Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 17, no. 1, pp. 1–10 (2013)
DOI: 10.7155/jgaa.00281

Orthogonal-Ordering Constraints are Tough

Ulrik Brandes   Barbara Pampel

Department of Computer & Information Science
University of Konstanz

Abstract

We show that rectilinear graph drawing, the core problem of bend-minimum orthogonal graph drawing, and uniform edge-length drawing, the core problem of force-directed placement, are $\mathcal{NP}$-hard even for embedded paths if subjected to orthogonal-ordering constraints.

Submitted:	Reviewed:	Revised:	Accepted:	Final:
June 2010	April 2011	September 2012	November 2012	December 2012

Published: January 2013

Article type:	Communicated by:
Concise paper	G. Di Battista

A preliminary version of this paper appeared as U. Brandes and B. Pampel: On the hardness of orthogonal-order preserving graph drawing, in *Proc. Intl. Symp. Graph Drawing (GD '08)*, Springer Lecture Notes in Computer Science vol. 5417, pp. 266–277, 2009. We are grateful to two anonymous reviewers whose comments helped to improve the exposition.

E-mail addresses: ulrik.brandes@uni-konstanz.de (Ulrik Brandes) barbara.pampel@uni-konstanz.de (Barbara Pampel)

1 Introduction

In some scenarios, graph drawing algorithms receive not only a graph as input, but also an initial (possibly partial) drawing. The task then is to redraw the graph while maintaining selected features of the input drawing. Examples of this kind are embedding-constrained graph layout, shape simplification, sketch-based drawing, and dynamic graph layout.

Similar situations are encountered in cartographic applications, with the simplification of lines being a fundamental example. Given a polygonal path, the task is to generate a simpler representation of the path, for instance by omitting vertices [6, 14] (level of detail) or by restricting the feasible types of segments [19, 17] (schematization).

A common constraint in all these scenarios is the preservation of horizontal and vertical ordering of vertices or points because this is thought to assist viewers in maintaining a mental map [18, 3, 4].

We show, however, that it is $\mathcal{NP}$-hard to draw a graph subject to common drawing standards when orthogonal-ordering constraints are introduced, even if restricted to simple embedded paths.

A drawing is called *rectilinear*, if all edges are axis-parallel. It is $\mathcal{NP}$-complete to decide whether any graph with maximum degree four has a rectilinear drawing [9]. The problem remains $\mathcal{NP}$-complete for instances in which each edge is prescribed to be either horizontal or vertical, but becomes polynomial when the prescriptions also include directions [16]. Without orthogonal-ordering constraints, every path has a rectilinear drawing. We show that even for paths it is $\mathcal{NP}$-complete to decide whether there is a rectilinear drawing that satiesfies given orthogonal-ordering constraints. This result implies that bend-minimum orthogonal graph drawing becomes hard when orthogonal-ordering constraints are introduced. While it is $\mathcal{NP}$-hard in general even without such ordering constraints [11], it is polynomial for embedded planar 4-graphs [21]. The paths we construct in the proof of Theorem 1 are embedded, however, and their embedding is preserved by any feasible drawing.

Our second result relates to force-directed graph drawing, where uniform edge-length is a central objective [15]. In a *uniform edge-length drawing*, all edges have the same length. While it is $\mathcal{NP}$-hard to decide whether a planar graph can be drawn with uniform edge lengths [8], paths and even trees can be drawn with any edge lengths [1]. We show that even for paths it is $\mathcal{NP}$-hard to decide whether there is a uniform edge-length drawing that satiesfies orthogonal-ordering constraints. An algorithm for orthogonal-order preserving drawing of general graphs with approximately uniform edge lengths is given in [7].

2 Preliminaries

All graphs considered in this paper are simple and undirected. Since we require edges to be represented as straight-lines, a *drawing* of a graph $G = (V, E)$ is a pair $p = (x, y)$ of coordinate vectors $x = (x_v)_{v \in V}$ and $y = (y_v)_{v \in V}$ such that no two vertices are in the same position and no edge intersects a vertex that it is not incident to.

