).

Definition: For any pair (x, y) of vertices in G , we say that a path $P : x \rightsquigarrow y$ in G is irregular iff x and y have degree at least 3, all other vertices on P have degree 2 in G , vertices x and y are regular, but their orientations differ in φ and ψ . That is, the orderings of the edges adjacent to x and to y are identical in both drawings, but the pairwise orientations are different: for one of the two vertices, the orientations are

identical in both drawings (say clock-wise), while for the other vertex, the orientations are opposite (one is clock-wise, and the other is counter-clock-wise). An edge e is an irregular edge iff it is the first or the last edge on an irregular path. In particular, if the irregular path only consists of edge e , then e is an irregular edge (see Figure 2(b)). If an edge is not irregular, then we say that it is regular. We denote the set of irregular edges by $\text{IRG}_E(\varphi, \psi)$.

Routing along Paths. One of the central concepts in our proof is that of routing along paths. Let G be any graph, and φ any drawing of G . Let $e = (u, v)$ be any edge of G , and let $P : u \rightsquigarrow v$ be any path connecting u to v in $G \setminus \{e\}$. It is possible that the image of P crosses itself in φ . We will first define a very thin strip S_P around the image of P in φ . We then say that the edge e is routed along the path P , iff its drawing follows the drawing of the path P inside the strip S_P , possibly crossing P .

In order to formally define the strip S_P , we first consider the graph G' , obtained from G , by replacing every edge of G with a 2-path containing $2\text{cr}_\varphi(G)$ inner vertices. The drawing φ of G then induces a drawing φ' of G' , such that, if P' is the path corresponding to P in G' , then every edge of G' crosses the image of P' at most once; every edge of $G' \setminus P'$ has at most one endpoint that belongs to P' ; and if an image of $e \notin P'$ crosses P' , then no endpoint of e belongs to P' . Let E_1 denote the subset of edges of $G' \setminus P'$ whose images cross the image of P' , let E_2 denote the subset of edges of P' whose images cross the images of other edges in P' , and let E_3 denote the set of edges in G' that have exactly one endpoint belonging to P' .

We now define a thin strip $S_{P'}$ around the drawing of path P' in φ' , by adding two curves, γ'_L and γ'_R , immediately to the left and to the right of the image of P' respectively, that follow the drawing of P' . Each edge in E_1 is crossed exactly once by γ'_L , and once by γ'_R . Each edge in E_3 is crossed exactly once by either γ'_R or γ'_L . For each pair (e, e') of edges in E_2 whose images cross, γ'_L and γ'_R will both cross each one of the edges e and e' exactly once. Curves γ'_L and γ'_R do not have any other crossings with the edges of G' . The region of the plane between the drawings of γ'_L and γ'_R , which contains the drawing of P' , defines the strip $S'_{P'}$. We let S_P denote the same strip, only when added to the drawing φ of G . Let γ_L and γ_R denote the two curves that form the boundary of S_P , and let $\gamma \in \{\gamma_L, \gamma_R\}$. Then the crossings between γ and the edges of G can be partitioned into four sets, C_1, C_2, C_3, C_4 (see Figure 4), where: (1) There is a 1 : 1 mapping between C_1 and the crossings between the edges of P and the edges of $G \setminus P$; (2) For each edge $e' \notin P$ that has exactly one endpoint in P , there is at most one crossing between γ and e' in C_2 , and there are no other crossings in C_2 ; (3) For each edge $e' \notin P$ that has exactly two endpoints in P , there are at most two crossings of γ and e' in C_3 , and there are no other crossings in C_3 ; and (4) for each crossing between a pair $e, e' \in P$ of edges, there is one crossing between γ and e , and one crossing between γ and e' . Additionally, if P crosses itself c times, then γ also crosses itself c times.

Definition: We say that the edge e is *routed along the path P* , iff its drawing follows the drawing of path P inside the strip S_P , in parallel to the drawing of P , except that it is allowed to cross the path P .

3 Proof of Theorem 1

The proof consists of two steps. We first assume that the input graph \mathbf{G} is 3-vertex connected, and prove a slightly stronger version of Theorem 1 for this case. Next, we show how to reduce the problem on general graphs to the 3-vertex connected case, while only losing a small additional factor in the number of crossings.

3.1 Handling 3-connected Graphs

In this section we assume that the input graph \mathbf{G} is 3-vertex connected, and we prove a slightly stronger version of Theorem 1 for this special case, that is summarized below.

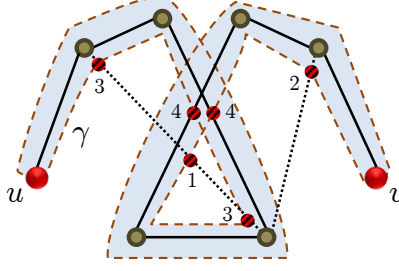


Figure 4: Strip S_p and the four types of crossing between γ and edges of G . Crossings in C_1 , C_2 , C_3 and C_4 are labeled with “1”, “2”, “3” and “4” respectively. Path P is shown in solid line, dotted lines correspond to other edges of G .

Theorem 6 *Let \mathbf{G}, \mathbf{H} and E^* be as in Theorem 1, and assume that \mathbf{G} is 3-connected and has no parallel edges. Then we can efficiently find a drawing of \mathbf{G} with at most $O(d_{\max} \cdot k \cdot (\text{OPT}_{\text{cr}}(\mathbf{G}) + k))$ crossings.*

Notice that we can assume w.l.o.g. that graph \mathbf{H} is connected. Otherwise, we can choose an edge $e \in E^*$ whose endpoints belong to two distinct connected components of \mathbf{H} , remove e from E^* and add it to \mathbf{H} . It is easy to see that this operation preserves the planarity of \mathbf{H} , and we can repeat it until \mathbf{H} becomes connected. We therefore assume from now on that \mathbf{H} is connected.

Recall that φ denotes the optimal drawing of \mathbf{G} , and $\varphi_{\mathbf{H}}$ is the drawing of \mathbf{H} induced by φ . Since the graph \mathbf{H} is planar, we can efficiently find its planar drawing, using Theorem 5. However, since \mathbf{H} is not necessarily 3-connected, there could be a number of such drawings, and we need to find one that is “close” to $\varphi_{\mathbf{H}}$. We use the following theorem, whose proof appears in Appendix.

Theorem 7 *We can efficiently find a planar drawing ψ of \mathbf{H} , such that*

$$\begin{aligned} |\text{IRG}_V(\psi, \varphi_{\mathbf{H}})| &= O(\text{OPT}_{\text{cr}}(\mathbf{G}) + k) \\ |\text{IRG}_E(\psi, \varphi_{\mathbf{H}})| &= O(d_{\max})(\text{OPT}_{\text{cr}}(\mathbf{G}) + k). \end{aligned}$$

We are now ready to describe the algorithm for finding a drawing of \mathbf{G} . We start with the planar embedding ψ of \mathbf{H} , guaranteed by Theorem 7. For every edge $e = (u, v) \in E^*$, we add an embedding of e to the drawing ψ of \mathbf{H} , via a curve $\gamma_e \in \Gamma(\psi)$, $\gamma_e : u \rightsquigarrow v$, that crosses the minimum possible number of edges of \mathbf{H} . Such a curve can be computed as follows. Let \mathbf{H}^{dual} be the dual graph of the drawing ψ of \mathbf{H} . Every curve $\gamma \in \Gamma(\psi)$, $\gamma : u \rightsquigarrow v$, defines a path in \mathbf{H}^{dual} . The length of the path, measured in the number of edges of \mathbf{H}^{dual} it contains, is exactly the number of edges of \mathbf{H} that γ crosses. Similarly, every path in \mathbf{H}^{dual} corresponds to a curve in $\Gamma(\psi)$. Let \mathcal{U} be the set of all faces of ψ (equivalently, vertices of \mathbf{H}^{dual}) whose boundaries contain u , and let \mathcal{V} be the set of all faces whose boundaries contain v . We find the shortest path $P_{(u,v)}$ between sets \mathcal{U} and \mathcal{V} in \mathbf{H}^{dual} , and the corresponding curve $\gamma_{(u,v)} : u \rightsquigarrow v$ in $\Gamma(\psi)$. Clearly, the number of crossings between $\gamma_{(u,v)}$ and the edges of \mathbf{H} is minimal among all curves connecting u and v in $\Gamma(\psi)$. By slightly perturbing the lengths of edges in \mathbf{H}^{dual} , we may assume that for every pair of vertices in \mathbf{H}^{dual} , there is exactly one shortest path connecting them. In particular, any pair of such shortest paths may share at most one consecutive segment. Consequently, for any pair $e, e' \in E^*$ of edges, the drawings $\gamma_e, \gamma_{e'}$ that we have obtained cross at most once.

Let ψ' denote the union of ψ with the drawings γ_e of edges $e \in E^*$ that we have computed. It now only remains to bound the number of crossings in ψ' . Clearly, $\text{cr}_{\psi'}(\mathbf{G}) = \text{cr}_{\psi'}(E^*) + \text{cr}_{\psi'}(E^*, E(\mathbf{H})) \leq k^2 + \sum_{e \in E^*} \text{cr}_{\psi'}(\gamma_e, E(\mathbf{H}))$. In order to bound $\text{cr}_{\psi'}(\gamma_e, E(\mathbf{H}))$, we use the following theorem, whose proof appears in the next section.

Theorem 8 *Let φ and ψ be two drawings of any planar connected graph H , whose maximum degree is d_{\max} , where ψ is a planar drawing. Then for every curve $\gamma \in \Gamma(\varphi)$, $\gamma : u \rightsquigarrow v$ there is a curve $\gamma' \in \Gamma(\psi)$, $\gamma' : u \rightsquigarrow v$, that participates in at most $O(\text{cr}_\varphi(H) + \text{cr}_\varphi(\gamma, E(H)) + |\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|)$ crossings.*

In other words, the number of additional crossings incurred by γ' is roughly bounded by the total number of crossings in φ , and the difference between the two drawings, that is, the number of irregular vertices and edges.

Since the optimal embedding φ of G contains an embedding of every edge $e \in E^*$, Theorem 8 guarantees that for every edge $e = (u, v) \in E^*$, there is a curve $\gamma'_e : u \rightsquigarrow v$ in $\Gamma(\psi)$, that participates in at most $O(\text{cr}_\varphi(\mathbf{H}) + \text{cr}_\varphi(e, E(\mathbf{H})) + |\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|) \leq O(\text{OPT}_{\text{cr}}(\mathbf{G}) + |\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|)$ crossings. Combining this with Theorem 7, the number of crossings between γ'_e and $E(H)$ is bounded by $O(d_{\max})(\text{OPT}_{\text{cr}}(\mathbf{G}) + k)$. Since for each edge $e \in E^*$, our algorithm chooses the optimal curve γ_e , we are guaranteed that γ_e participates in at most $O(d_{\max})(\text{OPT}_{\text{cr}}(\mathbf{G}) + k)$ crossings with edges of H . Summing up over all edges $e \in E^*$, we obtain that $\text{cr}_{\psi'}(\mathbf{G}) \leq k^2 + k \cdot O(d_{\max})(\text{OPT}_{\text{cr}}(\mathbf{G}) + k) \leq O(d_{\max} \cdot k \cdot (\text{OPT}_{\text{cr}}(\mathbf{G}) + k))$, as required. In order to complete the proof of Theorem 6, it now only remains to prove Theorem 8.

3.2 Proof of Theorem 8: Routing along Paths

The proof consists of two steps. In the first step, we focus on the drawing φ of H , and we show that for any curve $\gamma : u \rightsquigarrow v$ in $\Gamma(\varphi)$, there is a path $P : u \rightsquigarrow v$ in H , and another curve $\gamma^* : u \rightsquigarrow v$ in $\Gamma(\varphi)$ routed along P in φ , such that the number of crossings in which γ^* is involved is small. In the second step, we consider the planar drawing ψ of H , and show how to route a curve $\gamma' : u \rightsquigarrow v$ along the same path P in ψ , so that the number of crossings is suitably bounded. The next proposition handles the first step of the proof.

Proposition 1 *Let $\gamma : u \rightsquigarrow v$ be any curve in $\Gamma(\varphi)$, where φ is a drawing of H . Then there is a path $P : u \rightsquigarrow v$ in H , and a curve $\gamma^* : u \rightsquigarrow v$ in $\Gamma(\varphi)$ routed along P , such that $\text{cr}_\varphi(\gamma^*, H) \leq O(\text{cr}_\varphi(H) + \text{cr}_\varphi(\gamma, H))$. Moreover, γ^* does not cross the images of the edges of P . Path P is not necessarily simple, but an edge may appear at most twice on P .*

Proof: Consider the drawing φ of H , together with the curve γ . Let $E_1 \subseteq E(H)$ be the subset of edges whose images cross the images of other edges of H , and let $E_2 \subseteq E(H) \setminus E_1$ be the subset of edges whose images cross γ and that are not in E_1 . Let $H' = H \setminus (E_1 \cup E_2)$. Note that $\varphi_{H'}$ is a planar drawing of H' , and γ does not cross any edges of H' . Therefore, vertices u and v lie on the boundary of one face, denoted by F , of $\varphi_{H'}$. Without loss of generality, we may assume that F is the outer face of $\varphi_{H'}$. The boundary of F consists of one or several connected components. Let B_1, \dots, B_r be the boundary walks of the face F (where $r \geq 1$ is the number of connected components): each B_i is the (not necessarily simple) cycle obtained by walking around the boundary of the i th connected component, if the component contains at least 2 vertices; and it is a single vertex otherwise.

Consider two cases. First, assume that u and v are connected in H' , and so they both belong to the same component B_i . We then let P be one of the two segments of B_i that connect u and v . Notice that while P is not necessarily simple, each edge appears at most twice on it. We let γ^* be a curve drawn along the path P inside the face F . Notice that the only edges that γ^* crosses in the drawing φ of H , are the edges of $E_1 \cup E_2$ that have at least one endpoint on P . Each such edge is crossed at most twice by γ^* (once for each endpoint that belongs to P). Therefore, $\text{cr}_\varphi(\gamma^*, H) \leq O(|E_1| + |E_2|) \leq O(\text{cr}_\varphi(H) + \text{cr}_\varphi(\gamma, H))$. Assume now that u and v are not connected in H' , and assume w.l.o.g. that $u \in B_1$ and $v \in B_2$. Let L be a minimal set of edges of $E_1 \cup E_2$, such that u and v are connected in $H' \cup L$. Each edge $e \in L$ connects two distinct components B_i and B_j (as otherwise we could remove e without affecting the connectivity of $H' \cup L$). In particular, the drawings of all edges of L in φ lie inside the face F . Consider the following graph H^* : each vertex of H^* corresponds to a component B_i , for $1 \leq i \leq r$, and the edges of H^* are the edges of L connecting these

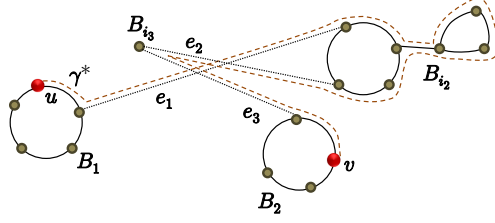


Figure 5: Routing the curve $\gamma^* : u \rightsquigarrow v$.

components. Since u and v are connected in $H' \cup L$, vertices representing B_1 and B_2 belong to the same connected component C of H^* . Moreover, because of the minimality of L , this connected component is a simple path P' connecting the vertices representing B_1 and B_2 in H^* .

Denote the edges of this path by e_1, \dots, e_{q-1} ; denote its vertices by $B_{i_1} \equiv B_1, B_{i_2}, \dots, B_{i_q} \equiv B_2$; each edge e_j , for $1 \leq j \leq q-1$, corresponds to an edge $e'_j \in L$, that connects a pair (x_j, y_j) of vertices, where $x_j \in B_{i_j}$, $y_j \in B_{i_{j+1}}$, in H . We denote $u = y_0, v = x_q$. Notice that since P' is simple, each component B_j , $1 \leq j \leq r$ appears at most once on the path. We now define the path P and the curve γ^* . The path P is defined as follows: $P = (P_1, e'_1, P_2, e'_2, \dots, e'_{q-1}, P_q)$, where for each $j : 1 \leq j \leq q$, P_j is obtained by traversing the boundary B_{i_j} in the clock-wise direction from y_{j-1} to x_j . The curve $\gamma^* : u \rightsquigarrow v$ is simply routed along P_j on the inside of the face F . That is, curve γ^* never crosses the images of the edges of H' (see Figure 5).

We now bound the number of edges of H , whose images in φ are being crossed by γ^* . We partition the crossings in which γ^* participates into four sets C_1, C_2, C_3, C_4 , like in the definition of routing along paths in Section 2. The only edges incident on vertices of path P that γ^* crosses are the edges in $E_1 \cup E_2$, and each such edge contributes at most two crossings to $C_2 \cup C_3$, while the number of crossings in $C_1 \cup C_4$ is bounded by $2\text{cr}_\varphi(H)$. Therefore, $\text{cr}_\varphi(\gamma^*, H) \leq O(|E_1| + |E_2| + \text{cr}_\varphi(H)) = O(\text{cr}_\varphi(H) + \text{cr}_\varphi(\gamma, H))$. \square

We now focus on the other embedding, ψ of H , and show how to obtain the final curve $\gamma' : u \rightsquigarrow v$, $\gamma' \in \Gamma(\psi)$, that participates in a small number of crossings.

Proposition 2 *There is a curve $\gamma' : u \rightsquigarrow v$ in $\Gamma(\psi)$, that has no self-crossings, and participates in at most $\text{cr}_\varphi(\gamma^*, H) + O(|\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|)$ crossings with the edges of H .*

Proof: We will route γ' along the path P in ψ . Since an edge may appear at most twice on P , path P may visit a vertex at most d_{\max} times. We will assume however that P visits every irregular vertex at most once, by changing P as follows: whenever an irregular vertex x appears more than once on P , we create a shortcut, by removing the segment of P that lies between the two consecutive appearances of x on P . As a result, in the final path P , each edge appears at most twice, and each irregular vertex at most once.

We will route the curve γ' along P , but we will allow it to cross the image of the path P . Therefore, we only need to specify, for each edge $e \in P$, whether γ' crosses it, and if not, on which side of e it is routed. Since ψ is planar, the edges of P do not cross each other.

We partition the path P into consecutive segments $\tau_0, \sigma_1, \tau_1, \sigma_2, \dots, \sigma_t, \tau_t$, where for each $j : 1 \leq j \leq t$, σ_j contains regular edges only, and all its vertices are regular, except perhaps the first and the last. For each $j : 0 \leq j \leq t$, either τ_j contains one or several consecutive irregular edges connecting the last vertex of σ_j and the first vertex of σ_{j+1} ; or it contains a single irregular vertex, which serves as the last vertex of σ_j and the first vertex of σ_{j+1} .

