Recognizing String Graphs in **NP**.

Marcus Schaefer
Department of Computer Science
DePaul University
243 South Wabash
Chicago, Illinois 60604, USA
mschaefer@cs.depaul.edu

Eric Sedgwick
Department of Computer Science
DePaul University
243 South Wabash
Chicago, Illinois 60604, USA
esedgwick@cs.depaul.edu

Daniel Štefankovič
Department of Computer Science
University of Chicago
1100 East 58th Street
Chicago, Illinois 60637, USA
stefanko@cs.uchicago.edu

October 8, 2001

Abstract

A *string graph* is the intersection graph of a set of curves in the plane. Each curve is represented by a vertex, and an edge between two vertices reflects that the corresponding curves intersect. We show that string graphs can be recognized in **NP**. The recognition problem was not known to be decidable until very recently, when two independent papers established exponential upper bounds on the number of intersections needed to realize a string graph [PT01, SŠ01]. These results implied that the recognition problem lies in **NEXP**. In the present paper we improve this by showing that the recognition problem for string graphs is in **NP**, and therefore **NP**-complete, since Kratochvíl [Kra91b] showed that the recognition problem is **NP**-hard. The result has consequences for the computational complexity of problems in graph drawing, and topological inference.

1 Strings, Drawings, and Diagrams

A *string graph* is the intersection graph of a set of curves in the plane. A *curve*, or *string*, is a set homeomorphic to $[0, 1]$. Given a collection of curves $(C_i)_{i\in I}$ in the plane, the corresponding intersection graph is $(I, \{\{i, j\} : C_i \text{ and } C_j \text{ intersect}\})$. The *size* of a collection of curves is the number of intersection points (we assume that no three curves intersect in the same point). A graph isomorphic to the intersection graph of a collection of curves in the plane is called a *string graph*.

The string graph problem is the problem of recognizing string graphs. The problem made its first explicit appearance in a 1966 paper by Sinden [Sin66] on circuit layout, although a similar question had been suggested earlier by Benzer [Ben59] on genetic structures. The string graph problem was introduced to the combinatorial community by Ron Graham in 1976 at a conference.

From a combinatorial point of view we are interested in $c_s(G)$, the smallest number of intersections of a set of curves realizing a string graph. For graphs G that are not string graphs, we let $c_s(G)$ be infinity. Define $c_s(n) = \max\{c_s(G) : G$ is a string graph on n vertices$\}$. A computable upper bound on $c_s(n)$ would imply decidability of the string graph problem. In 1991 Kratchovíl and Matoušek [KM91] showed rather surprisingly, that $c_s(n) \geq 2^{cn}$ for some constant c, and conjectured that $c_s(n) \leq 2^{cn^k}$ for some k. The papers by Pach and Tóth [PT01], and Schaefer and Štefankovič [SŠ01] established upper bound of this form, implying decidability of the string graph problem in nondeterministic exponential time.

The string graph problem is related to a graph drawing problem which we will need. Given a graph $G = (V, E)$ and a set $R \subseteq \binom{E}{2} = \{\{e, f\} : e, f \in E\}$ on E, we call a drawing D of G in the plane a *weak realization* of (G, R) if only pairs of edges which are in R are allowed to intersect in D (they do not have to intersect, however). In this case we call (G, R) *weakly realizable*. We say that D is a *realization* of G if exactly the pairs of edges in R intersect in D.[1] Let us define $c_w(G, R)$ as the smallest number of intersections in a weak realization of (G, R), $c_w(G) = \max\{c_w(G, R) : (G, R)$ has a weak realization$\}$, and $c_w(m) = \max\{c_w(G) : G$ has m edges$\}$.

The string graph problem can be reduced in polynomial time to the weak realizability problem [MNT88, Kra91b]. The reduction is as follows. Given a graph $G = (V, E)$, let $G' = (V \cup E, \{\{u, e\} : u \in e \in E\})$, and $R = \{\{\{u, e\}, \{v, f\}\} : \{u, v\} \in E\}$. Then G is a string graph if and only if (G', R) is weakly realizable.

