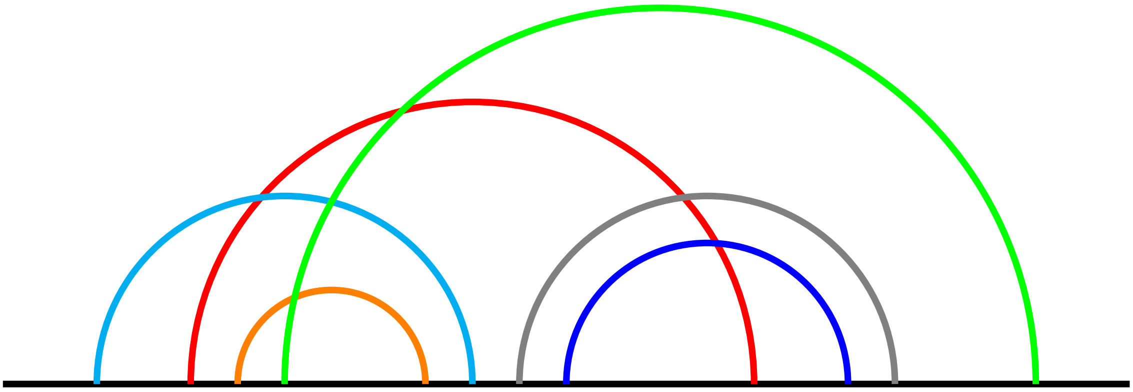
# Geometric Representations of Graphs

Tomáš Turek[1]



April 12, 2024

[1] Many parts of text are taken from the handouts made by Jan Kratochvíl. I also add some of my notes from the lectures and some pictures. Second part is made from my notes from the course lectured by Vít Jelínek. Keep in mind there may be some mistakes. You may visit GitHub.

# Contents

# Part I

# Geometric Representations of Graphs I

# Chapter 1

# Introducing

Firstly we will start by the introduction to the main characters, which are firstly intersection defined graph classes and secondly characterization of chordal graphs.

## 1.1 Intersection defined graph classes

**Definition 1.** *The intersection graph of a set family $\mathcal{A}$ is the graph*

$$IG(\mathcal{A}) = (\mathcal{A}, \{ab : a \neq b, a \cap b \neq \emptyset, a, b \in \mathcal{A}\}).$$

**Definition 2.** *Let $\mathcal{M}$ be a family of sets. We say that a graph $G$ is an intersection graph of (members of) $\mathcal{M}$ if $G$ is isomorphic to the graph $IG(\mathcal{A})$ for some family $\mathcal{A}$ whose all elements belong to $\mathcal{M}$. We write*

$$\mathcal{IG}(\mathcal{M}) = \{IG(\mathcal{A}) : \mathcal{A} \subseteq \mathcal{M}\}.$$

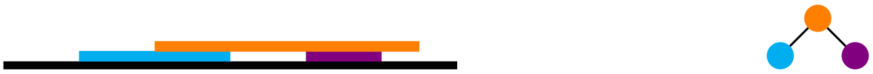
**Observation.** *For every graph $G$ and every set family $\mathcal{M}, G \in \mathcal{IG}(\mathcal{M})$ if and only if there is a mapping $f : V(G) \to \mathcal{M}$ such that $uv \in E(G)$ iff $f(u) \cap f(v) \neq \emptyset$ holds true for all pairs of distinct vertices $u, v$ of $G$.*

**Observation.** *For every family $\mathcal{M}$ (containing at least one nonempty set), it holds that $\mathcal{IG}(\mathcal{M}$ contains all complete graphs and is hereditary (i.e., every induced subgraph of every graph from $\mathcal{IG}(\mathcal{M})$ also belongs to $\mathcal{IG}(\mathcal{M})$).*

### 1.1.1 Examples

In many cases, the members of $\mathcal{M}$ are defined by their geometric shape. And in most of these cases, the members of $\mathcal{M}$ are arc-connected sets in the plane.