Consider some such segment σ_j , and a thin strip $S = S_{\sigma_j}$ around this segment. Then the parts of the drawings of the edges incident on the vertices of σ_j , that fall inside S are identical in both φ and ψ (except possibly for the edges incident on the first and the last vertex of σ_j). We can therefore route γ' along the

same side of σ_j along which γ^* is routed. If necessary, we may need to cross the path P once for each consecutive pair of segments, if the routings are performed on different sides of P . Let γ_j^* and γ_j' denote the segments of γ^* and γ' , respectively, that are routed along σ_j , and include crossings with all edges incident on σ_j . It is easy to see that the difference $\text{cr}_\psi(\gamma_j', H) - \text{cr}_\psi(\gamma_j^*, H)$ is bounded by $2d_{\max}$: we pay at most d_{\max} for crossing the edges incident on each endpoint of σ_j , which may be an irregular vertex. We may additionally pay 1 crossing for each irregular edge on P . Since each irregular vertex appears at most once on P , and each irregular edge at most twice, $\text{cr}_\psi(\gamma', H) - \text{cr}_\psi(\gamma^*, H) \leq O(|\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|)$. Finally, if γ' crosses itself, we can simply short-cut it by removing all resulting loops. \square

Combining Propositions 1 and 2, we get that $\text{cr}_\varphi(\gamma^*, H) \leq O(\text{cr}_\varphi(H) + \text{cr}_\varphi(\gamma, H))$, and $\text{cr}_\psi(\gamma', H) \leq \text{cr}_\varphi(\gamma^*, H) + O(|\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|) \leq O(\text{cr}_\varphi(H) + \text{cr}_\varphi(\gamma, H) + |\text{IRG}_E(\varphi, \psi)| + d_{\max}|\text{IRG}_V(\varphi, \psi)|)$.

3.3 Non 3-Connected Graphs

We briefly explain how to reduce the general case to the 3-connected case. We decompose the graph into a collection of sub-graphs. For each sub-graph, we find a drawing separately, and then combine them together to obtain the final solution. Each one of the sub-graphs is either a 3-connected graph, for which we can find a drawing using Theorem 6, or it can be decomposed into a planar graph plus one additional edge. In the latter case, we employ the algorithm of Hlineny and Salazar [21] to find an $O(d_{\max})$ -approximate drawing. The detailed proof of this part is presented in Section C in the Appendix.

4 Improved Algorithm for General Graphs

In this section we prove Theorem 2 and Corollary 1. We will rely on the Planar Separator Theorem of Lipton and Tarjan [28], and on the approximation algorithm for the Balanced Cut problem of Arora, Rao and Vazirani [3], that we state below.

Theorem 9 (Planar Separator Theorem [28]) *Let G be any n -vertex planar graph. Then there is an efficient algorithm to partition the vertices of G into three sets A, B, C , such that $|A|, |C| \leq 2n/3$, $|B| \leq O(\sqrt{n})$, and there are no edges in G connecting the vertices of A to the vertices of C .*

Theorem 10 (Balanced Cut [3]) *Let G be any n -vertex graph, and suppose there is a partition of vertices of G into two sets, A and C , with $|A|, |C| \leq 2n/3$, and $|E(A, C)| = c$. Then there is an efficient algorithm to find a partition (A', C') of vertices of G , such that $|A'|, |C'| \leq \alpha n$ for some constant $\alpha < 1$, and $|E(A', C')| \leq O(c\sqrt{\log n})$.*

Combining the two theorems together, we get the following corollary:

Corollary 2 *Let G be any n -vertex graph with maximum degree d_{\max} . Then there is an efficient algorithm to partition the vertices of G into two sets A', C' , with $|A'|, |C'| \leq \alpha n$ for some constant α , such that $|E(A', C')| \leq O(\sqrt{\log n})(d_{\max}\sqrt{n} + \text{OPT}_{\text{MP}}(G))$.*

Proof: Let E^* be an optimal solution for the Minimum Planarization problem on G , $|E^*| = \text{OPT}_{\text{MP}}(G)$, and let $H = G \setminus E^*$. Since H is a planar graph, there is a partition (A, B, C) of its vertices as in Theorem 9. Assume w.l.o.g. that $|A| \leq |C|$, and consider the partition $(A \cup B, C)$. Then $|A \cup B|, |C| \leq 2n/3$, and $|E_G(A \cup B, C)| \leq |E_H(A \cup B, C)| + |E^*| \leq O(d_{\max}\sqrt{n}) + \text{OPT}_{\text{MP}}(G)$. We can now apply Theorem 10 to obtain the desired partition of G . \square

We are now ready to describe the algorithm from Theorem 2. The algorithm consists of $O(\log n)$ iterations, and in each iteration i , we are given a collection $G_1^i, \dots, G_{k_i}^i$ of disjoint sub-graphs of G , with $k_i \leq \text{OPT}_{\text{MP}}(G)$.

The number of vertices in each such sub-graph is bounded by $n_i = \alpha^{i-1}n$, where $\alpha < 1$ is the constant from Corollary 2. In the input to the first iteration, $k_1 = 1$, and $G_1^1 = G$. Iteration i , for $i \geq 1$ is performed as follows. Consider some graph G_j^i , for $1 \leq j \leq k_i$. We apply Corollary 2 to this graph, and denote by H_j, H'_j the two sub-graphs of G_j^i induced by A' and C' , respectively. The number of vertices in each one of the subgraphs is at most $\alpha \cdot |V(G_j^i)| \leq \alpha n_i = n_{i+1}$. We denote by E_j^i the corresponding set of edges $E(A', C')$, and let $E^i = \bigcup_{j=1}^{k_i} E_j^i$. Since for all j , $|E_j^i| \leq O(\sqrt{\log n})(d_{\max}\sqrt{n_i} + \text{OPT}_{\text{MP}}(G_j^i))$, and $\sum_{j=1}^{k_i} \text{OPT}_{\text{MP}}(G_j^i) \leq \text{OPT}_{\text{MP}}(G)$, we get that $|E^i| \leq O(\sqrt{\log n})(k_i d_{\max}\sqrt{n_i} + \text{OPT}_{\text{MP}}(G)) \leq O(d_{\max}\sqrt{\log n} \cdot \sqrt{n_i})\text{OPT}_{\text{MP}}(G)$, as $k_i \leq \text{OPT}_{\text{MP}}(G)$. Finally, consider the collection $\mathcal{G}_{i+1} = \{H_1, H'_1, \dots, H_{k_i}, H'_{k_i}\}$ of the new graphs, and let $\mathcal{G}'_{i+1} \subseteq \mathcal{G}_{i+1}$ contain the non-planar graphs. Then $|\mathcal{G}'_{i+1}| \leq \text{OPT}_{\text{MP}}(G)$, and the graphs in \mathcal{G}'_{i+1} become the input to the next iteration. Since we can efficiently check whether a graph is planar, the set \mathcal{G}'_{i+1} can be computed efficiently.

The algorithm stops, when all remaining sub-graphs contain at most $O(\sqrt{\log n})$ edges. We then add the edges of all remaining sub-graphs to set E^{i^*} , where $i^* = O(\log n)$ is the last iteration. Our final solution is $E' = \bigcup_{i=1}^{i^*} E^i$, and its cost is bounded by $|E'| \leq \sum_{i=1}^{i^*} |E^i| \leq \sum_{i=1}^{i^*} O(d_{\max}\sqrt{\log n} \cdot \sqrt{n_i})\text{OPT}_{\text{MP}}(G) \leq O(d_{\max}\sqrt{n \log n})\text{OPT}_{\text{MP}}(G)$, since the values $\sqrt{n_i}$ form a decreasing geometric series for $i \geq 1$. This finishes the proof of Theorem 2. We now show how to obtain Corollary 1. Combining Theorems 1 and 2, we immediately obtain an efficient algorithm for drawing any graph G with at most $O(n \log n \cdot d_{\max}^5)\text{OPT}_{\text{cr}}^2(G)$ crossings. In order to get the approximation guarantee of $O(n \cdot \text{poly}(d_{\max}) \cdot \log^{3/2} n)$, we use an extension of the result of Even et al. [12] to arbitrary graphs, that we formulate in the next theorem, whose proof appears in Appendix.

Theorem 11 (Extension of [12]) *There is an efficient algorithm that, given any n -vertex graph G with maximum degree d_{\max} , outputs a drawing of G with $O(\text{poly}(d_{\max}) \log^2 n)(n + \text{OPT}_{\text{cr}}(G))$ crossings.*

We run our algorithm, and the algorithm given by Theorem 11 on the input graph G , and output the better of the two solutions. If $\text{OPT}_{\text{cr}}(G) \geq \sqrt{\log n}$, then the algorithm of Even et al. is an $O(n \cdot \text{poly}(d_{\max}) \cdot \log^{3/2} n)$ -approximation. Otherwise, our algorithm gives an $O(n \cdot \text{poly}(d_{\max}) \cdot \log^{3/2} n)$ -approximation.

References

- [1] M. AJTAI, V. CHVÁTAL, M. NEWBORN, AND E. SZEMERÉDI, *Crossing-free subgraphs*, Theory and Practice of Combinatorics, (1982), pp. 9–12.
- [2] C. AMBUHL, M. MASTROLILLI, AND O. SVENSSON, *Inapproximability results for sparsest cut, optimal linear arrangement, and precedence constrained scheduling*, in Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science, 2007, pp. 329–337.
- [3] S. ARORA, S. RAO, AND U. V. VAZIRANI, *Expander flows, geometric embeddings and graph partitioning*, J. ACM, 56 (2009), pp. 1–37.
- [4] S. N. BHATT AND F. T. LEIGHTON, *A framework for solving VLSI graph layout problems*, J. Comput. Syst. Sci., 28 (1984), pp. 300–343.
- [5] K. J. BÖRÖZKY, J. PACH, AND G. TÓTH, *Planar crossing numbers of graphs embeddable in another surface*, Int. J. Found. Comput. Sci., 17 (2006), pp. 1005–1016.
- [6] S. CABELLO AND E. W. CHAMBERS, *Multiple source shortest paths in a genus g graph*, in SODA, 2007, pp. 89–97.
- [7] S. CABELLO AND B. MOHAR, *Adding one edge to planar graphs makes crossing number hard*, in Proc. ACM Symp. on Computational Geometry, 2010, pp. 68–76.

- [8] M. CHIMANI, P. HLINĚNÝ, AND P. MUTZEL, *Approximating the crossing number of apex graphs*, in Graph Drawing, I. Tollis and M. Patrignani, eds., vol. 5417 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2009, pp. 432–434.
- [9] H. DJIDJEV AND S. M. VENKATESAN, *Planarization of graphs embedded on surfaces*, in WG '95: Proceedings of the 21st International Workshop on Graph-Theoretic Concepts in Computer Science, vol. 1017 of Lecture Notes in Computer Science, London, UK, 1995, Springer-Verlag, pp. 62–72.
- [10] R. G. DOWNEY AND M. R. FELLOWS, *Parameterized Complexity*, Springer-Verlag, 1999.
- [11] P. ERDŐS AND R. GUY, *Crossing number problems*, Amer. Math. Monthly, 80 (1973), pp. 52–58.
- [12] G. EVEN, S. GUHA, AND B. SCHIEBER, *Improved approximations of crossings in graph drawings and VLSI layout areas*, SIAM J. Comput., 32 (2002), pp. 231–252.
- [13] M. R. GAREY AND D. S. JOHNSON, *Crossing number is NP-complete*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 312–316.
- [14] I. GITLER, P. HLINĚNÝ, J. LEAÑOS, AND G. SALAZAR, *The crossing number of a projective graph is quadratic in the face-width*, Electronic Notes in Discrete Mathematics, 29 (2007), pp. 219–223.
- [15] M. GROHE, *Computing crossing numbers in quadratic time*, J. Comput. Syst. Sci., 68 (2004), pp. 285–302.
- [16] C. GUTWENGER AND P. MUTZEL, *A linear time implementation of spqr-trees*, in GD '00: Proceedings of the 8th International Symposium on Graph Drawing, vol. 1984 of Lecture Notes in Computer Science, London, UK, 2001, Springer-Verlag, pp. 77–90.
- [17] C. GUTWENGER, P. MUTZEL, AND R. WEISKIRCHER, *Inserting an edge into a planar graph*, Algorithmica, 41 (2005), pp. 289–308.
- [18] P. HLINĚNÝ, *Crossing number is hard for cubic graphs*, J. Comb. Theory, Ser. B, 96 (2006), pp. 455–471.
- [19] P. HLINĚNÝ AND M. CHIMANI, *Approximating the crossing number of graphs embeddable in any orientable surface*, in Proc. 21st Annual ACM-SIAM Symposium on Discrete Algorithms, 2010, pp. 918–927.
- [20] P. HLINĚNÝ AND G. SALAZAR, *Approximating the crossing number of toroidal graphs*, Lecture Notes in Computer Science, 4835/2007 (2007), pp. 148–159.
- [21] P. HLINĚNÝ AND G. SALAZAR, *On the crossing number of almost planar graphs*, in Graph Drawing, M. Kaufmann and D. Wagner, eds., vol. 4372 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007, pp. 162–173.
- [22] J. HOPCROFT AND R. TARJAN, *Efficient planarity testing*, J. ACM, 21 (1974), pp. 549–568.
- [23] K. KAWARABAYASHI, B. MOHAR, AND B. A. REED, *A simpler linear time algorithm for embedding graphs into an arbitrary surface and the genus of graphs of bounded tree-width*, in FOCS '08: Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science, 2008, pp. 771–780.
- [24] K. KAWARABAYASHI AND B. A. REED, *Computing crossing number in linear time*, in STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing, 2007, pp. 382–390.
- [25] P. KOLMAN AND J. MATOUŠEK, *Crossing number, pair-crossing number, and expansion*, J. Comb. Theory, Ser. B, 92 (2004), pp. 99–113.
- [26] F. T. LEIGHTON, *Complexity issues in VLSI: optimal layouts for the shuffle-exchange graph and other networks*, MIT Press, 1983.

- [27] F. T. LEIGHTON AND S. RAO, *Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms*, Journal of the ACM, 46 (1999), pp. 787–832.
- [28] R. J. LIPTON AND R. E. TARJAN, *A separator theorem for planar graphs*, SIAM Journal on Applied Mathematics, 36 (1979), pp. 177–189.
- [29] J. MATOUŠEK, *Lectures on discrete geometry*, Springer-Verlag, 2002.
- [30] B. MOHAR, *A linear time algorithm for embedding graphs in an arbitrary surface*, SIAM J. Discrete Math., 12 (1999), pp. 6–26.
- [31] ———, *On the crossing number of almost planar graphs*, Informatica (Slovenia), 30 (2006), pp. 301–303.
- [32] J. PACH AND G. TÓTH, *Thirteen problems on crossing numbers*, Geombinatorics, 9 (2000), pp. 194–207.
- [33] R. B. RICHTER AND G. SALAZAR, *Crossing numbers*, in Topics in Topological Graph Theory, L. W. Beineke and R. J. Wilson, eds., Cambridge University Press, 2009, ch. 7, pp. 133–150.
- [34] A. RISKIN, *The crossing number of a cubic plane polyhedral map plus an edge*, Studia Sci. Math. Hungar., 31 (1996), pp. 405–413.
- [35] J. SPENCER AND G. TÓTH, *Crossing numbers of random graphs*, Random Struct. Algorithms, 21 (2002), pp. 347–358.
- [36] L. A. SZEKELY, *Progress on crossing number problems*, in SOFSEM, vol. 3381 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2005, pp. 53–61.
- [37] C. THOMASSEN, *The graph genus problem is np-complete*, J. Algorithms, 10 (1989), pp. 568–576.
- [38] P. TURÁN, *A note of welcome*, J. Graph Theory, 1 (1977), pp. 1–5.
- [39] I. VRTO, *Crossing numbers of graphs: A bibliography*. <http://www.ifi.savba.sk/~imrich>.
- [40] H. WHITNEY, *Congruent graphs and the connectivity of graphs*, American Journal of Mathematics, 54 (1932), pp. 150–168.
- [41] D. R. WOOD AND J. A. TELLE, *Planar decompositions and the crossing number of graphs with an excluded minor*, in Graph Drawing 2006, vol. 4372 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2007, pp. 150–161.

A Block Decompositions

In this section we introduce the notion of *blocks*, and present a theorem for computing block decompositions of graphs, that we will later use to handle graphs that are not 3-connected.

Definition: Let $G = (V, E)$ be a 2-connected graph. A subgraph $B = (V', E')$ of G is called a *block* iff:

- $V \setminus V' \neq \emptyset$ and $|V'| \geq 3$;
- There are two special vertices $u, v \in V'$, called *block end-points* and denoted by $I(B) = (u, v)$, such that there are no edges connecting vertices in $V \setminus V'$ to $V' \setminus \{u, v\}$ in G , that is, $E(V \setminus V', V' \setminus \{u, v\}) = \emptyset$. All other vertices of B are called *inner vertices*;
- B is the subgraph of G induced by V' , except that it **does not** contain the edge $\{u, v\}$ even if it is present in G .

Notice that every 2-separator (u, v) of G defines at least two internally disjoint blocks B', B'' with $I(B'), I(B'') = (u, v)$.

Definition: Let \mathcal{F} be a laminar family of sub-graphs of G , and let \mathcal{T} be the decomposition tree associated with \mathcal{F} . We say that \mathcal{F} is a *block decomposition* of G , iff:

- The root of the tree \mathcal{T} is G , and all other vertices of \mathcal{T} are blocks. For consistency, we will call the root vertex “block” as well.
- For each block $B \in \mathcal{F}$, let \tilde{B} be the graph obtained by replacing each child B' of B with an artificial edge connecting its endpoints. Let \tilde{B}' be the graph obtained from \tilde{B} by adding an artificial edge connecting the endpoints of B (for the root vertex G , $\tilde{G} = \tilde{G}$). Then \tilde{B}' is 3-connected.
- If a block $B \in \mathcal{F}$ has exactly one child B' then $I(B) \neq I(B')$.

The next theorem states that we can always find a good block decomposition for any 2-connected graph.

Theorem 12 *Given a 2-connected graph $G = (V, E)$ with $|V| \geq 3$, we can efficiently find a laminar block decomposition \mathcal{F} of G , such that for every vertex $v \in V$ that participates in any 2-separator (u, v) of G , one of the following holds:*

- *Either v is an endpoint of a block $B \in \mathcal{F}$;*
- *or v has exactly two neighbors in G , and there is an edge $(u', v) \in E$, such that u' is an endpoint of a block $B \in \mathcal{F}$.*

We will use the notion of *SPQR-trees* in the proof. Recall that an SPQR tree \mathcal{T}' for the graph G defines a recursive decomposition of G , as follows. Each node x of the tree \mathcal{T}' is associated with a graph $G_x = (V_x, E_x)$, where $V_x \subseteq V$, and E_x consists of edges of G (called *actual* edges), and some additional edges, called *artificial* or *virtual* edges. Graph G_x is allowed to contain parallel edges. Additionally, we are given a bijection f_x between the artificial edges of G_x , and the edges adjacent to the vertex x in the tree \mathcal{T}' .

Each actual edge of graph G belongs to exactly one of the graphs G_x , for $x \in V(\mathcal{T}')$. The edges of the tree \mathcal{T}' , together with the artificial edges of the graphs G_x , show how to compose the graphs G_x together to obtain the original graph G . More specifically, if $e = (x, y)$ is an edge in the tree \mathcal{T}' , then there is a unique artificial edge e_x in G_x associated with it, and a unique artificial edge e_y in G_y associated with it (that is, $f_x(e_x) = f_y(e_y) = e$). The endpoints of both these artificial edges are copies of the same two vertices, that

is, $e_x = (u, v) = e_y$, for $u, v \in V(G)$. The graphs G_x and G_y share no vertices other than u and v . The removal of the edge (x, y) from \mathcal{T}' decomposes the tree into two connected components, C_1 and C_2 . Let $V_1 \subseteq V$ denote the set of all vertices appearing in the graphs G_z , for $z \in C_1$, and similarly, let $V_2 \subseteq V$ denote the vertices appearing in the graphs G_z where $z \in C_2$. Then $V_1 \cap V_2 = \{u, v\}$, and $V_1 \cup V_2 = V(G)$.

The vertices of the tree \mathcal{T}' belong to one of the four types: S , P , Q , and R .

