Simultaneous Geometric Graph Embeddings*

Alejandro Estrella-Balderrama[1], Elisabeth Gassner[3], Michael Jünger[4], Merijam Percan[4], Marcus Schaefer[2], and Michael Schulz[4]

[1] Department of Computer Science, University of Arizona
aestrell@cs.arizona.edu
[2] School of CTI, DePaul University
mschaefer@cs.depaul.edu
[3] Institut für Mathematik B, Technische Universität Graz
gassner@opt.math.tu-graz.ac.at
[4] Institut für Informatik, Universität zu Köln
{mjuenger,percan,schulz}@informatik.uni-koeln.de

Abstract. We consider the following problem known as simultaneous geometric graph embedding (SGE). Given a set of planar graphs on a shared vertex set, decide whether the vertices can be placed in the plane in such a way that for each graph the straight-line drawing is planar. We partially settle an open problem of Erten and Kobourov [5] by showing that even for two graphs the problem is **NP**-hard.

We also show that the problem of computing the rectilinear crossing number of a graph can be reduced to a simultaneous geometric graph embedding problem; this implies that placing SGE in **NP** will be hard, since the corresponding question for rectilinear crossing number is a long-standing open problem. However, rather like rectilinear crossing number, SGE can be decided in **PSPACE**.

1 Introduction

Simultaneous drawing deals with the problem of drawing two or more graphs at the same time such that all drawings satisfy specific requirements. When two planar graphs are given, the natural question arises whether a combined drawing leads to two planar drawings [2,5,6,8,9,10]. This problem has been studied in different variations. While most work has been spent on deciding whether different kinds of graphs allow such drawings, this paper focuses on the complexity question. We study the *geometric* version which restricts the problem to straight-line drawings.

Problem:	*Simultaneous Geometric Embedding Problem (SGE)*
Instance:	A set of planar graphs $G_i = (V, E_i)$ on the same vertex set V.
Question:	Are there plane straight-line drawings D_i of G_i such that each vertex is mapped to the same point in the plane in all such D_i?

* Partially supported by Marie-Curie Research Training Network (ADONET) and by the German Science Foundation (JU204/10-1).

S.-H. Hong, T. Nishizeki, and W. Quan (Eds.): GD 2007, LNCS 4875, pp. 280–290, 2007.
© Springer-Verlag Berlin Heidelberg 2007

The complexity of the SGE problem for two graphs is mentioned as an open problem in [5]. We settle part of the problem by showing that it is **NP**-hard. It remains open whether the problem lies in **NP**, but we show by a comparison to the rectilinear crossing number and the existential theory of the real numbers that settling the complexity of SGE will be hard, since determing the complexity of calculating the rectilinear crossing number is a long-standing open problem. Our result is related to an earlier paper, in which we showed that deciding the simultaneous embeddability with fixed edges is **NP**-complete for *three* graphs (Gassner et al. [8]).

It is easy to see that SGE is non-trivial; that is, there are two planar graphs without a simultaneous geometric embedding. More surprisingly, there are even two trees that cannot be simultaneously embedded geometrically [9].

2 NP-Hardness Proof

Theorem 1. *Deciding whether two graphs have a simultaneous geometric embedding is* **NP***-hard.*

Proof. We show that there exists a polynomial transformation from 3SAT, which is well-known to be **NP**-complete, to SGE for two planar graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$.

Problem:	*3-Satisfiability Problem (3SAT)*
Instance:	A CNF-system with a set U of boolean variables and a set C of clauses over U such that each clause in C has exactly three literals.
Question:	Is there a satisfying truth assignment for U?

Given an instance of 3SAT, we construct an instance (G_1, G_2) of SGE. Then we prove that the instance of 3SAT is satisfiable if and only if there exists a simultaneous geometric embedding of (G_1, G_2).

Construction: Let $U = \{u_1, u_2, \dots, u_n\}$ be the variable set and $C = \{c_1, c_2, \dots, c_m\}$ be the clause set where $c_j = (l_1^j \vee l_2^j \vee l_3^j)$ for literals $l_i^j = u_h$ or $l_i^j = \bar{u}_h$ for some variable u_h ($j \in \{1, 2, \dots, m\}$, $h \in \{1, 2, \dots, n\}$, $i \in \{1, 2, 3\}$). The 3SAT formula f can then be written $f = c_1 \wedge c_2 \wedge \dots \wedge c_m$.