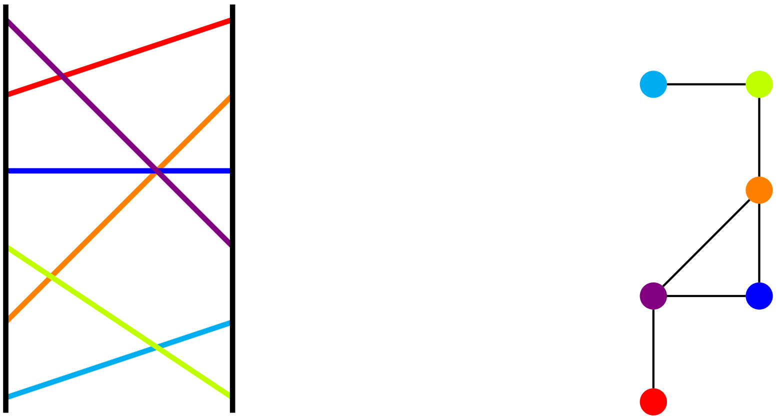
- **Interval graphs** INT $= \mathcal{IG}(\{$intervals on a line$\})$
- **Circle graphs** CIR $= \mathcal{IG}(\{$chords of a circle$\})$
- **Circular arc graphs** CA $= \mathcal{IG}(\{$arcs on a circle$\})$
- **Permutation graphs** PER $= \mathcal{IG}(\{$segments connecting two parallel lines$\})$
- **Function graphs** FUN $= \mathcal{IG}(\{$curves connecting two parallel lines$\})$
- **Polygon circle graphs** PC $= \mathcal{IG}(\{$polygons inscribed in a circle$\})$
- **Segment graphs** SEG $= \mathcal{IG}(\{$straight-line segments in the plane$\})$
- **Convex graphs** CONV $= \mathcal{IG}(\{$convex sets in the plane$\})$
- **String graphs** STRING $= \mathcal{IG}(\{$curves in the plane$\})$



(a) Drawn intervals.

(b) Corresponding graph.

Figure 1.1: Example of a graph from INT class.



(a) Drawn permutations.

(b) Corresponding graph.

Figure 1.2: Example of a graph from PER class.

## 1.2 Chordal graphs

**Definition 3.** *A graph is* ***chordal*** *if it does not contain any cycle of length greater than three as an induced subgraph.*

**Definition 4.** *A vertex $u$ of a graph $G$ is* ***simplicial*** *if $G[N_G(u)]$ is a clique.*

**Definition 5** (PES 🐕)**.** *A* ***perfect elimination scheme*** *for a graph $G$ is a linear ordering $u_1, u_2, \ldots, u_n$ of its vertices such that for every $i, u_i$ is simplicial in the induced subgraph $G[\{u_1, u_2, \ldots, u_i\}]$.*

**Lemma 1.** *Every minimal vertex cut in a chordal graph induces a clique.*

*Proof.* Let $A \subset V(G)$ be a minimal vertex cut, and suppose $u, v$ be two vertices of $A$. These vertices are connected by a path in each component of $G \setminus A$. If $u$ and $v$ were not adjacent, a pair of shortest such paths would give rise to an induced cycle of length greater than 3 in $G$. □

**Lemma 2.** *Every chordal graph, which is not a complete graph, contains two nonadjacent simplicial vertices.*

*Proof.* By induction. If $G$ is a complete graph, the claim of the lemma is fulfilled. If $G$ is not complete, it has a vertex cut, say $A$. Let $B$ be a connected component of $G \setminus A$, and set $G_1 = G[B \cup A]$ and $G_2 = G \setminus B$. By induction hypothesis, each of $G_1, G_2$ is either complete or has two nonadjacent simplicial vertices. Thus each of them has a simplicial vertex which does not belong to $A$. Each of these is then also simplicial in entire $G$, and they are clearly nonadjacent. □

*Corollary.* Every nonempty chordal graph contains a simplicial vertex.

**Definition 6** (Clique-tree decomposition)**.** *A* ***clique-tree decomposition*** *of a graph $G$ is a tree $T = (\mathcal{Q}, F)$, with $\mathcal{Q}$ being the set of all maximal cliques of $G$, such that for every vertex $u \in V(G)$, the subgraph $T[\{Q : u \in Q \in \mathcal{Q}\}]$ is connected.*

**Warning!! The vertex set of a clique-tree decomposition of a graph G is uniquely defined, but not necessarily the edge set!!**

**Theorem 1.** *For any graph $G$, the following statements are equivalent:*

1. *$G$ is chordal,*
2. *$G$ allows a PES.*
3. *$G$ has a clique-tree decomposition, and*
4. *$G$ is an intersection graph of subtrees of a tree.*

*Proof.* "1. $\Rightarrow$ 2." By induction on the number of vertices, using Lemma 2.

