



Balanced Schnyder Woods for Planar Triangulations: An Experimental Study with Applications to Graph Drawing and Graph Separators

Luca Castelli Aleardi^(✉) 

LIX, Ecole Polytechnique, Institut Polytechnique de Paris, Palaiseau, France
amturing@lix.polytechnique.fr

Abstract. In this work we consider balanced Schnyder woods for planar graphs, which are Schnyder woods where the number of incoming edges of each color at each vertex is balanced as much as possible. We provide a simple linear-time heuristic leading to obtain well balanced Schnyder woods in practice. As test applications we consider two important algorithmic problems: the computation of Schnyder drawings and of small cycle separators. While not being able to provide theoretical guarantees, our experimental results (on a wide collection of planar graphs) suggest that the use of balanced Schnyder woods leads to an improvement of the quality of the layout of Schnyder drawings, and provides an efficient tool for computing short and balanced cycle separators.

1 Introduction

Schnyder woods [27] and its generalizations are a deep tool for dealing with the combinatorics of planar [13] and surface maps [10, 11, 18]. They lead to efficient algorithmic and combinatorial solutions for a broad collection of problems, arising in several domains, from enumerative combinatorics to graph drawing and computational geometry. For instance, the use of Schnyder woods has led to linear-time algorithms for grid drawing [3, 18, 27], to the optimal encoding and uniform sampling of planar maps [26], to the design of compact data structures [9] and to deal with geometric spanners [5]. Schnyder woods lead to fast implementations (also integrated in open source libraries [25]) and provide strong tools for establishing rigorous theoretical guarantees that hold in the worst case, even for irregular, random or pathological cases. The main idea motivating this work is that, in practice, most real-world graphs exhibit strong regularities which make them far from the random and pathological cases. Based on this remark, many geometry processing algorithms try to exploit this regularity in order to obtain better results in practice. For instance, when applied to regular graphs, many mesh compression schemes [19] achieve good compression rates, well below

This work is supported by the French ANR GATO (ANR-16-CE40-0009-01).

© Springer Nature Switzerland AG 2019

D. Archambault and C. D. Tóth (Eds.): GD 2019, LNCS 11904, pp. 114–121, 2019.

https://doi.org/10.1007/978-3-030-35802-0_9

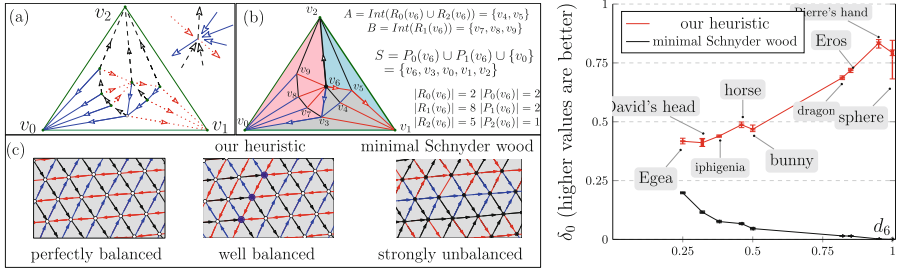


Fig. 1. (a) A planar triangulation endowed with a Schnyder wood. (b) a separator (A, B, S) obtained from the Schnyder wood. (c) three Schnyder woods of the same portion of a spherical grid: our heuristic leads to a majority of balanced vertices (white circles), while the minimal Schnyder wood is strongly unbalanced. (Right chart) Evaluation of the balance of Schnyder woods (tests are repeated with 500 random seeds) (Color figure online).

the worst-case optimal bound guaranteed by [26]. As far as we know, the problem of providing an adaptive analysis of Schnyder woods taking into account the graph regularity has not been investigated so far. This work provides empirical evidence about the fact that balanced Schnyder woods can lead to fast solutions achieving good results in practice, especially for real-world graphs. As test applications, we evaluate the layout quality of a Schnyder drawing depending on the balance of the underlying Schnyder wood, and we consider the problem of computing small separators for planar graphs, which has been extensively investigated [22–24, 28], due to its relevance for many graph algorithms.