- If x is an S -node, then the graph G_x is a cycle, with no parallel edges.
- If x is a P -node, then G_x consists of a pair of vertices connected by at least three parallel edges, and at most one of these edges is an actual edge.
- If x is an R -node, then G_x is a 3-connected graph, with more than 3 vertices and no parallel edges.
- If x is a Q -node, then G_x is just a single edge. The tree \mathcal{T}' has Q -nodes only if G itself is an edge, and then \mathcal{T}' has no other nodes.

No two P -nodes and no two S -nodes are adjacent in \mathcal{T}' .

For every 2-connected graph G , there is a unique SPQR tree. Moreover, this tree can be found in linear time, as was shown by Gutwenger and Mutzel [16]. The SPQR tree of G describes the set of all 2-separators of G , as follows: a pair of vertices u and v of G is a 2-separator if and only if either (1) there is an artificial edge (u, v) in some graph G_x , for $x \in \mathcal{T}'$, or (2) u and v are non-adjacent vertices in some graph G_x , where x is an S -node.

We now describe how to construct the laminar block decomposition \mathcal{F} , and the associated decomposition tree \mathcal{T} , for G . Since $|V| \geq 3$ and G is 2-connected, the tree \mathcal{T}' does not contain any Q -nodes. We assume first that G is not a cycle, and we treat the case of the cycle graph separately. We start by computing the SPQR tree \mathcal{T}' of G . We then choose an arbitrary P or R -node r of \mathcal{T}' to serve as the root of the tree \mathcal{T}' . For each node $x \in V(\mathcal{T}')$, we denote the subtree rooted at x by \mathcal{T}'_x . If x is the parent of y in the tree \mathcal{T}' , we call the artificial edge of e_x of G_x that is mapped to (x, y) a *child edge*; we call the edge of G_y that is mapped to (x, y) a *parent edge*. For each $x \in V(\mathcal{T}') \setminus \{r\}$, G_x has exactly one parent edge. If (u, v) is the parent edge of G_y and graph G contains an actual edge (u, v) , we can assume w.l.o.g that G_y does not contain the actual edge (u, v) , as we can remove it from G_y and add it to G_x . This operation may introduce parallel edges in graphs G_x corresponding to S -nodes or R -nodes.

We now proceed in two steps. First, for each node $x \in V(\mathcal{T}')$, we define a block B_x , that is added to \mathcal{F} . This will define a valid laminar block decomposition \mathcal{F} , except that if x is an S -node, then \tilde{B}'_x is not necessarily 3-connected. For each such S -node x , we then add additional blocks to \mathcal{F} in the second step, to avoid this problem.

Step 1: Consider a node x of \mathcal{T}' . If $x \neq r$, denote the parent edge of G_x by $(u, v) \in E(G_x)$ (if $x = r$, we do not define u and v). Let B_x be the union of all graphs G_y associated with the nodes y in \mathcal{T}'_x , with all artificial edges removed.

Since B_x does not contain any artificial edges, it is a subgraph of G . Clearly, $B_r = G$ since every edge of G appears in some graph G_z , for $z \in \mathcal{T}'$. We now verify that for every node $x \in V(\mathcal{T}') \setminus \{r\}$, G_x is indeed a block. Let $V_x = V(B_x)$, and let (u, v) be the parent edge of G_x . Since the graph does not contain any Q -nodes, $|V(B_y)| \geq 3$ for all $y \in V(\mathcal{T}')$, and since for every adjacent pair (y, z) of vertices on \mathcal{T}' , G_y and G_z only share two vertices, $V \setminus V(B_x) \neq \emptyset$. We now show that B_x is an induced subgraph of G (except that edge (u, v) is not in B_x , even if it is present in G). Let e be the edge of \mathcal{T}' , connecting G_x to its parent. Recall that e decomposes the tree \mathcal{T}' into two connected components, C_1 and C_2 , where one of the components is \mathcal{T}'_x . Assume it is C_1 . We have also defined $V_1 \subseteq V$ to be the union of $V(G_z)$ for $z \in C_1$, and similarly $V_2 \subseteq V$ is the union of $V(G_z)$ for $z \in C_2$. Recall that $V_1 \cap V_2 = \{u, v\}$, and by definition of B_x , $V(B_x) = V_1$. Consider some edge $(w_1, w_2) \neq (u, v)$ of G that connects two vertices of V_x . Then since $V_1 \cap V_2 = \{u, v\}$,

vertices w_1, w_2 do not both belong to V_2 . By the definition of SPQR trees, the edge (w_1, w_2) belongs to some graph G_z , for $z \in V(\mathcal{T}')$. Therefore $z \in C_1$ must hold, and $(w_1, w_2) \in E(B_x)$. Also, as we have observed before, there are no edges connecting $V_1 \setminus \{u, v\}$ to $V_2 \setminus \{u, v\}$ in G . Therefore, B_x is indeed a block. Note that if x is a P -node with only one child then G_x consists of 2 edges: one parent artificial edge, and one child artificial edge (in this case, originally G_x contained a third, actual edge, that we have moved to the graph of its father). Therefore for the child node y of x , B_y contains all vertices and edges that B_x contains. We add all blocks B_x , for all $x \in V(\mathcal{T}')$, to \mathcal{F} except if x is a P -node with only one child. Notice that under this definition of \mathcal{F} , for each block $B_x \in \mathcal{F}$, $\tilde{B}'_x = G_x$. Clearly, if x is not an S -node, then \tilde{B}'_x is 3-connected.

Step 2 In this step we take care of the S -nodes. Consider an S -node x , and assume that G_x is a cycle (a_1, \dots, a_s) , where (a_1, a_s) is the parent artificial edge of G_x . We define a nested set $B_x \supset B_x^1 \supset B_x^2 \supset \dots \supset B_x^{s-3}$ of blocks, where $B_x^1 \subset B_x$, that will be added to \mathcal{F} , in addition to B_x . In order to define the blocks B_x^i , we define a collection P_0, P_1, \dots, P_{s-3} paths, as follows. Path P_0 is obtained from G_x by removing the artificial parent edge (a_s, a_1) , so $P_0 = (a_1, a_2, \dots, a_s)$. Path P_i , for $i > 0$, is obtained from P_{i-1} as follows: if i is even, remove the first edge of P_{i-1} , and if it is odd, remove the last edge of P_{i-1} . Therefore, P_i is the portion of P_0 between $a_{1+\lceil i/2 \rceil}$ and $a_{s-\lceil i/2 \rceil}$. Let Y_i be the set of all child nodes of x in \mathcal{T}' , corresponding to the artificial edges of P_i . We are now ready to define B_x^i , for $1 \leq i \leq s-3$: it contains all actual edges of P_i and all blocks B_y for $y \in Y_i$. Graph B_x^i has two types of vertices: inner vertices of blocks B_y , for $y \in Y_i$, and the vertices of P_i . We now show that B_x^i is a block. Since each B_y , for $y \in Y_i$, is a block, no edge connects the interior of B_y to $G \setminus B_y$ (and therefore to $G \setminus B_x^i$). So in order to prove that B_x^i is a block with endpoints $a_{1+\lceil i/2 \rceil}$ and $a_{s-\lceil i/2 \rceil}$, it remains to show that there is no edge connecting an internal vertex a_j of P_i and a vertex in $V \setminus B_x^i$. But this is clearly true since we have already proved that B_x is a block. We add all blocks B_x^i , for $1 \leq i \leq s$ to \mathcal{F} . For convenience, we denote $B_x^0 = B_x$ and $B_x^{s-2} = \text{“EMPTY GRAPH”}$.

This completes the description of the family of blocks \mathcal{F} . It is clear from the construction, that \mathcal{F} is a laminar family of blocks. Now consider a block $B \in \mathcal{F}$, and the corresponding graph \tilde{B}' . We need to prove that \tilde{B}' is 3-connected. First, if $B = B_x$, where x is an R or P -node then $\tilde{B}' = G_x$, and therefore it is 3-connected. If $B = B_x^i$, for $i \in \{0, \dots, s-3\}$, for an S -node x , where G_x is a cycle on s vertices, then \tilde{B}' is obtained from P_i by replacing P_{i+1} with an artificial edge, and connecting the endpoints of P_i with another artificial edge. Therefore, \tilde{B}' is the triangle graph, which is 3-connected.

We now prove that every vertex w that belongs to some 2-separator of G is an endpoint of a block in \mathcal{F} , or it is a degree-2 vertex, and it has a neighbor that serves as an endpoint of a block in \mathcal{F} . By the properties of the SPQR trees, every such vertex w is either an endpoint of some artificial edge e , lying in some graph G_x , for $x \in V(\mathcal{T}')$, or it belongs to some graph G_y associated with an S -node y . In the former case, let G_x be the graph for which the artificial edge e , containing w , is the parent edge. Then B_x belongs to \mathcal{F} , and w is one of its endpoints. The only exception is when x is a P -node, with a unique child x' . But then x' cannot be a P -node, and w is one of its endpoints. In the latter case, we consider the graph G_y , containing w , where y is an S -node. As before, we denote the vertices of G_y by a_1, \dots, a_s . Assume that $w = a_i$. If $i \neq \lceil (s-1)/2 \rceil$, then a_i is an endpoint of some path P_j , $1 \leq j \leq s-3$, and thus, it is an endpoint of the block B_x^j . Assume now that $i = \lceil (s-1)/2 \rceil$. Then the vertex $w = a_{\lceil (s-1)/2 \rceil}$ is connected to $w' = a_{\lceil (s-1)/2 \rceil + 1}$ and $w'' = a_{\lceil (s-1)/2 \rceil - 1}$ by edges e' and e'' , respectively, in G_x . If either of these edges is an artificial edge, then there is a block $B \in \mathcal{F}$, such that $I(B) = (w, w')$, or $I(B) = (w, w'')$. Otherwise, if both edges are actual edges, then the degree of w is 2, and w' is a neighbor of w that serves as an endpoint of a block in \mathcal{F} .

Finally, we show that if a block B has only one child B' then $I(B) \neq I(B')$. Observe that if $B = B_x$ for a P or R -node x then $I(B)$ is the set of endpoints of the parent artificial edge, and $I(B')$ is the set of endpoints of the only child artificial edge. Therefore, if $I(B) = I(B')$, then G_x has two parallel artificial edges, and thus x must be a P -node. However, we add a block B_x associated with a P -node to \mathcal{F} only if it has more than one child. Now let B be either B_x or B_x^i for some S -node x . Since all paths P_i (defined for x) have distinct pairs of endpoints, and they differ from the endpoints of edges, it is straightforward that $I(B) \neq I(B')$.

In the proof, we did not consider the case where G is the cycle graph. We now briefly address this

case. Denote the vertices of the cycle by a_1, \dots, a_s . We create blocks B_1, \dots, B_{s-4} defined by $B_i = \{a_{1+\lfloor i/2 \rfloor}, \dots, a_{s-1-\lfloor i/2 \rfloor}\}$. Additionally, we create a block $\hat{B} = \{a_{s-1}, a_s\}$ if $s > 3$. It is straightforward to verify that this family of blocks together with G satisfies the conditions of the theorem.

B Proof of Theorem 7

We subdivide the sets of irregular vertices and edges into several subsets, that are then bounded separately. We start by defining the following sets of vertices and edges.

$$\begin{aligned} S_1 &= \{u \in V(\mathbf{H}) : u \text{ is a 1-separator in } \mathbf{H}\} \\ E_1 &= \{e \in E(\mathbf{H}) : e \text{ is incident on some } u \in S_1\} \end{aligned}$$

Let \mathcal{C} be the set of all 2-connected components of \mathbf{H} . For every 2-connected component $X \in \mathcal{C}$, we define

$$\begin{aligned} S_2(X) &= \{u \in V(X) \setminus S_1 : \exists v \in V(X) \text{ s.t. } (u, v) \text{ is a 2-separator in } X\} \\ E_2(X) &= \{e \in E(X) : e \text{ has both end-points in } S_2(X)\} \end{aligned}$$

Let $S_2 = \cup_{X \in \mathcal{C}} S_2(X)$ and $E_2 = \cup_{X \in \mathcal{C}} E_2(X)$. We start by showing that the number of vertices and edges in sets S_1 and E_1 , respectively, is small, in the next lemma, whose proof appears in Section B.1.

Lemma 1 (Irregular 1-separators) *We can bound the sizes of sets S_1 and E_1 as follows: $|S_1| = O(|E^*|)$ and $|E_1| = O(d_{\max} \cdot |E^*|)$. Moreover, $\sum_{C \in \mathcal{C}} |S_1 \cap V(C)| \leq 9|E^*|$.*

Next, we show that for **any** planar drawing ψ of \mathbf{H} , the number of irregular vertices and edges that do not belong to sets $S_1 \cup S_2$, and $E_1 \cup E_2$, respectively is small, in the next lemma, whose proof appears in Section B.2. Given any drawing φ of any graph H , we denote by $\text{pcr}_\varphi(H)$ the number of pairs of crossing edges in the drawing φ of H . Clearly, $\text{pcr}_\varphi(H) \leq \text{cr}_\varphi(H)$ for any drawing φ of H .

Lemma 2 *Let H be any planar graph, and let the sets S_1, S_2 of vertices and the sets E_1, E_2 of edges be defined as above for H . Let φ be an arbitrary drawing of H and ψ be a planar drawing of H . Then*

$$|\text{IRG}_V(\psi, \varphi) \setminus (S_1 \cup S_2)| + |\text{IRG}_E(\psi, \varphi) \setminus (E_1 \cup E_2)| = O(\text{pcr}_\varphi(H)) = O(\text{cr}_\varphi(H)).$$

Finally, we need to bound the number of irregular vertices in S_2 and irregular edges in E_2 . The bound does not necessarily hold for **every** drawing ψ . However, we show how to efficiently find a planar drawing, for which we can bound this number, in the next lemma.

Lemma 3 (Irregular 2-separators) *Let $\mathbf{G}, \mathbf{H}, E^*$ and φ be as in Theorem 7. Given \mathbf{G}, \mathbf{H} and E^* (but not φ), we can efficiently compute a planar drawing ψ of \mathbf{H} , such that*

$$|\text{IRG}_V(\psi, \varphi_{\mathbf{H}}) \cap S_2| = O(\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|).$$

and

$$|\text{IRG}_E(\psi, \varphi_{\mathbf{H}}) \cap E_2| = O(d_{\max})(\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|)$$

Theorem 7 then immediately follows from Lemmas 1, 2, and 3, where we apply Lemma 2 to the drawings ψ and $\varphi_{\mathbf{H}}$ of the graph \mathbf{H} . In the following subsections, we present the proofs of Lemmas 1, 2 and 3.

B.1 Proof of Lemma 1

Consider the following tree \mathcal{T} : the vertices of \mathcal{T} are $S_1 \cup \{v_C \mid C \in \mathcal{C}\}$, and there is an edge between v_C and $u \in S_1$ iff $u \in V(C)$. We partition the set $\{v_C : C \in \mathcal{C}\}$ into three subsets: set D_1 contains the leaf vertices of \mathcal{T} , set D_2 contains vertices whose degree in \mathcal{T} is 2, and set D_3 contains all remaining vertices. Since \mathbf{G} is 3-connected, for every component C with $v_C \in D_1 \cup D_2$, there is an edge $e \in E^*$ with one end-point in C . We charge edge e for C . Clearly, we charge each edge at most twice (at most once for each of its endpoints), and therefore, $|D_1| + |D_2| \leq 2|E^*|$. Since the number of vertices of degree greater than 2 is bounded by the number of leaves in any tree, we get that $|D_3| \leq |D_1| \leq 2|E^*|$, and so $|\mathcal{C}| \leq |D_1| + |D_2| + |D_3| \leq 4|E^*|$. Since the parent of every vertex $u \in S_1$ in the tree is a vertex of the form v_C for $C \in \mathcal{C}$, this implies that $|S_1| \leq |\mathcal{C}| + 1 \leq 4|E^*| + 1$, and $|E_1| \leq d_{\max}|S_1| \leq O(d_{\max})|E^*|$.

We now bound the sum $\sum_{C \in \mathcal{C}} |S_1 \cap V(C)|$. The sum equals the number of pairs (C, u) , where $C \in \mathcal{C}$ and $u \in S_1 \cap V(C)$. The number of such pairs in the tree \mathcal{T} is bounded by the number of edges in the tree, which in turn is bounded by the number of vertices, $|S_1| + |\mathcal{C}| \leq 8|E^*| + 1$. This finishes the proof of Lemma 1.

B.2 Proof of Lemma 2

In this section we bound on the number of irregular vertices and irregular edges that do not belong to $S_1 \cup S_2$ and $E_1 \cup E_2$, respectively. Lemma 4 bounds the number of irregular vertices and Lemma 5 the number of irregular edges.

Lemma 4 *Let φ be an arbitrary drawing of H and let ψ be a planar drawing of H . Let $S = S_1 \cup S_2$. Then*

$$|\text{IRG}_V(\psi, \varphi) \setminus S| \leq 12\text{pcr}_\varphi(H) \leq 12\text{cr}_\varphi(H). \quad (1)$$

Proof: Note first that we may assume that no two adjacent edges cross each other in the drawing φ . Indeed, if the images of two edges incident to a vertex u cross, we can uncross their drawings, possibly changing the cyclic order of edges adjacent to u , and preserving the cyclic order for all other vertices. The right-hand side of (1) will then decrease by 12, and the left hand side by at most 1, so we only strengthen the inequality. We can also assume w.l.o.g. that the graph H is 2-connected: otherwise, if \mathcal{C} is the set of all 2-connected components of H , then, since $\text{pcr}_\varphi(H) \geq \sum_{C \in \mathcal{C}} \text{pcr}_\varphi(C)$, it is enough to prove the inequality (1) for each component $C \in \mathcal{C}$ separately. So we assume below that H is 2-connected.

Consider some vertex $u \in \text{IRG}_V(\psi, \varphi) \setminus S$. Let F be the face of $H \setminus \{u\}$ that contains the image of u in the drawing $\psi_{H \setminus \{u\}}$. Note that graph $H \setminus \{u\}$ is 2-connected: otherwise, if v is a vertex separator of $H \setminus \{u\}$ then $\{u, v\}$ is a 2-separator for H , contradicting the fact that $u \notin S$. Therefore, the boundary of F is a simple cycle, that we denote by γ . Let v_1, \dots, v_κ be the neighbors of u in the order induced by γ . Vertices v_i partition γ into κ paths P_1, \dots, P_κ , where path P_i connects vertices v_i and v_{i+1} (we identify indices $\kappa + 1$ and 1). Let F_i be the face of the planar drawing ψ , that is bounded by (u, v_i) , P_i and (v_{i+1}, u) . Note that since for all $i \neq j$, the two paths P_i and P_j do not share any internal vertices, the total number of vertices that the boundaries of F_i and F_j for $i \neq j$ share is at most 3, with the only possibilities being u , v_i and v_{i+1} (the endpoints of P_i).

Consider the graph W formed by γ , u , and edges (u, v_i) , for $1 \leq i \leq \kappa$. This graph is homeomorphic to the wheel graph on κ vertices. In any planar embedding of W , the ordering of the vertices $\{v_i\}_{i=1}^\kappa$ is (v_1, \dots, v_κ) . So if the drawing φ_W of W is planar, then the circular ordering of the edges adjacent to u in φ is $((u, v_1), \dots, (u, v_\kappa))$ – the same as in ψ , up to orientation. Therefore, if $u \in \text{IRG}_V(\psi, \varphi)$ then either there is a pair P_i, P_j of paths, with $i \neq j$, whose images cross in φ , or an image of an edge (u, v_i) crosses a path P_j (recall that we have assumed that no two edges (u, v_i) and (u, v_j) cross each other; all self-intersections of paths P_i can be removed without changing the rest of the embedding). We say that this crossing point pays for u . Thus every irregular vertex is paid for by a crossing in the drawing φ . It only remains to show that every pair of crossing edges pays for at most 12 vertices.

