Bundled Crossings Revisited

Steven Chaplick*[1], Thomas C. van Dijk†[1], Myroslav Kryven‡[1], Ji-won Park[2], Alexander Ravsky§[3], and Alexander Wolff¶[1]

1 **University of Würzburg, Würzburg, Germany**
`firstname.lastname@uni-wuerzburg.de`

2 **KAIST, Daejeon, Republic of Korea**
`wldnjs1727@kaist.ac.kr`

3 **Pidstryhach Institute for Applied Problems of Mechanics and Mathematics, Nat. Acad. Sciences of Ukraine, Lviv, Ukraine**
`alexander.ravsky@uni-wuerzburg.de`

Abstract

An effective way to reduce clutter in a graph drawing that has (many) crossings is to group edges into *bundles* when they travel in parallel. Each edge can participate in many such bundles. Any crossing in this bundled graph occurs between two bundles, i.e., as a *bundled crossing*. We minimize the number of bundled crossings in circular layouts, where vertices are placed on a circle and edges are routed inside the circle.

For a given graph the goal is to find a bundled drawing with at most k crossings. We show that the problem has an FPT algorithm (in k) when we require a simple circular layout.

1 Introduction

In traditional node–link diagrams, vertices are mapped to points in the plane and edges are usually drawn as straight-line segments connecting the vertices. For large and somewhat dense graphs, however, such layouts tend to be so cluttered that it is hard to see any structure in the data. For this reason, Holten [14] introduced *bundled drawings*, where edges that are close together and roughly go into the same direction are drawn using Bézier curves such that the grouping becomes visible. Due to the practical effectiveness of this approach, it has quickly been adopted by the information visualization community [9, 12, 15, 16, 19]. However, bundled drawings have only recently attracted study from a theoretical point of view. Nevertheless, in his survey on crossing minimization, Schaefer already listed bundled crossing minimization as an open problem [20, page 35].

Fink et al. [11] considered bundled crossings in the context of drawing metro maps. They suggested replacing the classical objective of crossing minimization [3, 13, 17] by what they called *block crossing minimization*. Given a set of x-monotone curves (the metro lines that go through two neighboring stations), a block crossing is the exchange of two adjacent blocks of curves. Fink et al. also introduced *monotone* block crossing minimization where each pair of lines can intersect at most once. They considered various network topologies: single edge, path, (upward) tree, planar graph, (bounded-degree) general graph.

Our research builds on recent work of Fink et al. [10] and Alam et al. [1] who extended the notion of block crossings from sets of x-monotone curves to general drawings of graphs.

* Supported by DFG grant WO 758/11-1.
† Supported by DFG grant DI 2161/2-1.
‡ Supported by DAAD.
§ Supported by Erasmus+.
¶ Supported by DFG grant WO 758/9-1.

35th European Workshop on Computational Geometry, Utrecht, The Netherlands, March 18–20, 2019.
This is an extended abstract of a presentation given at EuroCG'19. It has been made public for the benefit of the community and should be considered a preprint rather than a formally reviewed paper. Thus, this work is expected to appear eventually in more final form at a conference with formal proceedings and/or in a journal.

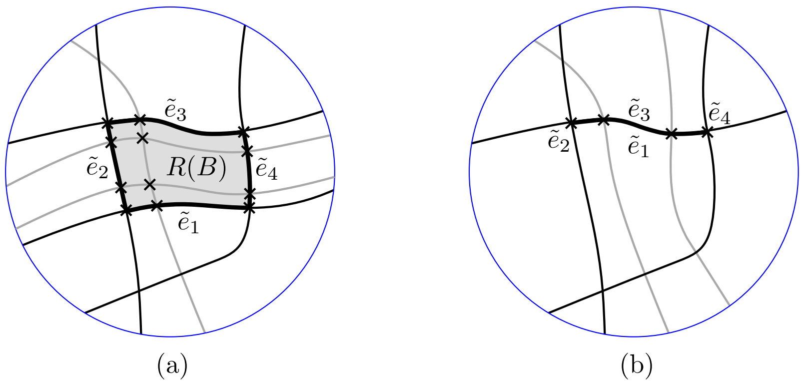


Figure 1 (a) A non-degenerate bundled crossing B and (b) a degenerate bundled crossing B' where one bundle consists of just one edge piece $\tilde{e}_1$ ($\tilde{e}_1 = \tilde{e}_3 = R(B')$) with endpoints $\tilde{e}_2$ and $\tilde{e}_4$.

