Graph Simultaneous Embedding Tool, GraphSET*

Alejandro Estrella-Balderrama, J. Joseph Fowler, and Stephen G. Kobourov

Department of Computer Science, University of Arizona
{aestrell,jfowler,kobourov}@cs.arizona.edu

Technical Report 07-03
September 3, 2007

Abstract. Problems in simultaneous graph drawing involve the layout of several graphs on a shared vertex set. This paper describes a *Graph Simultaneous Embedding Tool*, GraphSET, which can aid in the investigation of several embedding problems. In particular, GraphSET can be used in studies of simultaneous geometric embedding, simultaneous embedding with fixed edges, and colored simultaneous embedding. The tool can be used in two ways: (i) to study theoretical problems in simultaneous graph drawings by helping produce examples and counter-examples and (ii) to produce drawings of given input graphs using built-in implementations of known algorithms. GraphSET is available for download at http://graphset.cs.arizona.edu.

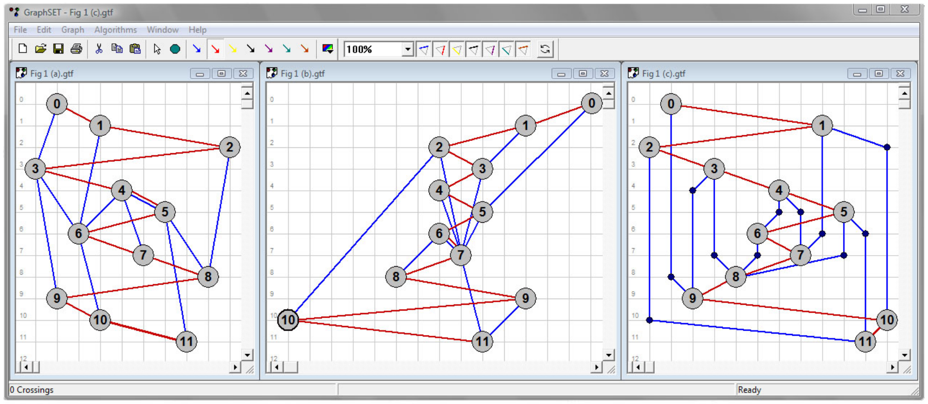
1 Introduction

Two n-vertex graphs $G_1(V, E_1)$ and $G_2(V, E_2)$ have a *simultaneous embedding with mapping* if each can be drawn in the xy-plane without crossings such that there exists a bijection $f : V \mapsto V$ in which v and $f(v)$ have the same xy-coordinates. If this can be done for some bijection f, then G_1 and G_2 are *simultaneously embedable without mapping*. If edges of both E_1 and E_2 are drawn with straight-line segments in which no pair of edges from the edge set cross, then G_1 and G_2 have a *simultaneous geometric embedding*.

If edges $e_1 \in E_1, e_2 \in E_2$ where $e_1 = (u, v)$ and $e_2 = (f(u), f(v))$ for some $u, v \in V$, then a *fixed edge* between u and v is one which is drawn the same way in both drawings (which holds trivially for straight-line edges but is not necessarily true when edge-bends are allowed). The problem of *colored simultaneous embedding* is a relaxation of the simultaneous embedding with mapping problem in which V is partitioned into k *colors* $V_1, V_2, \ldots, V_k$ such that $V_1 \cup V_2 \cup \cdots \cup = V$ and $V_i \cap V_j = \varnothing$ if $i \neq j$. Here the bijection f respects this coloring such that $f|_{V_i} : V_i \mapsto V_i$ in which $v \in V_i$ if and only if $f|_{V_i}(v) \in V_i$ for $i \in [1..k]$. Clearly, these definitions can be extended to an arbitrary number of graphs.

These problems are difficult to solve and require extensive manipulation of different instances in order to gain insight. A tool that allows dynamic manipulation of the vertices while keeping track of how the crossings change in each graph being simultaneously embedded can be very useful. We present our tool through a series of applications where the tool has benefited us. Our hope is that others investigating simultaneous embeddings can also find the tool beneficial.