A *preorder* (also called weak partial order) is a binary relation that is reflexive and transitive. We consider the following type of constraints.

Definition 1 (Orthogonal-order Preserving Graph Drawing) *Given a graph $G = (V, E)$ and preorders $\preceq_x, \preceq_y$ on V, a drawing $p = (x, y)$ of G is called* orthogonal-order preserving*, if*

$$u \preceq_x v \implies x(u) \leq x(v) \quad \text{and} \quad u \preceq_y v \implies y(u) \leq y(v)$$

for all pairs of vertices $u, v \in V$.

Note that, in particular, ties in a preorder must be preserved, but no drawing is feasible if two vertices are tied in both preorders.

For concreteness we will assume that the constraints $\preceq_x, \preceq_y$ arise from a given drawing for which the two preorders induced by the coordinate axes are to be preserved. While this is in fact a restriction to total preorders, it will turn out that the problems of interest here remain intractable.

The following observations are crucial for our subsequent arguments. Assume that the input to an orthogonal-order constrained graph drawing problem contains the edges e, e_h, and e_v as shown in Figure 1. For $e = \{u, v\}$, we call the unbounded area of points with x-coordinates in the interval $[x(u), x(v)]$ the *vertical strip* of e. Since e_v is contained in it, the orthogonal-ordering constraints derived from the drawing imply that a vertical edge e *forces* e_v to be vertical, too. If e_v in turn is not drawn vertically, it *prevents* e from being vertical. Similarly, the *horizontal strip* of e is defined by the y-coordinates in the interval $[y(v), y(u)]$ covered by e. Since e can force e_h to be horizontal, e_h can prevent e from being so.

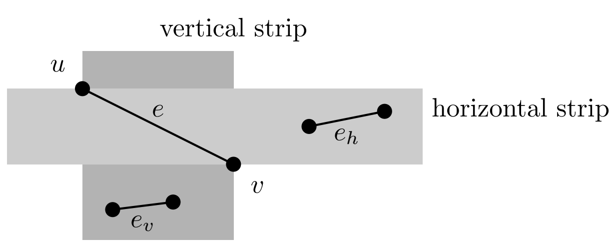


Figure 1: If e is horizontal (vertical), an edge in its horizontal (vertical) strip must be, too. If such an edge is not horizontal (vertical), e cannot be, either.

3 Rectilinear Drawings

Rectilinear drawings do not exist for graphs that contain a vertex of degree higher than four. As the proof of the following lemma indicates, there always exists, if any, a rectilinear *grid drawing*, where all vertices have integer coordinates. This holds independently of any orthogonal-ordering constraints.

Lemma *The orthogonal-order preserving rectilinear drawing problem is in $\mathcal{NP}$.*

Proof: Consider any input $(G, \preceq_x, \preceq_y)$ for which there is a feasible drawing. This drawing implies total preorders of the vertices along each axis. Now consider a vertical strip between any two consecutive x-coordinates. This strip separates the drawing into two parts that are connected by horizontal lines only. The width of the strip can therefore be adjusted arbitrarily by horizontally moving the two parts closer together or farther apart. The analogous observation holds for a horizontal strip between two consecutive y-coordinates. Assigning to each vertex its ranks in the two total preorders induced by the drawing therefore yields another rectilinear drawing that also satisfies the ordering constraints, but with integer coordinates in the range $\{1, \ldots, n\}$. A non-deterministic Turing machine can output such coordinates in polynomial time, and it can be verified in deterministic polynomial time whether they yield a feasible drawing. □

We show that the order-constrained rectilinear drawing problem is $\mathcal{NP}$-complete even for paths and with very restricted constraints. Specifically, we consider the problem that a planarly drawn graph is to be redrawn with axis-parallel edges while preserving its planar embedding and the orthogonal ordering of its vertices.

