Cross-coloring: improving the technique by Kolmogorov and Barzdin

Therese Biedl * Timothy M. Chan*

June 26, 2000

Abstract

In this paper, we study how to color *crosses*, i.e., pairs of rows and columns in a grid, such that any two overlapping crosses have a different color. We show that this problem can be solved by computing an edge-coloring in a bipartite graph. Using this result we reduce significantly the time complexity and the volume bounds of algorithms for three-dimensional orthogonal graph drawing that are based on the technique of Kolmogorov and Barzdin.

1 Background

In this paper, we study how to color crosses and its application to three-dimensional orthogonal graph drawing.

Assume that we have a rectangular grid with R rows and and C columns. A *cross* is a pair (r, c) of one row and one column. We say that two crosses (r_1, c_1) and (r_2, c_2) *overlap* if $r_1 = r_2$ or $c_1 = c_2$ or both. Given a collection $\mathcal{C}$ of crosses, a *cross-coloring with k colors* is an assignment of integers $\{1, \ldots, k\}$ to the crosses such that any two crosses that overlap have a different color.

We want to find a cross-coloring with few colors, and give two approaches for this problem. The first, straightforward, approach converts this problem into a vertex-coloring in a *conflict graph*, i.e., a graph that expresses when two crosses must not have the same color. We develop here a second approach which uses the geometric structure to convert the problem into an edge-coloring problem of a bipartite graph. The second approach is superior in that it uses the minimum number of colors (whereas the first approach only uses asymptotically the optimal number of colors). Also, depending on the configuration of the crosses, the second approach takes less time. We give these two approaches in Section 2.

Our interest in cross-coloring was motivated by three-dimensional orthogonal graph drawing. In [KB67], Kolmogorov and Barzdin introduced a very effective technique that is the basis for many subsequent algorithms for 3D orthogonal graph drawing [ESW96, ESW97,

*Department of Computer Science, University of Waterloo, Waterloo, ON N2L 3G1, Canada, e-mail {biedl,tmchan}@uwaterloo.ca.

BTW00]. As we will show, the problem to be solved can be expressed as a cross-coloring problem. Using our second approach, rather than the traditional first approach, drastically improves the time complexity of these algorithms from $O(m^{3/2})$ or higher to $O(m \log m)$. Since we use fewer colors, this also improves the volume bounds of these algorithms. We explain orthogonal graph drawings and our improvements in Section 3.

2 Cross-coloring

From now on, assume that $\mathcal{C}$ is a collection of N crosses in a grid with R rows and C columns; the collection is allowed to contain the same cross repeatedly. Let Δ_h be the maximum number of crosses that all have the same row, let Δ_v be the maximum number of crosses that all have the same column, and let $\Delta_m = \max\{\Delta_h, \Delta_v\}$. Clearly, no coloring of $\mathcal{C}$ can have fewer than Δ_m colors, because if Δ_m crosses all have the same, say, row, then these Δ_m crosses each must have a different color. We now give two approaches to coloring $\mathcal{C}$ with $O(\Delta_m)$ colors.

2.1 Reduction to vertex-coloring

The first, straightforward, approach to cross-coloring is to express the problem as a vertex-coloring problem in a conflict graph H defined as follows: The vertices of H are exactly the crosses $\mathcal{C}$. Two crosses (r_1, c_1) and (r_2, c_2) are connected by an edge in H if and only if they overlap. Clearly, any vertex-coloring of H gives a cross-coloring.

Assume that (r, c) is a cross. Then there are at most $\Delta_h - 1$ other crosses with row r and at most $\Delta_v - 1$ other crosses with column c. Hence, the degree of (r, c) in H is at most $\Delta_h - 1 + \Delta_v - 1 \leq 2\Delta_m - 2$, and the maximum degree of H is at most $2\Delta_m - 2$. With a simple greedy-method, we can find a vertex-coloring of H with $2\Delta_m - 1$ colors, hence a cross-coloring with $2\Delta_m - 1$ colors.

The time to compute this cross-coloring is proportional to $|V(H)| + |E(H)|$. The number of vertices of H is N, but the number of edges of H might be as high as $(2\Delta_m - 2)N/2$ if every row and every column is incident to exactly Δ_m crosses. Hence no better time complexity than $O(\Delta_m N)$ can be guaranteed to find the cross-coloring.

2.2 Reduction to edge-coloring

Now we reduce the cross-coloring problem to an edge-coloring in a bipartite graph H' defined as follows: H' has one vertex for every row r and one vertex for every column c, so $|V(H')| = R + C$. For every cross (r, c) we have one edge in H' that connects row r and column c; this edge will also be denoted (r, c). If there is more than one cross (r, c), then we repeatedly connected r and c in H', hence $|E(H')| = |\mathcal{C}| = N$.