It is common to define a drawing of a graph as a function that maps vertices to points in the plane and edges to Jordan arcs that connect the corresponding points. Here we will consider simple drawings, that is, any two edges intersect at most once and no edge self-intersects. We will often identify vertices with their points and edges with their curves.

Let D be a drawing, and let $I(D)$ be the set of intersection points among the edges in D. We say that a *bundling* of D is a partition of $I(D)$ into *bundled crossings*, where a set $B \subseteq I(D)$ is a bundled crossing if the following holds (see Fig. 1).

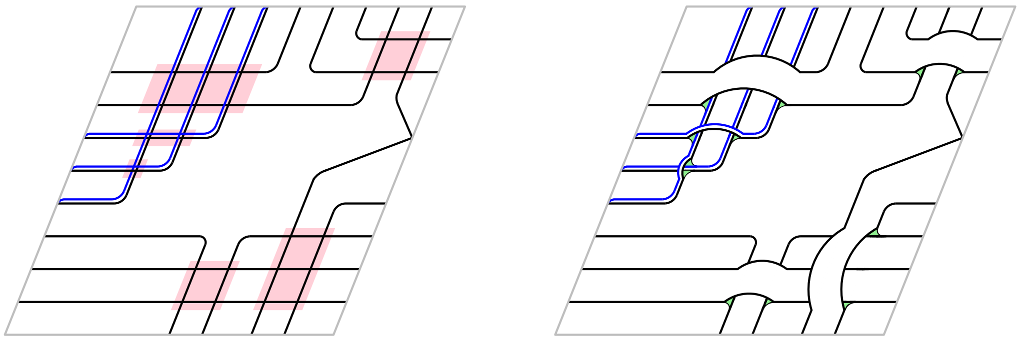
- B is contained in a closed Jordan region $R(B)$ whose boundary consists of four Jordan arcs $\tilde{e}_1$, $\tilde{e}_2$, $\tilde{e}_3$, and $\tilde{e}_4$ that are pieces of edges e_1, e_2, e_3, and e_4 in D.
- The pieces of the edges cut out by the region $R(B)$ can be partitioned into two sets $\tilde{E}_1$ and $\tilde{E}_2$ such that $\tilde{e}_1, \tilde{e}_3 \in \tilde{E}_1$, $\tilde{e}_2, \tilde{e}_4 \in \tilde{E}_2$, and each pair of edge pieces in $\tilde{E}_1 \times \tilde{E}_2$ has exactly one intersection point in $R(B)$, whereas no two edge pieces in $\tilde{E}_1$ (respectively $\tilde{E}_2$) have a common point in $R(B)$.

We call the sets of edges E_1 and E_2 corresponding to edge pieces $\tilde{E}_1$ and $\tilde{E}_2$ *bundles*. We call the edges that bound the two bundles of a bundled crossing *frame edges*. We say that a bundled crossing is *degenerate* if at least one of the bundles consists of only one edge piece; see Fig. 1(b). In this case, the region of the plane associated with the crossing coincides with that edge piece. In particular, any point in $I(D)$ by itself is a degenerate bundled crossing.

We consider circular layouts, where vertices are on a circle and edges are inside the circle. We denote by $\mathrm{bc}^\circ(G)$ the *circular bundled crossing numbers* of a graph G, i.e., the smallest number of bundled crossings over all bundlings of all simple circular layouts of G.

For computing $\mathrm{bc}^\circ(G)$, Alam et al. [1] gave an algorithm whose approximation factor depends on the density of the graph. They posed the existence of an FPT algorithm for $\mathrm{bc}^\circ(G)$ as an open question, which we answer in the affirmative. In this note, we first show how to decide whether $\mathrm{bc}^\circ(G) \leq 1$. Then, we generalize our result as follows.

▶ **Theorem 1.** *There is a computable function f such that for any n-vertex graph G and integer k, we can check, in $O(f(k)n)$ time, if $\mathrm{bc}^\circ(G) \leq k$, i.e., if G admits a circular layout with k bundled crossings. Within the same time bound, we can compute such a layout.*



(a)

(b)

Figure 2 (a) A bundled drawing D with six bundled crossings (pink); parallel (blue) edges can be inserted to avoid degenerate bundled crossings; (b) the corresponding surface of genus 6; the components of the surface which are not regions are marked in green

2 An FPT Algorithm for Simple Circular Layouts

Our algorithm is inspired by works on circular layouts with at most k crossings [2] and circular layouts where each edge is crossed at most k times [4]. Both first observed that the graphs admitting such circular layouts have treewidth $O(k)$, and then developed algorithms using Courcelle's theorem, which establishes that expressions in extended monadic second order logic can be evaluated efficiently. We define treewidth and MSO_2 in the full version [5].