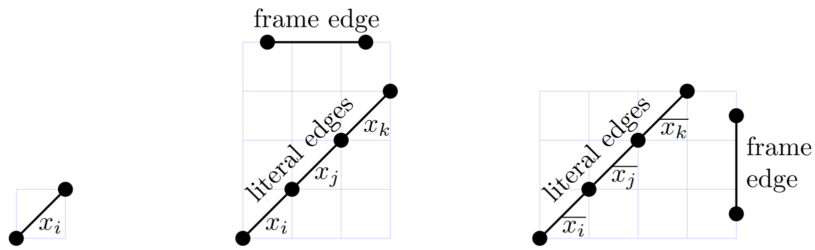
Theorem 1 *The orthogonal-order preserving rectilinear drawing problem is $\mathcal{NP}$-complete even for simple path drawings, and regardless of whether planarity is to be preserved or not.*

Proof: The problem is in $\mathcal{NP}$ by the previous lemma. Completeness is shown by reduction from MONOTONE 3-SAT, an $\mathcal{NP}$-complete variant of 3-SAT in which each clause contains either only positive or only negative literals [10]. An instance $F = C_1 \wedge \ldots \wedge C_m$ of MONOTONE 3-SAT over variables $x_1, \ldots, x_n$ thus consist of clauses $C_i = y_{i1} \vee y_{i2} \vee y_{i3}$, where either $y_{ij} \in \{x_1, \ldots, x_n\}$ (positive clause) or $y_{ij} \in \{\overline{x_1}, \ldots, \overline{x_n}\}$ (negative clause) for all $j = 1, 2, 3$.

We transform F into an instance of the drawing problem by constructing a path $P(F)$ using the gadgets shown in Figure 2 and requiring preservation of the orthogonal orderings implied by its drawing.

First, variable and clause gadgets are placed along a diagonal, line, so that initially the ordering constraints of each gadget are independent from those in other gadgets. We will ensure that variable edges are drawn horizontally (vertically) if and only if the corresponding variable is assigned true (false).

For each occurrence of a variable in a clause we then place a pair of *consistency edges* in the empty intersections of the horizontal and vertical strips



(a) variable (b) positive clause (c) negative clause

Figure 2: Gadgets for the rectilinear drawing problem: a variable can be drawn horizontally (true) or vertically (false), and frame edges prevent clauses from having three non-satisfied literals

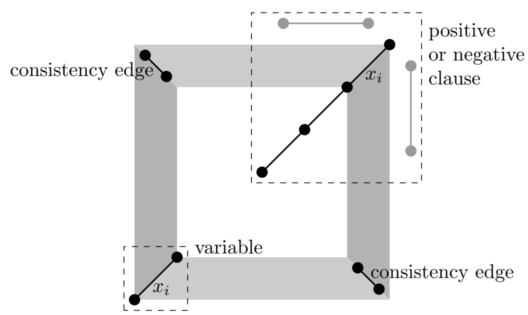


Figure 3: Ensuring consistency of a variable edge and an associated literal edge.

defined by a variable edge and one of its corresponding literal edges. In a feasible drawing all four edges are oriented alike because a variable edge forces one of the two consistency edges to be oriented alike which in turn prevents the literal edge from being oriented differently which then also forces the other consistency edge to be oriented alike.

The clause gadgets are designed such that a feasible drawing must have at least one horizontal (vertical) literal edge for each positive (negative) clause because otherwise the endpoints of their frame edge coincide.

Finally, linking paths connect all gadgets into a single path as shown in Figure 4. Since these linking paths do not introduce additional dependencies, a feasible drawing represents a consistent truth assignment with at least one satisfied literal in each clause, and a feasible drawing can be obtained from a satisfying truth assignment by orienting variable edges, consistency edges, and literal edges accordingly and routing the linking paths inbetween. Note that feasible drawings are necessarily planar. □