For our construction we assume an ordering of the clauses, say $(c_1, c_2, \dots, c_m)$. Furthermore we choose an order of the three literals in each clause c_j and hence get an order of all literals in the following way $(l_1^1, l_2^1, l_3^1, l_1^2, \dots, l_3^m)$.

For each clause c_j we define a *clause box* by introducing vertices $r_1^j, \dots, r_7^j$, $y^{1,j}$, $y^{2,j}$, $y^{3,j}$. These vertices are connected by edges of E_1 (solid) and E_2 (dashed) such as shown in Figure 1.

Next, we introduce two global vertices R_1 and R_2. We add an edge (R_1, R_2) to both graphs G_1 and G_2. Furthermore, R_1 is connected to the clause box of each clause c_j by edges (R_1, r_i^j) in $E_1 \cap E_2$ with $i = 2, \dots, 6$. We also connect R_2 to the clause box by edges (R_2, r_1^j) and (R_2, r_7^j) in E_1.

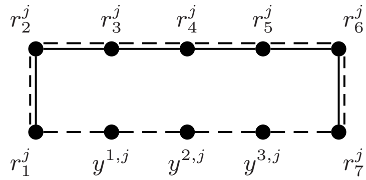

Fig. 1. The clause box of clause c_j. Edges of G_1 are solid and edges of G_2 are dashed.

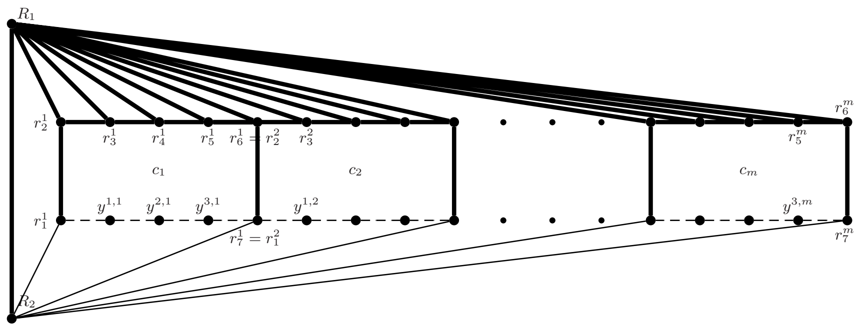

Fig. 2. The figure shows all vertices and all edges constructed so far. Edges which belong to both E_1 and E_2 are drawn bold and solid, edges of $E_1 \backslash E_2$ are thin and solid while edges of $E_2 \backslash E_1$ are dashed.

To make the construction more rigid we glue together neighboring clause boxes. This is done by identifying r_2^{j+1} with r_6^j and r_1^{j+1} with r_7^j for $j = 1, 2, \ldots, m-1$.

Figure 2 gives an idea of the construction so far. Notice that the graph given by the edges in E_1 is a subdivision of a triconnected graph which will be used later in the proof. Its planar embedding is unique up to a homomorphism of the plane.

For every literal l_i^j with $i = 1, 2, 3$, $j = 1, 2, \ldots, m$, we define a *literal gadget* that consists of thirteen vertices and eighteen edges in E_1 and fifteen edges in E_2 as shown in Figure 3. Notice that the edges in E_1 of each literal gadget are a subdivision of a triconnected graph. The only two possible embeddings are shown in Figure 3.

From now on in all figures the edges in E_1 are represented by solid lines while the edges in E_2 are drawn dashed.