Let r be a row. Then there are at most Δ_h many crosses that use row r; so r has degree $\leq \Delta_h$ in H'. Similarly any column has degree $\leq \Delta_v$ in H', so the maximum degree in H' is Δ_m. Since H' is bipartite, it has an edge-coloring with Δ_m colors. This edge-coloring can be found in $O(|E(H)| \log(|V(H)|)) = O(N \log(R + C))$ time [CH82]. (Alternatively, the edge-coloring could be found in $O(\Delta_m N)$ time [Sch98], or an edge-coloring with $\Delta_m + 2$

colors could be found in $O(\log(\Delta_m)N)$ time [KR00]. Neither of these algorithms gives an asymptotically better time complexity in our application below.)

Now, if two crosses (r_1, c_1) and (r_2, c_2) overlap, then they have (say) the same row $r_1 = r_2$, and hence the corresponding edges (r_1, c_1) and (r_2, c_2) in H' have a common endpoint at $r_1 = r_2$. Hence the two crosses obtain a different color in the edge-coloring, and the edge-coloring yields a cross-coloring.

Theorem 1 *Let $\mathcal{C}$ be a collection of N crosses in a grid with R rows and C columns. Then a cross-coloring of $\mathcal{C}$ with the minimum number of colors can be found in $O(N \log(R + C))$ time.*

3 Improving 3D orthogonal graph drawing algorithms

We now show how faster better cross-coloring can be used to improve those 3D orthogonal graph drawing algorithms that are based on the technique by Kolmogorov and Barzdin. We first explain this technique, then show how it relates to cross-coloring, and finally study improvements for specific algorithms.

3.1 A technique for 3D orthogonal graph drawings

In [KB67], Kolmogorov and Barzdin introduced a very effective technique for 3D orthogonal graph drawing, which is the basis for many 3D orthogonal graph drawing algorithms [ESW96, ESW97, BTW00]. We refer to these papers for motivation and complete definitions regarding orthogonal graph drawing, and review here only the necessary terms. In a 3D orthogonal graph drawing, edges are routed along the grid lines of an underlying rectangular grid. Also, edge routes should be disjoint except possibly at common endpoints.

The technique of Kolmogorov and Barzdin consists of defining an edge route that depends only on the begin point, end point, and the height assigned to the edge. More precisely:

- All edges are directed from one endpoint to the other.
- For any directed edge $e = vw$, assign a two-dimensional grid point $(x_e(v), y(v))$ for v and a two-dimensional grid point $(x_e(w), y(w))$ for w.
- For any directed edge vw, assign a height $h(vw)$.
- The direction, grid points and height determine the route of the edge.

The particular method of choosing directions, grid points and edge route varies from algorithm to algorithm and is irrelevant for our observations below; we refer to the original papers for details. In all cases, it can be shown that the edge routes are disjoint (except possibly at endpoints) as long as the following two rules hold for any two directed edges $e_1 = v_1w_1$ and $e_2 = v_2w_2$:

- $y_{e_1}(v_1) \neq y_{e_2}(v_2)$ or $h(v_1w_1) \neq h(v_2w_2)$.
- $x_{e_1}(w_1) \neq x_{e_2}(w_2)$ or $h(v_1w_1) \neq h(v_2w_2)$.

3.2 Application of cross-coloring

When using the technique of Kolmogorov and Barzdin, one must find a height for each edge, given the information on the grid points, such that the above two conditions hold. Kolmogorov and Barzdin [KB67] left unspecified how to do this. Eades, Stirk and Whitesides [ESW96] proposed to this with a vertex-coloring in a conflict graph, which is the approach taken in subsequent papers [ESW97, BTW00] as well. This step turns out to be the bottleneck in the time complexity of the orthogonal graph drawing algorithms, since it is super-linear, while all other steps are linear.

We now show how to reduce finding suitable heights to cross-coloring. The approach of [ESW96] corresponds to our first approach with vertex-coloring. As we will see, using the second approach – edge-coloring – to find the cross-coloring and hence the heights improves the time complexity and volume bounds of the above algorithms.

So assume that for every directed edge $e = vw$ we have grid points $(x_e(v), y_e(v))$ and $(x_e(w), y_e(w))$ assigned. Define a cross for every directed edge $e = vw$, which consists of the row with y-coordinate $y_e(v)$ and the column with x-coordinate $x_e(w)$. Assume that we have a cross-coloring of these crosses; we want to show that the cross-coloring then indicates a suitable assignment of heights to edges.