Suppose that u is paid for by a crossing of edges e_1 and e_2 . For each edge $e \in \{e_1, e_2\}$, there is a face F^e (in the embedding of ψ) such that e and u lie on the boundary of F^e : if e lies on path P_i then $F^e = F_i$; if $e = (u, v_j)$ then F^e is either F_{j-1} or F_j . Since in the latter case we have two choices for F^e , we can choose distinct faces F^{e_1} and F^{e_2} . Therefore, if a crossing of edges e_1 and e_2 pays for a vertex $u \in \text{IRG}_V(\psi, \varphi) \setminus S$, then there are two distinct faces F^{e_1} and F^{e_2} in ψ , incident to e_1 and e_2 respectively, such that u lies on the intersection of the boundaries of F^{e_1} and F^{e_2} . We say that the pair of faces F^{e_1} and F^{e_2} is the witness for the irregular vertex u . Since the boundaries of F^{e_1} and F^{e_2} may share at most 3 vertices that do not belong to S_2 , the pair (F^{e_1}, F^{e_2}) is a witness for at most 3 irregular vertices. Since each edge e_i is incident to at most two faces in ψ , there are at most 4 ways to choose F^{e_1} and F^{e_2} , and for each such choice (F^{e_1}, F^{e_2}) is a witness for at most 3 irregular vertices. We conclude that each pair of edges that cross in φ pays for at most 12 irregular vertices. \square

Lemma 5 *Let φ be an arbitrary drawing of H and ψ be its planar drawing. Let $E_S = E_1 \cup E_2$. Then*

$$|\text{IRG}_E(\psi, \varphi) \setminus E_S| \leq 8\text{pcr}_\varphi(H) \leq 8\text{cr}_\varphi(H).$$

Proof: We can assume w.l.o.g. that there are no vertices of degree 2 in H , by iteratively removing such vertices u , and replacing the two edges incident on u with a single edge. This operation may decrease the number of irregular edges by at most factor 2, and can only decrease the number of pairs of crossing edges. Similarly to the proof of Lemma 4, we assume that the graph H is 2-connected: otherwise, we can apply the argument below separately to each 2-connected component.

We say that the orientation of a regular vertex u is positive, if the ordering of the edges incident to u is the same in φ and ψ , including the flip. If the flips in φ and ψ are opposite, we say that the orientation is negative. For every irregular edge e , the orientation of one of its endpoints is positive, and of the other is negative.

Consider an irregular edge $e = (u, v) \in \text{IRG}_E(\psi, \varphi) \setminus E_S$, and assume w.l.o.g. that the orientation of u is positive and the orientation of v is negative. Let F_1 and F_2 be the two faces incident to e in the embedding ψ . Since H is 2-connected, the boundaries of F_1 and F_2 are simple cycles. Denote them by C_1 and C_2 . Let $P_i = C_i \setminus \{e\}$ be the sub-path of C_i that connects u to v . We now prove that P_1 and P_2 do not share any vertices except for u and v . Indeed, assume for contradiction that a vertex $w \notin \{u, v\}$ lies on both P_1 and P_2 . Since $e \notin E_S$, either u or v (or both) are not in S . Assume w.l.o.g. that $u \notin S$. We draw two curves, connecting w to the middle of the edge e inside the planar drawing ψ of H ; one of the two curves lies inside F_1 and the other lies inside F_2 . The union of the two curves defines a cycle that separates $H \setminus \{w\}$ into two pieces, with u belonging to one piece and v to the other. Denote these pieces by B_u and B_v , respectively. (We assume that $w \in B_u, w \in B_v$). We will now show that (u, w) is a 2-separator for H , leading to a contradiction. Observe first that since the degrees of u and v are at least 3, and the separating cycle only crosses one edge of H (the edge e), both B_u and B_v contain at least 3 vertices each. Since every path from B_u to B_v must cross the separating cycle, each such path either contains the vertex w or the edge e . Therefore, (u, w) is a 2-separator for H , contradicting our assumption that $u \notin S$.

We say that the pair of faces (F_1, F_2) is the witness for the irregular edge e . From the above discussion, each pair of faces is a witness for at most one irregular edge.

Let o_u^μ be the orientation — either clockwise or counterclockwise — in which paths P_1 , e , and P_2 leave u in the embedding μ (where μ is either φ or ψ). If the orientation is clockwise $o_u^\mu = 1$; otherwise $o_u^\mu = -1$. Similarly, we define o_v^μ . Note that in any embedding μ' in which paths P_1 , e and P_2 do not cross each other, $o_u^{\mu'} = -o_v^{\mu'}$. In particular, since ψ is a planar embedding, $o_u^\psi = -o_v^\psi$. But since the orientation of u is positive, and the orientation of v is negative, $o_u^\varphi = o_v^\varphi$. Therefore, there is a pair (e_1, e_2) of crossing edges in φ , where either $e_1 \in P_1, e_2 \in P_2$; or $e_1 \in P_1, e_2 = e$; or $e_1 = e$ and $e_2 \in P_2$. We say that the crossing of e_1 and e_2 pays for the irregular edge e . The edges e_1 and e_2 lie on the boundaries of F_1 and F_2 respectively. Similarly to the previous lemma, given two crossing edges e_1 and e_2 , there are at most 4 ways to choose the

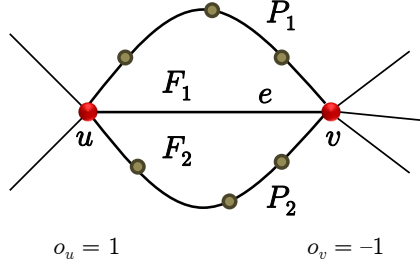


Figure 6: Graph H , irregular edge e , paths P_1 and P_2 .

faces (F_1, F_2) incident to them, and each such pair of faces is a witness for at most one edge. Therefore, each pair of crossing edges pays for at most 4 irregular edges. We conclude that the number of irregular edges is bounded by $4cr_\varphi(H)$. Replacing the edges back by the original 2-paths increases the number of irregular edges by at most factor 2, as each irregular 2-path contains two irregular edges. \square

B.3 Proof of Lemma 3

We start with a high level overview of the proof. Assume first that the graph \mathbf{H} is 2-connected. We can then use Theorem 12 to find a laminar block decomposition \mathcal{F} of \mathbf{H} . Moreover, each vertex $v \in S_2$ is either an endpoint of a block in \mathcal{F} , or it is a neighbor of an endpoint of a block in \mathcal{F} . Therefore, $|S_2|$ is roughly bounded by $O(|\mathcal{F}| \cdot d_{\max})$. On the other hand, since the graph \mathbf{G} is 3-connected, each block $B \in \mathcal{F}$ must contain an endpoint of an edge from E^* as an inner vertex, that can be charged for the block B , for its endpoints, and for the neighbors of its endpoints. This approach would work if we could show that every edge $e \in E^*$ is only charged for a small number of blocks. This unfortunately is not necessarily true, and an edge $e \in E^*$ may be charged for many blocks in \mathcal{F} . However, this may only happen if there is a large number of nested blocks, all of which contain the same endpoint of the edge e . We call such set of blocks a “tunnel”. We then proceed in two steps. First, we bound the number of blocks of \mathcal{F} that do not participate in such tunnels, by charging them to the edges of E^* , as above. Next, we perform some local changes in the embeddings of the tunnels (by suitably flipping the embedding of each block of the tunnel), so that we can charge the number of irregular vertices that serve as endpoints of blocks participating in the tunnels to the crossings in φ .

We now proceed with the formal proof. We start with an arbitrary planar drawing ψ_{init} of \mathbf{H} . Let \mathcal{C} be the set of all 2-connected components of \mathbf{H} . We consider each component $X \in \mathcal{C}$ separately. For a component $X \in \mathcal{C}$, let $cr_\varphi(\mathbf{G}, X)$ denote the number of crossings in φ in which edges of X participate, and let $E^*(X)$ denote the subset of edges of E^* that have at least one endpoint in X . We will modify ψ_{init} locally on each 2-connected component $X \in \mathcal{C}$ and obtain a planar drawing ψ of \mathbf{H} such that

$$\begin{aligned} |\text{IRG}_V(\varphi_{\mathbf{H}}, \psi) \cap S_2(X)| &= O(cr_\varphi(\mathbf{G}, X) + |E^*(X)| + |S_1 \cap X|) \\ |\text{IRG}_E(\varphi_{\mathbf{H}}, \psi) \cap E_2(X)| &= O(d_{\max}(cr_\varphi(\mathbf{G}, X) + |E^*(X)| + |S_1 \cap X|)). \end{aligned}$$

Summing up over all $X \in \mathcal{C}$, and using Lemma 1 gives the desired bound. Since we guarantee that the modifications of ψ_{init} are restricted to X , we can modify the 2-connected components $X \in \mathcal{C}$ independently to obtain the final desired drawing.

Fix a 2-connected component $X \in \mathcal{C}$. If X is 3-connected then $S_2(X) = E_2(X) = \emptyset$ and there is nothing to prove. So we assume below that X is not 3-connected. We compute the laminar block decomposition $\mathcal{F}(X)$ and the corresponding decomposition tree $\mathcal{T}(X)$ for X , given by Theorem 12. For convenience, we

use $\mathcal{F}'(X) = \mathcal{F}(X) \setminus \{X\}$ to denote the set of all blocks in $\mathcal{F}(X)$, excluding the whole component X . We now proceed in three steps. Our first step is to explore some structural properties of the blocks $B \in \mathcal{F}(X)$. We will use these properties, on the one hand, to bound the number of blocks that do not participate in tunnels, and on the other hand, to find the layout of the tunnels. In the second step, we define the subsets of blocks that we can charge to the edges in E^* . We then charge some of the vertices in S_2 and edges in E_2 to these blocks. In the last step, we define tunnels, to which all remaining blocks belong, and we show how to take care of them.

Step 1: Structural properties of blocks

Consider some block $B \in \mathcal{F}'(X)$, with endpoints u and v . Since X is 2-connected, there is a path $P_{out}^B : u \rightsquigarrow v$ in $(X \setminus B) \cup \{u, v\}$. Moreover, if B' is the parent of B in $\mathcal{T}(X)$, whose endpoints are u' and v' , we can ensure that $P_{out}^{B'} \subseteq P_{out}^B$, as follows. Consider the graph B^* obtained from B' after we remove all inner vertices of B from it. Since X is 2-connected, so is B' . Therefore, there are 2 vertex disjoint paths in B^* , connecting the vertices in $\{u', v'\}$ to the vertices in $\{u, v\}$. We assume w.l.o.g. that these paths are $P_1 : u' \rightsquigarrow u$ and $P_2 : v \rightsquigarrow v'$. We can then set $P_{out}^{B'} = (P_1, P_{out}^B, P_2)$ (see Figure 7). Therefore, from now on we assume that if B' is the parent of B , then $P_{out}^{B'} \subseteq P_{out}^B$.

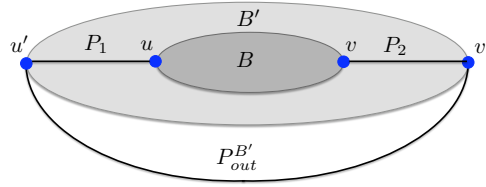


Figure 7: Paths $P_{out}^B, P_{out}^{B'}$.

Since we have assumed that \mathbf{G} is 3-vertex connected, for every block $B \in \mathcal{F}'(X)$, there is also a path Q in $\mathbf{G} \setminus \{u, v\}$, connecting an **inner** vertex of the block B , with an **inner** vertex of the path P_{out}^B . Let x_B be the last vertex on Q that belongs to B and y_B be the first vertex on Q that belongs to P_{out}^B (notice that $y_B \neq u, v$, since Q does not contain u or v). We denote the segment of Q between x_B and y_B by P_0^B , and we call the vertex x_B the *connector vertex* for the block B .

Note that if B'' is a child block of B and x_B is an inner vertex of B'' as well, then since $P_{out}^B \subseteq P_{out}^{B''}$, we can choose x_B to be the connector vertex of B'' as well, and use $P_0^{B''} = P_0^B$. So we assume that each connector vertex x appears contiguously in the tree \mathcal{T} . That is, if B is a descendant of B_1 and an ancestor of B_2 and $x_{B_1} = x_{B_2}$, then $x_B = x_{B_1} = x_{B_2}$. We also assume that in this case $P_0^{B_1} = P_0^B = P_0^{B_2}$. We denote the segment of P_{out}^B between u and y_B by $P_{1,out}^B$ and the segment between y_B and v by $P_{2,out}^B$.

Since X is 2-connected, there are two vertex disjoint paths between x_B and y_B in X . One of them must pass through u and the other through v . We denote the segment between u and x_B of the former path by $P_{1,in}^B$ and the segment between x_B and v of the latter path by $P_{2,in}^B$. Let P_{in}^B be the concatenation of $P_{1,in}^B, P_{2,in}^B$. Note that the paths $P_0^B, P_{1,in}^B, P_{2,in}^B, P_{1,out}^B$ and $P_{2,out}^B$ do not intersect, except at endpoints (see Figure 8). We emphasize that x_B is an **inner** vertex of B , and y_B is an **inner** vertex on path P_{out}^B — a fact that we use later.

For each component $X \in \mathcal{C}$, let \mathcal{S}_X be the union of (i) the set $S_1 \cap X$ and (ii) the set of vertices of X incident to edges of E^* . Using Lemma 1,

$$\sum_{X \in \mathcal{C}} |\mathcal{S}_X| \leq \sum_{X \in \mathcal{C}} (|E^*(X)| + |S_1 \cap X|) \leq O(|E^*|). \quad (2)$$

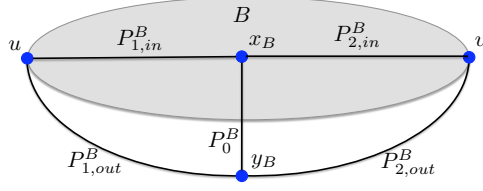


Figure 8: Paths P_0^B , $P_{1,in}^B$, $P_{2,in}^B$, $P_{1,out}^B$ and $P_{2,out}^B$. Vertex x_B is an inner vertex of B , and vertex y_B is an inner vertex of P_{out}^B . All five paths are non-empty and completely disjoint except for their endpoints.

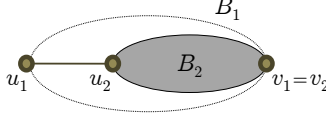


Figure 9: A simple block B_1 .

We now show that for each block $B \in \mathcal{F}'(X)$, the connector vertex $x_B \in \mathcal{S}_X$. Indeed, consider the first edge (x_B, z) of the path P_0^B . If $z \in X$, then $(x_B, z) \in E^*(X)$, as by the definition of the block, no edges of X connect inner vertices of B to $X \setminus B$. Otherwise, if $z \notin X$, then x_B must be a 1-separator, so $x_B \in S_1$.

Finally, we study structural properties of chains of nested blocks. We also introduce a notion of a *simple* block, and show that all non-simple blocks contain a certain useful structure.

Definition: Let $B_1 \in \mathcal{F}'(X)$ be any block, whose endpoints are denoted by u_1 and v_1 . We say that B_1 is a *simple block* iff it contains exactly three vertices, u_1, v_1 , and u_2 , and has exactly one child in $\mathcal{T}(X)$, denoted by B_2 (assume w.l.o.g. that the endpoints of B_2 are (u_2, v_1)). Moreover, B_1 is obtained by adding exactly one edge, (u_1, u_2) , to B_2 (see Figure 9). If $B_1 \in \mathcal{F}'(X)$ has exactly one child in $\mathcal{T}(X)$, but it is not a simple block, then we say that it is *complex*.

We need the following two claims.

Claim 1 Consider a chain of 5 nested blocks: B_1, B_2, B_3, B_4 and B_5 , where B_{i+1} is the only child of B_i (for $i \in \{1, \dots, 4\}$). Assume that no vertices in $V(B_1) \setminus V(B_5)$ have degree 2 in X . Then one of the blocks B_1, B_2, B_3 , or B_4 is complex.

Proof: Notice that from the definition of simple blocks, if all blocks B_1, B_2, B_3, B_4 are simple, at least one vertex $z \in \{u_2, v_2, u_3, v_3\} \setminus V(B_5)$ must have degree 2 in X (where u_i and v_i are endpoints of B_i), contradicting the fact that $V(B_1) \setminus V(B_5)$ cannot contain such vertices. \square

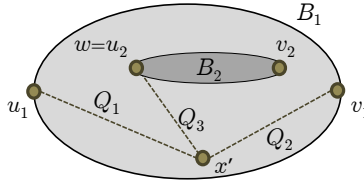


Figure 10: A complex block. Paths Q_1, Q_2, Q_3 are pairwise vertex disjoint, except for containing x' as a common endpoint.

Claim 2 Suppose that a non-simple block $B_1 \in \mathcal{F}'(X)$ has exactly one child B_2 in $\mathcal{T}(X)$. Denote the endpoints of B_1 by u_1 and v_1 , and the endpoints of B_2 by u_2 and v_2 . Then for every vertex $x' \in V(\tilde{B}_1) \setminus \{u_1, v_1\}$, there are three paths $Q_1 : x' \rightsquigarrow u_1$, $Q_2 : x' \rightsquigarrow v_1$, and $Q_3 : x' \rightsquigarrow w$, with $w \in \{u_2, v_2\}$, and all three paths are contained in $\tilde{B}_1 \setminus \{(u_2, v_2)\}$. Moreover, Q_1, Q_2 and Q_3 do not share any vertices, except for the vertex x' that serves as their endpoint. (See Figure 10 for an illustration.)

Proof: Since B_1 has only one child, $u_2 \notin \{u_1, v_1\}$ or $v_2 \notin \{u_1, v_1\}$ (or both). Let us assume w.l.o.g. that $u_2 \notin \{u_1, v_1\}$. In particular, B_1 contains at least 3 vertices.

We consider two cases. Assume first that \tilde{B}_1 contains exactly 3 vertices. Then these vertices must be u_1, v_1 and u_2 . The only valid choice for the vertex x' is $x' = u_2$. From the definition of blocks, B_1 cannot contain the edge (u_1, v_1) . But since it is connected, it must contain the edge (u_1, u_2) . Therefore, the only way for B_i not to be simple (since we have assumed that \mathbf{G} contains no parallel edges) is if B_1 contains the edge (u_2, v_1) . But in this case, we get the following three paths: $Q_1 = (u_1, u_2)$, $Q_2 = (u_2, v_1)$, and $Q_3 = \emptyset$.

