Efficient Algorithms for Ortho-Radial Graph Drawing*

Benjamin Niedermann[1], Ignaz Rutter[2], and Matthias Wolf[3]

1 **University of Bonn**
niedermann@uni-bonn.de
2 **TU Eindhoven**
i.rutter@tue.nl
3 **Karlsruhe Institute of Technology (KIT)**
matthias.wolf@kit.edu

Abstract

An ortho-radial drawing is an embedding of a graph into an ortho-radial grid, whose gridlines are concentric circles around the origin and straight-line spokes emanating from the origin but excluding the origin itself. Recently, Barth et al. [1] showed that such drawings can be described combinatorially by so-called *valid ortho-radial representations*, which only specify angles at vertices and bends on the edges, but neglects any kind of geometric information. A similar representation for embeddings in orthogonal grids is a central ingredient for bend minimization in this setting [5]. However, the result of Barth et al. [1] is existential and does not provide an efficient algorithm for testing whether a given ortho-radial representation is valid, let alone actually obtaining a drawing from a valid ortho-radial representation. In this paper, we provide efficient algorithms (with quadratic running time) for these two problems.

1 Introduction

Grid drawings of graphs embed graphs into grids such that vertices map to grid points and edges map to internally disjoint curves on the grid lines that connect their endpoints. Orthogonal grids, whose grid lines are horizontal and vertical lines, are popular and widely used in graph drawing. Ortho-radial drawings are a generalization of this concept to grids that are formed by concentric circles and straight-line spokes from the center but excluding the center. Among other applications, they are used to visualize network maps; see Fig. 1. Equivalently, they can be viewed as graphs drawn in an orthogonal fashion on the surface of a standing cylinder; see Fig. 2. We will use these different points of view interchangeably.

The main objective in orthogonal graph drawing is to minimize the number of bends on the curves. The core of a large fraction of the algorithmic work on this problem is the Topology-Shape-Metrics framework (TSM) introduced by Tamassia [5], which shows that orthogonal drawings can be described purely combinatorially by giving an *orthogonal representation*, which describes (i) the angles formed by consecutive edges around each vertex and (ii) the directions of bends along the edges. Such a representation is *valid* if (I) the angles around each vertex sum to 360°, and (II) the sum of the angles around each face with k vertices is $(2k-2)\cdot 180°$ for internal faces and $(2k+2)\cdot 180°$ for the outer face.

Recently Barth et al. [1] generalized orthogonal representations to ortho-radial drawings, called *ortho-radial* representations. Previously such representations were only known to exist for cycles and Θ-graphs [4] as well as cubic graphs with rectangular faces [3].

Let $G=(V,E)$ be a planar graph with an ortho-radial drawing Δ. We distinguish two types of simple cycles in G. If the center of the grid lies in the interior of a simple cycle, the

* This research was funded in part by Humility & Conviction in Public Life, a project of the University Connecticut sponsored by the John Templeton Foundation and the Helmholtz Program Storage and Cross-linked Infrastructures, Topic 6 Superconductivity, Networks and System Integration.

34th European Workshop on Computational Geometry, Berlin, Germany, March 21–23, 2018.
This is an extended abstract of a presentation given at EuroCG'18. 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.

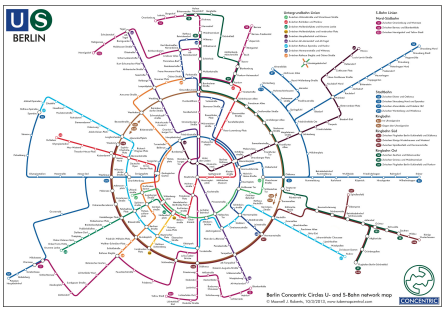
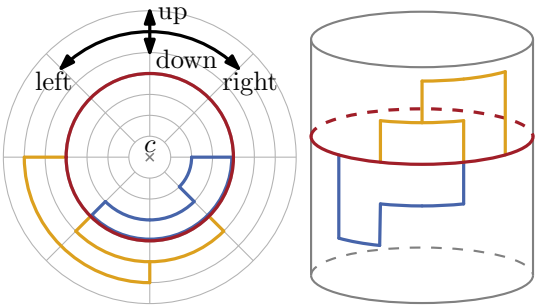


Figure 1 Metro map of Berlin using an ortho-radial layout. Image copyright by Maxwell J. Roberts. Reproduced with permission.



(a) Ortho-radial grid. **(b)** Cylinder drawing.