▶ **Theorem 2** (Courcelle [7,8]). *For any integer $t \geq 0$ and any MSO_2 formula ψ of length ℓ, an algorithm can be constructed which takes a graph G with n vertices, m edges, and treewidth at most t and decides in $O(f(t,\ell) \cdot (n+m))$ time whether $G \models \psi$ where the function f from this time bound is a computable function of t and ℓ.*

We recall the observation of Alam et al. [1] that a drawing with k bundled crossings can be lifted onto a surface of genus k; see Fig. 2. Then we examine the structure of such a surface and present our algorithm for the case of one bundled crossing and finally for k bundled crossings.

2.1 Constructing the surface determined by a bundled drawing

Consider a bundled circular drawing D, i.e., it is drawn on a disk $\mathfrak{D}$ residing on a sphere, where the boundary of $\mathfrak{D}$ is the circle of D. Note that inserting parallel edges into the drawing (i.e., making our graph a multi-graph) can be done without modifying the bundled drawing, but allows us to assume that every bundled crossing has four distinct frame edges; see Fig. 2. Each bundled crossing B defines a Jordan curve C_B made up of the four Jordan arcs $\tilde{e}_1$, $\tilde{e}_2$, $\tilde{e}_3$, $\tilde{e}_4$ in clockwise order taken from its four frame edges $e_1, \dots, e_4$ respectively (here (e_1, e_3) and (e_2, e_4) frame the two bundles; $e_i = u_i v_i$). Let C'_B (see Fig. 3) denote a Jordan curve on $\mathfrak{D}$ outside of C_B where every point on C'_B lies at a sufficiently small distance $\epsilon > 0$ from C_B so that C'_B only contains the crossings in B and the distance from C'_B to the crossings outside of C'_B is at least $\frac{2}{3}$ of the distance from C_B to these crossings. Note that C'_B consists of eight Jordan arcs (in clockwise order) $c'_{2,1}, c'_{1,3}, c'_{3,2}, c'_{2,4}, c'_{4,3}, c'_{3,1}, c'_{1,4}, c'_{4,2}$, where $c'_{i,j}$ goes from e_i to e_j. The surface $\mathfrak{D}'$ is constructed by creating a flat handle on top of $\mathfrak{D}$ which connects $c'_{1,3}$ to $c'_{3,1}$ (when we lift the drawing onto this surface the bundle containing e_1 and e_3 will go over this handle), and doing so for each bundled crossing. We lift the drawing D

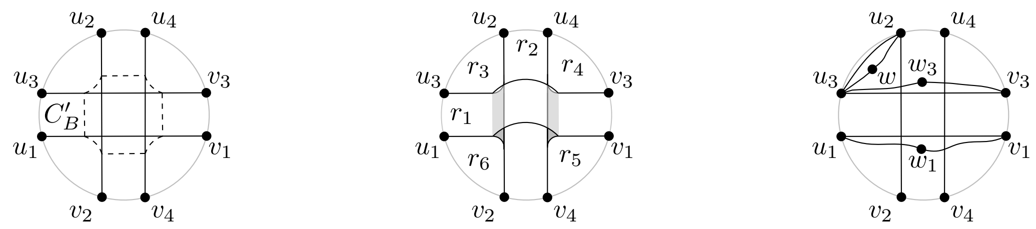


Figure 3 The curve C'_B; the regions $r_1, \ldots, r_6$; augmented graphs G'_{r_1} and G'_{r_3}