In Theorem 4.4 we show that the weak realizability problem is in **NP**. Because of the reduction of the string graph problem to the weak realizability problem, and Kratchovíl's proof of **NP**-hardness of the string graph problem [Kra91b] this implies the following corollary.

Corollary 1.1 *The string graph problem is* **NP***-complete.*

The string graph problem is also related to Euler (or Venn) diagrams, and through these to topological inference. Given a specification of the relationships of concepts, such as "some A is B, some B is C, but no A is C", we can ask whether there is a diagram illustrating the relationship of the concepts (regions). In this particular case Figure 1 does illustrate this situation.

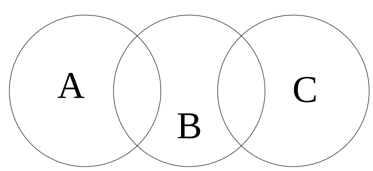

Figure 1: Some A is B, some B is C, but no A is C.

This problem is polynomial-time equivalent to the string graph problem, even in the more general case in which we allow more refined predicates to describe the relationship between regions (see [SŠ01] for detailed definitions and the reductions). Hence we also get the following result.

Corollary 1.2 *The existential theory of diagrams is* **NP***-complete.*

For the proof of our main theorem, Theorem 4.4, the same approach as in our earlier paper [SŠ01] proves successful: we reinterpret the problem as a problem over words. The

[1] Kratochvíl [Kra98, Kra91a, Kra91b] calls (G, R) an *abstract topological graph*, and uses the word *feasible* for weakly realizable. A drawing with intersections is also called an *immersion*.

necessary background material on words and word equations is covered in Section 2. The topological aspects of the proof are covered in Section 3.

2 Word Equations

Let Σ be an alphabet of symbols and Θ be an alphabet of variables. The alphabets Σ and Θ are disjoint. A *word equation* $u = v$ is a pair of words $(u, v) \in (\Sigma \cup \Theta)^* \times (\Sigma \cup \Theta)^*$. The *size of the equation* $u = v$ is $|u| + |v|$. A *solution of the word equation* $u = v$ is a morphism $h : (\Sigma \cup \Theta)^* \to \Sigma^*$ such that $h(a) = a$ for all $a \in \Sigma$ and $h(u) = h(v)$ (h being a morphism means that $h(wz) = h(w)h(z)$ for any $w, z \in (\Sigma \cup \Theta)^*$). The *length of the solution* h is $\sum_{x\in\Theta} |h(x)|$.

A *word equation with specified lengths* is a word equation $u = v$ and a function $f : \Theta \to \mathbb{N}$. The solution h has to respect the lengths, i.e. we require $|h(x)| = f(x)$ for all $x \in \Theta$.

Let w be a word in Σ^*. Let $w = c_1 f_1 c_2 \dots c_k f_k$ where $c_1 = w[1]$ and f_i is the longest prefix of $f_i c_{i+1} \dots f_k$ which occurs in $c_1 f_1 \dots f_{i-1} c_i$. The *Lempel-Ziv (LZ) encoding* of w is $LZ(w) = c_1[a_1, b_1]c_2 \dots c_k[a_k, b_k]$ where $f_i = w[a_i...b_i]$. The *size of the encoding* is $|LZ(w)| = k(\log |w| + \log |\Sigma| + 1)$. Note that some words can be compressed exponentially.

Let $h : (\Sigma \cup \Theta)^* \to \Sigma^*$ be a solution of an equation $u = v$. The LZ encoding of h is the sequence of LZ encodings of $h(x)$ for all $x \in \Theta$. The size of the encoding is $|LZ(h)| = \sum_{x\in\Theta} |LZ(h(x))|$. The usefulness of LZ encoding for word equations is demonstrated by following two results.

Theorem 2.1 ([GKPR96]) *Let $u = v$ be a word equation. Given a LZ encoding of a morphism h we can check whether h is a solution of the equation in time polynomial in $|LZ(h)|$.*

Theorem 2.2 ([PR98]) *Let $u = v$ be a word equation with lengths specified by a function f. Assume that $u = v$ has a solution respecting the lengths given by f. Then there is a solution h respecting the lengths such that $|LZ(h)|$ is polynomial in the size of the binary encoding of f and the size of the equation. Moreover, a lexicographically least such solution can be found in polynomial time.*

Given an equation with specified lengths there might be solutions which can not be LZ compressed. However Theorem 2.2 says that there is a solution which can be LZ compressed. In particular if the equation has a unique solution then that solution can be LZ compressed. Note that it is easy to encode several equations into one equation [Lot, Proposition 12.1.8], hence Theorems 2.1 and 2.2 hold for systems of equations as well.