Assume now that \tilde{B}_1 contains at least 4 vertices. From Theorem 12, the graph \tilde{B}'_1 is 3-connected. Let x' be an arbitrary inner vertex of B_1 . Assume first that $x' \notin \{u_2, v_2\}$. Recall that the Fan Lemma states that for every r -connected graph A , a vertex a in A and a set of r vertices $B \subset V(A) \setminus \{a\}$, there exist r paths that connect a to vertices of B that have no common vertices other than a . We apply the Fan Lemma in graph \tilde{B}'_1 to x' and $\{u_1, v_1, u_2\}$. Let Q_1 be the resulting path between x' and u_1 , Q_2 the path between x' and v_1 , and Q'_3 the path between x' and u_2 . Note that paths Q_1 and Q_2 do not contain the artificial edge (u_2, v_2) , as otherwise they would contain u_2 . Notice also that none of the three paths contains the artificial edge (u_1, v_1) , as this would violate their disjointness. Finally, let Q_3 be equal to either Q'_3 , if Q'_3 does not visit v_2 , or the segment of Q'_3 between x' and v_2 , if it does (the latter can only happen if $v_2 \notin \{u_1, v_1\}$). We have thus constructed the required paths Q_1, Q_2 and Q_3 . Assume now that $x' \in \{u_2, v_2\}$. Since \tilde{B}'_1 is 3-vertex connected (and $\tilde{B}'_1 \neq K_3$), the graph $\tilde{B}'_1 \setminus \{(u_2, v_2)\}$ is 2-vertex connected. We again apply the Fan Lemma to w and $\{u_1, v_1\}$ in this graph and find the desired paths Q_1 and Q_2 . We let Q_3 to be the trivial path of length 0. \square

Step 2: Blocks we can pay for

Fix a 2-connected component $X \in \mathcal{C}$. In this step, we define three subsets $\mathcal{R}_1(X), \mathcal{R}_2(X), \mathcal{R}_3(X)$ of $\mathcal{F}(X)$, and bound the number of blocks contained in them. We also define a subset $\tilde{\mathcal{S}}_2 \subseteq \mathcal{S}_2$ of vertices and a subset $\tilde{E}_2 \subseteq E_2$ of edges, that can be charged to these blocks. The remaining blocks of $\mathcal{F}(X)$ will be partitioned into structures called tunnels, and we take care of them in the next step.

Set $\mathcal{R}_1(X)$: Let $\mathcal{R}_1(X)$ denote the set of blocks $B \in \mathcal{F}(X)$, such that B is either the root of $\mathcal{T}(X)$, or it is one of its leaves, or it has a degree greater than 2 in $\mathcal{T}(X)$, or it contains a vertex from \mathcal{S}_X that does not belong to any of its child blocks. We also add five immediate ancestors of every such block to $\mathcal{R}_1(X)$.

Claim 3 $\sum_{X \in \mathcal{C}} |\mathcal{R}_1(X)| = O(|E^*|)$.

Proof: Denote the number of leaves in $\mathcal{T}(X)$ by L_X . For each leaf block B , we charge the connector vertex $x_B \in \mathcal{S}_X$ for B . For each non-leaf block B , such that B contains a vertex $x \in \mathcal{S}_X$ that does not belong to any of its children, we charge x for B (even if $x_B \neq x$). Since $\mathcal{F}(X)$ is a laminar family, it is easy to see that each vertex $x \in \mathcal{S}_X$ is charged at most once. The number of vertices of degree at least 3 in $\mathcal{T}(X)$ is at most $L_X - 1$. By adding five ancestors of each block, we increase the size of $\mathcal{R}_1(X)$ by at most a factor of 5. Therefore, $\sum_{X \in \mathcal{C}} |\mathcal{R}_1(X)| \leq \sum_{X \in \mathcal{C}} O(|\mathcal{S}_X|) = O(|E^*|)$. \square

Set $\mathcal{R}_2(\mathbf{X})$: Consider a vertex $x \in \mathcal{S}_X$. Notice that the set of blocks $B \in \mathcal{F}(X)$ with $x_B = x$ must be a nested set. We add the smallest such block and its five immediate ancestors to $\mathcal{R}_2(X)$.

Claim 4 $\sum_{X \in \mathcal{C}} |\mathcal{R}_2(X)| = O(|E^*|)$.

Proof: For each block $B \in \mathcal{R}_2(X)$, we charge the connector vertex x_B for B . By the definition of $\mathcal{R}_2(X)$, each connector vertex pays for at most 6 blocks. Therefore, $\sum_X |\mathcal{R}_2(X)| \leq \sum_X O(|\mathcal{C}_X|) = O(|E^*|)$. \square

Set $\mathcal{R}_3(\mathbf{X})$: Note that the blocks of $\mathcal{F}(X)$ that do not belong to $\mathcal{R}_1(X) \cup \mathcal{R}_2(X)$ all have degree exactly 2 in $\mathcal{T}(X)$, and therefore the sub-graph of $\mathcal{T}(X)$ induced by such blocks is simply a collection of disjoint paths. Consider some block $B \in \mathcal{F}(X) \setminus (\mathcal{R}_1(X) \cup \mathcal{R}_2(X))$. It has exactly one child in $\mathcal{T}(X)$, that we denote by B' . Let u and v be the endpoints of B , and let u' and v' be the endpoints of B' . Consider the graph \tilde{B}' obtained from B by first replacing B' with an artificial edge (u', v') and then by adding a new artificial edge (u, v) . By Theorem 12, the graph \tilde{B}' is 3-vertex connected. Therefore, it has a unique planar drawing $\pi_{\tilde{B}'}$. We add B to $\mathcal{R}_3(X)$ iff the four vertices u, v, u', v' **do not lie** on the boundary of the same face in this drawing.

Lemma 6 $\sum_{X \in \mathcal{C}} (|\mathcal{R}_3(X)|) = O(\text{cr}_\varphi(\mathbf{G}))$.

Proof: Consider some block $B \in \mathcal{R}_3(X)$. Denote $B_0 = B$, and for $i = 1, \dots, 5$, let B_i be the child of B_{i-1} in $\mathcal{T}(X)$. For each $i : 1 \leq i \leq 5$, let (u_i, v_i) denote the endpoints of the block B_i . Since when we added a block to $\mathcal{R}_1(X)$ or $\mathcal{R}_2(X)$, we also added five its immediate ancestors to $\mathcal{R}_1(X)$ or $\mathcal{R}_2(X)$, respectively, each of the blocks B_i , for $0 \leq i \leq 5$, has a unique child, and moreover, for $i = 1, \dots, 5$, $x_{B_i} = x_B$ and $P_0^{B_i} = P_0^B$. Let \hat{E}_B denote the edges of B that do not belong to B_5 , that is, $\hat{E}_B = E(B) \setminus E(B_5)$. We will show that for each $B \in \mathcal{R}_3$, there is at least one crossing in φ , in which the edges of \hat{E}_B participate. Since every edge may belong to at most 5 such sets \hat{E}_B , it will follow that $|\mathcal{R}_3(X)| \leq O(\text{cr}_\varphi(X, \mathbf{G}))$, and $\sum_{X \in \mathcal{C}} (|\mathcal{R}_3(X)|) = O(\text{cr}_\varphi(\mathbf{G}))$. Therefore, it now only remains to show that for each block $B \in \mathcal{R}_3(X)$, the edges of \hat{E}_B participate in at least one crossing in φ . Assume for contradiction that this is not true, and let B be the violating block. We will show that we can find a planar drawing of \tilde{B}' , in which the vertices (u, v, u_1, v_1) all lie on the boundary of the same face, contradicting the fact that $B \in \mathcal{R}_3(X)$.

We denote by B^* the graph obtained from B after we remove all inner vertices of B_1 and their adjacent edges from it. Notice that all edges of B^* belong to \hat{E}_B . We also denote $x_B = x, y_B = y$ and $P_0^B = P_0$. Recall that for all $1 \leq i \leq 5$, $x_i = x, y_i = y$ and $P_0^{B_i} = P_0$. Recall that by definition, x is an **inner** vertex on $P_{in}^{B_i}$ for all $1 \leq i \leq 5$, and y is an **inner** vertex on P_{out}^B .

We start with a high-level intuition for the proof. Let $P_{in} = P_{in}^{B_1} \subseteq B_1$, and assume for now that P_{in} only contains the edges of \hat{E}_B (this is not necessarily true in general). Observe that P_{in} contains no edges of $B \setminus B_1$. Therefore, the sets $E(B^*), E(P_{in}), E(P_0)$ and $E(P_{out}^B)$ of edges are completely disjoint. Consider the drawing φ of \mathbf{G} , and erase from it all edges and vertices, except those participating in B^*, P_{in}, P_0 and P_{out}^B . Let φ' be the resulting drawing. For convenience, we call the edges of \hat{E}_B *blue edges*, and the remaining edges *red edges*. By our assumption, the blue edges do not participate in any crossings. Since we have assumed that P_{in} only consists of blue edges, all crossings in φ' are between the edges of $P_0, P_{1,out}^B$ and $P_{2,out}^B$. All these three paths share a common endpoint, y , and they are completely disjoint otherwise. Therefore, we can uncross their drawings in φ' , and obtain a planar drawing φ'' of $B^* \cup P_{in} \cup P_{out}^B \cup P_0$. Erase the drawing of P_0 from φ'' , and replace the drawings of paths P_{out}^B and P_{in} by drawings of edges $e : u \rightsquigarrow v$, $e' : u_1 \rightsquigarrow v_1$, respectively, to obtain a planar drawing π' of \tilde{B}' . Note that in π' , the drawings of edges (u, v) and (u_1, v_1) (and therefore their endpoints) lie on the boundary of one face, since the drawing of the path P_0 in φ'' connects **internal** points of edges (u_1, v_1) and (u, v) and does not cross the images of any edges. Therefore, we have found a planar drawing of \tilde{B}' , in which the vertices u, v, u', v' lie on the boundary of the

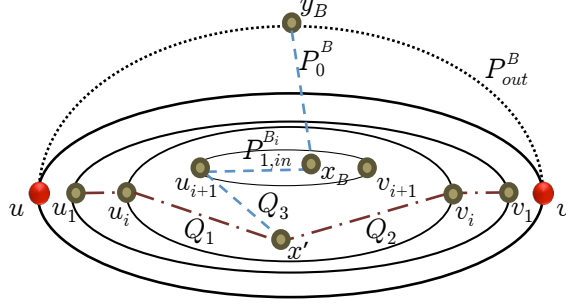


Figure 11: Paths Q_1, Q_2 (and their extensions Q'_1 and Q'_2), and Q_3 . Recall that path $P'_{in} = (Q'_1, Q'_2)$, and path $P'_0 = (Q_3, P_{1,in}^{B_i}, P_0)$.

same face, contradicting the fact that $B \in \mathcal{R}_3(X)$. The only problem with this approach is that P_{in} does not necessarily only consist of edges of $\hat{E}_B \setminus E(B^*)$. We overcome this by finding a new path $P'_{in} : v \rightsquigarrow u$ that only contains edges of \hat{E}_B but no edges of B^* , and another path P'_0 connecting an inner vertex x' of P'_{in} to the vertex y . If we ensure that (1) $P'_{in} : v \rightsquigarrow u$ only contains edges of \hat{E}_B but no edges of B^* ; (2) path $P'_0 : x' \rightsquigarrow y$ connects an inner vertex x' of P'_{in} to y and contains no edges of B^* ; and (3) The paths P'_{in}, P'_0 and P_{out}^B are completely disjoint, except for possibly sharing endpoints, then we can again apply the above argument, while replacing the path P'_{in} with P_{in} , and path P_0 with P'_0 . We now provide the formal proof.

We first note that at least one of the four blocks B_1, B_2, B_3, B_4 is complex. Indeed, by Claim 1 it suffices to show that $V(B_1) \setminus V(B_5)$ does not contain a vertex w whose degree is 2 in X . Note that if $w \in V(B_1) \setminus V(B_5)$ and the degree of w in X is 2, then $w \in \mathcal{S}_X$. This is since \mathbf{G} is 3-connected, and so all degree-2 vertices in X must either be incident on an edge of E^* , or belong to S_1 . Therefore, one of the blocks B_1, \dots, B_4 must have been added to $\mathcal{R}_1(X)$, together with its five immediate ancestors.

We finally show that since one of the blocks B_i , for $1 \leq i \leq 4$, is complex, we can find the planar drawing of \tilde{B}' in which u, v, u_1, v_1 lie on the same face, thus leading to contradiction.

Claim 5 *If at least one of the blocks B_i , for $1 \leq i \leq 4$ is complex, then there is a planar drawing of \tilde{B}' , in which u, v, u_1, v_1 all lie on the boundary of the same face.*

Proof: Let B_i be the first complex block among B_1, B_2, B_3 and B_4 . Notice that since B_i has only one child in $\mathcal{T}(X)$, it must contain at least one inner vertex. Choose an arbitrary inner vertex x' of \tilde{B}_i . Since B_i is complex, there are three paths $Q_1 : x' \rightsquigarrow u_i$, $Q_2 : x' \rightsquigarrow v_i$, and $Q_3 : x' \rightsquigarrow w$, as in Claim 2. We assume w.l.o.g. that $w = u_{i+1}$. We extend paths Q_1 and Q_2 to paths Q'_1 and Q'_2 , connecting x' to vertices u_1 and v_1 , as follows. Since X is 2-connected, there are two vertex disjoint paths connecting $\{u_i, v_i\}$ to $\{u_1, v_1\}$ in B_1 . We assume w.l.o.g. that these paths are $\Delta_1 : u_i \rightsquigarrow u_1$ and $\Delta_2 : v_i \rightsquigarrow v_1$. We append these paths to Q_1 and Q_2 , obtaining the desired paths $Q'_1 : x' \rightsquigarrow u_1$ and $Q'_2 : x' \rightsquigarrow v_1$. Finally, we define paths P'_{in} and P'_0 , as follows. Let $P'_{in} : u_1 \rightsquigarrow v_1$ be the union of paths $Q'_1 : x' \rightsquigarrow u_1$ and $Q'_2 : x' \rightsquigarrow v_1$. Let $P'_0 : x' \rightsquigarrow y_B$ be the union of paths $Q_3 : x' \rightsquigarrow u_{i+1}$, $P_{1,in}^{B_i} : u_{i+1} \rightsquigarrow x$ and $P_0^B : x \rightsquigarrow y$ (see Figure 11). Observe that x' is indeed an inner vertex of P'_{in} , so P'_0 connects an inner vertex of P'_{in} to an inner vertex of P_{out}^B , as required.

We now verify that paths P'_{in} and P'_0 satisfy other required conditions. First, P'_{in} only contains edges of \hat{E}_B but no edges of B^* , since all paths Q_1, Q'_1, Q_2, Q'_2 lie in B_1 but do not contain edges of $B_{i+1} \supseteq B_5$. Next, path $P'_0 : x' \rightsquigarrow y_B$ does not contain edges of B^* , since it is the concatenation of the path $Q_3 \subseteq B_i \subseteq B_1$, the path $P_{1,in}^{B_i} \subseteq B_i \subseteq B_1$ and the path P_0 , that does not contain edges of B . It is straightforward to verify that paths P'_{in}, P'_0 , and P_{out}^B share no vertices except for y and x' . Therefore, the sets $E(B^*), E(P'_0), E(P'_{in})$ and $E(P_{out}^B)$ of edges are completely disjoint, as required.

We now consider the drawing φ' obtained from φ , after we remove all edges and vertices, except those participating in B^* , P_{out}^B, P_{in}' and P_0' . We call the edges of \tilde{E}_B blue, and the remaining edges red. Then P_{in}' only consists of blue edges, but it does not contain edges of B^* . Since in the resulting drawing, φ' , no blue edges participate in crossings, the only crossings involve paths $P_{1,out}^B, P_{2,out}^B$ and P_0' . As before, we can uncross them and obtain a planar drawing φ'' , which gives a planar drawing π' of \tilde{B}' , in which the vertices u, v, u_1, v_1 all lie on the same face. \square

\square

Let $\mathcal{R}(X) = \mathcal{R}_1(X) \cup \mathcal{R}_2(X) \cup \mathcal{R}_3(X)$, and let $\mathcal{R}'(X)$ be the set of all blocks $B \in \mathcal{F}(X)$, whose parent belongs to $\mathcal{R}(X)$. Since all leaves of tree $\mathcal{T}(X)$ belong to $\mathcal{R}_1(X)$, it is easy to see that $|\mathcal{R}'(X)| \leq |\mathcal{R}_1(X)|$. Therefore, we get the following corollary:

Corollary 3

$$\sum_{X \in \mathcal{C}} (|\mathcal{R}(X)| + |\mathcal{R}'(X)|) \leq O(\text{cr}_\varphi(\mathbf{G}) + |E^*|).$$

By Theorem 12, every vertex in $S_2(X)$ is an endpoint of a block in $\mathcal{F}(X)$, or it has degree 2 in X . Let $\tilde{S}_2(X) \subseteq S_2(X)$ denote the set of vertices of $S_2(X)$ that either have degree 2 in X , or serve as endpoints of blocks in $\mathcal{R}(X) \cup \mathcal{R}'(X)$, and let $S_2'(X) = S_2(X) \setminus \tilde{S}_2(X)$. Additionally, let $\tilde{S}_2 = \bigcup_{X \in \mathcal{C}} \tilde{S}_2(X)$, and $S_2' = S_2 \setminus \tilde{S}_2$. Since, as we already observed, vertices that have degree 2 in X belong to \mathcal{S}_X , we have that:

$$|\tilde{S}_2| \leq \sum_{X \in \mathcal{C}} (2|\mathcal{R}(X)| + 2|\mathcal{R}'(X)| + |\mathcal{S}_X|) \leq O(\text{cr}_\varphi(\mathbf{G}) + |E^*|).$$

We let $\tilde{E}_2(X) \subseteq E_2(X)$ denote the edges of $E_2(X)$ that have at least one endpoint in $\tilde{S}_2(X)$, and $E_2'(X) = E_2(X) \setminus \tilde{E}_2(X)$. Additionally, let $\tilde{E}_2 = \bigcup_{X \in \mathcal{C}} \tilde{E}_2(X)$, and $E_2' = E_2 \setminus \tilde{E}_2$. Clearly,

$$|\tilde{E}_2| \leq d_{\max} |\tilde{S}_2| \leq O(d_{\max})(\text{cr}_\varphi(\mathbf{G}) + |E^*|).$$

It now only remains to bound the number of irregular vertices in set S_2' , and the number of irregular edges in set E_2' . From our definitions, for each $X \in \mathcal{C}$, for each $v \in S_2'(X)$, there is a block $B \in \mathcal{F}(X) \setminus (\mathcal{R}(X) \cup \mathcal{R}'(X))$, such that v is an endpoint of B . Moreover, for each $e \in E_2'(X)$, both endpoints of e belong to $S_2'(X)$.

Step 3: Taking care of tunnels

We now consider blocks of $\mathcal{F}(X) \setminus \mathcal{R}(X)$. The degree of each such block in $\mathcal{T}(X)$ is 2. A *tunnel* Z is a maximal path in $\mathcal{T}(X)$ containing blocks in $\mathcal{F}(X) \setminus \mathcal{R}(X)$. Let $\mathcal{Z}(X)$ denote the set of all such tunnels in $\mathcal{T}(X)$, and let $\mathcal{Z} = \bigcup_{X \in \mathcal{C}} \mathcal{Z}(X)$. Notice that each pair of tunnels is completely disjoint in the tree $\mathcal{T}(X)$ (but their blocks may share vertices: if the first block of one of the tunnels is a descendant of the last block of another in $\mathcal{T}(X)$, then the blocks are nested; also, the first blocks of two tunnels can share endpoints).