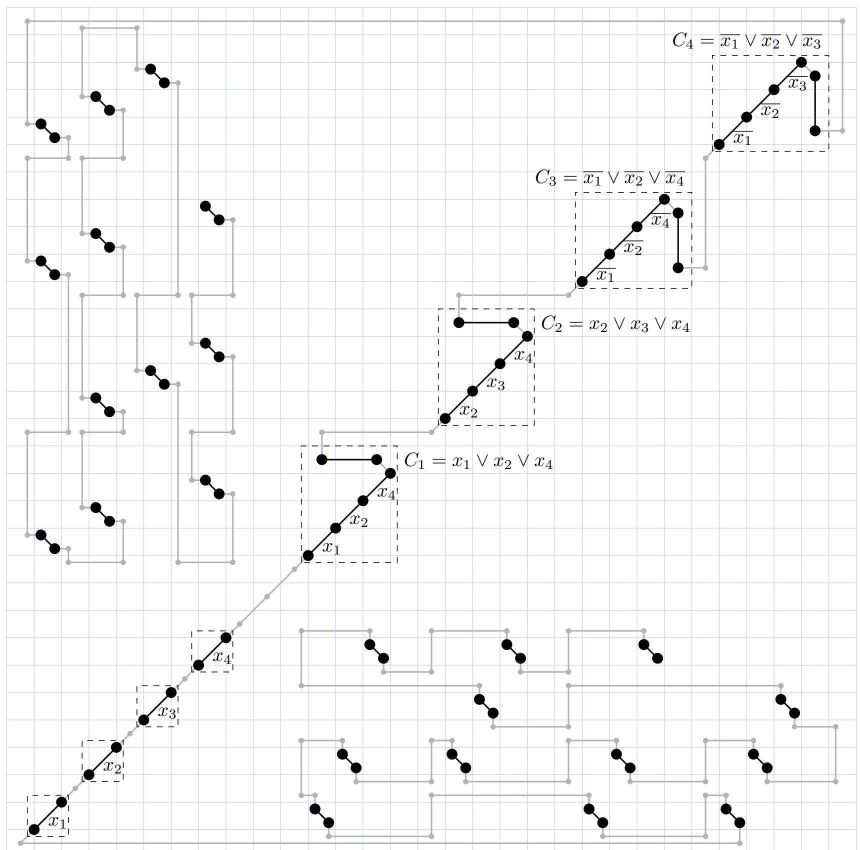


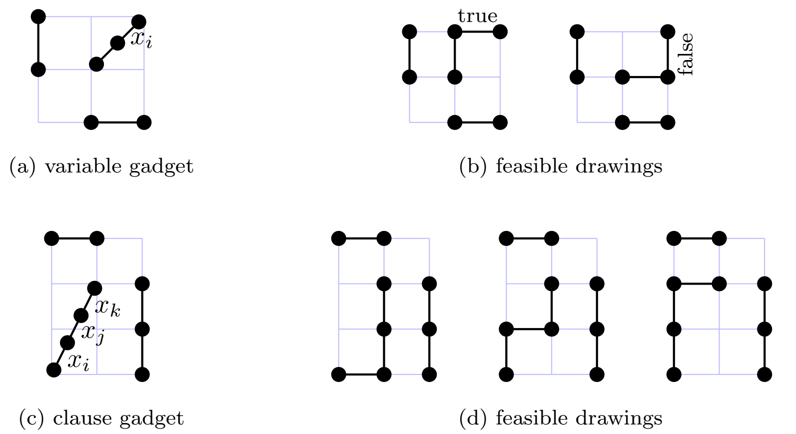
Figure 4: Drawing for path $P(F)$ constructed from a MONOTONE 3-SAT instance $F = (x_1 \vee x_2 \vee x_4) \wedge (x_2 \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_4}) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3})$.

4 Drawings with Uniform Edge Lengths

Theorem 2 *The orthogonal-order preserving uniform edge-length drawing problem is $\mathcal{NP}$-hard even for simple path drawings, and regardless of whether planarity is to be preserved or not.*

Proof: The proof is by reduction from ALL-POSITIVE ONE-IN-THREE-SAT, an $\mathcal{NP}$-complete variant of 3-SAT in which there are only positive literals and exactly one literal must be satisfied in each clause [20].