We will need the following two results which easily follow from [GKPR96].

Lemma 2.3 *Given an LZ-encoding $LZ(w)$ of w we can check whether w is a palindrome in time polynomial in $|LZ(w)|$.*

Lemma 2.4 *Given an LZ encoding $LZ(w)$ of w and $a \in \Sigma$, we can compute the number of occurrences of a in w in time polynomial in $|LZ(w)|$.*

3 Computational Topology

In the following let M be an orientable 2-manifold with compact boundary. A simple arc γ such that both its endpoints $\gamma(0), \gamma(1)$ are on the boundary ∂M and the internal points $\gamma(x), 0 < x < 1$ are in the interior $\dot{M}$ is called a *properly embedded arc*. Two properly embedded arcs γ_1, γ_2 are *isotopic* ($\gamma_1 \sim \gamma_2$) if there is a continuous deformation of γ_1 to γ_2 which does not move the endpoints. The *isotopy class* of γ is the set of properly embedded arcs isotopic to γ. Given two properly embedded arcs γ_1, γ_2 the *intersection number* of γ_1 and γ_2 is

$$i(\gamma_1, \gamma_2) = \min_{c_i \sim \gamma_i} |c_1 \cap c_2|.$$

We say that two properly embedded arcs γ_1, γ_2 are *isotopically disjoint* if $i(\gamma_1, \gamma_2) = 0$.

Let γ_1, γ_2 be two properly embedded arcs. A *bigon* B bounded by γ_1, γ_2 is a disc which has exactly two intersections (of γ_1 and γ_2) on the boundary ∂B.

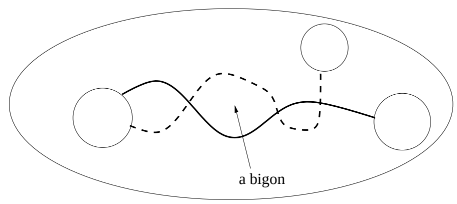

Figure 2: Example of a bigon.

We will need the following standard result (see e.g. [FT91]). Note that the condition that the manifold M is orientable is necessary.

Lemma 3.1 *If two properly embedded arcs γ_1, γ_2 intersect more than $i(\gamma_1, \gamma_2)$ times then they bound a bigon.*

Lemma 3.2 *Let $\gamma_1, \dots, \gamma_n$ be properly embedded arcs. Then there are $c_1, \dots, c_n$ in general position such that $c_i \sim \gamma_i,\ i \in [n]$ and $|c_i \cap c_j| = i(\gamma_i, \gamma_j),\ i, j \in [n]$.*

Proof Let $c_1, \dots, c_n$ be in general position such that $\sum |c_i \cap c_j|$ is minimized. If there are two properly embedded arcs c_i, c_j which intersect more than $i(\gamma_i, \gamma_j)$ times then they bound a bigon B. Let e, f be the sides of the bigon. W.l.o.g. assume that B is the smallest bigon w.r.t. containment. Then any properly embedded arc c which crosses e also crosses f, otherwise B would not be smallest. However then we can reroute the properly embedded arc c_i to decrease the total number of intersections, a contradiction. See Figure 3. $\square$

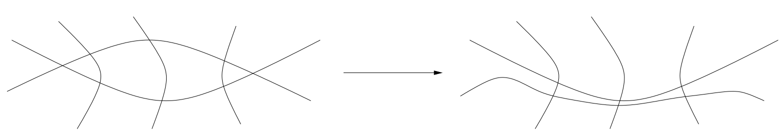

Figure 3: Decreasing the number of intersections.

Let T be a triangulation of the manifold M. Let γ be a properly embedded arc. We say that γ is in general position w.r.t. T if all the intersections with T are transversal and if γ enters a triangle $t \in T$ via edge e then it leaves t via an edge different from e.