The parent of the first block (closest to the root of $\mathcal{T}(X)$) in a tunnel belongs to $\mathcal{R}(X)$. Therefore, by Corollary 3, the total number of tunnels is at most

$$|\mathcal{Z}| \leq \sum_{X \in \mathcal{C}} |\mathcal{R}'(X)| = O(\text{cr}_\varphi(\mathbf{G}) + |E^*|). \quad (3)$$

Consider some tunnel $Z = B_1 \supset \dots \supset B_\kappa$. Denote the endpoints of the block B_i by (u_i, v_i) , for $1 \leq i \leq \kappa$. Let $B' \subseteq B_\kappa$ be the unique child of block B_κ in $\mathcal{T}(X)$, and denote its endpoints by (u', v') . Since a tunnel consists of consecutive blocks in $\mathcal{T}(X)$, none of which are in $\mathcal{R}_2(X)$, all blocks in the tunnel have the same connector vertex. Denote $x = x_{B_1}$, $y = y_{B_1}$, $P_0 = P_0^{B_1}$, and recall that for all $1 \leq i \leq \kappa$, $x_{B_i} = x$, $y_{B_i} = y$,

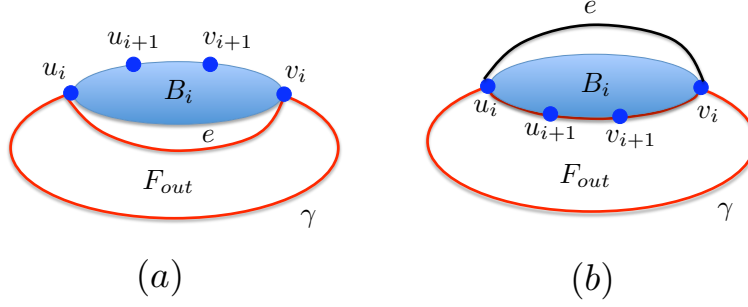


Figure 12: Iteration i .

and $P_0^{B_i} = P_0$. Let $P_{in} = P_{in}^{B'}$ and $P_{out} = P_{out}^{B_i}$. Note that x is an inner vertex of P_{in} , and y is an inner vertex of P_{out} . All three paths $P_0 : x \rightsquigarrow y$, $P_{in} : u' \rightsquigarrow v'$ and $P_{out} : u \rightsquigarrow v$ share no vertices except for x and y .

We define two auxiliary graphs corresponding to the tunnel Z . First, we remove all inner vertices of B' from B_1 , to obtain the graph \mathbf{H}_Z . We then add paths P_0, P_{out}, P_{in} to \mathbf{H}_Z , contracting all degree-2 vertices in the subgraph $P_0 \cup P_{out} \cup P_{in}$, to obtain the graph J_Z . Therefore, the paths P_0, P_{out} and P_{in} are represented by 5 edges in J_Z (see Figure 13). We call these edges *artificial edges*.

Observe that ψ_{init} induces a planar drawing ψ_Z of the graph $\mathbf{H}_Z \cup P_{in} \cup P_{out}$. However, in this drawing, we are not guaranteed that the vertices $(v_1, v_2, \dots, v_\kappa, v', u', u_\kappa, \dots, u_1)$ all lie on the boundary of the same face. Our next goal is to change the drawing ψ_Z to ensure that all these vertices lie on the boundary of the same face. We can then extend this drawing to obtain a planar drawing of J_Z . Combining the final drawings ψ_Z for all tunnels Z will give the final drawing ψ of the whole graph.

We start with the drawing ψ_Z of $\mathbf{H}_Z \cup P_{in} \cup P_{out}$, induced by ψ_{init} . We then perform κ iterations. In iteration $i : 1 \leq i \leq \kappa$, we ensure that all vertices in $(v_1, v_2, \dots, v_{i+1}, u_{i+1}, \dots, u_1)$ lie on the boundary of the same face. We refer to this face as the outer face. For convenience, we denote v' and u' by $v_{\kappa+1}$ and $u_{\kappa+1}$, respectively.

Consider some iteration $i : 1 \leq i \leq \kappa$, and assume that we are given a current drawing ψ_Z of $H_Z \cup P_{in} \cup P_{out}$, in which the vertices in $(v_1, v_2, \dots, v_i, u_i, \dots, u_1)$ lie on the boundary γ of the outer face F_{out} of the drawing. Let ψ_i be the drawing, induced by ψ_Z , of the graph $B_i \cup \gamma$. Let ψ'_i be the drawing obtained from ψ_i after we replace B_{i+1} with a single edge. Notice that (u_i, v_i) both lie on γ , so we can view γ as the drawing of the path $P_{out}^{B_i}$. Recall that in the unique planar drawing $\pi_{\tilde{B}'_i}$ of \tilde{B}'_i , the four vertices $u_i, v_i, u_{i+1}, v_{i+1}$ all lie on the boundary of the same face. In particular, there is a cycle $C_i \subseteq B_i$, such that $u_i, v_i, u_{i+1}, v_{i+1} \in C_i$, and if γ_i denotes the drawing of C_i given by $\pi_{\tilde{B}'_i}$, then all edges and vertices of $B_i \setminus C_i$ are drawn inside γ_i . Let C'_i, C''_i be the two segments connecting u_i to v_i in C_i . Notice that both u_{i+1} and v_{i+1} must belong to the same segment, since otherwise, the ordering of the four vertices along C_i is either $(v_i, v_{i+1}, u_i, u_{i+1})$, or $(u_i, v_{i+1}, v_i, u_{i+1})$, and the images of the artificial edges (u_i, v_i) and (u_{i+1}, v_{i+1}) would cross in $\pi_{\tilde{B}'_i}$. Assume w.l.o.g. that $u_{i+1}, v_{i+1} \in C'_i$. We have three possibilities. The first possibility is that the vertices u_{i+1}, v_{i+1} belong to γ – in this case we do nothing. The second possibility is that the segment $C'_i \subseteq \gamma$. In this case we can “flip” the drawing of B_i , so that now C'_i lies on the boundary of the outer face of the drawing of \mathbf{H}_Z , thus ensuring that all vertices $(v_1, v_2, \dots, v_{i+1}, u_{i+1}, \dots, u_1)$ lie on the boundary of the outer face. The third possibility is that there is an edge $e = (u_i, v_i)$ that belongs to γ . In this case, we “flip” the image of the edge e (possibly together with the image of B_i), so that C'_i becomes the part of the boundary of the outer face (see Figure 12).

Let ψ_Z be this new embedding of the graph \mathbf{H}_Z . Since different tunnels are completely disjoint (except

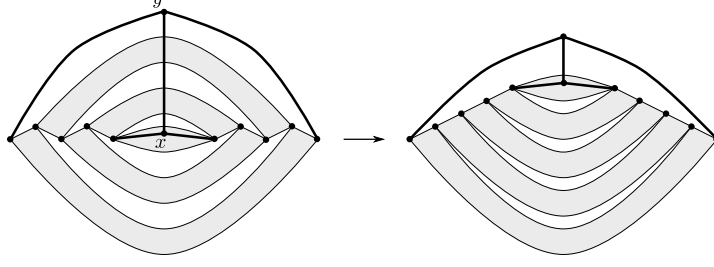


Figure 13: Graph J_Z . Bold lines are the artificial edges, representing the paths P_0, P_{in} and P_{out} . The second figure shows the outcome of the flipping procedure, where all vertices $u_1, u_2, \dots, u_\kappa, v_\kappa, \dots, v_1$ lie on the boundary of one face.

that it is possible that the last block of one tunnel contains the first block of another), we can perform this operation independently for each tunnel $Z \in \mathcal{Z}(X)$, for all $X \in \mathcal{C}$ and the resulting planar embedding ψ is our final planar embedding of \mathbf{H} . Notice that for every tunnel Z , we can naturally extend ψ_Z to a planar embedding $\psi(J_Z)$ of J_Z , by adding a planar drawing of the 5 artificial edges of J_Z inside the face on whose boundary the vertices $u_1, u_2, \dots, u_\kappa, v_\kappa, \dots, v_1$ lie.

It now only remains to bound the number of irregular vertices in $\text{IRG}_V(\varphi, \psi) \cap S'_2$, and the number of irregular edges in $\text{IRG}_E(\varphi, \psi) \cap E'_2$.

For every tunnel $Z \in \mathcal{Z}$, let $\hat{S}_2(Z) = \{u \in V(J_Z) : \exists v \in V(J_Z) \text{ s.t. } (u, v) \text{ is a 2-separator for } J_Z\}$. We need the following lemma, whose proof appears in the end of this section.

Lemma 7 *For every tunnel $Z \in \mathcal{Z}$, $|\hat{S}_2(Z)| \leq 8$.*

We now show how to complete the proof of Lemma 3, using Lemma 7.

Recall that φ is the optimal embedding of \mathbf{G} . For each tunnel $Z \in \mathcal{Z}$, we define the following drawing $\varphi(J_Z)$: first, erase from φ all edges and vertices, except those participating in Z, P_0, P_{in} and P_{out} (that have been defined for Z). Next, route the five artificial edges of J_Z along the images of the paths P_0, P_{in} and P_{out} . Finally, if any pair of artificial edges crosses more than once in the resulting embedding, perform uncrossing, that eliminates such multiple crossings, without increasing the number of other crossings in the drawing. Let $\text{cr}_{\varphi(J_Z)}$ denote the number of crossings in the resulting drawing. Since the five artificial edges may have at most 25 crossings with each other, we have that:

$$\text{cr}_{\varphi(J_Z)} \leq \text{cr}_{\varphi(\mathbf{H}_Z, \mathbf{G})} + 25$$

and

$$\sum_{Z \in \mathcal{Z}} \text{cr}_{\varphi(J_Z)} \leq O(\text{cr}_{\varphi(\mathbf{G})}) + O(|\mathcal{Z}|) \leq O(\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|).$$

Fix some tunnel $Z \in \mathcal{Z}$. Since the drawing $\psi(J_Z)$ is planar, we can apply Lemma 2 to the drawings $\psi(J_Z), \varphi(J_Z)$ of J_Z , and get that:

$$|\text{IRG}_V(\psi(J_Z), \varphi(J_Z))| \leq O(\text{cr}_{\varphi(J_Z)}(J_Z) + |\hat{S}_2(Z)|)$$

and

$$|\text{IRG}_E(\psi(J_Z), \varphi(J_Z))| \leq O(d_{\max})(\text{cr}_{\varphi(J_Z)}(J_Z) + |\hat{S}_2(Z)|).$$

Summing up over all tunnels $Z \in \mathcal{Z}$, we get that:

$$\sum_{Z \in \mathcal{Z}} |\text{IRG}_V(\psi(J_Z), \varphi(J_Z))| \leq O(\text{OPT}_{\text{cr}}(G) + |E^*|) + O(\mathcal{Z}) = O(\text{OPT}_{\text{cr}}(G) + |E^*|)$$

and

$$\sum_{Z \in \mathcal{Z}} |\text{IRG}_E(\psi(J_Z), \varphi(J_Z))| \leq O(d_{\max})(\text{OPT}_{\text{cr}}(G) + |E^*|)$$

Finally, we observe that since the tunnels are disjoint, if $v \in S'_2$, $v \in V(Z)$, and $v \in \text{IRG}_V(\varphi, \psi)$, then either $v \in \text{IRG}_V(\psi(J_Z), \varphi(J_Z))$, or v is an endpoint of the first block of the tunnel Z . Therefore,

$$|\text{IRG}_V(\varphi, \psi) \cap S'_2| \leq \sum_{Z \in \mathcal{Z}} (|\text{IRG}_V(\psi(J_Z), \varphi(J_Z))| + 2) \leq O(\text{cr}_{\varphi(J_Z)}(J_Z) + |E^*|).$$

Each edge in E'_2 has both endpoints in S'_2 , and therefore must be either completely contained in some tunnel, or be adjacent to an endpoint of the first block of a tunnel. So if $e \in E'_2$, and $e \in \text{IRG}_E(\varphi, \psi)$, then either $e \in \text{IRG}_E(\psi(J_Z), \varphi(J_Z))$ for some tunnel Z , or it is adjacent to an endpoint of the first block of some tunnel Z . Therefore,

$$|\text{IRG}_E(\varphi, \psi) \cap E'_2| \leq \sum_{Z \in \mathcal{Z}} (|\text{IRG}_V(\psi(J_Z), \varphi(J_Z))| + 2d_{\max}) \leq O(d_{\max})(\text{cr}_{\varphi(J_Z)}(J_Z) + |E^*|).$$

It now only remains to prove Lemma 7.

Proof of Lemma 7

Consider a tunnel $Z = (B_1, \dots, B_\kappa)$, $Z \subseteq X$ for some $X \in \mathcal{C}$. We will show that if we remove any pair of vertices, except for, possibly, pairs in the set $\mathcal{P} = \{(x, u_\kappa), (x, v_\kappa), (y, u_2), (y, v_2)\}$, the graph J_Z remains connected. For convenience, we denote $u_{\kappa+1} = u'$ and $v_{\kappa+1} = v'$, the endpoints of the unique child B' of B_κ .

Observe that every vertex $a \notin \{u_1, v_1, u_{\kappa+1}, v_{\kappa+1}\}$ of J_Z has degree at least 3 in J_Z : otherwise we would have a vertex $a \in V(Z) \setminus \{u_1, v_1, u_{\kappa+1}, v_{\kappa+1}\}$ with $\deg_X a \leq 2$. Then either a is incident on an edge in $E^*(X)$, or it belongs to S_1 , and therefore $a \in \mathcal{S}_X$. Then the smallest B_i of Z that contains a would belong to $\mathcal{R}_1(X)$.

Consider now a vertex $w \notin \{u_i, v_i, x, y : 1 \leq i \leq \kappa + 1\}$. Let B_j be the smallest block that contains w . Since $V(B_j) \setminus V(B_{j+1})$ contains an inner vertex w , B_j must be a complex block. Therefore, from Claim 2, w is connected to u_j, v_j and either u_{j+1} or v_{j+1} by three vertex disjoint paths. It is obvious that each of the vertices x and y is connected to u_1, v_1 , and either $u_{\kappa+1}$ and $v_{\kappa+1}$ by three vertex disjoint paths.

Let $L = \{u_i, v_i : 1 \leq i \leq \kappa + 1\}$. Observe that it is enough to show that for any pair $(p, q) \notin \mathcal{P}$ of vertices, all vertices in $L \setminus \{p, q\}$ remain connected in the graph $J_Z \setminus \{p, q\}$. Indeed, assume that all vertices in set $L \setminus \{p, q\}$ remain connected in graph $J_Z \setminus \{p, q\}$. Let w, w' be any pair of vertices of $J_Z \setminus \{p, q\}$. We show that w, w' remain connected as well in the resulting graph. Indeed, if both $w, w' \notin L$, each one of these vertices has three paths disjoint paths connecting them to vertices of L , and at least one of the three paths must survive even after the removal of p, q from J_Z . Similarly, if one of vertices w, w' belongs to L , they remain connected as long as all vertices of L remain connected.

We now show that for any pair (p, q) of vertices, $(p, q) \notin \mathcal{P}$, the vertices in $L \setminus \{p, q\}$ all remain connected in graph $J_Z \setminus \{p, q\}$.

Consider the unique planar embedding of J_Z . If we remove the edge (x, y) from this embedding, we obtain a cycle C containing the vertices of L : this cycle is simply the boundary of the face that contained the edge (x, y) . Denote the ordering of vertices on the cycle by $(v_1, v_2, \dots, v_{\kappa+1}, x, u_{\kappa+1}, \dots, u_1, y)$, where (u_i, v_i) are endpoints of block B_i (but observe that some consecutive vertices in this ordering may coincide, e.g. it is possible that $u_i = u_{i+1}$). Let $L_u = \{u_i : 1 \leq i \leq \kappa + 1\}$, and $L_v = \{v_i : 1 \leq i \leq \kappa + 1\}$.

Consider a vertex u_i with $\deg_X u_i \geq 3$. Denote its neighbors on the cycle by u_{i_1} and u_{i_2} (since some vertices u_j might coincide, the vertex u_{i_1} is not necessarily equal to u_{i-1}). Let w be a neighbor of u_i other than u_{i_1} and u_{i_2} . Clearly, $w \notin L_u$, because of definition of block.

Assume first that $w \notin L_v$ either. Consider the smallest block B_h that contains w . Again, since $w \notin L$, block B_h has to be complex. Therefore, there is a path from w to v_h that does not visit any vertex in L_u . By concatenating this path with the edge (u_i, w) , we get a path $P(u_i)$ from u_i to v_h that does not visit any other vertex in L_u . If $w \in L_v$, then the path $P(u_i)$ is simply the edge (u_i, w) . Similarly, there is a path $P(v_i)$ from every vertex v_i of degree at least 3 to some $u_h \in L_u$, that does not visit any other vertex in L_v .

Assume for contradiction that there is some pair $(p, q) \notin \mathcal{P}$, such that in the graph $J_Z \setminus \{p, q\}$, the set $L \setminus \{p, q\}$ is not connected. Since all vertices of L lie on the cycle C , it is clear that p and q must belong to C . Moreover, both of them must lie on the same of the two arcs connecting x and y in C . Let us say that (p, q) belong to the arc on which the vertices L_u lie. Then the other arc of C remains connected. But then for each vertex $u_j \notin \{p, q\}$ of degree at least 3, the path $P(u_j)$ will connect it to the vertices in L_v . Similarly, if (p, q) belong to the arc containing vertices of L_v , each vertex $v_j \notin \{p, q\}$ of degree at least 3, remains connected to the vertices of L_u via the path $P(v_j)$.

Now if u_i has degree 2, then it must be either u_1 or $u_{\kappa+1}$. It is clear that we can disconnect u_1 from u_2 and y only by removing both u_2 and y (since B_i is 2-vertex connected). Similarly, we can disconnect $u_{\kappa+1}$ from u_κ and x only by removing both u_κ and x . If vertex v_i has degree 2, then it must be either v_1 , or $v_{\kappa+1}$, and so the only pairs of vertices that can disconnect it are (x, v_κ) and (y, v_2) .

C Handling Non 3-Connected Graphs

In this section we prove Theorem 1, by describing a reduction from the general case — when the graph G is not necessarily 3-connected — to the 3-connected case. Our algorithm consists of two parts. In the first part, we decompose the original graph \mathbf{G} into a number of sub-graphs, and find a drawing for each one of the sub-graphs separately. In the second part, we combine these drawings together to obtain the final drawing.

C.1 Part 1: Decomposition

We first note that we can assume w.l.o.g. that the input graph \mathbf{G} is 2-connected: Otherwise, we can separately embed the 2-connected components of \mathbf{G} and then combine their embeddings. We also assume that the graph $\mathbf{H} = \mathbf{G} \setminus E^*$ is connected, since otherwise we can start removing edges from E^* and adding them to \mathbf{H} , until it becomes connected, as in Section 3. Finally, we can assume w.l.o.g. that the input graph contains no parallel edges: otherwise, if there is a collection (e_1, \dots, e_κ) of parallel edges, we can subdivide each edge e_i by adding a vertex v_i to it, and add edges connecting every consecutive pair (v_i, v_{i+1}) of vertices, for $1 \leq i \leq \kappa$ (we identify $v_{\kappa+1}$ and v_1). It is easy to verify that this transformation does not increase the maximum vertex degree in \mathbf{G} , and does not increase the cost of the optimal solution. We will use the following theorem of Hliněný and Salazar [21].

Theorem 13 ([21]) *Let G be any graph of maximum degree d_{\max} , $e \in E(G)$, such that $G \setminus \{e\}$ is planar. Then we can efficiently find a drawing ψ of G with at most $O(d_{\max} \cdot \text{OPT}_{\text{cr}}(G))$ crossings.*

Our high-level idea is to decompose the graph \mathbf{G} into blocks (see Section A for the definition). For each such block B , we will add an artificial edge connecting its endpoints, that will “simulate” the rest of the graph, $\mathbf{G} \setminus B$. Similarly, we will add an artificial edge connecting the endpoints of B to the remaining graph, $\mathbf{G} \setminus B$, that will simulate B . In the course of such recursive decomposition, a block may end up containing a number of such artificial edges. We will then try to find drawings of each such augmented block separately. We need to argue that the total optimal solution cost in these new sub-problems does not increase by much. This is done as follows. We will have two types of blocks. The first type is blocks that have at most two artificial edges, and when one of these edges is removed from the block, we obtain a planar graph. For such blocks, we will argue that their total solution cost is bounded by $O(\text{OPT}_{\text{cr}}(\mathbf{G}))$, and then use Theorem 13

to find their drawings. For the remaining blocks, we will show that we can augment the set E^* of edges, to set \hat{E}^* , with $|\hat{E}^*| \leq O(|E^*|)$, such that for each such block B , $B \setminus \hat{E}^*$ is planar. (Observe that now \hat{E}^* may have to contain artificial edges). We then use Theorem 6 to find the drawing of each such block separately. We now describe the decomposition procedure in more detail.