Preliminaries and Related Works. In this work we deal with *planar triangulations*, which are genus 0 simple maps where every face is triangulated (we will denote by n the number of vertices and by m the number of edges). Given a planar triangulation with a distinguished root (outer) face (v_0, v_1, v_2) , a Schnyder wood [27] is defined as a coloring (with colors 0, 1 or 2) and orientation of the inner edges such that each inner vertex has exactly one outgoing incident edge for each color, and the remaining incident edges must satisfy the local Schnyder rule (see Fig. 1(a)). A given rooted triangulation may admit many Schnyder woods [1, 2, 14]: among them, the *minimal* one (without ccw oriented triangles) plays a fundamental role [9, 26]. Here we focus on balanced Schnyder woods, for which the ingoing edges are evenly distributed around inner vertices. A related problem concerns the computation of egalitarian orientations: unfortunately the results in [6] only apply to unconstrained orientations. Schnyder woods have led to a linear-time algorithm providing an elegant solution to the grid drawing problem (solved independently also in [17]): in its pioneristic work [27] Schnyder showed that a planar graph with n vertices admits a straight-line drawing on a grid of size $O(n) \times O(n)$. Schnyder drawings have a number of nice properties that make them useful for addressing problems [4, 5, 12] involving planar graphs in several distinct domains. While recent works [21] provide a probabilistic study

of the converge for uniformly sampled triangulations endowed with a Schnyder wood, as far as we know there are no theoretical or empirical evaluations of the quality of Schnyder drawings for regular graphs. Given a graph G we consider *small separators* which are defined by a partition (A, B, S) of all vertices such that S is a separating vertex set of small size (usually $|S| = O(\sqrt{m})$), and the remaining vertices in $G \setminus S$ belong to a balanced partition (A, B) satisfying $|A| \leq \alpha n$, $|B| \leq \alpha n$ (usually, for planar graphs, the *balance ratio* is $\alpha = \frac{2}{3}$). Here we focus on *simple cycle separators* [24], for which fast implementations [16, 20] have been recently proposed (some of them [16] are provided with a worst-case bound of $\sqrt{8m}$ on the cycle size).

2 Contribution

2.1 Balanced Schnyder Woods

Our first step is to measure the balance of a Schnyder wood: given an inner vertex v of degree $\deg(v)$ having $\text{indeg}_i(v)$ incoming edges of color i (for $i \in \{0, 1, 2\}$), we define its *defect* as $\delta(v) = \max_i \text{indeg}_i(v) - \min_i \text{indeg}_i(v)$ if $\deg(v)$ is a multiple of 3, and $\delta(v) = \max_i \text{indeg}_i(v) - \min_i \text{indeg}_i(v) - 1$ otherwise. We say that a vertex is *balanced* if $\delta(v) = 0$ and a Schnyder wood is *well balanced* if a majority of vertices have a small defect. For regular graphs is possible, in principle, to get a Schnyder wood that is perfectly balanced ($\delta(v) = 0$ everywhere) as shown in Fig. 1(c). In practice many Schnyder woods are unbalanced and we are not aware of existing theoretical or empirical results on the balance of Schnyder woods.

An Heuristic for Well Balanced Schnyder Woods. We make use of the well known incremental vertex shelling procedure [7] that computes a Schnyder wood with a sequence of vertex removals. This procedure has many degrees of freedom: at each step the choice of the vertex to be removed can possibly lead to a different Schnyder wood. In order to get as much as possible balanced vertices, we retard the removal of some vertices according to a balance priority, defined as the total number of ingoing edges incident to a vertex during the shelling procedure. The balance can be further improved by performing the reversal of oriented triangles in a post-processing¹ step. We refer to [8] for more details.