Figure 2 An ortho-radial drawing of a graph on a grid (a) and its equivalent interpretation as an orthogonal drawing on a cylinder (b).

cycle is *essential*; otherwise it is *non-essential*. Further, there is an unbounded face in Δ and a face that contains the center of the grid; we call the former the *outer face* and the latter the *central face*, which we mark by a small "x"; see Fig. 2a. All other faces are called *regular*.

Consider now an ortho-radial drawing Γ of G (as in [1] we restrict our attention to drawings without bends). An *ortho-radial representation* can be obtained from Γ by measuring for each incidence of a vertex v to a face f that lies to the right of the edges uv and vw the clockwise angle $a \in \{90°, 180°, 270°, 360°\}$. We call such a set of angles to vertex–face incidences an *angle assignment*. Given an angle assignment, for two edges uv and vw incident to the same vertex v, we define the *rotation* $\mathrm{rot}(uvw)$ as 1 if there is a right turn at v, 0 if uvw is straight, and -1 if there is a left turn at v. In the special case that $u = w$, we define $\mathrm{rot}(uvw) = -2$. The rotation of a path $P = v_1, \ldots, v_k$ is $\mathrm{rot}(P) = \sum_{i=1}^{k-1} \mathrm{rot}(v_{i-1}v_iv_{i+1})$ and the rotation of a cycle $C = v_1, \ldots, v_k, v_1$ is $\mathrm{rot}(C) = \sum_{i=1}^{k} \mathrm{rot}(v_{i-1}v_iv_{i+1})$, where we take $v_0 = v_k$ and $v_{k+1} = v_1$. For a face f we use $\mathrm{rot}(f)$ to denote the rotation of the facial cycle that bounds f (oriented such that f lies on the right side of the cycle).

A representation obtained from an ortho-radial drawing satisfies the following conditions.

(I) The sum of all angles around each vertex is 360°.

(II) For each face f: $\mathrm{rot}(f) = \begin{cases} 4, & f \text{ is a regular face} \\ 0, & f \text{ is the outer or the central face but not both} \\ -4, & f \text{ is both the outer and the central face.} \end{cases}$

We call a representation satisfying these properties an *ortho-radial representation*. However, not every ortho-radial representation has a corresponding ortho-radial drawing [1]. Barth et al. show that this requires a third condition, which essentially states that each essential cycle that contains an edge that points upward (on the cylinder) also has to contain an edge that points downward (and vice versa). More formally, fix a horizontal edge $e^\star = st$ on the outer face that points to the right as *reference* edge.

For a simple, essential cycle C in G and a path P from the endpoint s of the reference edge $e^\star$ to a vertex v on C the *labeling* ℓ_C^P assigns to each edge e on C the label $\ell_C^P(e) = \mathrm{rot}(e^\star + P + C[v, e])$. Note that we use $+$ to denote concatenation of paths and $C[v, e]$ to denote the subpath of C starting at v and ending with the edge e. In this paper we always assume that P is *elementary*, i.e., P intersects C only at its endpoints. For these paths the labeling is independent of the actual choice of P [1]. We therefore drop the superscript P and write $\ell_C(e)$ for the labeling of an edge e on an essential cycle C. We call an essential cycle *monotone* if either all its labels are non-negative or all its labels are non-positive. A

monotone cycle is a *decreasing* cycle if it has at least one strictly positive label, and it is an *increasing* cycle if C has at least one strictly negative label. An ortho-radial representation is *valid* if it contains neither decreasing nor increasing cycles. The validity of an orthor-radial representation ensures that on each essential cycle with at least one non-zero label there is at least one edge pointing up and one pointing down. Barth et al. [1] prove the following theorem[1].

▶ **Proposition 1** (Theorem 5 in [2]). *A 4-plane graph admits a bend-free ortho-radial drawing if and only if it admits a valid ortho-radial representation.*

To that end, Barth et al. [2] prove the following results among others. Since we use them throughout this paper, we restate them for the convenience of the reader. Both assume ortho-radial representations that are not necessarily valid.