Furthermore, we define edge sets that link all literal gadgets that belong to the same variable u_h. Let $l_{i_1}^{j_1}$, $l_{i_2}^{j_2}$, …, $l_{i_{\omega_h}}^{j_{\omega_h}}$ be the set of all literals that belong to variable u_h, that is either $l_{i_\alpha}^{j_\alpha} = u_h$ or $l_{i_\alpha}^{j_\alpha} = \bar{u}_h$. Assume that these literals are given in the order defined above. Then we will link the gadgets of each pair of literals neighbored in this ordered list by edges in E_2 in the following way:

Let $l_{i_k}^{j_k}$ and $l_{i_{k+1}}^{j_{k+1}}$ with $k \in \{1, 2, \ldots, m-1\}$ be two literals neighbored in the ordered list. We add three edges in E_2. Their endpoints depend on the fact whether

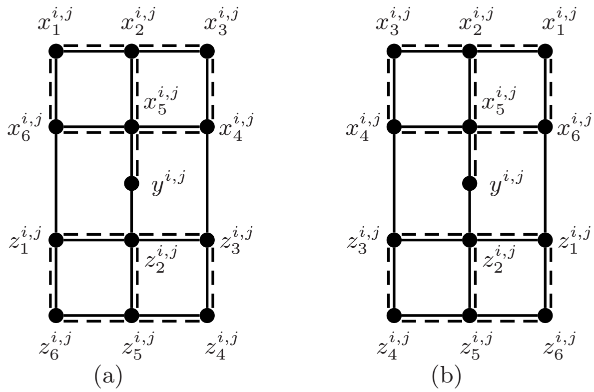

Fig. 3. Literal gadget for l_i^j with corresponding variable u_h. The edges in E_1 are solid and those in E_2 are dashed. The two different drawings (a) and (b) will become important later.

the two literals are negated or unnegated. If both literals are negated or both are unnegated, then we add the three edges $(z_1^{i_k,j_k}, z_6^{i_{k+1},j_{k+1}})$, $(z_2^{i_k,j_k}, z_5^{i_{k+1},j_{k+1}})$, $(z_3^{i_k,j_k}, z_4^{i_{k+1},j_{k+1}})$. If one of the literals is negated and one is unnegated, we add the three edges $(z_1^{i_k,j_k}, z_4^{i_{k+1},j_{k+1}})$, $(z_2^{i_k,j_k}, z_5^{i_{k+1},j_{k+1}})$, $(z_3^{i_k,j_k}, z_6^{i_{k+1},j_{k+1}})$ to graph G_2. For an example with three literals ($\omega_h = 3$) the linking edges are visualized in Figure 4.

For each clause we define a *clause gadget* consisting of three literal gadgets, the clause box and some additional vertices and edges. Let c_j be a clause with literals l_1^j, l_2^j and l_3^j. Notice that the three literal gadgets are already connected to the clause box using the vertices $y^{i,j}$ with $i = 1, 2, 3$. Further connections are given by the additional edges $(r_3^j, x_2^{1,j})$, $(r_4^j, x_2^{2,j})$ and $(r_5^j, x_2^{3,j})$ in E_2. We also add two vertices s^j, t^j and connect them to the literal gadgets via the new edges $(x_3^{1,j}, s^j) \in E_2$, $(s^j, x_1^{2,j})$, $(x_3^{2,j}, t^j) \in E_1$ and $(t^j, x_1^{3,j}) \in E_2$. A possible simultaneous embedding of a clause gadget is shown in Figure 5.

In order to connect the clause gadget to the global vertex R_2 we add vertices w^j, $w^{1,j}$, $w^{2,j}$ and $w^{3,j}$ and connect them to vertices R_2, $z_5^{1,j}$, $z_5^{2,j}$ and $z_5^{3,j}$ and to each other as shown in Figure 5.

This completes the construction.

1. Assume that the 3SAT-instance is satisfiable. Thus we can fix a true/false-assignment of the variables that satisfies the given formula and we construct an instance of SGE as explained above. We prove that there exists a simultaneous geometric embedding of the constructed instance. We say that a variable u makes a clause c `true` if either u is a literal in c and $u =$ `true` or if $\bar{u}$ is a literal in c and $u =$ `false`. Since the instance of 3SAT is satisfiable there exists at least one variable u in each clause c that makes c `true`. If variable u makes its clause `true` we draw the corresponding literal gadget as shown in Figure 3 (a). Otherwise we draw the gadget as shown in Figure 3 (b). The clause gadgets are drawn side by side in their specific ordering with the global vertices R_1 and R_2 being