2.2 From Schnyder Drawings to Small Simple Cycle Separators

Schnyder woods provides a very fast procedure for partitioning, given an arbitrary inner vertex v , the set of inner faces of a triangulation into three sets $R_0(v)$, $R_1(v)$ and $R_2(v)$ (respectively blue, red and gray triangles in Fig. 1(b)), whose boundaries consist of the three disjoint paths $P_0(v)$, $P_1(v)$ and $P_2(v)$ emanating from v . The computation of simple cycle separators can be done as follows: for each vertex v check whether the two sets $A = \text{Int}(R_i(v) \cup R_{i+1}(v))$ and $B = \text{Int}(R_{i+2}(v))$ satisfy the prescribed balance ratio for at least one index

¹ The results presented in Sect. 2.3 are obtained without post-processing step.

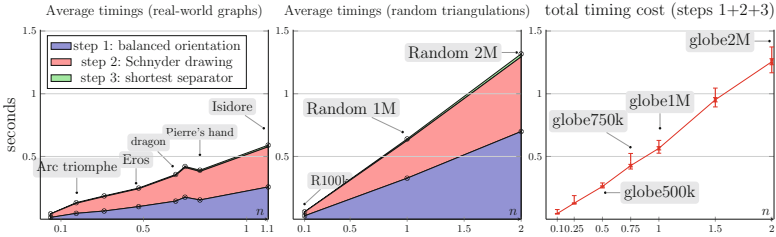


Fig. 2. Evaluation of timing costs over 100 executions (allocating 1 GB of RAM for the JVM): timings are expressed as a function of the size (millions of vertices).

$i \in \{0, 1, 2\}$ (indices are modulo 3, and $Int(R)$ denotes the set of inner vertices of a region R): then select the vertex for which the corresponding cycle length $|P_i(v)| + |P_{i+1}(v)| + 1$ is minimal. All this steps can be performed almost instantaneously, since all the quantities above are encoded in the Schnyder drawing itself (see [27] for more details). As far as we know there are no theoretical guarantees on both the partition balance and boundary size: as observed in practice, most vertices lead to unbalanced partitions whose boundary size can be very large.

2.3 Experimental Results

Datasets and Experimental Setting. We run our experimental evaluations² on a broad variety of graphs³, including real-world meshes used in geometry processing (made available by the `aim@shape` and `Thing10k` repositories), synthetic regular graphs with different shapes (`sphere`, `cylinder`, ...), random planar triangulations (generated with uniform random sampling [26]), and Delaunay triangulations of random points. As done in geometric modeling, we use the proportion of degree 6 vertices, denoted by d_6 , to measure the regularity of a graph: d_6 is close to 1 for regular meshes, while is usually below 0.3 for irregular and random graphs. To evaluate the balance of a Schnyder woods we use the proportion of balanced vertices, denoted by δ_0 , and the average defect computed on all vertices, denoted by δ_{avg} . As for previous works [16, 20], the results (e.g. the size of the separator) can depend on the choice of the initial seed (the root face in our case). We perform tests with hundreds of random seeds: for each choice of the seed, we adopt whisker plots to show the entire range of computed values, while each box represents the middle 50% of values (as in Figs. 1 and 4).

Balance of Schnyder Woods. To evaluate the balance quality of the Schnyder woods we plot the value δ_0 as a function of d_6 : our balanced Schnyder woods are compared to minimal ones in Fig. 1. Experimental results strongly suggests that our heuristic leads to well balanced Schnyder woods. Our heuristic performs

² Our datasets and code can be found at <http://www.lix.polytechnique.fr/~amturning/software.html>.

³ Previous works [16, 20] triangulate the input graph in a preprocessing phase.