If $e_1 = v_1w_1$ and $e_2 = v_2w_2$ are two directed edges with $y_{e_1}(v_1) = y_{e_2}(v_2)$, then the two crosses assigned to these two edges have the same row. Hence these two crosses have different colors, which implies that $h(v_1w_1) \neq h(v_2w_2)$. Similarly, if $x_{e_1}(w_1) = x_{e_2}(w_2)$, then the two crosses have the same column, hence different colors, and $h(v_1w_1) \neq h(v_2w_2)$. So this is a suitable assignment of heights for the directed edges.

Note that for every edge we have exactly one cross; hence finding the cross-coloring takes $O(m \log(R + C))$ time, where m is the number of edges in the graph and R and C are the number of rows and columns, respectively, that contain a grid point assigned to one edge. In particular, this time complexity is independent of the number Δ_m of edges that start in a common row or end in a common column. For the orthogonal graph drawing algorithms below, this is an improvement of the time complexity, since $\log(R + C) < \Delta_m$.

3.3 Improvements of specific algorithms

Now we study the effect of our observation onto various 3D orthogonal graph drawing algorithms.

The algorithm in [ESW96] With this algorithm, Eades, Stirk and Whitesides find a 3D orthogonal drawing of any n-vertex graph with maximum degree 6 where vertices are represented by points. This algorithm uses the technique of Kolmogorov and Barzdin. Since they used the first approach (based on a vertex-coloring), the time to find these drawings is $O(m^{3/2})$, which is $O(n^{3/2})$ by $m \leq 3n$. Using edge-coloring instead, the time complexity reduces to $O(n \log n)$. This also decreases the number of different heights needed, and therefore the volume bounds of the drawing. We will not study this in detail, since the volume bound of this algorithm has been improved with the next algorithm.

The Compact-drawing Algorithm [ESW97] With this algorithm, Eades, Symvonis and Whitesides find a 3D orthogonal drawing of any graph with maximum degree 6 where vertices are represented by points. This algorithm uses the technique of Kolmogorov and Barzdin. The *Z-dimension* (i.e., the extend in the Z-direction) of their drawing is $4K+1$, where K is the number of different heights in one coloring step.

The authors used an assignment of heights obtained through a vertex-coloring, and hence have $K \leq 2\Delta_m - 1$. With $\Delta_m \leq \lceil\sqrt{n}\rceil$, their Z-dimension hence is $\leq 8\lceil\sqrt{n}\rceil - 3$.

Using edge-coloring to obtain heights, we can improve the bound to $K \leq \Delta_m$, hence the Z-dimension is $\leq 4\lceil\sqrt{n}\rceil + 1$. This reduces the overall volume bound from $60n^{3/2} + o(n^{3/2})$ to $30n^{3/2} + o(n^{3/2})$. It also reduces the bound on the maximum edge length from $16\lceil\sqrt{n}\rceil - 10$ units to $12\lceil\sqrt{n}\rceil - 6$ units.

Finally, since there are $O(\sqrt{n})$ rows and columns, the time to find this drawing decreases from $O(n^{3/2})$ to $O(n \log n)$. The revised version of Theorem 2 of [ESW97] hence appears as follows:

Theorem 2 *Every n-vertex maximum degree 6 graph G has a three-dimensional orthogonal grid drawing with the following characteristics:*

(i) at most 7 bends per edge route,

(ii) $12\lceil\sqrt{n}\rceil - 6$ maximum edge length, and

(iii) a bounding box of dimensions $(3\lceil\sqrt{n}\rceil + 2) \times 5\lceil\sqrt{n}\rceil \times (4\lceil\sqrt{n}\rceil + 1)$.

Moreover, the drawing can be obtained in $O(n \log n)$ time.

The algorithm Optimal Volume Cube-Drawing **[BTW00]** With this algorithm, Biedl, Thiele and Wood find a 3D orthogonal drawing of any graph with n vertices and m edges where vertices are represented by cubes. This algorithm uses the technique of Kolmogorov and Barzdin. The Z-dimension of their drawing is $4K + \sqrt{2m} + o(\sqrt{m})$, where K is the number of different heights in one coloring step.