* This work is supported in part by NSF grants CCF-0545743 and ACR-0222920.



(a) (b) (c)

Fig. 1. The three types of ULP trees: (a) caterpillar, (b) radius-2 star, and (c) 3-spider.

A related tool is the *Interactive Multi-User System for Simultaneous Graph Drawing* [7]. It only considers simultaneous geometric embedding of two graphs and the emphasis is on collaboration by using the DiamondTouch device [2]. Another related tool that can be used to obtain simultaneous drawing of graphs using force-directed methods is described in [3].

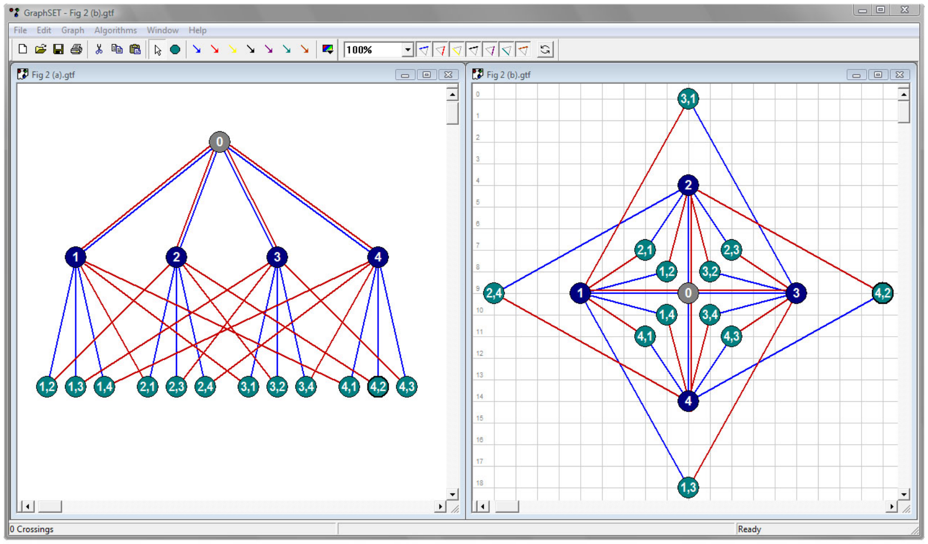
2 Applications

2.1 Unlabeled Level Planar Trees

A *level graph* $G(V, E, \phi)$ is a graph with mapping $\phi : V \mapsto [1..k]$ assigning every vertex to one of k levels so that $\phi(u) \neq \phi(v)$ for every edge. In a *level drawing* all vertices of a level share the same y-coordinate with each edge drawn in a y-monotonic fashion. If G can be drawn this way without edge crossings, then G is level planar. For our problem, we only consider bijections in which $\phi : V \mapsto [1..n]$ with one vertex per level. This allows us to restrict each vertex to its own horizontal line based upon its y-coordinate, called a *track*. If G is level planar overall all $n!$ labelings, then G is *unlabeled level planar* (ULP). The set of ULP graphs are precisely those that can be simultaneously embedded with a strictly monotonic path since one can draw the path as a y-monotonic zig-zag fashion. In [4] we characterized the set of ULP trees to be either

- (i) a *caterpillar* in which the removal of all degree-1 vertices leaves a path or empty graph,
- (ii) a *radius*-2 *star* that is subgraph of a double star of radius 2, or
- (iii) a *degree*-3 *spider* that is homeomorphic to a claw, $K_{1,3}$.

Fig. 1 show examples of each of these being simultaneously embedded with a strictly monotonic path. Our tool has the feature of allowing the user to snap and lock vertices to tracks to investigate not only unlabeled level planar graphs but the planarity of multiple level graphs being simultaneously embedded.



(a) (b)

Fig. 2. A pair of trees whose union has a subdivision of K_5 is shown in (a) in which one tree has the red edges and the other has blue edges. A simultaneous geometric embedding for this pair is given in (b).