We say that a path P in graph \mathbf{G} is *nice* if it does not contain any edge of E^* . We will be repeatedly using the following two easy observations.

Observation 1 *Let G be any graph, $E^* \subseteq E(G)$ a subset of edges, such that $G \setminus E^*$ is planar. Let B be a block of G , whose endpoints are $I(B) = (u, v)$, and assume that B contains a nice path connecting u to v . Let G' be the graph obtained by removing B from G , and adding an artificial edge $e = (u, v)$ to it. Then $G' \setminus E^*$ is also planar.*

Observation 2 *Let G be any 2-connected graph, and let B be any block of G with endpoints $I(B) = (u, v)$. Then there is a path $P : u \rightsquigarrow v$ contained in $(G \setminus B) \cup \{u, v\}$.*

We say that a block B of G is nice, iff it does not contain any edge of E^* . We start by iteratively removing **maximal** (w.r.t. inclusion) nice blocks from G , and adding each one of them to the set \mathcal{A} of nice blocks we construct. Consider one such block B with endpoints $I(B) = (u, v)$. We remove block B from G , and add it to the set \mathcal{A} of nice blocks. We then add an artificial edge (u, v) both to the remaining graph G , and to the block B . Let \mathcal{A} be the resulting set of these augmented nice blocks, and let G' be the resulting remaining graph. From Observation 1, graph $G' \setminus E^*$ is planar. It is also easy to see that G' does not contain any nice blocks, and it is 2-connected. Consider now some block $B \in \mathcal{A}$. Since the algorithm was repeatedly choosing maximal nice blocks, B contains a unique artificial edge, $e = (u, v)$, connecting its endpoints $I(B) = (u, v)$. Moreover, from Observation 2, the graph $(\mathbf{G} \setminus B) \cup \{u, v\}$ contains a path $P : u \rightsquigarrow v$. Therefore, $\text{OPT}_{\text{cr}}(B) \leq \text{cr}_{\varphi}(B, \mathbf{G})$, and $\sum_{B \in \mathcal{A}} \text{OPT}_{\text{cr}}(B) \leq O(\text{OPT}_{\text{cr}}(\mathbf{G}))$. The artificial edges that have been added to graph G' at this step are called type-1 artificial edges, and all artificial edges that will be added throughout the rest of the algorithm are called type-2 artificial edges. As our next step, we use Theorem 12 to find a laminar block decomposition \mathcal{F} for graph G' . Recall that for each block $B \in \mathcal{F}$, we denote by \tilde{B} the graph obtained from B by replacing its children with artificial edges, and \tilde{B}' is obtained from \tilde{B} by adding an artificial edge connecting the endpoints of B . For each $B \in \mathcal{F}$, we are guaranteed that \tilde{B}' is 3-connected. Intuitively, we would now like to solve each one of the blocks \tilde{B}' , for $B \in \mathcal{F}$, separately. However, since we have added artificial edges to such blocks, the graph $\tilde{B}' \setminus E^*$ may not be planar anymore. Of course, if we add all type-2 artificial edges to the set E^* , this problem will be resolved, but then the resulting set E^* may become too large. We show below how to avoid this problem. We start by defining a set \mathcal{B} of blocks, whose size will be bounded by $O(|E^*|)$. We show that the type-2 artificial edges belonging to all such blocks can be added to the set E^* without increasing its size by too much. We then show how to take care of remaining blocks.

- Let $\mathcal{B}_1 \subseteq \mathcal{F}$ be the set of blocks \tilde{B}' , for $B \in \mathcal{F}$, such that \tilde{B}' contains an edge of E^* . Clearly, $|\mathcal{B}_1| \leq O(|E^*|)$, and all the leaves of the tree \mathcal{T} belong to \mathcal{B}_1 (since otherwise such a leaf would be a nice block).
- Let \mathcal{B}_2 be the set of blocks \tilde{B}' , such that B has at least two children in the tree \mathcal{T} . Since all leaves of the tree \mathcal{T} belong to \mathcal{B}_1 , it is easy to see that $|\mathcal{B}_2| \leq |\mathcal{B}_1| \leq |E^*|$.

Consider the decomposition tree \mathcal{T} , and remove all the vertices corresponding to the blocks in \mathcal{B}_1 and \mathcal{B}_2 from it. The resulting sub-graph of \mathcal{T} is simply a collection \mathcal{P} of disjoint paths. Moreover, $|\mathcal{P}| \leq |\mathcal{B}_1| + |\mathcal{B}_2| \leq 2|E^*|$. Consider some such path $P \in \mathcal{P}$, and assume that $P = (B_1, B_2, \dots, B_k)$, where $B_k \subset B_{k-1} \subset \dots \subset B_1$. Let i^* be the largest index $i : 1 \leq i \leq k$, such that B_i contains a nice path connecting its endpoints. We add $\tilde{B}'_{i^*}, \tilde{B}'_{i^*+1}, \dots, \tilde{B}'_k$ to \mathcal{B}_3 . We show in the next claim that $i^* \geq k - 5$, so $|\mathcal{B}_3| \leq 6|\mathcal{P}| \leq 12|E^*|$.

Claim 6 $i^* \geq k - 5$.

Proof: Assume otherwise. Then the laminar family \mathcal{F} contains four blocks B^1, B^2, B^3, B^4 , such that for $1 \leq j < 4$, B^j is the father of B^{j+1} in \mathcal{T} , and $B^j \setminus B^{j+1}$ does not contain edges of E^* . Moreover, each one of the four blocks has exactly one child, and none of these blocks contains a nice path connecting its endpoints.

Denote $I(B^1) = (u, v)$ and $I(B^4) = (u', v')$. Since B^1 does not contain a nice path connecting its endpoints, and $B^1 \setminus B^2, B^2 \setminus B^3, B^3 \setminus B^4$ do not contain edges of E^* , all paths $P' : u \rightsquigarrow v$ in B^1 must contain u' and v' . Therefore, if we remove the vertices of $B^4 \setminus \{u', v'\}$ from B^1 , we will obtain two connected components, R and R' , where $u \in R$ and $v \in R'$. Each one of the two components contains exactly one vertex from $\{u', v'\}$, and we assume w.l.o.g. that $u' \in R, v' \in R'$.

We now claim that R does not contain any vertices outside of u and u' , that is, R is just a collection of parallel edges (or just a single edge): otherwise, R is a nice block with end-points at u and u' , and we have assumed that G' does not contain any nice blocks. Similarly, R' does not contain any vertices outside of v and v' . Therefore, the set of vertices of block B^1 is $V(B^4) \cup \{v, u\}$. But then it is impossible that there are two additional blocks, B^2, B^3 , such that $B^4 \subset B^3 \subset B^2 \subset B^1$, as every pair of distinct blocks must differ in their vertices. \square

Finally, we let $C_P = (B_1 \setminus B_{i^*}) \cup I(B_{i^*})$. We add to C_P two artificial edges: edge e connecting the endpoints of B_1 , and edge e' connecting the endpoints of B_{i^*} . For simplicity, we denote the new graph by C' , and the old graph by $C = C_P$. We then add C' to a new set \mathcal{C} of sub-graphs of G' . We need the following claim:

Claim 7 *Graph $C' \setminus \{e\}$ is planar. Moreover, $\text{OPT}_{\text{cr}}(C') \leq \text{cr}_{\varphi}(C, \mathbf{G}) + 1$.*

Proof: Since block B_{i^*} contained a nice path, denoted by P' , connecting v_{i^*} to u_{i^*} , and since $B_1 \setminus B_{i^*}$ did not contain edges of E^* , from Observation 1, $C \cup \{e'\} = C' \setminus \{e\}$ is planar.

For the second part, from Observation 2, there is a path P'' , connecting the endpoints of B_1 in graph $(\mathbf{G} \setminus B_1) \cup I(B_1)$. Consider now the optimal embedding φ of \mathbf{G} , and remove from it all edges and vertices except for those in C, P', P'' . This drawing gives a drawing φ of graph C' , where edge e is drawn along the image of P' , and edge e' along the image of P'' . The number of crossings in this drawing is at most $\text{cr}_{\varphi}(C, \mathbf{G}) + \text{cr}_{\varphi}(P', P'')$. Finally, if the images of edges e, e' cross multiple times in φ , we can un-cross them, without increasing the number of crossings between any other pair of edges. This will result in a drawing of C' with at most $\text{cr}_{\varphi}(C, \mathbf{G}) + 1$ crossings. \square

Since $|\mathcal{C}| \leq |\mathcal{P}| \leq 2|E^*|$, we have that $\sum_{C' \in \mathcal{C}} \text{OPT}_{\text{cr}}(C') \leq \sum_{C' \in \mathcal{C}} (\text{cr}_{\varphi}(C, \mathbf{G}) + 1) \leq O(\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|)$.

Finally, let $\mathcal{B} = \mathcal{B}_1 \cup \mathcal{B}_2 \cup \mathcal{B}_3$. From the above discussion, $|\mathcal{B}| \leq O(|E^*|)$. Moreover, the total number of children of blocks in \mathcal{B} in the tree \mathcal{T} is also bounded by $O(|\mathcal{B}|) \leq O(|E^*|)$. Let $\tilde{B}' \in \mathcal{B}$ be any such block, and let B_1, \dots, B_{κ} be its children. Recall that \tilde{B}' is obtained from B by replacing each child B_i by an artificial edge e_i , connecting the endpoints (v_i, u_i) of B_i . For each such child B_i , there is also a path $P_i : v_i \rightsquigarrow u_i$, $P_i \subseteq B_i$. Additionally, we have added an edge e connecting the endpoints (u, v) of block B . We associate this edge with a path $P : u \rightsquigarrow v$, $P \subseteq (\mathbf{G} \setminus B) \cup I(B)$, that is guaranteed by Observation 2. We now add the edges $e, e_1, \dots, e_{\kappa}$ to E^* , and we denote by \hat{E}^* be the resulting set of edges. We then have $|\hat{E}^*| \leq O(|E^*|)$.

Clearly, for each $\tilde{B}' \in \mathcal{B}$, $\tilde{B}' \setminus \hat{E}^*$ is planar. This is since $B \setminus E^*$ was planar, and all the edges that have been added to B in order to obtain \tilde{B}' , were also added to \hat{E}^* . Moreover, $\text{OPT}_{\text{cr}}(\tilde{B}') \leq \text{OPT}_{\text{cr}}(\mathbf{G})$, since we have a collection $P, P_1, \dots, P_{\kappa}$ of edge disjoint paths associated with the edges $e, e_1, \dots, e_{\kappa}$, that are contained in $\mathbf{G} \setminus \tilde{B}'$, connecting the endpoints of their corresponding edges.

Let $\Gamma = \mathcal{A} \cup \mathcal{B} \cup \mathcal{C}$. Notice that since we have added artificial edges, it is possible that graphs $A \in \Gamma$ now contain parallel edges. For each such graph $A \in \Gamma$, we let A' denote the corresponding graph with no parallel edges, that is, we replace every set of parallel edges with a single edge. Let $\mathcal{A}', \mathcal{B}', \mathcal{C}'$ and Γ' be the collections of these modified graphs, corresponding to the collections $\mathcal{A}, \mathcal{B}, \mathcal{C}$ and Γ , respectively.

As we have already observed, each graph $A' \in \mathcal{A}' \cup \mathcal{C}'$ can be decomposed into a planar graph plus one additional edge, and $\sum_{A \in \mathcal{A} \cup \mathcal{C}} \text{OPT}_{\text{cr}}(A) \leq O(\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|)$. Using Theorem 13, we can efficiently find drawings $\psi_{A'}$ of graphs $A' \in \mathcal{A}' \cup \mathcal{C}'$, with at most $O(d_{\max} \cdot (\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|))$ crossings in total. We can also use Theorem 6 to find a drawing $\psi_{A'}$ for each graph $A' \in \mathcal{B}'$, having at most $O(d_{\max} \cdot |E^*| \cdot (\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|))$ crossings in total. Overall, from the above discussion, for each graph $A' \in \Gamma'$, we can efficiently find a drawing $\psi_{A'}$ of A' , such that the total number of crossings in these drawings is bounded by $O(d_{\max} \cdot |E^*| \cdot (\text{OPT}_{\text{cr}}(\mathbf{G}) + |E^*|))$.

C.2 Part 2: Composition of Drawings

In this section we show how to compose the drawings of the graphs in Γ' , to obtain the final drawing of \mathbf{G} . We build a binary decomposition tree \mathcal{T}' corresponding to the collection Γ of sub-graphs of \mathbf{G} , as follows. The graph at the root of the tree is \mathbf{G} . The graphs at the leaves of \mathcal{T}' are the graphs in Γ . For every non-leaf node, the corresponding graph G_0 is the composition of its two child subgraphs G_1 and G_2 along the unique artificial edge that belongs to both G_1 and G_2 . Notice that our original decomposition tree \mathcal{T} can be turned into a binary tree whose leaves are graphs in $\mathcal{B} \cup \mathcal{C}$, and we can add graphs in \mathcal{A} to this tree one-by-one, as we merge them with the root of the tree, to obtain the final binary tree \mathcal{T}' .

Theorem 14 *Suppose that we are given the decomposition tree \mathcal{T}' , and drawings $\psi_{A'}$ of graphs $A' \in \Gamma'$. Then we can efficiently find a drawing of \mathbf{G} with at most $d_{\max}^2 \sum_{A' \in \Gamma'} \text{cr}_{\psi_{A'}}(A')$ crossings.*

Proof: We start by assigning weights to the edges of the graphs in the decomposition tree \mathcal{T}' . Once the weights are assigned, for each graph G_0 in the tree, the weighted degree of a vertex $x \in V(G_0)$, denoted by $\text{deg}_{G_0}^w x$, is the sum of the weights of the edges incident to x in G_0 . We assign the weights to the edges of the graphs from the top to the bottom of the tree \mathcal{T}' . For the root graph \mathbf{G} , the weights of all its edges (which are non-artificial edges), are 1. Let G_0 be the current graph, with its two children G_1 and G_2 , that share an artificial edge $e = (u, v)$. The weights of all edges, other than the edge e remain in graphs G_1 and G_2 the same as in graph G_0 . The weight of the edge e is set in both graphs G_1 and G_2 to be:

$$\text{weight}(e) = \min \left\{ \text{deg}_{G_1 \setminus e}^w(u), \text{deg}_{G_1 \setminus e}^w(v), \text{deg}_{G_2 \setminus e}^w(u), \text{deg}_{G_2 \setminus e}^w(v) \right\}.$$

It is easy to see that if, for all vertices $x \in V(G_0)$, $\text{deg}_{G_0}^w(x) \leq d_{\max}$, then for all vertices $y \in V(G_i)$, for $i \in \{1, 2\}$, $\text{deg}_{G_i}^w(y) \leq d_{\max}$ as well. Therefore, the weighted degrees of all vertices in all graphs in the tree \mathcal{T}' are bounded by d_{\max} . Finally, we assign weights to edges of graphs $A' \in \Gamma'$, as follows. Let $A \in \Gamma$ be the graph corresponding to A' . For each set e_1, \dots, e_κ of parallel edges in A , the weight of the corresponding edge in A' is the sum of the weights of the edges e_1, \dots, e_κ in A . The weights of all other edges are identical in A and A' .

We now define the weighted cost of a drawing ψ of any edge-weighted graph H , $\text{cr}_\psi^w(H)$, as follows. The weighted cost of a crossing of two edges of weights w_1 and w_2 is $w_1 w_2$. The cost of the drawing is the sum of weighted costs of all crossings.

Notice that for each graph $A' \in \Gamma'$, the drawing $\psi_{A'}$ of A' induces a drawing ψ_A of the corresponding graph $A \in \Gamma$, such that the weighted cost of ψ_A is bounded by that of $\psi_{A'}$. Since the weighted degrees of vertices in all graphs in Γ' are bounded by d_{\max} , we have

$$\sum_{A \in \Gamma} \text{cr}_{\psi_A}^w(A) \leq \sum_{A' \in \Gamma'} \text{cr}_{\psi_{A'}}^w(A') \leq d_{\max}^2 \sum_{A' \in \Gamma'} \text{cr}_{\psi_{A'}}(A').$$

We now combine all drawings of graphs in Γ as follows. We proceed from the bottom to the top of the tree \mathcal{T}' . At each node G_0 , we combine the two drawings ψ_1 and ψ_2 of its two children G_1 and G_2 into a drawing

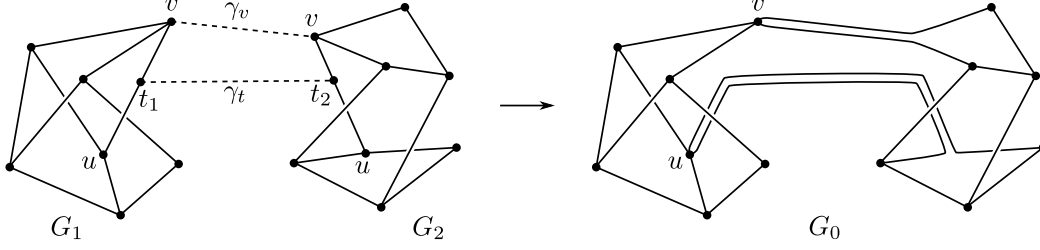


Figure 14: Obtaining a drawing for G_0 by composing the drawings for G_0 and G_1 .

ψ_0 of G_0 so that

$$\text{cr}_{\psi_0}^w(G_0) \leq \text{cr}_{\psi_1}^w(G_1) + \text{cr}_{\psi_2}^w(G_2).$$

Finally, we obtain a drawing ψ of \mathbf{G} with

$$\text{cr}_{\psi}(\mathbf{G}) \leq \text{cr}_{\psi}^w(\mathbf{G}) \leq \sum_{A' \in \Gamma'} \text{cr}_{\psi_{A'}}^w(A') \leq d_{\max}^2 \sum_{A' \in \Gamma} \text{cr}_{\psi_{A'}}(A').$$

We now show how to combine the drawings ψ_1 and ψ_2 of graphs G_1 and G_2 . Let $e = (u, v)$ be the unique artificial edge shared by G_1 and G_2 . Without loss of generality, we assume that $\text{weight}(e) = \deg_{G_2 \setminus e}^w(u)$.

We note that we can assume that the following properties hold (for each $i = 1, 2$):

- the vertex v lies on the external boundary of the drawing ψ_i ;
- there is a point t_i on the drawing of the edge e in ψ_i , such that the segment of the drawing of e between t_i and $\psi_i(v)$ lies on the external boundary of the drawing ψ_i .

If these properties do not hold, we transform each drawing ψ_i as follows. For convenience, assume that the drawing ψ_i is on the 2-sphere. We take a point t_i on the curve corresponding to the edge e in the drawing ψ_i , so that there are no crossing points on the segment of e between t_i and $\psi_i(v)$. Then we take a point t'_i that lies on the same face of ψ_i as v and t_i . Finally, we perform a stereographic projection from t'_i and obtain the desired drawing $\hat{\psi}_i$. Since v and t_i lie on the face bounding t'_i in ψ_i , it follows that they both lie on the outer face in $\hat{\psi}_i$.