"2. $\Rightarrow$ 3." By induction on the number of vertices again. Suppose $G' = G \setminus v_n$ has a clique-tree $T = (\mathcal{Q}', F')$. If $Q' = N_G(v_n) \in \mathcal{Q}'$ , then $Q = N_G[v_n]$ is a maximal clique in $G$, $\mathcal{Q} = (\mathcal{Q}' \setminus \{Q'\}) \cup \{Q\}$ and $T = (\mathcal{Q}, F)$ is a clique-tree for $G$, where $F = (F' \setminus \{Q'P : P \in \mathcal{Q}'\}) \cup \{QP : Q'P \in F'\}$. If, on the other hand, $Q' = N_G(v_n) \notin \mathcal{Q}'$ , then $\mathcal{Q} = \mathcal{Q}' \cup N_G[v_n]$ and $(\mathcal{Q}, F' \cup N_G[v_n]P)$ is a clique-tree for $G$ for any $P \in \mathcal{Q}'$ such that $N_G(v_n) \subset P$.

"3. $\Rightarrow$ 4." Given a clique-tree decomposition $T = (\mathcal{Q}, F)$, define $T_u = T[\{Q : u \in Q \in \mathcal{Q}\}]$ for $u \in V(G)$. Clearly $V(T_u) \cap V(T_v) \neq \emptyset$ iff $u$ and $v$ belong to the same maximal clique of $G$, which happens if and only if $u$ and $v$ are adjacent in $G$.

"4. $\Rightarrow$ 1." Let $G$ be the intersection graph of a collection $\{T_u\}_{u \in V(G)}$ of subtrees of a tree $T$. Suppose $v_1, v_2, \ldots, v_k$ be an induced cycle in $G$, with $k > 3$. Then the subtrees $T_{v_1}$ and $T_{v_3}$ are vertex disjoint, and hence there is an edge $e \in E(T)$ which lies on every path connecting $T_{v_1}$ and $T_{v_3}$ in $T$ . This edge separates $T$ into $T_1$ and $T_2$ such that $T_{v_1}$ and $T_{v_3}$ belong to different components of $T \setminus e$, say, $T_{v_1} \subseteq T_1$ and $T_{v_3} \subseteq T_2$. One can show by induction on $i$ that for every $i \geq 3$, $T_{v_i} \subseteq T_2$. But then $T_{v_k}$ and $T_{v_1}$ must be disjoint, contradicting the assumption that $v_1 v_k \in E(G)$. □

*Corollary.* Chordal graphs are perfect (i.e., $\chi(H) = \omega(H)$ for every induced subgraph $H$ of $G$).

*Proof.* Consider a PES $u_1, u_2, \ldots, u_n$ for $G$ and color the vertices from $u_1$ to $u_n$ by the First Fit Method (we try to use minimal color if we cannot use any of them, create a new color). □

# Chapter 2

# Interval, permutation and function graphs

## 2.1 Interval graphs

**Definition 7.** *A graph is an* ***interval graph*** *if it is isomorphic to the intersection graph of a collection of intervals on a line.*

**Observation.** *Every interval graph has an interval representation in which all of the intervals are closed.*

**Definition 8** (Clique-path decomposition)**.** *A* ***clique-path decomposition*** *of a graph is a clique-tree decomposition in which the underlying tree is a path.*

**Theorem 2.** *For any graph $G$, the following statements are equivalent:*

1. *$G$ is an interval graph,*
2. *$G$ has a clique-path decomposition, and*
3. *$G$ is an intersection graph of subpaths of a path.*

*Proof.* "1. ⇔ 3." is obvious.

"1. ⇒ 2." Assume $I_u, u \in V(G)$ is an interval representation of $G$. Use the fact that intervals on a line have the Helly property, i.e., if any two of a collection of intervals have a nonempty intersection, then all of them have a nonempty intersection. In other words, if $Q_i \in \mathcal{Q}$ is a maximal clique of $G$, then there exists a point $P_i$ which belongs to $\bigcap_{u \in Q_i} I_u$. Moreover, for every $v \notin Q_i, P_i \notin I_v$, since $Q_i$ is a maximal clique. (E.g., the rightmost of the left endpoints of the intervals $I_u, u \in Q_i$ is a good candidate for $P_i$.) Order the cliques of $Q$ as $Q_1, Q_2, \ldots, Q_k$ so that $P_1 < P_2 < \cdots < P_k$. Then the path $Q_1 Q_2 \ldots Q_k$ is a clique-path decomposition of $G$.