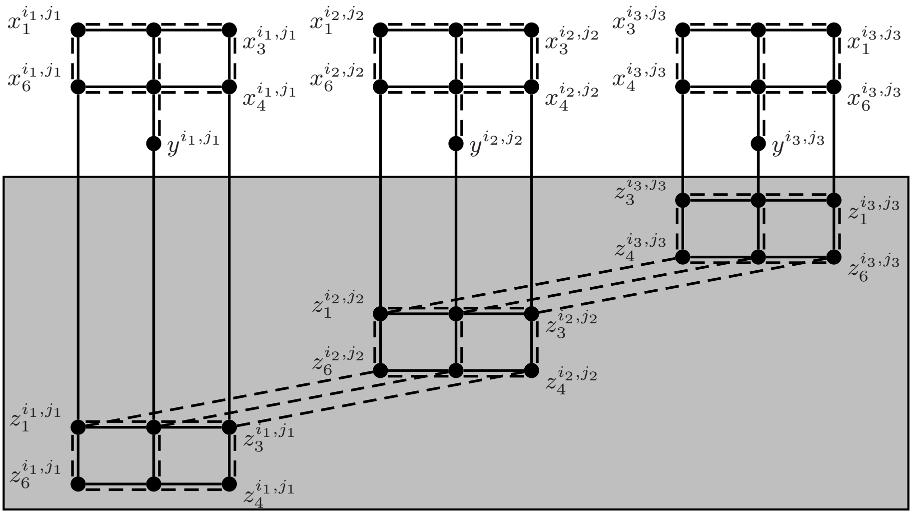

Fig. 4. All literal gadgets that belong to the same variable u_h are linked with edges in E_2. Here, the first two gadgets belong to an unnegated literal u_h whereas the third belongs to a negated literal $\bar{u}_h$.

positioned at the outer face as shown in Figure 2. Furthermore, the x-vertices of each literal gadget lie inside the clause box of its corresponding clause and the z-vertices lie outside. Moreover, every variable u gets its own horizontal region for the z-vertices to avoid crossings of linking edges of different variables. In Figure 4 the horizontal level is marked gray. Linking edges belonging to a different variable are either positioned above or below this region.

Consider now different literal gadgets corresponding to one variable u. Either all the unnegated or all the negated literals (if there exist such literals) make their clauses true but not both. But that is sufficient for the linking edges to be drawn without crossings (not counting crossings between an edge of G_1 and an edge of G_2) as shown in Figure 4.

It remains to show that we can draw the edges inside the clause gadgets without crossings of edges of the same graph.

Consider clause c_j with literals l_1^j, l_2^j and l_3^j and corresponding variables u_l, u_m, u_r. If u_l makes c_j `true`, there exists a simultaneous geometric embedding. See Figure 6 for the case where u_l is the only variable that makes c_j `true`. Simple modifications yield a simultaneous embedding for the case where u_l is not the only variable that makes c_j `true`. Due to symmetry an analogous drawing can be found for the case where u_r makes c_j `true`.

Finally, if u_m makes c_j `true`, we can find a simultaneous embedding as shown in Figure 5. Hence, we have found a simultaneous geometric embedding of the constructed instance.

2. Now assume that we are given a 3SAT-formula and the constructed SGE instance allows a simultaneous geometric embedding. We show that we can find a satisfying truth assignment for the 3SAT-instance.