onto $\mathfrak{D}'$ obtaining the lifted drawing D'. Clearly, D' is crossing-free. Note that each Jordan curve C'_B remains on our original disk. We will now cut $\mathfrak{D}'$ into *components* (maximal connected subsets) using the frame edges and the Jordan curves C'_B for each B. Namely, for each bundled crossing B, we first cut $\mathfrak{D}'$ along each of the frame edges $e_1, \ldots, e_4$ of B. We additionally cut $\mathfrak{D}'$ along the four *corner* Jordan curves $c'_{2,1}, c'_{3,2}, c'_{4,3}$, and $c'_{1,4}$ of C'_B. This results in a subdivision of $\mathfrak{D}'$ which we call $\mathfrak{S}$. Here, we also use $D_{\mathfrak{S}}$ to denote the sub-drawing of D' on $\mathfrak{S}$, i.e., $D_{\mathfrak{S}}$ is missing the frame edges since these have been cut out. Let us now consider the components of $\mathfrak{S}$. Notice that every edge of $D_{\mathfrak{S}}$ is contained in one component of $\mathfrak{S}$. Furthermore, in order for a component $\mathfrak{s}$ of $\mathfrak{S}$ to contain an edge of $D_{\mathfrak{S}}$, $\mathfrak{s}$ must have two endpoints on its boundary—to be precise, we consider the boundary of $\mathfrak{s}$ in $\mathfrak{D}'$ whenever we think of the boundary of such a component of $\mathfrak{S}$. With this in mind, we focus on each component of $\mathfrak{S}$ with a vertex of G on its boundary and call it a *region.* Observe that a crossing in D which does not involve a frame edge corresponds, in $D_{\mathfrak{S}}$, to a pair of edges where one goes over a handle and the other goes underneath.

2.2 Recognizing a graph with k bundled crossings

Consider a bundled circular drawing D of G consisting of one bundled crossing. The bundled crossing consists of two bundles, so we have up to four frame edges, whose set will be denoted by $\mathcal{F}$. By $V(\mathcal{F})$, we denote the set of vertices incident to frame edges. Via the construction above, we obtain the subdivided surface $\mathfrak{S}$; see Fig. 3. Let r_1 and r_2 be the regions each bounded by the pair of frame edges corresponding to one of the bundles, and let $r_3, \ldots, r_6$ be the regions each bounded by one edge from one pair and one from the other pair; see Fig. 3. These are all the regions of $\mathfrak{S}$. Since, as mentioned before, each of the non-frame edges of G (i.e., each $e \in E(G) \setminus \mathcal{F}$) along with two endpoints are contained in exactly one of these regions, each component of $G \setminus V(\mathcal{F})$ including the edges connecting it to vertices of $V(\mathcal{F})$ is drawn in $D_{\mathfrak{S}}$ in some region of $\mathfrak{S}$. In this sense, for each region r of $\mathfrak{S}$, we use G_r to denote the subgraph of G induced by the components of $G \setminus V(\mathcal{F})$ contained in r in $D_{\mathfrak{S}}$ including the edges connecting them to elements of $V(\mathcal{F})$. Additionally, each vertex of G is incident to an edge in $\mathcal{F}$ (in which case it is on the boundary of at least two regions) or it is on the boundary of exactly one region.

Notice that there are two types of regions: $\{r_1, r_2\}$ and $\{r_3, r_4, r_5, r_6\}$. Consider a region of the first type, for example r_1, and note that it is a topological disk[1], i.e., G_{r_1} is outerplanar. Moreover, it has a special drawing where the two frame edges e_1 and e_3 bounding the region r_1 are on the outerface. Now, consider adding a new vertex w_j, for $j = 1, 3$ adjacent to both u_j and v_j so that w_j is placed slightly outside of the region; see Fig. 3. Denote the resulting augmented graph by $G^*_{r_1}$ and the corresponding drawing by

[1] We slightly abuse this notion to also mean a simply connected set.

$D^*_{r_1}$ – it is easy to see that $D^*_{r_1}$ is outerplanar. Moreover, in every outerplanar embedding of $G^*_{r_1}$, the vertices u_j, w_j, v_j, $j = 1, 3$, occur consecutively on the outerface.

Similarly for a region of the second type, for example r_3, the graph G_{r_3} is outerplanar also with a special drawing where all the vertices must be on the arc u_3u_2 of the disk subtended by the two frame edges e_3 and e_2 bounding the region r_3. We construct the augmented graph $G^*_{r_3}$ by adding to G_{r_3} an edge u_3u_2 and adding a vertex w adjacent to both u_3 and u_2. Again, $G^*_{r_3}$ is outerplanar as r_3 is a topological disk. Moreover, in every outerplanar embedding of $G^*_{r_3}$, the vertices u_3, w, u_2 occur consecutively on the outerface.

In other words, G_{r_i} "fits" into r_i because its augmented graph $G^*_{r_i}$ is outerplanar ($\star$) – note: that we do not require the specific outerplanar embedding of G_{r_i} for this augmentation.