▶ **Proposition 2** (Lemma 12 in [2]). *Let C_1 and C_2 be two essential cycles and let $H = C_1 + C_2$ be the subgraph of G formed by these two cycles. For any common edge vw of C_1 and C_2 where v lies on the central face of H, the labels of vw are equal, i.e., $\ell_{C_1}(vw) = \ell_{C_2}(vw)$.*

▶ **Proposition 3** (Lemma 16 in [2]). *Let C and C' be two essential cycles that have at least one common vertex. If all edges on C are labeled with 0, C' is neither increasing nor decreasing.*

Proposition 2 is a useful tool for comparing the labels of two interwoven essential cycles. For example, if C_1 is decreasing, we can conclude for all edges of C_2 that also lie on C_1 and that are incident to the central face of H that they are non-negative. Proposition 3 is useful in the scenario where we have an essential cycle C with non-negative labels, and a decreasing cycle C' that shares a vertex with C. We can then conclude that C is also decreasing. In particular, these two propositions together imply that the central face of the graph H formed by two decreasing cycles is bounded by a decreasing cycle.

While it is not hard to test whether a given angle assignment is an ortho-radial representation, Barth et al. left open the problem of testing whether a given ortho-radial representation is valid. Moreover, following their proof of Proposition 1, which constructs from a valid ortho-radial representation a corresponding ortho-radial drawing, requires a quadratic number of such validity tests. In this paper, we show that such a validity test can indeed be implemented to run in quadratic time. Used naively, this would result in a quartic algorithm for constructing a drawing for a valid ortho-radial drawing. We further improve on this and reduce also that running time to quadratic. Thus, our main result is that, given an ortho-radial representation of a graph, we can test in quadratic time whether it is valid, and in the positive case we can produce a corresponding drawing, whereas in the negative case, we find a monotone cycle witnessing that the representation is not valid.

2 Finding Monotone Cycles

The two conditions for ortho-radial representations are local and checking them can easily be done in linear time. We therefore assume in this section that we are given a planar graph G with an ortho-radial representation Γ. The condition for validity however references all essential cycles of which there may be exponentially many. We present an algorithm that checks whether Γ contains a monotone cycle and computes such a cycle if one exists. The main difficulty is that the labels on a decreasing cycle C depend on an elementary path P

[1] In the following we refer to the full version [2] of [1], when citing lemmas and theorems.

from the reference edge to C. However, we know neither the path P nor the cycle C in advance, and choosing a specific cycle C may rule out certain paths P and vice versa.

We only describe how to search for decreasing cycles; increasing cycles can be found by searching for decreasing cycles in a suitably defined mirrored representation. A decreasing cycle C is *outermost* if it is not contained in the interior of any other decreasing cycle. Clearly, if Γ contains a decreasing cycle, then it also has an outermost one, and in fact it can be shown that this cycle is uniquely determined.

▶ **Lemma 4.** *If Γ contains a decreasing cycle, there is a unique outermost decreasing cycle.*

The core of our algorithm is an adapted left-first DFS. Given a directed edge e, it determines the outermost decreasing cycle C in Γ such that C contains e in the given direction and e has the smallest label among all edges on C, if such a cycle exists. By running this test for each directed edge of G as the start edge, we find a decreasing cycle if one exists.

Our algorithm is based on a DFS that visits each vertex at most once. A left-first search maintains for each visited vertex v a reference edge $\mathrm{ref}(v)$, the edge of the search tree via which v was visited, and whenever it has a choice which vertex to visit next, it picks the first outgoing edge in clockwise direction after the reference edge that leads to an unvisited vertex. In addition to that, we employ a filter that ignores certain outgoing edges during the search. To that end, we define for all outgoing edges e incident to a visited vertex v a *search label* $\tilde{\ell}(e)$ by setting $\tilde{\ell}(e) = \tilde{\ell}(\mathrm{ref}(v)) + \mathrm{rot}(\mathrm{ref}(v) + e)$ for each outgoing edge e of v. In our search we ignore edges with negative search labels. For a given directed edge $e = vw$ in a graph G with ortho-radial representation Γ, we initialize the search by setting $\mathrm{ref}(w) = vw$, $\tilde{\ell}(e) = 0$ and then start searching from w.

Let T denote the directed search tree with root w constructed by the DFS in this fashion. If T contains v, then this determines a *candidate cycle* C containing the edge vw. If C is a decreasing cycle, which we can easily check by determining an elementary path from the reference edge to C, we report it. Otherwise, we show that there is no outermost decreasing cycle C such that vw is contained in C and has the smallest label among all edges on C.