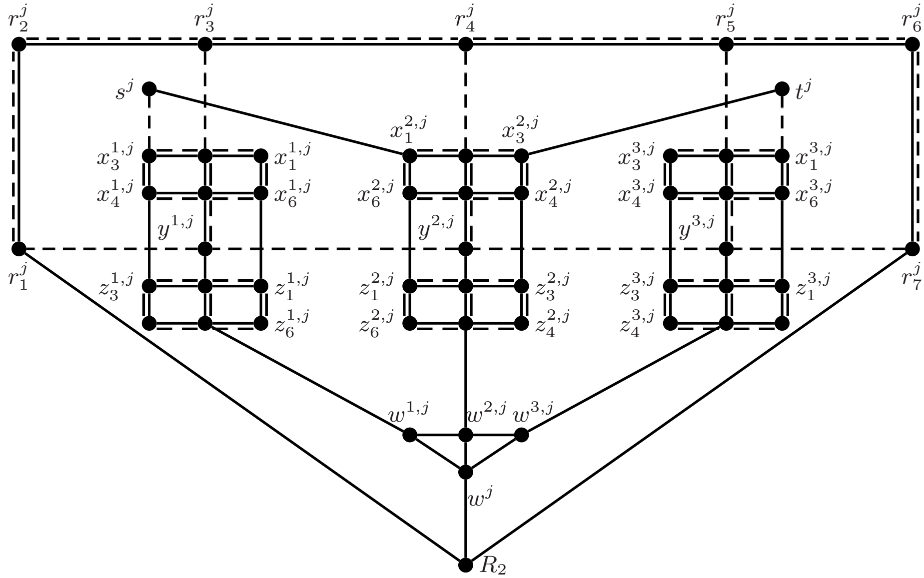

Fig. 5. Clause gadget for clause c_j plus global vertex R_2

Notice that the subgraph of G_1 shown in Figure 2 is a triconnected subdivision. Consequently, it has a unique combinatorial embedding up to homomorphisms of the plane. We choose the planar embedding with the edge (R_1, R_2) on the boundary of the outer face such that the cycle $(R_1, r_2^1, r_1^1, R_2, r_7^m, r_6^m)$ has the same order as visualized in Figure 2.

Observe that each literal gadget in the construction has one of exactly two possible planar embeddings shown in Figure 3. Let $l_{i_1}^{j_1}$, $l_{i_2}^{j_2}$, …, $l_{i_{\omega_h}}^{j_{\omega_h}}$ be the set of all literals that belong to variable u_h. Then due to the edges in E_2 shown in Figure 4 all unnegated literals of u_h have the same embedding and all negated literals have just the opposite embedding. We assign the value `true` to variable u_h if the ordering for unnegated literals is the same as in Figure 3 (a) and `false` otherwise.

For each literal l_i^j in each clause c_j the vertex $y^{i,j}$ lies on the boundary of the clause box. The edge $(r_3^j, x_2^{1,j})$ is not allowed to cross any of the edges incident to global vertex R_1 (which is positioned outside the clause box). Hence $x_2^{1,j}$ and thus all vertices $x_i^{1,j}$, with $i = 1, \ldots, 6$, have to lie within the clause box. With similar arguments the x-vertices of l_2^j and l_3^j lie within the clause box. But now the vertices s^j and t^j must lie within the clause box which is surrounded by edges in E_2.

As soon as a literal gadget l_i^j is connected to a literal gadget of the same variable (see Figure 4) the vertices $z_k^{i,j}$, with $k = 1, \ldots, 6$, lie outside the corresponding clause box. This is particularly the case for all literal gadgets that belong to a clause which is not `true`.