"2. ⇒ 3." Given a clique-path decomposition $P = (\mathcal{Q}, F)$, define $P_u = P[\{Q : u \in Q \in \mathcal{Q}\}]$ for $u \in V(G)$. Clearly $V(P_u) \cap V(P_v) \neq \emptyset$ iff $u$ and $v$ belong to the same maximal clique of $G$, which happens if and only $u$ and $v$ are adjacent in $G$. □

## 2.2 Comparability graphs

**Definition 9.** *A graph $G$ is a* ***comparability graph*** *if there exists a partial order $P = (V(G), \leq)$ on the vertex set of $G$ (i.e., an* ***antireflexive***, ***antisymmetric*** *and* ***transitive binary relation****) such that for any two vertices $u, v \in V(G)$, $uv \in E(G)$ if and only if $u \leq v$ or $v \leq u$ (i.e., if $u$ and $v$ are comparable in $P$). The class of comparability graphs will be denoted by $CO$.*

**Observation.** *A graph is a comparability graph if and only if its edges can be transitively oriented.*

**Theorem 3.** *Comparability graphs are perfect.*

*Proof.* Because $G \in CO$ then $\exists P = (V, \leq)$. We create a Haase diagram. Then we may see that cliques are chains and also independent sets are the antichains. Therefore we color every layer with one color and it is the same as the size of the largest clique. □

**Notation.** *If $\mathcal{A}$ is a graph class, the symbol $co - \mathcal{A}$ is used to denote the class containing the complements of the graphs in $\mathcal{A}$.*

**Observation.** *If $\mathcal{A} \subseteq \mathcal{B}$, then $co - \mathcal{A} \subseteq co - \mathcal{B}$.*

**Theorem 4.** *All equivalencies hold:*

1. *FUN = co − CO*
2. *PER = CO ∩ co − CO*
3. *INT = CHOR ∩ co − CO*

*Proof.* 1. "FUN ⊆ co-CO": Given a collection of curves joining two vertical parallel lines (and lying in the stripe between them), for any two non-crossing curves, it is uniquely determined which one lies above the other one (this follows from the Jordan curve theorem), and this gives a transitive orientation of the complement of the intersection graph of this collection.

"co-CO ⊆ FUN": Let $G = (V, E)$ be a graph and let $P = (V, \leq)$ be a partial order which corresponds to a transitive orientation of the complement of $G$. If $d$ is the dimension of $P$, $P$ is the intersection of $d$ linear orders $L_1, L_2, \ldots, L_d$ of $V$. In the plane, draw d distinct parallel (vertical) lines $l_1, l_2, \ldots, l_d$, and on each $l_i$, mark distinct points $P_{iu}, u \in V$ bottom up in the order $L_i$. Consider piece-wise linear curves $c(u) = P_{1u}P_{2u} \ldots P_{du}$, for $u \in V$. If $uv \in E$, $u$ and $v$ are incomparable in $P$, and hence there are indices $i$ and $j$ such that $u <_{L_i} v$ and $v <_{L_j} u$, in other words $P_{iu}$ is below $P_{iv}$, while $P_{ju}$ is above $P_{jv}$. Hence the curves $c(u)$ and $c(v)$ cross somewhere between $l_i$ and $l_j$. If, on the other hand, $uv \notin E$, $uv$ is an edge of the complement of $G$ and hence $u$ and $v$ are comparable in $P$, say, $u \leq v$. But then $u <_{L_i} v$ for every $i = 1, 2, \ldots, d$, and for each $i = 1, 2, \ldots, d - 1$, the curve $c(u)$ lies below the curve $c(v)$ in the stripe between $l_i$ and $l_{i+1}$. Thus $c(u)$ and $c(v)$ are disjoint.

2. Note first that co-PER ⊆ PER. Indeed, given a permutation representation of a graph, swap the order of the endpoints on one of the bounding lines to obtain a representation of the complement of the given graph. Then PER = co-(co-PER) ⊆ co-PER, and hence PER = co-PER.