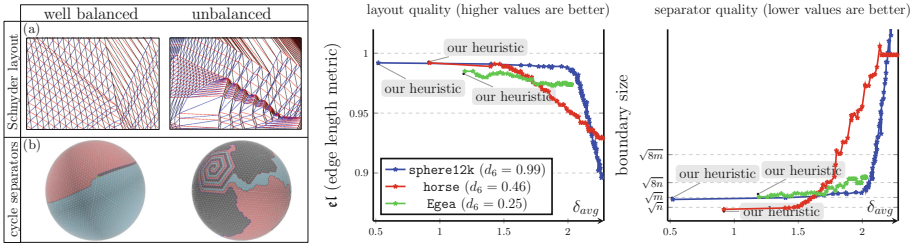


Fig. 3. For a fixed initial seed, we generate a sequence of Schnyder woods by starting from a well balanced Schnyder wood (computed with our heuristic) and by randomly reversing ccw oriented triangles. In the charts we plot the layout and separator quality as functions of the average defect δ_{avg} of the corresponding Schnyder wood.

particularly well for regular graphs, for which a large majority of vertices are balanced (79% in average for the **sphere** graph). The results are good also for irregular graphs (**egea**), where about 45% of vertices are balanced. Also observe that the choice of the initial seed has a limited effect on the balance of the resulting Schnyder wood. Minimal Schnyder woods represent a bad case, especially for regular graphs: most vertices have a large defect and the resulting paths $P_0(v)$, $P_1(v)$ and $P_2(v)$ resemble very long spirals.

Runtime Performances. The algorithmic solutions relying on Schnyder woods are simple to implement and extremely fast. As observed in practice (see Fig. 2), our Java implementation allows processing between 1.43M and 1.92M vertices per second: we run our tests on an EliteBook with a core i7-5600U 2.60 GHz (with Ubuntu 16.04 and 1 GB of RAM allocated for the JVM). This has to be compared to the C implementations of previous results on cycle separators [16, 20], running on an Intel Xeon X5650 2.67 GHz (with 48.4 GB of RAM): the fastest variant of the procedures tested in [16] allows processing between 0.54M and 0.62M vertices per second for the case of square **grids**. Our timing costs are little affected by the choice of the initial seed and the structural properties of the graph. Observe that once the Schnyder drawing is given, the extraction of the cycle separator is instantaneous (0.01 s for a 1M vertices graph).

Layout Quality. A qualitative evaluation of the graph layouts based on the balance Schnyder woods is provided by the pictures in Fig. 3(a) showing two portions of the Schnyder drawings of a regular sphere graph. When starting from a our well balanced Schnyder woods the shape of triangles is much more balanced, and the resulting drawing partially captures the regularity of the grid. When starting from an unbalanced Schnyder wood the drawing exhibits many long edges and flat triangles, a typical drawback of Schnyder drawings. In order to provide a quantitative measure of the layout quality we consider the *edge lengths aesthetic metric* defined by $\epsilon_l = 1 - d_{el}$, where d_{el} is the average percent deviation of edge lengths: values close to 1 mean that most edges have the same length (see [15] for more details). For a fixed initial seed, we start from a

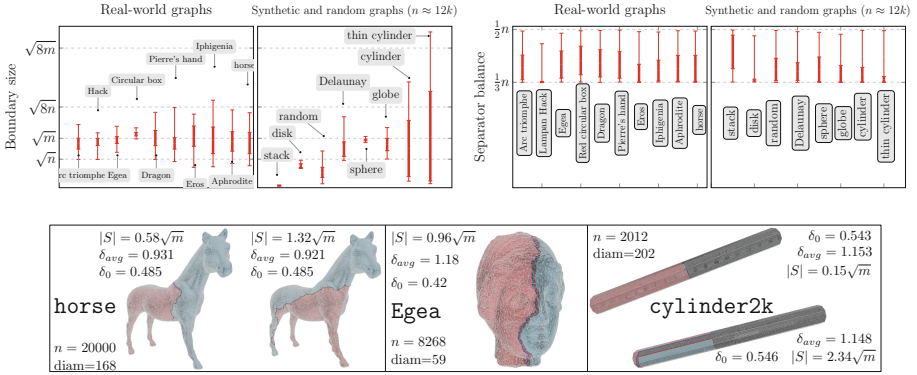


Fig. 4. We evaluate the quality of our simple cycle separators obtained from our balanced Schnyder woods (tests are repeated using 200 random seeds). The left charts report the boundary sizes, while the right charts show the plots of the separator balance (the normalized size of the smallest of the two sets A and B). The graphs are listed from left to right according to the increasing values of their relative diameter.