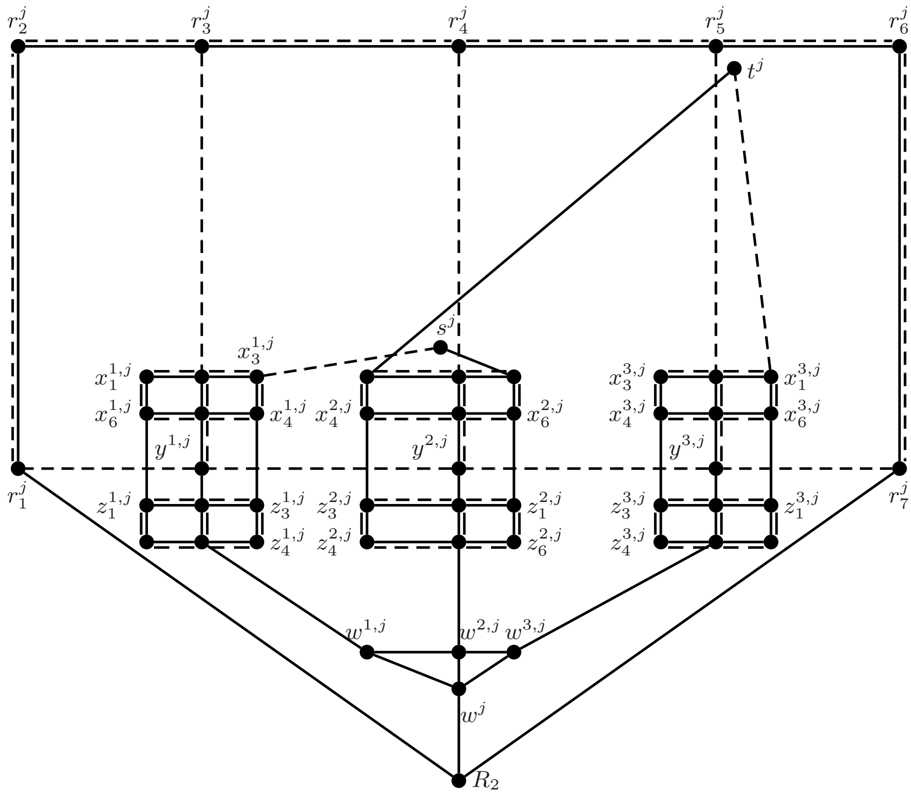

Fig. 6. SGE of the clause gadget when u_l is the only variable that makes c_j `true`

Assume that there exists a clause c_j that is not `true`. Since no variable makes c_j `true` all gadgets are of the form in Figure 3 (b). This case is shown in Figure 7.

Notice that in Figure 7 vertex s_j must be placed in the light gray area as vertex $x_3^{1,j}$ lies in this area. Otherwise the edge $(x_3^{1,j}, s_j) \in E_2$ crosses one edge of the cycle that surrounds the gray area, which is a contradiction. With similar arguments t_j lies inside the dark gray area on the right of this figure. Hence the edge pair $(r_5^j, x_2^{3,j})$ and $(s_j, x_1^{2,j})$ or the edge pair $(r_3^j, x_2^{1,j})$ and $(x_3^{2,j}, t_j)$ must cross twice in order to avoid a crossing of two edges of the same graph. But this is not possible in a straight-line drawing and leads to a contradiction to the assumption that clause c_j is not `true`. Thus all clauses are `true` and hence we have found a satisfying truth assignment. □

3 Simultaneous Straight-Line Drawings and the Rectilinear Crossing Number

In this section we discuss the relationship between simultaneous geometric embeddings and two famous problems, the rectilinear crossing number and

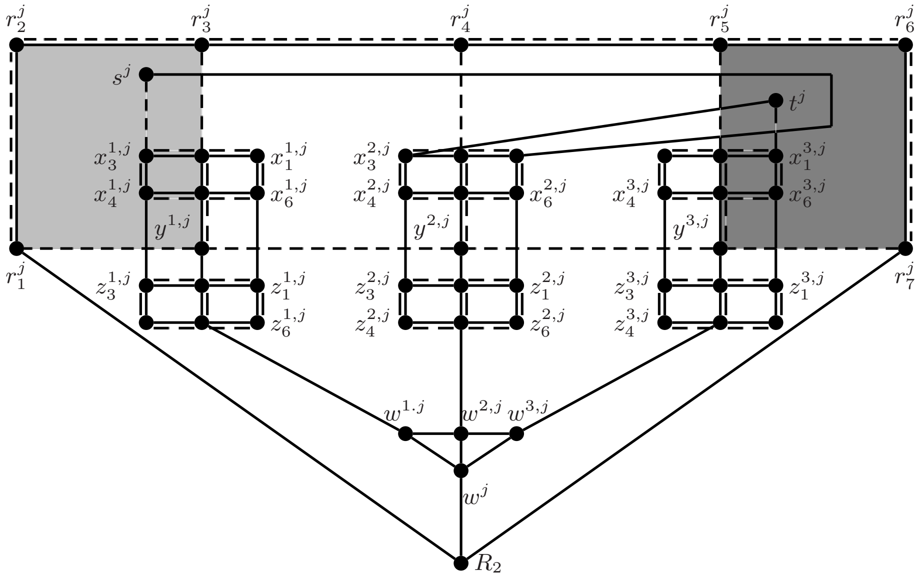

Fig. 7. If a clause is `false` then there exist two edges in the corresponding clause gadget that cross twice

existential theory of the reals. We show, that the complexity of SGE can be placed in between these two problems.