2.2 Simultaneous Geometric Embedding

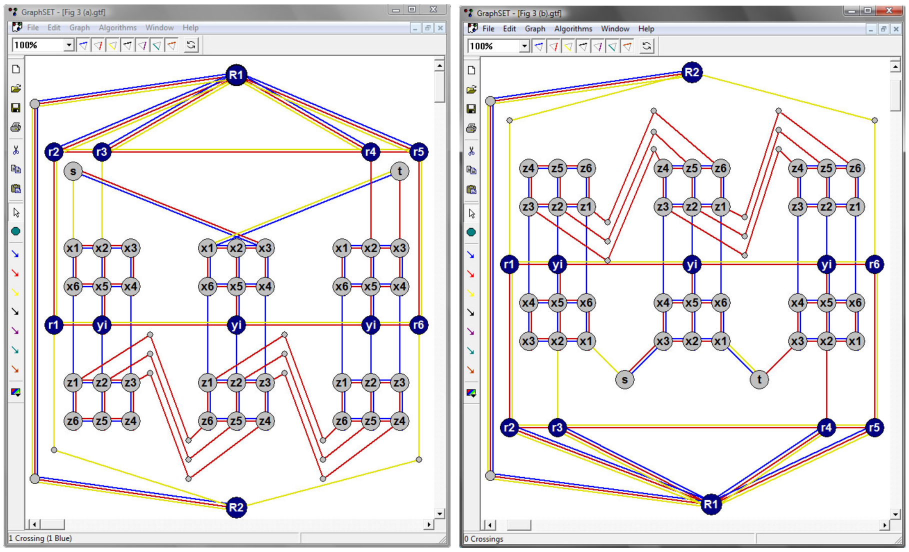
For this example we consider simultaneous geometric embedding of two trees $T_1(V, E_1)$ and $T_2(V, E_2)$ on $n^2 - 2n + 2$ vertices whose union contains a subgraph homeomorphic to the complete graph K_n on n vertices for a given n. Both T_1 and T_2 have a root vertex labeled '0' that is adjacent to $n-1$ vertices of V labeled '1', '2', ..., '$n-1$'. In each tree these $n-1$ vertices have $n-2$ leaves so that each non-leaf vertex has degree $n-1$. The leaves are labeled i, j for $i, j \in [1..n-1]$ such that $i \neq j$. In T_1 the vertex labeled $i \in [1..n-1]$ has leaves i, j for $j \in [1..n-1]$ such that $i \neq j$. Similarly, in T_2 the vertex labeled $j \in [1..n-1]$ has leaves i, j for $i \in [1..n-1]$ such that $i \neq j$.

Fig. 2 shows two trees for case of $n = 5$ on 17 vertices that illustrates a schema for generating a layout that works up to $n = 5$. We show another more complex example for $n = 7$ on the website `http://graphset.cs.arizona.edu`. For large values of n these tree pairs do not have a simultaneous geometric embedding, as show by Geyer *et al.* [6]. However, it is not known what is the smallest value of n that forces a crossing; for example, the case $n = 8$ is open.

2.3 Simultaneous Embedding with Fixed Edges

GraphSET can handle multiple edges with different colors. These edges can include bends and can be treated as a single edge (for fixed edges) or as different edges (for multi-graphs). An application of this tool for simultaneous embedding with fixed edges is the manipulation of gadgets for complexity proofs.

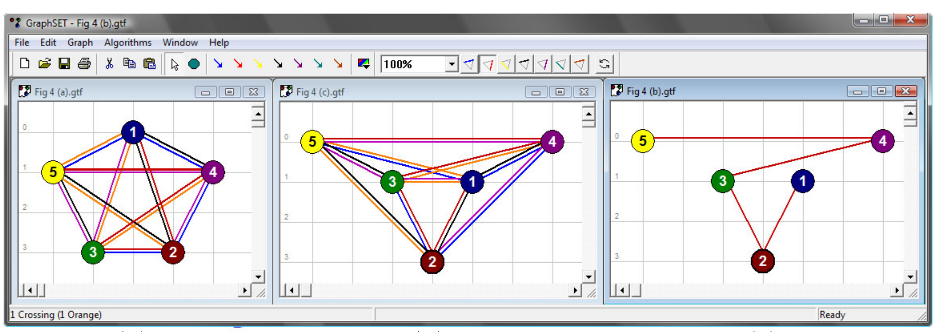
In [5] Gassner *et al.* proved that simultaneous embedding with fixed edges is NP-Complete for 3 graphs. The proof is a reduction to 3-SAT using clause