balanced Schnyder wood obtained with our heuristic and we randomly reverse ccw oriented triangles, obtaining a sequence of Schnyder woods which are more and more unbalanced. The middle chart in Fig. 3 reports the values of ϵl as a function of the average defect: the layout quality tends to deteriorate as soon as Schnyder woods get more unbalanced (high values of δ_{avg}).

Length and Balance of Separators. We look for separators with a balance ratio $\alpha = \frac{2}{3}$ that are *short*: the boundary size is at most $|S| \leq \sqrt{8m}$, as required in [16]. We plot in the charts of Fig. 4 the boundary sizes and partition balances of the separators obtained from a Schnyder drawing as described in Sect. 2.1. Our tests, repeated over several tens of graphs, confirm our intuition that balanced Schnyder woods lead to good separators for a large majority of classes of graphs. As for the layout quality, the separator size and balance strongly depend on the balance of the underlying Schnyder wood (right chart in Fig. 3). The boundary size of the separator is affected by the choice of the seed for graphs with large diameter: a good choice of the seed would prevent from getting too long cycles. For graphs with small diameter (e.g. random triangulations) Schnyder woods lead to very short separators, while the size is closed to \sqrt{m} for most real-world graphs. Our separators are often longer when compared with the results obtained in [16], but well below the prescribed bound of $\sqrt{8m}$.

References

1. Bernardi, O., Bonichon, N.: Catalan’s intervals and realizers of triangulations. *J. Comb. Theory Ser. A* **116**(1), 55–75 (2009). <https://hal.archives-ouvertes.fr/hal-00143870>(22 pages)