Lemma 3.3 *Let T be triangulation of a 2-manifold M. Let γ be a properly embedded arc. Then there is $c \sim \gamma$ which is in general position w.r.t. T.*

Proof Let $c \sim \gamma$ which minimizes the number of intersections with T. If c enters and leaves $t \in T$ through the same edge e then the number of intersections of c with T can be reduced, a contradiction. $\square$

Given a properly embedded arc γ in general position w.r.t. T we can label each edge of the triangulation with the number of intersections of γ with the edge. In each triangle the labels of its edges determine the behavior of the curve inside the triangle up to isotopy (for this we need that γ is in general position w.r.t. T and that γ is not self-intersecting). Hence the numbering on T determines the isotopy class of γ. The size of the representation is the total bitlength of the labels. Each isotopy class has many different representations.

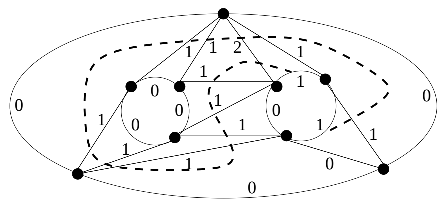

Figure 4: Example of a numbering.

Call a numbering $\ell : E_T \to \mathbb{N}$ of T *valid* if there is a properly embedded arc γ in general position w.r.t. T which intersects each $e \in T$, $\ell(e)$ times.

Let ℓ be a valid numbering. The sum of the labels of edges from $E_T \cap \partial M$ is 2. For each triangle $t \in T$ the labels a, b, c of edges of t satisfy $a + b \geq c, a + c \geq b, b + c \geq a$ and $a+b+c$ is even. These conditions are necessary for validity, but not sufficient. Call a labeling satisfying the conditions *semi-valid*. Any semi-valid labeling defines a properly-embedded arc and a (possible empty) set of closed curves.

We associate the following system of word equations with the triangulation T. The system will emulate behavior of a set of labeled curves on M. We assume that the curves do not intersect. For each oriented edge $(u, v) \in T$ there is a variable $x_{u,v}$ encoding the order in which the curves intersect on (u, v). Let $t \in T$ be a triangle with vertices u, v, w. We add six variables $y_{t,u}$, $y_{t,v}$, $y_{t,w}$, $y_{u,t}$, $y_{v,t}$, $y_{w,t}$. See Figure 5. For example, the variable $y_{t,u}$ encodes the segments of curves between the oriented edges (w, u) and (v, u). We have the following equations $x_{u,v} = y_{u,t}y_{t,v}$, $x_{v,u} = y_{v,t}y_{t,u}$, $x_{v,w} = y_{v,t}y_{t,w}$, $x_{w,v} = y_{w,t}y_{t,v}$, $x_{u,w} = y_{u,t}y_{y,w}$, $x_{w,u} = y_{w,t}y_{t,u}$. Note that if we know then lengths of the x variables, then we can calculate the lengths of the other variables, for example $|y_{u,t}| = (|x_{u,v}| + |x_{u,w}| - |x_{v,w}|)/2$.

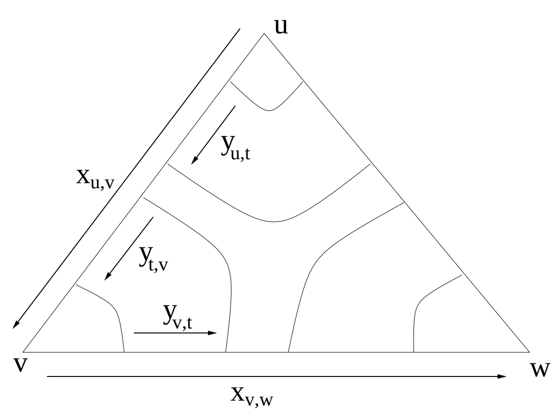

Figure 5: The variables for a triangle t.

Lemma 3.4 *Given a numbering $\ell : E_T \to \mathbb{N}$ we can test whether ℓ is valid in polynomial time.*

Proof We first verify that ℓ is semi-valid, and reject ℓ if it is not.