"PER ⊆ CO ∩ co-CO": Obviously PER ⊆ FUN = co-CO. Then the above small observation implies PER = co-PER ⊆ co-(co-CO) = CO as well.

"CO ∩ co-CO ⊆ PER": Suppose both $G$ and its complement can be transitively oriented, say $\overrightarrow{E_1}$ be a transitive orientation of $G$ and $\overrightarrow{E_2}$ a transitive orientation of the complement $-G$ of $G$. Then $\overrightarrow{E_1} \cup \overrightarrow{E_2}$ is a transitive orientation of the complete graph $K_{V(G)}$ on the vertex set of $G$, i.e., a linear ordering of the vertices of $G$. And so is $\overrightarrow{E_1}^{-1} \cup \overrightarrow{E_2}$. Place the vertices of $G$ on two parallel lines, on one of them in the linear order given by $\overrightarrow{E_1} \cup \overrightarrow{E_2}$, on the other one in the order given by $\overrightarrow{E_1}^{-1} \cup \overrightarrow{E_2}$, and connect the two occurrences of a vertex $u$ by a straight-line segment called $s(u)$, for every vertex $u \in V(G)$. If $uv \in E(G)$, then the pair $u, v$ is ordered differently on the two lines (by $\overrightarrow{E_1}$ on one of them and by $\overrightarrow{E_1}^{-1}$ on the other one) and the segments $s(u), s(v)$ cross each other somewhere between the two lines. If $uv \notin E(G)$, the pair $u, v$ is ordered the same way (by $\overrightarrow{E_2}$) on both of the lines, and thus the segments $s(u)$ and $s(v)$ are disjoint. So $\{s(u)\}_{u \in V(G)}$ is a permutation representation of $G$.

3. "INT ⊆ CHOR ∩ co-CO": Let $\{I(u)\}_{u \in V(G)}$ be an interval representation of a graph $G$. Define a transitive orientation $\overrightarrow{E_2}$ of the non-edges of $G$ by setting $uv \in \overrightarrow{E_2}$ if $\max I(u) < \min I(v)$. Thus $G \in$ co-CO. The fact that $G \in$ CHOR follows from the fact that

$$\text{INT} = \mathcal{IG}(\{\text{connected subgraphs of paths}\}) \subseteq$$

$$\subseteq \mathcal{IG}(\{\text{connected subgraphs of trees}\}) = \text{CHOR}.$$

"CHOR ∩ co-CO ⊆ INT": Let $G$ be a chordal graph which allows a transitive orientation $\overrightarrow{E_2}$ of its non-edges. Define a binary relation $<$ on the set $\mathcal{Q}$ of maximal cliques of $G$ by setting

$$Q < Q' \Leftrightarrow \exists u \in Q \exists v \in Q' : uv \in \overrightarrow{E_2}.$$

**Claim 5.** *The relation $<$ is a partial order on $\mathcal{Q}$.*

*Proof of claim.* We will show all properties of partial order.

- Antireflexivity: Each $Q \in \mathcal{Q}$ is a clique, so there are no two vertices $u, v \in Q$ that would form a non-edge of $G$. Hence $Q \not< Q$.

- Antisymmetry: Suppose for the contrary that there are $u \in Q, v \in Q'$ s.t. $uv \in \overrightarrow{E_2}$, and another pair $x \in Q, y \in Q'$ s.t. $yx \in \overrightarrow{E_2}$. First observe that $u \neq x$ and $v \neq y$ (if $u = x$, the transitivity of $\overrightarrow{E_2}$ would imply $yv \in \overrightarrow{E_2}$, which is impossible; if $v = y$, the transitivity of $\overrightarrow{E_2}$ would imply $ux \in \overrightarrow{E_2}$, which is again impossible). Next observe that both $uy$ and $xv$ must be edges of $G$ (if $uy \notin E(G)$, then either $uy \in \overrightarrow{E_2}$, or $yu \in \overrightarrow{E_2}$, yielding $ux \in \overrightarrow{E_2}$ in the former case and $yv \in \overrightarrow{E_2}$ in the letter one, both contradicting the fact that $Q$ and $Q'$ are cliques of $G$; the case of $xv \notin E(G)$ is analogous). Lastly, we conclude that $G[\{u, v, x, y\}] \simeq C_4$ , contradicting the assumption that $G$ is chordal.
- Transitivity: Suppose $Q < Q' < Q''$ and let $u \in Q, v, x \in Q'$ and $y \in Q''$ be vertices such that $uv, xy \in \overrightarrow{E_2}$. If $v = x$, the transitivity of $\overrightarrow{E_2}$ implies $uy \in \overrightarrow{E_2}$, hence $Q < Q''$. If $v \neq x$, one of $ux$, $vy$ must be a non-edge (otherwise $G[\{u, v, x, y\}]$ would be an induced cycle of length 4, contradicting the assumption that $G$ is chordal). If $ux \notin E(G)$, $ux \in \overrightarrow{E_2}$ and the transitivity of $\overrightarrow{E_2}$ implies $uy \in \overrightarrow{E_2}$. If $vy \notin E(G)$, $vy \in \overrightarrow{E_2}$ and the transitivity of $\overrightarrow{E_2}$ implies $uy \in \overrightarrow{E_2}$. In either case, $Q < Q''$.