To sum up, G has a circular drawing D with at most one bundled crossing, because there exist (i) a set of $\beta \leq 4$ frame edges $\mathcal{F} = \{e_1, e_2, \ldots, e_\beta\}$, (ii) a particular circular drawing $D_\mathcal{F}$ of frame edges, (iii) the drawing of the one bundled crossing B, and (iv) corresponding regions $r_1, \ldots, r_\gamma (\gamma \leq 6)$ of the subdivided surface $\mathfrak{S}$ such that the following properties hold:

1. The set of edges $E(G)$ is partitioned into $E_0, E_1, \ldots, E_\gamma$.
2. There is a bijection from E_0 to $\mathcal{F}$ so that the subgraph of G formed by E_0 is isomorphic to the graph formed by $\mathcal{F}$.
3. No vertex in $V(G) \setminus V(E_0)$ has incident edges $e \in E_i$, $e' \in E_j$ for $i \neq j$.
4. For each $v \in V(E_0)$, and each edge e incident to v, exactly one of the following is true: (i) $e \in E_0$ or (ii) $e \in E_i$ and v is on the boundary of r_i.
5. For each region r_i, let G_i be the graph formed by E_i and vertices in $V(E_0)$ on the boundary of r_i (even if they are not incident to an edge in E_i), and let G^*_i be the corresponding augmented graph (i.e., as in $\star$ above). Then, G^*_i must be outerplanar.

To test for a drawing with one bundled crossing, we first enumerate drawings $D_\mathcal{F}$ of up to four lines in the circle. For each drawing $D_\mathcal{F}$ that is valid for frame edges of one bundled crossing, we define our surface and its regions (which will allow the augmentation to be well-defined). Then, we will build an MSO_2 formula to express Properties 1–5 above. We have intentionally phrased these properties in a logical way so that it is clear that they are expressible in MSO_2. The only condition which is not obviously expressible is the outerplanarity check. For this, we recall that outerplanarity is characterized by two forbidden minors (i.e., K_4 and $K_{2,3}$) [6] and that, for every fixed graph H, there is an MSO_2 formula $\textsc{minor}_H$ so that for all graphs G, $G \models \textsc{minor}_H$ if and only if G contains H as a minor [8, Corollary 1.15]. Thus, Properties 1–5 can be expressed as an MSO_2 formula ψ and, by Courcelle's theorem, there is a computable function f such that we can test if $G \models \psi$ in $O(f(\psi, t)n)$ time for input graphs of treewidth at most t. Since outerplanar graphs have treewidth at most two [18], the region graphs are outerplanar, and adding the (up to) 8 frame vertices raises the treewidth by at most 8, G must also have treewidth at most 10.

The key ingredient above was that every region was a topological disk. However, in a subdivision $\mathfrak{S}$ constructed from a bundled drawing with k bundled crossings this is not trivial as regions can go over and under many handles; see Fig. 2. We state this result in the following lemma, whose proof can be found in the full version [5].

▶ **Lemma 3.** *Each region r of $\mathfrak{S}$ is a topological disk.*

Properties 1–5 and Lemma 3 allow us to construct an MSO_2 formula φ to test whether $\mathrm{bc}^\circ(G) \leq k$ with pattern $\mathcal{F}$ of frame edges. The size of φ depends only on k. This lemma further implies that the treewidth of a graph G with $\mathrm{bc}^\circ(G) \leq k$ is at most $8k + 2$ since deleting a vertex from a graph lowers its treewidth by at most one and the treewidth of an outerplanar graph is at most two [18]. So, applying Courcelle's Theorem (2) on φ of each pattern $\mathcal{F}$ leads to an FPT algorithm to test whether $\mathrm{bc}^\circ(G) \leq k$. This proves Theorem 1.