(a) (b)

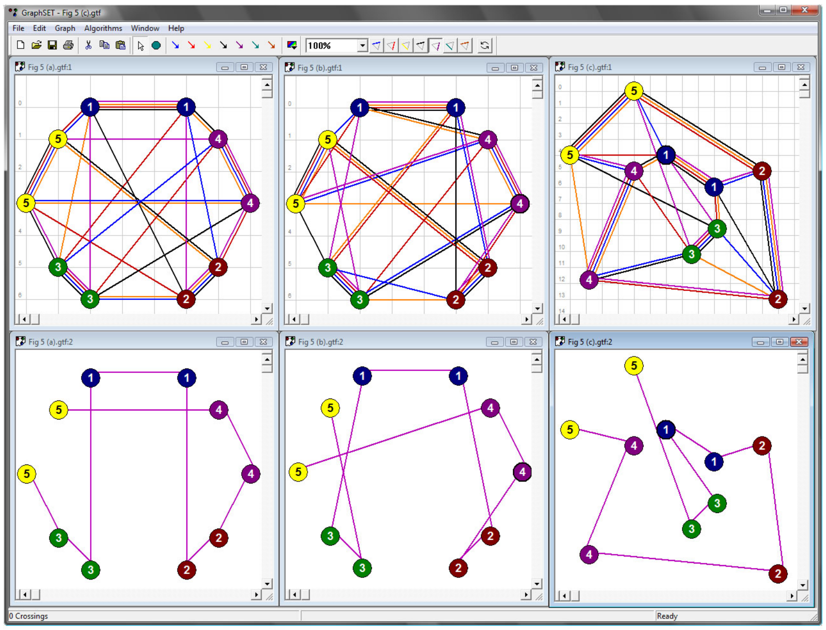
Fig. 3. (a) Gadget for a clause with 3 literals in a 3-SAT reduction; (b) A non-obvious crossings-free drawing of the gadget.

gadgets and literal gadgets (See Fig. 3(a)). There are two possible embeddings for each literal gadget, these embeddings corresponds to true or false values in the corresponding literals. The argument is that a drawing of the clause without crossings is only possible if one of the literals is true. In the drawing this implies that we can only get rid of a crossings by flipping a literal gadget (changing the embedding of the gadget). The tool is useful in finding problems in the gadget construction by exhibiting non-obvious embeddings that may break the argument (see for example Fig. 3(b)). The flipping operations included in the tool are helpful in this situation.



(a) (b) (c)

Fig. 4. Example of fives paths on five colors that cannot be simultaneously embedded.



(a) (b) (c)

Fig. 5. Example of how swapping adjacencies between vertices of the same color allow for simultaneous embedding.

2.4 Colored Simultaneous Geometric Embedding

Fig. 4(a) shows an example of five paths on five colors on distinctly colored vertices that was used in [1] to show that there does not exist a universal pointset for 5-colored paths. Each path is represented by edges of the same color. Here edge color is used to distinguish between the different paths being simultaneously embedded, which is unrelated to the coloring of the vertices characterizing the mappings between the five paths. In this case of distinct colors, this is equivalent to the problem of simultaneous embedding with mapping.

The tool allows one to easily verify that regardless of the placement of vertices, at least one of the graphs has a crossing, as seen in Fig. 4(b) with only one crossing, the minimum possible. By showing and hiding various edge colors, one can elect to see only a subset of graphs where the crossings occur, as in Fig. 4(c).