▶ **Lemma 5.** *Assume Γ contains a decreasing cycle. Let C be the outermost decreasing cycle of Γ and let vw be an edge on C with the minimum label, i.e., $\ell_C(vw) \leq \ell_C(e)$ for all edges e of C. Then the left-first DFS from vw finds C.*

Proof Sketch. To prove this we assume for a contradiction that the search does not find C. Then, starting from vw there is a first edge xy of C that is not followed by the search. This can be for one of two reasons. Either it was filtered out, since the search label $\tilde{\ell}(xy)$ is less then 0, or, at the point the edge xy is considered by the search vertex y has already been visited. The first case is easily excluded using the assumption that $\ell_C(vw)$ is minimal. In the second case, we find that our search contains a search path Q that at some point deviates from C and then reaches y, before it eventually backtracks and reaches xy.

Let P be the subpath of Q from the vertex b where Q first leaves C to the first vertex p that again lies on C; see Fig. 3. The subgraph $H = P + C$ that is formed by the decreasing cycle C and the path P is a Θ-graph consisting of the three internally vertex-disjoint paths $P[b, p]$, $C[b, p]$ and $\overline{C}[b, p]$ between b and p. Due to the left-first rule, the circular ordering of the edges around the vertex b is fixed, and C is essential. Consequently the interior of C is the central face. We denote the cycle bounding the outer face but in which the edges are directed such that the outer face lies locally to the left by C'. Depending on which of the two remaining faces is the outer face, the situation is as shown in Fig. 4. One can then show that in either case the cycle bounding the outer face is a decreasing cycle, thus contradicting the assumption that C is outermost. In the remainder we sketch the proof of the easier case.

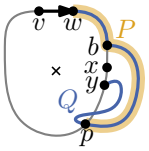


Figure 3 Path Q and its prefix P that leaves C once and ends at a vertex p of C.

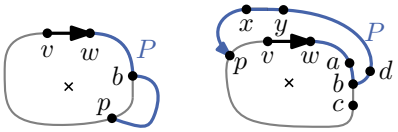


Figure 4 The two possible embeddings of the subgraph formed by the decreasing cycle C and the path P, which was found by the search.

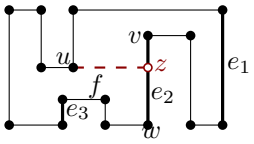


Figure 5 Candidate edges (bold) for a port u.

If $\overline{C'} = \overline{C}[b,p] + \overline{P}[p,b]$ forms the outer face of H, vw lies on C'; see Fig. 4, left. We show that C' is a decreasing cycle, which contradicts the assumption that C is the outermost decreasing cycle. Since P is simple and lies in the exterior of C, the path P is a part of C', which means $C'[w,p] = P$. The other part of C' is formed by $C[p,w]$. Since C forms the central face of H, the labels of the edges on $C[p,w]$ are the same for C and C' by Proposition 2. In particular, $\ell_C(vw) = \ell_{C'}(vw)$ and all the labels of edges on $C[p,w]$ are non-negative because C is decreasing. The label of any edge e on both C' and P is $\ell_{C'}(e) = \ell_{C'}(vw) + \mathrm{rot}(vw + P[w,e]) = \ell_C(vw) + \tilde{\ell}(e) \geq 0$. Thus, the labeling of C' is non-negative. Further, not all labels of C' are 0 since otherwise C is not a decreasing cycle by Proposition 3. Hence, C' is decreasing and contains C in its interior, a contradiction. ◀

The left-first DFS clearly runs in $O(n)$ time. We run it for each of the $O(n)$ directed edges of G. Since some edge must have the lowest label on the outermost decreasing cycle, Lemma 5 guarantees that we eventually find a decreasing cycle if one exists. Decreasing cycles can be detected symmetrically.

▶ **Theorem 6.** *Let G be a 4-planar graph on n vertices and let Γ be an ortho-radial representation of G. It can be determined in $O(n^2)$ time whether Γ is valid.*

3 Rectangulation

The core of the algorithm for drawing a valid ortho-radial representation Γ of a graph G by Barth et al. [1] is a *rectangulation procedure* that successively augments G with new vertices and edges to a graph $G^\star$ along with a valid ortho-radial representation $\Gamma^\star$ where every face of $G^\star$ is a *rectangle*, i.e., a face with no concave angles. In particular the algorithm only inserts edges that preserve the validity of the ortho-radial representation. To that end it applies $O(n^2)$ validity tests. Using the validity test from Section 2 the rectangulation algorithm runs in $O(n^4)$ time. We now sketch the procedure in more detail and briefly describe how to improve its running time to $O(n^2)$ time.