The reduction is essentially the same as the one for rectilinear drawings, though with the gadgets shown in Figure 5. Both gadgets ensure that in a feasible drawing exactly one of the diagonal edges is drawn horizontally and the others vertically. This is because the frame edges in a clause define a rectangular region to which the diagonal path is confined. Since edges must be drawn with uniform length, this region is at most one unit wide and one (variable gadget)



(a) variable gadget (b) feasible drawings

(c) clause gadget (d) feasible drawings

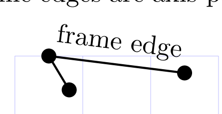
Figure 5: Gadgets modified for the uniform edge-length drawing problem

or two units (clause gadget) high. The diagonal path must be of length two or three because of the drawing convention, and it must be monotone in both x- and y-direction because of the ordering constraints. This leaves only the alternatives shown. Note that the diagonal paths do not fit into smaller regions and, therefore, bordering edges are necessarily vertical or horizontal. By adding extra rows and columns between gadgets, we can create enough space to add linking paths that are long and flexible without adding undesired constrains.

Again, a horizontal (vertical) variable edge corresponds to an assignment of true (false) and the same kind of consistency edges ensure that variable edges and literal edges are oriented alike. Hence there is a feasible drawing if and only if exactly one of the variables in each clause is assigned the value true. Again, these feasible drawings are planar. □

5 Discussion

We have shown that orthogonal-ordering constraints render rectilinear and uniform edge-length graph drawing problems intractable even for the simplest of graphs, single paths. By small perturbation of the instances created, both results generalize to inputs with vertices in general position, i.e. with no three vertices on a line. Similarly, the hardness of uniform edge-length drawing generalizes to total orders $\preceq_x, \preceq_y$ because the axis-alignment in feasible drawings is not enforced by coordinate ties in the input. This is different in the clause gadgets used for the rectilinear case, where it was essential that frame edges are axis-parallel. Since pairs of edges are forbidden to overlap in more than one point, the frame edges in clause gadgets of Figure 2 can be replaced by pairs of edges as shown here.



The embedding preserving variant of Theorem 1 has been generalized to slopes that are multiples of $90/d$ degrees for all $d \geq 1$ [12]. This is in contrast to an optimal algorithm for paths that are monotone with respect to one axis [5] and direction-restricted models in which paths or vertices are constrained to lie within a given distance of the input position and the number of bends can be minimized in polynomial time [19, 17].

To proof Theorem 2 we constructed gadgets that essentially enforced rectilinear drawings on the grid, for which the edge-length problem has already been studied in [2, 13].

For practical purposes it will be interesting to characterize realizable families of orthogonal-ordering constraints. Of the many conceivable variants, one that may be particularly interesting to study is the following: under which conditions can a graph be drawn in $d+1$ dimensions subject to d preorders?

References

[1] C. Bachmaier, U. Brandes, and B. Schlieper. Drawing phylogenetic trees. In X. Deng and D.-Z. Du, editors, *Proceedings of the 16th Annual International Symposium on Algorithms and Computation (ISAAC '05)*, volume 3827 of *LNCS*, pages 1110–1121. Springer, 2005. doi:10.1007/11602613_110.

[2] S. Bhatt and S. Cosmadakis. The complexity of minimizing wire lengths in VLSI layouts. *Information Processing Letters*, 25(4):263–267, 1987. doi:10.1016/0020-0190(87)90173-6.

[3] S. Bridgeman and R. Tamassia. Difference metrics for interactive orthogonal graph drawing algorithms. In S. H. Whitesides, editor, *Proceedings of the 6th International Symposium on Graph Drawing (GD '98)*, volume 1547 of *LNCS*, pages 57–71. Springer, 1998. doi:10.1007/3-540-37623-2_5.

[4] S. Bridgeman and R. Tamassia. A user study in similarity measures for graph drawing. *Journal of Graph Algorithms and Applications*, 6(3):225–254, 2002. doi:10.7155/jgaa.00053.

[5] D. Delling, A. Gemsa, M. Nöllenburg, and T. Pajor. Path schematization for route sketches. In *Algorithm Theory - SWAT 2010*, volume 6139 of *LNCS*, pages 285–296. Springer, 2010. doi:10.1007/978-3-642-13731-0_27.