Let $\Sigma = \{a, b\}$. Take the set of equations associated with T over Σ. For each $e = (u, v) \in E_T$ specify $|x_{u,v}| = \ell(e)$. For each edge $e = (u, v) \in E_T \cap \partial M$ we specify $x_{u,v} = b^{\ell(e)}$.

We claim that if ℓ is valid then the system of equations has a unique solution. Take the properly embedded arc γ which realizes ℓ. Number the intersections of γ with T in the order in which they occur on γ. Each intersection corresponds to a position in some variable. By induction on the number of intersections it follows that each position in every variable is forced to be b.

On the other hand if ℓ is not valid then, because it is semi-valid, there is a solution to the set of word equations. However, a lexicographically smallest solution will now contain as (corresponding to the closed curves, disconnected from the endpoints b). Because of Theorem 2.2 we can compute the lexicographically least solution in polynomial time, and—using Lemma 2.4—check that it does not contain any occurrences of a.

Thus by solving the system we can check whether ℓ is valid. □

Lemma 3.5 *Let γ_1, γ_2 be properly embedded arcs given by valid numberings ℓ_1, ℓ_2. If γ_1, γ_2 do not intersect then we can verify that $i(\gamma_1, \gamma_2) = 0$ in polynomial time.*

Proof Let $\Sigma = \{a, b\}$. Take the set of equations associated with T over Σ. For each $e = (u, v) \in E_T$ specify $|x_{u,v}| = \ell_1(e) + \ell_2(e)$. For each edge $e = (u, v) \in E_T \cap \partial M$ we specify $x_{u,v} = w$ where w represents the order in which γ_1 and γ_2 occur on e.

Let h be a solution of the system. Assume that the number of occurrences of a in $h(x_{u,v})$ is $\ell_1(u, v)$ and $x_{u,v} = x^R_{v,u}$ for all $(u, v) \in E_T$. Then γ_1, γ_2 are isotopically disjoint.

If γ_1, γ_2 are disjoint then the system has a unique solution h. The proof is analogous to the argument in the proof of Lemma 3.4.

Since the solution h is unique we can find $LZ(h)$ in polynomial time by Theorem 2.2. For each $e = (u, v) \in E$ we verify that the number of occurrences of a in $h(x_{u,v})$ is $\ell_1(e)$, and $x_{u,v} = x^R_{v,u}$ using Lemma 2.3, and Lemma 2.4. □

4 Weak Realizability

We need the following bound on the number of intersections in a drawing of a weak realization with the smallest number of intersections.

Theorem 4.1 ([SŠ01]) *Let G be a graph with m edges, $R \subseteq \binom{E}{2}$ such that (G, R) is weakly realizable, and let D be a weak realization of (G, R) with the minimal number of intersections. Then for any edge $e \in G$ there are less than 2^m intersections on the curve realizing e in D.*

To apply our topological results, we have will translate the weak realizability problem into a more topological version. Let $G = (V, E)$ be a graph. Let M be the manifold obtained from the plane by drilling $|V|$ holes, each hole is labeled by a vertex of G. Let $R \subseteq \binom{E}{2}$. A set S of properly embedded arcs on M is called a *weak realization with holes* of (G, R) if for each $e = \{u, v\} \in E$ there is a properly embedded arc in S connecting hole u to hole v and if $(e, f) \notin R$ then the properly embedded arcs e, f are isotopically disjoint.

Given a weak realization D we can drill small holes in place of the vertices to obtain a weak realization with holes. Given a weak realization with holes by Lemma 3.2 there is a weak realization with holes in which for $(e,f) \notin R$ the properly embedded arcs e,f are disjoint. Contracting the holes we obtain a weak realization of (G,R).

Proposition 4.2 *(G,R) is weakly realizable iff there is a weak realization with holes.*

Lemma 4.3 *Let G be a graph with m edges and n vertices. Assume that (G,R) has a weak realization with holes. Let M be the manifold obtained form the plane by drilling $|V|$ holes. Let T be a minimal triangulation of M. Then there is a weak realization with holes of (G,R) such that there are at most 2^{12n+m} intersections on each edge of T.*

Proof Since T is minimal it has $3n$ vertices (3 on each hole). By a simple application of Euler's formula T has $9n-6$ edges.