The authors used an assignment of heights obtained through a vertex-coloring, and hence have $K \leq \Delta_v + \Delta_h - 1$. With $\Delta_v \leq 4/\sqrt{3} \cdot \sqrt{m} + o(\sqrt{m})$ and $\Delta_h \leq 2\sqrt{2} \cdot \sqrt{m} + o(\sqrt{m})$, the Z-dimension hence is at most $(16/\sqrt{3} + 9\sqrt{2})\sqrt{m} + o(\sqrt{m})$.

Using edge-coloring to obtain heights, we can improve the bound to $K \leq \max\{\Delta_v, \Delta_h\} \leq 2\sqrt{2}\sqrt{m} + o(\sqrt{m})$, hence the Z-dimension is $\leq 9\sqrt{2}\sqrt{m} + o(\sqrt{m})$. This reduces the overall volume from $\approx 143m^{3/2} + o(m^{3/2})$ to $\approx 83m^{3/2} + o(m^{3/2})$.

Finally, since there are $O(\sqrt{m})$ rows and columns, the time to find this drawing decreases from $O(m^{3/2})$ to $O(m \log m)$. The revised version of Theorem 3 of [BTW00] hence appears as follows:

Theorem 3 *There exists an algorithm to determine a 12-degree-restricted orthogonal cube-drawing of any loopless graph G in $O(m \log m)$ time, with $\approx 83m^{3/2} + o(m^{3/2})$ bounding box volume and at most six bends per edge route.*

The algorithm Optimal Volume Box-Drawing **[BTW00]** With this algorithm, Biedl, Thiele and Wood find a 3D orthogonal drawing of any graph with n vertices and m edges where vertices are represented by boxes of arbitrary aspect ratios. This algorithm uses the technique by Kolmogorov and Barzdin. The Z-dimension of their drawing is $2K+1$, where K is the number of different heights in one coloring step.

The authors used an assignment of heights obtained through a vertex-coloring, and hence have $K \leq \Delta_v + \Delta_h - 1$. With $\Delta_v \leq 10m/\lceil\sqrt{n}\rceil$ and $\Delta_h \leq 6m/\lceil\sqrt{n}\rceil$, the Z-dimension hence is at most $32m/\lceil\sqrt{n}\rceil - 1$.

Using edge-coloring to obtain heights, we can improve the bound to $K \leq \max\{\Delta_v, \Delta_h\} \leq 10m/\lceil\sqrt{n}\rceil$, hence the Z-dimension is $\leq 20m/\lceil\sqrt{n}\rceil + 1$. This reduces the overall volume from $144m\sqrt{n} + o(m\sqrt{n})$ to $90m\sqrt{n} + o(m\sqrt{n})$.

Finally, since there are $O(\sqrt{n})$ rows and columns, the time to find this drawing decreases from $O(m^2/\sqrt{n})$ to $O(m \log n)$. The revised version of Theorem 5 of [BTW00] hence appears as follows:

Theorem 4 *There exists an algorithm to determine an orthogonal drawing of any simple graph G in $O(m \log n)$ time, with $90m\sqrt{n} + o(m\sqrt{n})$ bounding box volume and at most four bends per edge route.*

Acknowledgments

We thank Erik Demaine for useful discussions, and Peter Eades, Antonios Symvonis and Sue Whitesides for kindly sharing their revised version. Research of both authors was supported by NSERC.

References

[BTW00] T. Biedl, T. Thiele, and D. Wood. Three-dimensional orthogonal graph drawing with optimal volume. Technical Report CS-2000-12, Department of Computer Science, University of Waterloo, May 2000.

[CH82] R. Cole and J. Hopcroft. On edge coloring bipartite graphs. *SIAM J. Comput.*, 11(3):540–546, 1982.

[ESW96] P. Eades, C. Stirk, and S. Whitesides. The techniques of Kolmogorov and Bardzin for three-dimensional orthogonal graph drawings. *Information Processing Letters*, 60(2):97–103, 1996.

[ESW97] P. Eades, A. Symvonis, and S. Whitesides. Two algorithms for three dimensional orthogonal graph drawing. In *Graph Drawing (GD'96)*, volume 1190 of *Lecture Notes in Computer Science*, pages 139–154. Springer-Verlag, 1997. Extended version to appear in *Discr. Applied Math.*

[KB67] A.N. Kolmogorov and Ya.M. Barzdin. Realization of nets in three-dimensional space. *Problems in Cybernetics*, pages 261–268, March 1967.

[KR00] A. Kapoor and R. Rizzi. Edge-coloring bipartite graphs. *J. Algorithms*, 34(2):390–396, 2000.

[Sch98] A. Schrijver. Bipartite edge-coloring in $O(\Delta m)$ time. *SIAM Journal on Computing*, 28(3):841–846, 1998.