Problem:	*Rectilinear Crossing Number Problem (RCR)*
Instance:	A graph G.
Question:	What is the minimum number of crossings in a straight-line drawing of G?

RCR is well-known to be **NP**-hard [7,1]. We will show that RCR reduces to SGE via **NP**-many-one reductions, which are many-one reductions computed by an **NP**-machine rather than a polynomial time machine:

Theorem 2. *RCR* **NP***-many-one reduces to SGE for an unbounded number of graphs.*

Proof. Let $G = (V, E)$ be a graph. Guess k pairs of edges that are the potential crossing pairs and let M be the set of these edge pairs.

We define graphs $G_{e,f} = (V, E_{e,f})$ with $E_{e,f} = \{e, f\}$ for each pair of edges e and f which is not in M. If there exist an edge d which is not part of any of the new graphs $G_{e,f}$ we define a graph $G_d = (V, E_d)$ with $E_d = \{d\}$.

Notice, that each edge (and each vertex) has been added to at least one graph $G_{e,f}$ or G_d. Furthermore, if one of the graphs $G_{e,f}$ contains two edges e and f they are not allowed to cross in a straight-line drawing as this pair is not one of the k guessed pairs. Thus the decision problem whether G can be drawn straight-line with only edge-crossings in M is equivalent to the problem of finding a simultaneous geometric embedding of the graphs $G_{e,f}$ and G_d. □

Since **NP** is closed under **NP**-many-one reductions, placing SGE in **NP** has immediate consequence for RCR:

Corollary 1. *If SGE lies in* **NP** *then RCR lies in* **NP**.

Since placing RCR in **NP** is a long-standing open problem, we should not expect any easy resolution of the complexity of SGE [3, pg. 389].

Next, we will show that SGE can be expressed in the language of the existential theory of the reals. More, formally, SGE reduces to $\mathbb{R}_\exists$, the set of existential first-order sentences true over the real numbers.

Problem: *Existential Theory of the Real Numbers* ($\mathbb{R}_\exists$)
Instance: An expression of the form

$$(\exists x_1 \in \mathbb{R}) \ldots (\exists x_n \in \mathbb{R})\ P(x_1, \ldots, x_n)$$

where P is a quantifier-free Boolean formula with atomic predicates of the form $g(x_1, \ldots, x_n) \Delta 0$ where g is a real polynomial and $\Delta \in \{>, =\}$. Atomic predicates can be combined using $\vee$, $\wedge$ and $\neg$.
Question: Is the given formula true?

Theorem 3. *There exists a polynomial transformation from SGE to* $\mathbb{R}_\exists$.

Proof. Let $G_1 = (V, E_1), \ldots, G_k = (V, E_k)$ be an instance of SGE. Edge pairs $\{e, f\}$ belonging to the same graph G_i are not allowed to cross; we call such a pair a *forbidden* pair. We define the graph $G = (V, E)$ by $E := \bigcup_{i=1,\ldots,k} E_i$.

We construct an instance of $\mathbb{R}_\exists$ in the following way. For each vertex $v \in V$ we let two variables $x_v, y_v \in \mathbb{R}$ represent the coordinates of the vertex in the final drawing (which leads to the embedding that we are looking for). An edge $(u, v) \in E$ is then represented by the set of points $(x_u + t(x_v - x_u), y_u + t(y_v - y_u))$ where $t \in [0, 1]$.

We need to write constraints ensuring that the resulting drawing of G is good. In particular, we have to guarantee that no two vertices coincide, that no edge contains a vertex other than its endpoints, and that no two forbidden edges intersect.

The constraints are all of the same form: two geometric objects are apart from each other; we express this by requiring there to be a line separating them. For example, for an edge e between points $u = (x_u, y_u)$ and $w = (x_w, y_w)$ and a vertex v at (x_v, y_v) we can use the formula $A(v, e)$:

$$\begin{array}{ll} (\ y_u > a_{v,e} x_u + b_{v,e} & \wedge \\ \ \ y_w > a_{v,e} x_w + b_{v,e} & \wedge \\ \ \ y_v < a_{v,e} x_v + b_{v,e}\ ) & \vee \\ (\ y_u < a_{v,e} x_u + b_{v,e} & \wedge \\ \ \ y_w < a_{v,e} x_w + b_{v,e} & \wedge \\ \ \ y_v > a_{v,e} x_v + b_{v,e}\ ). & \end{array}$$

Then $A(v, e)$ is true if and only if v and e lie on opposite sides of the line $y = a_{v,e}x + b_{v,e}$, that is, if v does not lie on e. Similarly, we can write formulas $B(e, f)$ that express that e and f do not intersect and $C(u, v)$ expressing that u and v are distinct.