Consider the following weak realization problem. Graph H has $V_H = V_T \cup V_G$. We have $E_T \cup E_G \subseteq E_H$. Moreover, there are edges connecting $v \in V_G$ to all vertices of T which lie on the hole labeled by v. The pairs P of edges which are allowed to intersect are the following.

- All pairs in R are allowed to intersect.
- For every $v \in G$ fix an edge $e_v \in T$ on the boundary of the hole labeled by v. Then e_v can intersect with edges in E_G going to v.
- Any edge in T which is not on the boundary ∂M can intersect any edge in E_G.

Clearly (G,R) is weakly realizable iff (H,P) is weakly realizable.

Consider the realization of H with the smallest number of intersections. By Theorem 4.1 there is a realization such that on each edge there are at most $2^{|E_H|} \leq 2^{12n+m}$ intersections. $\square$

Theorem 4.4 *The weak realizability problem is in* **NP**.

Proof Let (G,R) be an instance of the weak realizability problem. Assume that (G,R) is weakly realizable. Hence by proposition 4.2 it has a weak realization with holes. Let T be a minimal triangulation of the manifold M. By Lemma 4.3 there is a weak realization with holes in which each edge of T is intersected at most 2^{12n+m} times. Hence for any arc γ and any edge $e \in T$ the binary encoding of the number of intersections of γ with e has size polynomial in n. To verify weak realizability with holes it is therefore sufficient to guess, for each edge of G, a labeling of the edges of T whose numbers are at most 2^{12n+m}. We can then use Lemma 3.5, for any $(e,f) \notin R$ to check in polynomial time that e,f are isotopically disjoint for the set of labelings we guessed. This, by Lemma 3.2, is sufficient to guarantee the existence of a weak realization of (G,R). $\square$

References

[Ben59] S. Benzer. On the topology of the genetic fine structure. *Proceedings of the National Academy of Science*, 45:1607–1620, 1959.

[FT91] B. Farb and B. Thurston. Homeomorphisms and simple closed curves. *unpublished manuscript*, 1991.

[GKPR96] L. Gąsieniec, M. Karpinski, W. Plandowski, and W. Rytter. Efficient algorithms for lempel-ziv encoding. *in Proceedings of SWAT'96, LNCS 1097*, pages 392–403, 1996.

[KM91] Jan Kratochvíl and Jiří Matoušek. String graphs requiring exponential representations. *Journal of Combinatorial Theory, Series B*, 53:1–4, 1991.

[Kra91a] Jan Kratochvíl. String graphs. I. the number of critical nonstring graphs is infinite. *Journal of Combinatorial Theory, Series B*, 52, 1991.

[Kra91b] Jan Kratochvíl. String graphs. II. recognizing string graphs is NP-hard. *Journal of Combinatorial Theory, Series B*, 52, 1991.

[Kra98] Jan Kratochvíl. Crossing number of abstract topological graphs. *Lecture Notes in Computer Science*, 1547:238–245, 1998.

[Lot] M. Lothaire. Algebraic combinatorics on words. To be published by Campbridge University Press, 2002.

[MNT88] J. Matoušek, J. Nešetřil, and R. Thomas. On polynomial time decidability of induced-minor-closed classes. *Comment. Math. Univ. Carolin.*, 29(4):703–710, 1988.

[PR98] Wojciech Plandowski and Wojciech Rytter. Application of lempel-ziv encodings to the solution of words equations. In *Automata, Languages and Programming*, pages 731–742, 1998.

[PT01] János Pach and Géza Tóth. Recognizing string graphs is decidable. In P. Mutzel, editor, *Graph Drawing 2001*, Lecture Notes in Computer Science. Springer, 2001. to appear.

[Sin66] Frank W. Sinden. Topology of thin film circuits. *Bell System Technical Journal*, 45:1639–1662, 1966.

[SŠ01] Marcus Schaefer and Daniel Štefankovič. Decidability of string graphs. In *Proceedings of the 33th Annual ACM Symposium on Theory of Computing (STOC-2001)*, pages 241–246, 2001.