[6] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973. doi:10.3138/FM57-6770-U75U-7727.

[7] T. Dwyer, Y. Koren, and K. Marriott. Stress majorization with orthogonal ordering constraints. In P. Healy and N. S. Nikolov, editors, *Proceedings of the 13th International Symposium on Graph Drawing (GD '05)*, volume 3843 of *LNCS*, pages 141–152. Springer, 2006. doi:10.1007/11618058_14.

[8] P. Eades and N. C. Wormald. Fixed edge-length graph drawing is np-hard. *Discrete Applied Mathematics*, 28:111–134, 1990. doi:10.1016/0166-218X(90)90110-X.

[9] M. Formann, T. Hagerup, J. Haralambides, M. Kaufmann, F. T. Leighton, A. Symvonis, E. Welzl, and G. Woeginger. Drawing graphs in the plane with high resolution. *SIAM Journal on Computing*, 22(5):1035–1052, 1993. doi:10.1137/0222063.

[10] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of $\mathcal{NP}$–Completeness.* W.H. Freeman & Co, 1979.

[11] A. Garg and R. Tamassia. On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.*, 31(2):601–625, 2002. doi:10.1137/S0097539794277123.

[12] A. Gemsa, M. Nöllenburg, T. Pajor, and I. Rutter. On d-regular schematization of embedded paths. In I. Cerná, T. Gyimóthy, J. Hromkovic, K. Jefferey, R. Královic, M. Vukolic, and S. Wolf, editors, *Proc. 37th Intl. Conf. Current Trends in Theory and Practice of Computer Science (SOFSEM 2011)*, volume 6543 of *LNCS*, pages 260–271. Springer, 2011. doi:10.1007/978-3-642-18381-2_22.

[13] A. Gregori. Unit-length embedding of binary trees on a square grid. *Information Processing Letters*, 31(4):167–173, 1989. doi:10.1016/0020-0190(89)90118-X.

[14] H. Imai and M. Iri. An optimal algorithm for approximating a piecewise linear function. *Journal of Information Processing*, 9(3):159–162, 1986.

[15] M. Kaufmann and D. Wager, editors. *Drawing Graphs: Methods and Models.*, volume 2025 of *LNCS*, chapter Drawing on physical analogies., pages 71–86. Springer, 2001. doi:10.1007/3-540-44969-8_4.

[16] J. Maňuch, M. Patterson, S.-H. Poon, and C. Thachuk. Complexity of finding non-planar rectilinear drawings of graphs. In U. Brandes and S. Cornelsen, editors, *Proceedings of the 18th International Symposium on Graph Drawing (GD '11)*, volume 6502 of *LNCS*, pages 305–316. Springer, 2011. doi:10.1007/978-3-642-18469-7_28.

[17] D. Merrick and J. Gudmundsson. Path simplification for metro map layout. In M. Kaufmann and D. Wagner, editors, *Proceedings of the 14th International Symposium on Graph Drawing (GD '06)*, volume 4372 of *LNCS*, pages 258–269. Springer, 2007. doi:10.1007/978-3-540-70904-6_26.

[18] K. Misue, P. Eades, W. Lai, and K. Sugiyama. Layout adjustment and the mental map. *Journal on Visual Languages and Computing*, 6(2):183–210, 1995. doi:10.1006/jvlc.1995.1010.

[19] G. Neyer. Line simplification with restricted orientations. In F. Dehne, A. Gupta, J.-R. Sack, and R. Tamassia, editors, *Proceedings of the 6th Workshop on Algorithms and Data Structures (WADS '99)*, volume 1663 of *LNCS*, pages 13–24. Springer, 1999. doi:10.1007/3-540-48447-7_2.

[20] T. J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the 10Th annual ACM symposium on Theory of computing*, STOC '78, pages 216–226. ACM, 1978. doi:10.1145/800133.804350.

[21] R. Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM Journal on Computing*, 16:421–444, 1987. doi:10.1137/0216030.