Consider a face f with a concave angle at u such that the following two turns when walking along f (in clockwise direction) are right turns; see Fig. 5. We call u a *port* of f and define a set of *candidate edges* that contains those edges e of f, for which $\mathrm{rot}(f[u,e]) = 2$. We treat this set as a sequence $e_1, \ldots, e_l$, where the edges appear in the same order as in f beginning with the first candidate after u. The *augmentation* Γ^u_{vw} with respect to a candidate edge vw is obtained by splitting vw into the edges vz and zw, where z is a new vertex, and adding the edge uz in the interior of f such that the angle formed by zu and the edge following u on f is 90°. The direction of the new edge uz in Γ^u_{vw} is the same for all candidate edges. If this direction is vertical, we call u a *vertical port* and otherwise a *horizontal port*.

The rectangulation algorithm successively removes ports by inserting edges. For a vertical

port u it inserts a vertical edge from u to its first candidate e_1, which yields a valid ortho-radial representation $\Gamma^u_{e_1}$ [2, Lemma 21]. For a horizontal port u it successively goes through candidates $e_1, \ldots, e_l$ of u and takes the first valid augmentation $\Gamma^u_{e_i}$, which always exists [2] and can be identified by the validity test of Section 2. Since there are $O(n)$ concave angles to remove and $O(n)$ candidates per augmentation, the algorithm runs in $O(n^4)$.

Exploiting structural insights we speed up each validity test to $O(n)$ time obtaining $O(n^3)$ running time in total. Since validity tests are only performed for horizontal ports, we focus on removing the concave angle at a horizontal port u in the remainder of the section. We further assume without loss of generality that the new edge uz points right. The key insight to improve the running time of a single validity tests is the following structural observation.

▶ **Lemma 7.** *If the first augmentation $\Gamma^u_{e_1}$ contains a decreasing cycle, the new edge uz in any augmentation Γ^u_e has label $\ell_C(uz) = 0$ on any decreasing cycle C.*

Hence, uz has the minimum label on all decreasing cycles and therefore one DFS from uz suffices to check for decreasing cycles. For increasing cycles a similar observation does not hold but in the rectangulation procedure only the final test for u needs to test for increasing cycles. This test can be replaced by a test for a horizontal path [2, Lemmas 25, 26]. This improves the running time of each applied validity test to $O(n)$.

Finally, we reduce the number of validity tests to $O(n)$ obtaining $O(n^2)$ running time for the rectangulation algorithm. We introduce a post-processing phase after each augmentation step, which ensures that all but a constant number of candidates that were considered for u lie on rectangular faces. Thus, these edges will not be candidates for any future port. All concave angles introduced in the post-processing phase become vertical ports, for which no validity tests are performed. Hence, the total number of validity tests is reduced to $O(n)$.

▶ **Theorem 8.** *Given a valid ortho-radial representation Γ of a graph G, a corresponding rectangulation can be computed in $O(n^2)$ time.*

In particular, using Corollary 19 from [2], given a graph G with valid ortho-radial representation Γ, a corresponding ortho-radial drawing Δ can be computed in $O(n^2)$ time. This concludes the construction of a TSM framework for ortho-radial drawings. An interesting open problem is to devise algorithms that construct valid ortho-radial representations using a small number of bend vertices. In particular, what is the complexity of testing whether a given graph with a fixed embedding has an ortho-radial drawing without bends?

Acknowledgement. We thank Lukas Barth for interesting discussions.

References

1 L. Barth, B. Niedermann, I. Rutter, and M. Wolf. Towards a Topology-Shape-Metrics framework for ortho-radial drawings. In B. Aronov and M.J. Katz, editors, *SoCG 2017*, volume 77 of *LIPIcs*, pages 14:1–14:16, 2017.

2 L. Barth, B. Niedermann, I. Rutter, and M. Wolf. Towards a Topology-Shape-Metrics framework for ortho-radial drawings. *CoRR*, arXiv:1703.06040, 2017.

3 M. Hasheminezhad, S. M. Hashemi, B. D. McKay, and M. Tahmasbi. Rectangular-radial drawings of cubic plane graphs. *Computational Geometry: Theory and Applications*, 43:767–780, 2010.

4 M. Hasheminezhad, S. M. Hashemi, and M. Tahmasbi. Ortho-radial drawings of graphs. *Australasian Journal of Combinatorics*, 44:171–182, 2009.

5 R. Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM Journal on Computing*, 16(3):421–444, 1987.