Fig. 5 shows a more complex example of five paths on five colors on ten vertices in which each color has two vertices. Here the tool provides a special command to swap the adjacency lists between two vertices of the same color for one of the graphs. As given in Fig. 5(a) a crossing will always occur regardless of placement of vertices. However, swapping the adjacencies of vertices of same color amongst the five graphs in Fig. 5(b) allows one to obtain a simultaneous geometric embedding without crossings in Fig. 5(c).

3 Implementation

GraphSET is a Windows application written in C++. It is a stand-alone application that does not require any third-party libraries and can be launched from `http://graphset.cs.arizona.edu`, where the source code is available.

GraphSET includes other algorithms for graph drawing in addition to those described above. For example, a PQ-tree implementation and a planarity testing algorithm that draws the PQ-trees at each step of the reduction is included; see Fig. 6.

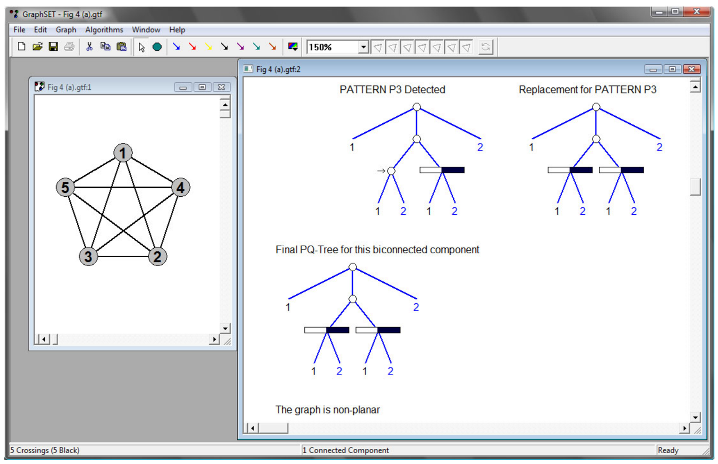


Fig. 6. PQ-tree reduction to show that K_5 is non-planar.

4 Conclusions and Future Work

We presented GraphSET, a tool that has been very valuable to us in the research of various problems related to simultaneous embedding. We hope that other researchers interested in these problems will find this tool useful.

While currently GraphSET includes the recognition and drawing algorithms for ULP trees, in the future we plan to incorporate algorithms for all ULP graphs. The addition of colored tracks and operations to swap vertices between tracks can help in the research of colored simultaneous embedding.

References

1. U. Brandes, C. Erten, J. Fowler, F. Frati, M. Geyer, C. Gutwenger, S. Hong, M. Kaufmann, S. Kobourov, G. Liotta, P. Mutzel, and A. Symvonis. Colored simultaneous geometric embeddings. In *13th Computing and Combinatorics Conference (COCOON)*. To appear in 2007.
2. P. Dietz and D. Leigh. Diamondtouch: a multi-user touch technology. In *14th ACM Symposium on User interface software and technology*, pages 219–226, 2001.

3. C. Erten, S. G. Kobourov, A. Navabia, and V. Le. Simultaneous graph drawing: Layout algorithms and visualization schemes. In *11th Symposium on Graph Drawing (GD)*, pages 437–449, 2003.
4. A. Estrella-Balderrama, J. J. Fowler, and S. G. Kobourov. Characterization of unlabeled level planar trees. In *14th Symposium on Graph Drawing (GD)*, pages 367–369, 2006.
5. E. Gassner, M. Jünger, M. Percan, M. Schaefer, and M. Schulz. Simultaneous graph embeddings with fixed edges. In *32nd International Workshop on Graph-Theoretic Concepts in Computer Science*, 2006.
6. M. Geyer, M. Kaufmann, and I. Vrto. Two trees which are self-intersecting when drawn simultaneously. In *13th Symposium on Graph Drawing*, pages 201–210, 2005.
7. S. G. Kobourov and C. Pitta. An interactive multi-user system for simultaneous graph drawing. In *12th Symposium on Graph Drawing (GD)*, pages 492–501, 2004.