2. Bonichon, N.: A bijection between realizers of maximal plane graphs and pairs of non-crossing Dyck paths. *Discret. Math.* **298**, 104–114 (2005). <https://hal.archives-ouvertes.fr/hal-00307593>
3. Bonichon, N., Felsner, S., Mosbah, M.: Convex drawings of 3-connected plane graphs. *Algorithmica* **47**(4), 399–420 (2007). <https://doi.org/10.1007/s00453-006-0177-6>
4. Bonichon, N., Gavoille, C., Hanusse, N., Ilcinkas, D.: Connections between theta-graphs, delaunay triangulations, and orthogonal surfaces. In: Thilikos, D.M. (ed.) *WG 2010. LNCS*, vol. 6410, pp. 266–278. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16926-7_25
5. Bonichon, N., Gavoille, C., Hanusse, N., Perković, L.: Plane spanners of maximum degree six. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) *ICALP 2010. LNCS*, vol. 6198, pp. 19–30. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14165-2_3
6. Borradaile, G., Iglesias, J., Migler, T., Ochoa, A., Wilfong, G.T., Zhang, L.: Egalitarian graph orientations. *J. Graph Algorithms Appl.* **21**(4), 687–708 (2017). <https://doi.org/10.7155/jgaa.00435>
7. Brehm, E.: 3-orientations and Schnyder 3-tree-decompositions. Master’s thesis, FB Mathematik und Informatik, Freie Universität Berlin (2000)
8. Castelli Aleardi, L.: Balanced Schnyder woods for planar triangulations: an experimental study with applications to graph drawing and graph separators (2019). <https://arxiv.org/abs/1908.06688>
9. Castelli Aleardi, L., Devillers, O.: Array-based compact data structures for triangulations: practical solutions with theoretical guarantees. *JoCG* **9**(1), 247–289 (2018). <https://doi.org/10.20382/jocg.v9i1a8>
10. Castelli Aleardi, L., Fusy, É., Lewiner, T.: Schnyder woods for higher genus triangulated surfaces, with applications to encoding. *Discret. Comput. Geom.* **42**(3), 489–516 (2009). <https://hal.inria.fr/hal-00712046v1>
11. Despré, V., Gonçalves, D., Lévêque, B.: Encoding toroidal triangulations. *Discret. Comput. Geom.* **57**(3), 507–544 (2017). <https://doi.org/10.1007/s00454-016-9832-0>
12. Dhandapani, R.: Greedy drawings of triangulations. *Discret. Comput. Geom.* **43**(2), 375–392 (2010). <https://doi.org/10.1007/s00454-009-9235-6>
13. Felsner, S.: Lattice structures from planar graphs. *Electr. J. Comb.* **11**(1) (2004)
14. Felsner, S., Zickfeld, F.: On the number of planar orientations with prescribed degrees. *Electr. J. Comb.* **15**(1) (2008)
15. Fowler, J.J., Kobourov, S.G.: Planar preprocessing for spring embedders. In: 20th International Symposium Graph Drawing, pp. 388–399 (2012)
16. Fox-Epstein, E., Mozes, S., Phothilimthana, P.M., Sommer, C.: Short and simple cycle separators in planar graphs. *ACM J. Exp. Algorithm* **21**(1), 2:2:1–2:2:24 (2016)
17. de Fraysseix, H., Pach, J., Pollack, R.: How to draw a planar graph on a grid. *Combinatorica* **10**(1), 41–51 (1990)
18. Gonçalves, D., Lévêque, B.: Toroidal maps: Schnyder woods, orthogonal surfaces and straight-line representations. *Discret. Comput. Geom.* **51**(1), 67–131 (2014). <https://doi.org/10.1007/s00454-013-9552-7>
19. Gotsman, C.: On the optimality of valence-based connectivity coding. *Comput. Graph. Forum* **22**(1), 99–102 (2003). <https://doi.org/10.1111/1467-8659.t01-1-00649>

20. Holzer, M., Schulz, F., Wagner, D., Prasinou, G., Zaroliagis, C.D.: Engineering planar separator algorithms. *ACM J. Exp. Algorithm* **14** (2009). <https://doi.org/10.1145/1498698.1571635>
21. Li, Y., Sun, X., Watson, S.S.: Schnyder woods, $\text{sle}(16)$, and liouville quantum gravity. Technical report [arXiv:1705.03573v1](https://arxiv.org/abs/1705.03573v1) [math.PR], ArXiv, May 2016
22. Lipton, R.J., Tarjan, R.E.: A separator theorem for planar graphs. *SIAM J. Appl. Math.* **36**(2), 177–189 (1979)
23. Lipton, R.J., Tarjan, R.E.: Applications of a planar separator theorem. *SIAM J. Comput.* **9**(3), 615–627 (1980). <https://doi.org/10.1137/0209046>
24. Miller, G.L.: Finding small simple cycle separators for 2-connected planar graphs. *J. Comput. Syst. Sci.* **32**(3), 265–279 (1986). [https://doi.org/10.1016/0022-0000\(86\)90030-9](https://doi.org/10.1016/0022-0000(86)90030-9)
25. PIGALE, Public Implementation of a Graph Algorithm Library and Editor. <http://pigale.sourceforge.net/>
26. Poulalhon, D., Schaeffer, G.: Optimal coding and sampling of triangulations. *Algorithmica* **46**(3–4), 505–527 (2006)
27. Schnyder, W.: Embedding planar graphs on the grid. In: Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, vol. 90, pp. 138–148 (1990). <http://departamento.us.es/dma/leuita/PAIX/Referencias/schnyder.pdf>
28. Spielman, D.A., Teng, S.: Disk packings and planar separators. In: Proceedings of the Twelfth Annual Symposium on Computational Geometry, pp. 349–358 (1996). <https://doi.org/10.1145/237218.237404>