Define

$$A := \bigwedge_{v \in V, e \in E, v \notin e} (\exists a_{v,e}, b_{v,e} \in \mathbb{R})\ A_{v,e},$$

$$B := \bigwedge_{(e,f) \in X} (\exists a_{e,f}, b_{e,f} \in \mathbb{R})\ B_{e,f},$$

$$C := \bigwedge_{u,v \in V} (\exists a_{u,v}, b_{u,v} \in \mathbb{R})\ C_{u,v},$$

where we let X be the set of forbidden edge pairs.

Let $V = \{v_1, \ldots, v_n\}$ and let (x_i, y_i) be the coordinates of vertex v_i for $i = 1, \ldots, n$, then

$$(\exists x_1, y_1, \ldots, x_n, y_n \in \mathbb{R})\ A \wedge B \wedge C$$

expresses that there exists a good straight-line drawing of G in which no forbidden pair of edges crosses. The drawing of G gives rise to a set of drawings for each graph G_i (by deleting all other edges) and thus to a simultaneous geometric embedding. As the forbidden edge pairs do not cross, each graph G_i has a planar drawing.

Finally, note that the formula can easily be brought into the normal form required for $\mathbb{R}_\exists$. □

Since it is known that $\mathbb{R}_\exists$ can be decided in **PSPACE** [4], we can draw the following conclusion about the complexity of SGE:

Corollary 2. *SGE, for an arbitrary number of graphs, is* **NP***-hard and lies in* **PSPACE**.

References

1. Bienstock, D.: Some provably hard crossing number problems. Discrete Comput. Geom. 6(5), 443–459 (1991)
2. Brass, P., Cenek, E., Duncan, C.A., Efrat, A., Erten, C., Ismailescu, D., Kobourov, S.G., Lubiw, A., Mitchell, J.S.B.: On simultaneous planar graph embeddings. In: Dehne, F., Sack, J.-R., Smid, M. (eds.) WADS 2003. LNCS, vol. 2748, pp. 243–255. Springer, Heidelberg (2003)
3. Brass, P., Moser, W., Pach, J.: Research problems in discrete geometry. Springer, New York (2005)
4. Canny, J.: Some algebraic and geometric computations in pspace. In: STOC 1988. Proceedings of the twentieth annual ACM symposium on Theory of computing, pp. 460–469 (1988)
5. Erten, C., Kobourov, S.G.: Simultaneous embedding of planar graphs with few bends. In: Pach, J. (ed.) GD 2004. LNCS, vol. 3383, pp. 195–205. Springer, Heidelberg (2005)

6. Frati, F.: Embedding graphs simultaneously with fixed edges. In: Kaufmann, M., Wagner, D. (eds.) GD 2006. LNCS, vol. 4372, pp. 108–113. Springer, Heidelberg (2007)
7. Garey, M.R., Johnson, D.S.: Crossing number is NP-complete. SIAM Journal on Algebraic and Discrete Methods 4(3), 312–316 (1983)
8. Gassner, E., Jünger, M., Percan, M., Schaefer, M., Schulz, M.: Simultaneous graph embeddings with fixed edges. In: Fomin, F.V. (ed.) WG 2006. LNCS, vol. 4271, pp. 325–335. Springer, Heidelberg (2006)
9. Geyer, M., Kaufmann, M., Vrt'o, I.: Two trees which are self-intersecting when drawn simultaneously. In: Healy, P., Nikolov, N.S. (eds.) GD 2005. LNCS, vol. 3843, pp. 201–210. Springer, Heidelberg (2006)
10. Di Giacomo, E., Liotta, G.: A note on simultaneous embedding of planar graphs. In: 21st European Workshop on Comp.Geometry, pp. 207–210 (2005)