We superimpose drawings ψ_1 and ψ_2 so that drawings of G_1 and G_2 do not overlap and points $\psi_1(v)$, $\psi_2(v)$, t_1 , and t_2 lie on the external boundary of the drawing. We then connect points t_1 and t_2 with a curve γ_t and points $\psi_1(v)$ and $\psi_2(v)$ with a curve γ_v so that curves γ_t and γ_v do not cross each other and do not cross the drawings of G_1 and G_2 (see Figure 14). Now, we erase the drawings of segments of $\psi_i(e)$ between points t_i and v . Let γ_u be the concatenation of remaining pieces of $\psi_1(e)$ and $\psi_2(e)$ and γ_t . The curve γ_u connects $\psi_1(u)$ and $\psi_2(u)$. Finally, we “contract” curves γ_u and γ_v : we move points $\psi_2(u)$ and $\psi_2(v)$ along the curves γ_u and γ_v , until they reach $\psi_1(u)$ and $\psi_1(v)$. We route each edge e incident to u (respectively v) in $G_2 \setminus e$: first along the curve γ_u (respectively γ_v) and then along the original drawing ψ_2 of e . (If edges parallel to e belong to G_2 , we re-route them in the same way: first along γ_v , then along their original drawing in ψ_2 , and finally along γ_u). We obtain an embedding ψ_0 of G_0 (curves γ_u , γ_v and the embeddings of the edge e are not parts of ψ_0). Figure 14 depicts an example of the above composition step.

Let us compute the cost of drawing ψ_0 . Since γ_v does not cross the drawings ψ_1 and ψ_2 , we do not introduce any new crossings when we contract γ_v . For every crossing of an edge $e' \in E(G_1 \cup G_2)$ with $\psi_1(e)$ or $\psi_2(e)$, we introduce crossings between all edges incident to u in $G_2 \setminus e$ and e' . The total weighted cost of these crossings is $\deg_{G_2 \setminus e}^w(u) \cdot \text{weight}(e')$. It is equal to $\text{weight}(e) \cdot \text{weight}(e')$, the cost of the crossing between e

and e' . Therefore, the total weighted cost of the drawing does not increase, that is,

$$\text{cr}_{\psi_0}^w(G_0) \leq \text{cr}_{\psi_1}^w(G_1) + \text{cr}_{\psi_2}^w(G_2).$$

□

D Algorithms for bounded-genus graphs

In this section we present the proof of Theorem 3.

Remark 2 *Note that for any fixed $\gamma \geq 0$, given a graph G of genus γ , we can find an embedding into a surface of genus γ in linear time [30, 23]. However, in our algorithm we do not assume that the input graph is embedded into a surface of minimum genus. Therefore, if the input graph G has genus γ , but we are only given an embedding into a surface of genus $g > \gamma$, the approximation guarantee of the drawing produced by our algorithm will depend on g . This is in particular interesting when the genus of the graph γ is super-constant, in which case computing an embedding of minimum genus becomes NP-hard [37].*

For any graph H , and an embedding τ of H into a surface of genus at least 1, the *nonseparating dual edge-width* of (H, τ) , denoted by $\text{ndew}(H, \tau)$, is the length of the shortest surface-nonseparating cycle in the dual of H , w.r.t. the embedding τ . We also write $\text{ndew}(H)$ when τ is clear from the context. We will use the following algorithmic result by Hliněný and Chimani [19].

Theorem 15 (Hliněný & Chimani [19]) *Let G be a graph embedded in an orientable surface of genus $g \geq 1$, with $\text{ndew}(G) \geq 2^{2g+2} \cdot d_{\max}$. Then there is an efficient algorithm that computes a drawing of G in the plane with at most $3 \cdot 2^{3g+2} \cdot d_{\max}^2 \cdot \text{OPT}_{\text{cr}}(G)$ crossings.*

Let G be a graph, and let σ be an embedding of G into an orientable surface S of genus g . We begin by computing an integer $\kappa \in \{0, \dots, g\}$, and a sequence of graphs G_0, \dots, G_κ . For each graph G_i we also compute a drawing σ_i of G_i into a surface of genus $g - i$. Initially, we set $G_0 = G$, and $\sigma_0 = \sigma$. For each $i : 0 \leq i < g$, if $\text{ndew}(G_i, \sigma_i) \geq 2^{2g+2} \cdot d_{\max}$, then we set $\kappa = i$, and we terminate the sequence G_0, \dots, G_κ . Otherwise, if $\text{ndew}(G_i, \sigma_i) < 2^{2g+2} \cdot d_{\max}$, then we first compute a shortest surface-nonseparating cycle C_i^* in the dual of G_i w.r.t. the embedding σ_i . Such a cycle can be found in time $O(g^3 n \log n)$ using the algorithm of Cabello and Chambers [6]. We construct G_{i+1} by removing from G_i all edges whose duals are in $E(C_i^*)$. We also construct an embedding of G_{i+1} by cutting the surface into which G_i is embedded along the cycle C_i^* . This gives us an embedding σ_{i+1} of G_{i+1} . As observed in [19], the graph G_{i+1} is a spanning subgraph of G_i , and the embedding σ_{i+1} is into a surface of genus $g - i - 1$.

Let us define

$$E_1^* = E(G) \setminus E(G_\kappa).$$

We have

$$|E_1^*| = \sum_{i=0}^{\kappa-1} |E(C_i^*)| = \sum_{i=0}^{\kappa-1} \text{ndew}(G_i, \sigma_i) < \kappa \cdot 2^{2g+2} \cdot d_{\max} \leq g \cdot 2^{2g+2} \cdot d_{\max}. \quad (4)$$

If $\kappa = g$, then the graph G_κ is drawn into a surface of genus 0, and therefore G_κ is planar. Otherwise, if $\kappa < g$, then we have a drawing σ_κ of the graph G_κ into a surface of genus $g - \kappa \geq 1$, and with

$$\text{ndew}(G_\kappa, \sigma_\kappa) \geq 2^{2g+2} \cdot d_{\max} > 2^{2(g-\kappa)+2} \cdot d_{\max}.$$

This means in particular that we can run the algorithm from Theorem 15 to obtain a drawing φ of G_κ into the plane with at most $3 \cdot 2^{3(g-\kappa)+2} \cdot d_{\max}^2 \cdot \text{OPT}_{\text{cr}}(G_\kappa)$ crossings. Define the set $E_2^* \subseteq E(G_\kappa)$ of edges to contain all edges participating in crossings in φ . We have

$$|E_2^*| \leq |\text{cr}_\varphi(G_\kappa)| \leq 3 \cdot 2^{3(g-\kappa)+2} \cdot d_{\max}^2 \cdot \text{OPT}_{\text{cr}}(G_\kappa) \leq 3 \cdot 2^{3g+2} \cdot d_{\max}^2 \cdot \text{OPT}_{\text{cr}}(G). \quad (5)$$

Let $E^* = E_1^* \cup E_2^*$. Observe that the graph $G \setminus E^* = G_\kappa \setminus E_2^*$ is planar. The assertion of Theorem 3 follows trivially if the graph G is planar, so we may assume that $\text{OPT}_{\text{cr}}(G) \geq 1$. By (4) and (5) we therefore have

$$|E^*| = |E_1^*| + |E_2^*| = 2^{O(g)} \cdot d_{\max}^2 \cdot \text{OPT}_{\text{cr}}(G).$$

Running the algorithm from Theorem 1 with the planarizing set E^* , we obtain a drawing of G into the plane with at most $d_{\max}^{O(1)} \cdot |E^*| \cdot (\text{OPT}_{\text{cr}}(G) + |E^*|) = d_{\max}^{O(1)} \cdot 2^{O(g)} \cdot \text{OPT}_{\text{cr}}^2(G)$ crossings.

To obtain an $\tilde{O}(2^{O(g)} \cdot \sqrt{n})$ -approximation for bounded-degree graphs, run the above algorithm, and the algorithm of Even et al. [12], and output the drawing with fewer crossings. \square

E Proof of Theorem 11

Recall that the algorithm of Even et al. [12] finds a drawing of a bounded degree graph with at most $O(\log^3 n \cdot (n + \text{OPT}_{\text{cr}}(G)))$ crossings. We first show that this algorithm can be extended to arbitrary graphs to produce drawings with at most $\text{poly}(d_{\max}) \log^3 n \cdot (n + \text{OPT}_{\text{cr}}(G))$ crossings, where d_{\max} is the maximum vertex degree in G . We then note that by using the approximation algorithm of Arora et al. [3] for **Balanced Separator** instead of the algorithm of Leighton and Rao [27], this guarantee can be improved to $\text{poly}(d_{\max}) \log^2 n \cdot (n + \text{OPT}_{\text{cr}}(G))$.

Lemma 8 *Suppose that there is a polynomial time algorithm \mathcal{A} , that, for any n -vertex graph $G = (V, E)$ with vertex degrees at most 3, finds a drawing of G with at most $\alpha(n + \text{OPT}_{\text{cr}}(G))$ crossings. Then there is a polynomial time algorithm that finds a drawing of any graph G with at most $O(d_{\max}^4 \cdot \alpha \cdot (n + \text{OPT}_{\text{cr}}(G)))$ crossings, where d_{\max} is the maximum vertex degree in G .*

Proof: The algorithm first constructs an auxiliary graph \tilde{G} with maximum vertex degree 3. Informally, \tilde{G} is the graph obtained from G by replacing every vertex v with a path P_v of length $\deg v$ (e.g., if G is a d -regular graph then \tilde{G} is the replacement product of G and the path of length d). Formally, the vertices of \tilde{G} are pairs (u, e) , where $u \in V$, $e \in E$ and e is incident on u . The edges of \tilde{G} consist of two subsets. First, for every edge $e = (u, v) \in E$, we connect the vertices (u, e) and (v, e) of \tilde{G} with an edge \tilde{e} . We call such edges “type 1 edges”. Additionally, for each $u \in V(G)$, if $e_1, \dots, e_{\deg u}$ is the list of all edges incident on u (in an arbitrary order), then we connect every consecutive pair $(u, e_i), (u, e_{i+1})$ of vertices, for $1 \leq i < \deg u$, with a type-2 edge. Let P_u denote the resulting path formed by these edges. This completes the description of \tilde{G} . Note that \tilde{G} has at most $d_{\max} \cdot n$ vertices, and every vertex of \tilde{G} has degree at most 3. Also note that if we contract every path P_u in \tilde{G} , for $u \in V(G)$, into a vertex, we obtain the graph G .

We now bound the crossing number of \tilde{G} . Observe that we can obtain a drawing $\varphi_{\tilde{G}}$ of \tilde{G} from any drawing φ_G of G as follows. We put each vertex (u, e) on the drawing of the edge e very close to the drawing of u . We draw each type-1 edge $\tilde{e} = ((u, e), (v, e))$ of \tilde{G} along the segment of the drawing of e in φ_G , connecting the images of (u, e) and (v, e) . We draw type-2 edges on the line segments connecting their endpoints. We now bound the number of crossings in this drawing. Notice that there are no crossings between the type-1 and the type-2 edges. The number of crossings between pairs of type-1 edges is bounded by the total number of crossings in φ_G . Finally, in order to bound the number of crossings between pairs of type-2 edges, we notice that if $u \neq v$, then the edges of P_u and P_v do not cross. Any pair of edges on path P_u may cross at most

once, since any pair of line segments crosses at most once. Therefore, there are at most $\binom{\deg u}{2}$ crossings among the edges of the path P_u for every vertex u . Overall,

$$\text{cr}_{\varphi_{\tilde{G}}}(\tilde{G}) \leq \text{cr}_{\varphi_G}(G) + \sum_{u \in V} \binom{\deg u}{2} \leq \text{cr}_{\varphi_G}(G) + nd_{\max}(d_{\max} - 1),$$

and

$$\text{OPT}_{\text{cr}}(\tilde{G}) \leq \text{OPT}_{\text{cr}}(G) + nd_{\max}(d_{\max} - 1).$$

Our algorithm runs \mathcal{A} on \tilde{G} and finds a drawing $\varphi'_{\tilde{G}}$ of \tilde{G} with at most

$$\alpha(d_{\max}n + \text{OPT}_{\text{cr}}(\tilde{G})) \leq \alpha(d_{\max}^2n + \text{OPT}_{\text{cr}}(G))$$

crossings. We now show how to transform the resulting drawing $\varphi'_{\tilde{G}}$ of \tilde{G} into a drawing φ'_G of G . Informally, this is done by contracting the drawing of every path P_u into a point. More precisely, we draw every vertex $u \in V(G)$ at the point $\varphi'_{\tilde{G}}((u, e_1))$ (where e_1 is the first edge in the incidence list for u). For each path P_u , for $u \in V(G)$, we construct $(\deg u)$ auxiliary curves $\gamma_{u, e_1}, \dots, \gamma_{u, e_{\deg u}}$, where for each $i : 1 \leq i \leq \deg u$, curve γ_{u, e_i} connects the images of (u, e_1) and (u, e_i) in $\varphi'_{\tilde{G}}$, and no pair of such curves cross (though curves that correspond to different vertices of V are allowed to cross). This is done as follows. First, we draw each curve γ_{u, e_i} along the image of the path P_u in $\varphi'_{\tilde{G}}$, following the segment that connects the images of (u, e_1) and (u, e_i) ; in a neighborhood of each vertex (u, e_j) of P_u , we draw the curve γ_{u, e_i} on the side of P_u opposite to the side where the edge \tilde{e}_j enters (u, e_j) (thus γ_{u, e_i} does not cross the drawing of e_j near the point $\varphi'_{\tilde{G}}((u, e_j))$). We make sure that all curves γ_{u, e_i} are drawn in general position. Next, if any of the resulting curves cross themselves, or cross each other, we perform uncrossing. Since all these curves start at the same point – the image of (u, e_1) – we can uncross them so that the final curves do not cross each other, and do not cross themselves.

We are now ready to describe the drawing of every edge $e = (u, v) \in E(G)$. The drawing of e is a concatenation of three curves: $\gamma_{u, e}$, $\varphi'_{\tilde{G}}(\tilde{e})$, and $\gamma_{v, e}$. The second segment of this drawing is called a type-1 segment, while the first and the third segments are called type-2 segments. Note that it is possible that some pairs of edges have more than one crossing, adjacent edges cross each other and some edges have self crossings in this drawing; we will fix that later. We now bound the number of crossings $\text{cr}_{\varphi'_G}(G)$.

- The number of crossings between all pairs of type-1 segments is bounded by $\text{cr}_{\varphi'_{\tilde{G}}}(\tilde{G})$.
- The number of crossings between any pair of type-2 segments $\gamma_{u, e}$ and $\gamma_{v, e'}$ (where $u \neq v$), is bounded by the number of crossings between paths P_u and P_v . Since every crossing between the paths P_u and P_v may pay for crossings of at most d_{\max}^2 such pairs of curves, the total number of such crossings is at most $d_{\max}^2 \text{cr}_{\varphi'_{\tilde{G}}}(\tilde{G})$.
- Similarly, the number of crossings between a type-2 curve $\gamma_{u, e}$ and a type-1 curve $\varphi'_{\tilde{G}}(\tilde{e}')$ (for an arbitrary edge e' of G) is at most the number of crossings between P_u and \tilde{e}' in $\varphi'_{\tilde{G}}$. So the total number of such crossings is at most $d_{\max} \cdot \text{cr}_{\varphi'_{\tilde{G}}}(\tilde{G})$.

We conclude that the number of crossings is $O(d_{\max}^2 \text{cr}_{\varphi'_{\tilde{G}}}(\tilde{G})) \leq O(\alpha d_{\max}^4 (n + \text{OPT}_{\text{cr}}(G)))$. Finally, the algorithm uncrosses drawings of edges that cross more than once, crossing pairs of adjacent edges, and edges that cross themselves. During this step the number of crossings can only go down. \square

The algorithm of Even et al. [12] uses an algorithm for **Balanced Separator** as a subroutine. We need a few definitions. Suppose we are given a graph $G = (V, E)$ with non-negative vertex weights w . For each subset $A \subseteq V$ of vertices, let $w(A)$ denote the total weight of vertices in A . We say that a cut (S, \bar{S}) is b -balanced

w.r.t. the weights w , iff $w(S), w(\bar{S}) \geq bw(V)$. The cost of the cut is $|E(S, \bar{S})|$. Even et al. prove the following theorem.

Theorem 16 *Suppose that there is some function $\beta : \mathbb{Z} \rightarrow \mathbb{R}^+$, and an efficient algorithm for Balanced Separator, with the following property. Given any n -vertex, vertex-weighted graph $G = (V, E)$ and values $0 \leq b \leq 1/2$, C_b , such that every sub-graph of G has a b -balanced separator of size at most C_b , the algorithm returns a $1/3$ -balanced cut of G , whose cost is $O(\beta(n)C_b)$ (the constant in the O -notation may depend on b). Then there is an efficient algorithm to find a drawing of any bounded degree graph G with $O(\beta^2(n) \log n) \cdot (n + \text{OPT}_{\text{cr}}(G))$ crossings.*

Even et al. use the algorithm of Leighton and Rao [27] that gives an algorithm for balanced separators with approximation factor $\beta(n) = O(\log n)$. We note that the results of Arora, Rao and Vazirani [3] gives an improved algorithm, with $\beta(n) = O(\sqrt{\log n})$, and thus we can efficiently find a drawing of a bounded degree graph with at most $O(\log^2 n) \cdot (n + \text{OPT}_{\text{cr}}(G))$ crossings.

Theorem 17 (Arora et al. [3]) *For every constant $0 < b < 1/2$ and some $0 < b' < b$ (that depends on b), there is a bi-criteria approximation algorithm for the Balanced Cut Problem with the following approximation guarantee. Given a graph $G = (V, E)$ and a set of vertex weights w , the algorithm finds a b' -balanced cut w.r.t. w of cost at most $O(\sqrt{\log n} \cdot C_b)$, where C_b is the cost of the optimal b -balanced cut w.r.t. w . (The constant in the O -notation depends on b .)*

We point out that the algorithm in Theorem 17 does not directly satisfy the requirements of Theorem 16, since it finds a b' -balanced cut only for *some* $b' \in (0, b)$ that depends on b , while we are required to produce a $1/3$ -balanced cut. For completeness, we show that the algorithm of [3] can still be used to obtain an algorithm for balanced separator as required in the statement of Theorem 16, for $\beta(n) = O(\sqrt{\log n})$.

We iteratively apply the algorithm of Arora et al. We start with $S = V$ and $\bar{S} = \emptyset$. We first find a b' -balanced cut in $G[S] = G$ (where b' is the constant guaranteed by Theorem 17). If the larger side of the cut contains at most $2/3$ of the total weight, then the cut is $1/3$ -balanced and we are done. Otherwise, we let S be the larger side of the cut (w.r.t. weights w), and we add the smaller side of the cut to \bar{S} . Then we iteratively apply the ARV-algorithm to $G[S]$, update sets S and \bar{S} , and repeat. We stop when S contains at most $2/3$ of the total weight of G . Note that after each iteration the weight of S decreases by at least a factor $(1 - b')$, since the algorithm of [3] finds a b' -balanced cut. Therefore, the algorithm stops in at most $\lceil \log_{1-b'} 2/3 \rceil$ steps. Observe that after each iteration, $w(S) \geq w(V)/3$, since before each iteration $w(S) \geq 2w(V)/3$ and we let S to be the larger of the two sides of the cut. Hence, $w(V)/3 \leq w(S) \leq 2w(V)/3$ when the algorithm terminates. That is, the cut (S, \bar{S}) is $1/3$ -balanced.

In every iteration, we cut at most $O(\sqrt{\log n} \cdot C_b)$ edges, and the total number of iterations is at most $\lceil \log_{1-b'} 2/3 \rceil$. Thus the cost of the cut is $O(\sqrt{\log n} \cdot C_b)$.