References

1. M. Alam, M. Fink, and S. Pupyrev. The bundled crossing number. In Y. Hu and M. Nöllenburg, editors, *GD*, volume 9801 of *LNCS*, pages 399–412. Springer, 2016. URL: `http://arxiv.org/abs/1608.08161`, `doi:10.1007/978-3-319-50106-2_31`.
2. M. J. Bannister and D. Eppstein. Crossing minimization for 1-page and 2-page drawings of graphs with bounded treewidth. *Journal of Graph Algorithms and Applications*, 22(4):577–606, 2018. `doi:10.7155/jgaa.00479`.
3. C. Buchheim, D. Ebner, M. Jünger, G. W. Klau, P. Mutzel, and R. Weiskircher. Exact crossing minimization. In P. Healy and N. S. Nikolov, editors, *GD*, volume 3843 of *LNCS*, pages 37–48. Springer, 2006. `doi:10.1007/11618058_4`.
4. S. Chaplick, M. Kryven, G. Liotta, A. Löffler, and A. Wolff. Beyond outerplanarity. In F. Frati and K.-L. Ma, editors, *GD*, volume 10692 of *LNCS*, pages 546–559. Springer, 2018. URL: `http://arxiv.org/abs/1708.08723`, `doi:10.1007/978-3-319-73915-1_42`.
5. S. Chaplick, T. C. van Dijk, M. Kryven, J. won Park, A. Ravsky, and A. Wolff. Bundled crossings revisited. Arxiv report, 2018. URL: `http://arxiv.org/abs/1812.04263`.
6. G. Chartrand and F. Harary. Planar permutation graphs. *Annales de l'I.H.P. Probabilités et statistiques*, 3(4):433–438, 1967. URL: `http://eudml.org/doc/76875`.
7. B. Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Inform. Comput.*, 85(1):12–75, 1990. `doi:10.1016/0890-5401(90)90043-H`.
8. B. Courcelle and J. Engelfriet. *Graph Structure and Monadic Second-Order Logic: A Language-Theoretic Approach.* Cambridge Univ. Press, 2012.
9. W. Cui, H. Zhou, H. Qu, P. C. Wong, and X. Li. Geometry-based edge clustering for graph visualization. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1277–1284, 2008. `doi:10.1109/TVCG.2008.135`.
10. M. Fink, J. Hershberger, S. Suri, and K. Verbeek. Bundled crossings in embedded graphs. In E. Kranakis, G. Navarro, and E. Chávez, editors, *LATIN*, volume 9644 of *LNCS*, pages 454–468. Springer, 2016. `doi:10.1007/978-3-662-49529-2_34`.
11. M. Fink, S. Pupyrev, and A. Wolff. Ordering metro lines by block crossings. *J. Graph Algorithms Appl.*, 19(1):111–153, 2015. `doi:10.7155/jgaa.00351`.
12. E. R. Gansner, Y. Hu, S. North, and C. Scheidegger. Multilevel agglomerative edge bundling for visualizing large graphs. In G. D. Battista, J.-D. Fekete, and H. Qu, editors, *PACIFICVIS*, pages 187–194. IEEE, 2011. `doi:10.1109/PACIFICVIS.2011.5742389`.
13. M. R. Garey and D. Johnson. Crossing number is NP-complete. *SIAM J. Algebr. Discrete Meth.*, 4:312–316, 1983. `doi:10.1137/0604033`.
14. D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Trans. Vis. Comput. Graph.*, 12(5):741–748, 2006. `doi:10.1109/TVCG.2006.147`.
15. C. Hurter, O. Ersoy, S. I. Fabrikant, T. R. Klein, and A. C. Telea. Bundled visualization of dynamicgraph and trail data. *IEEE Trans. Vis. Comput. Graphics*, 20(8):1141–1157, 2014. `doi:10.1109/TVCG.2013.246`.
16. C. Hurter, O. Ersoy, and A. Telea. Graph bundling by kernel density estimation. *Comput. Graph. Forum*, 31:865–874, 2012. `doi:10.1111/j.1467-8659.2012.03079.x`.
17. K. Kawarabayashi and B. Reed. Computing crossing number in linear time. In *STOC*, pages 382–390. ACM, 2007. `doi:10.1145/1250790.1250848`.
18. S. L. Mitchell. Linear algorithms to recognize outerplanar and maximal outerplanar graphs. *Inform. Process. Lett.*, 9(5):229–232, 1979. `doi:10.1016/0020-0190(79)90075-9`.

19 S. Pupyrev, L. Nachmanson, S. Bereg, and A. E. Holroyd. Edge routing with ordered bundles. In M. van Kreveld and B. Speckmann, editors, *Proc. 19th Int. Symp. Graph Drawing (GD'11)*, volume 7034 of *LNCS*, pages 136–147. Springer, 2012. doi:10.1007/978-3-642-25878-7_14.

20 M. Schaefer. The graph crossing number and its variants: A survey. *Electr. J. Combin.*, Dynamic Survey DS21, 2017. URL: http://www.combinatorics.org/ojs/index.php/eljc/article/view/DS21.