□

**Claim 6.** *The relation $<$ is a linear ordering of $\mathcal{Q}$.*

*Proof of claim.* Let $Q \neq Q'$ be two different maximal cliques of $G$. Their maximality implies that none of them is a subset of the other one. Hence there is a vertex $u \in Q$ which does not belong to $Q'$ . If $u$ were adjacent to all vertices of $Q'$, $Q' \cup \{u\}$ would be a clique of $G$, contradicting the maximality of $Q'$ . Hence there is a $v \in Q'$ such that $uv \notin E(G)$. Then either $uv \in \overrightarrow{E_2}$ or $vu\overrightarrow{E_2}$, thus $Q < Q'$ or $Q' < Q$. □

**Claim 7.** *Let $\mathcal{Q} = \{Q_1 < Q_2 < \cdots < Q_k\}$ be the maximal cliques of $G$ ordered by $<$. Then $P_G = (\mathcal{Q}, \{Q_iQ_{i+1} : i = 1, 2, \ldots, k-1\})$ is a clique-path decomposition of $G$, and hence $G \in INT$.*

*Proof of claim.* Indeed $P_G$ is a path whose nodes are the maximal cliques of $G$. It remains to show that vertices of $G$ appear in these cliques consecutively. Suppose $Q < Q' < Q''$ and $u \in Q \cap Q''$. If there were a vertex $v \in Q'$ nonadjacent to $u$, we would have $uv \in \overrightarrow{E_2}$ because of $Q < Q'$ and $vu \in \overrightarrow{E_2}$ because of $Q' < Q''$, contradicting the antisymmetry of $\overrightarrow{E_2}$. Hence $u$ is adjacent to all vertices of $Q'$ , and thus $u \in Q'$ follows from the maximality of $Q'$. □

The last claim proved the theorem as well. □

# Chapter 3

# Interval filament graphs

**Definition 10.** *Given a half-plane $\pi$ with a border-line $l$, an **interval filament** in $\pi$ is a simple curve with endpoints on $l$, its interior lying in $\pi$ and within the stripe determined by lines perpendicular to $l$ and passing through the endpoints of the curve. The class of **interval filament graphs** is $IFA = \mathcal{IG}\{$interval filaments in a half-plane$\}$.*

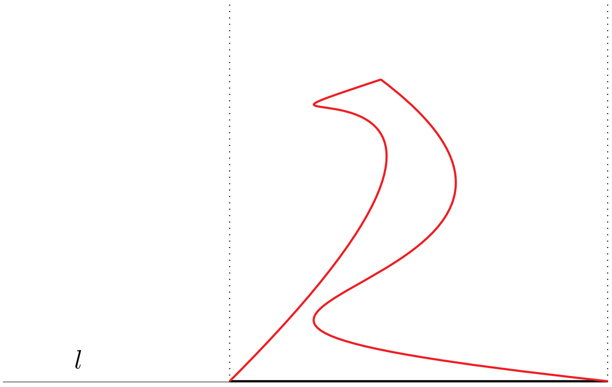


Figure 3.1: An illustration to the definition. The name "interval-filament" comes from the fact that the curve "lives above" the interval that is determined by the endpoints of the filament on the boundary line $l$.

**Comment.** *If the base intervals of two interval-filaments overlap (i.e., they are not disjoint, but none of them is a subinterval of the other one), the filaments necessarily cross each other and the corresponding vertices in the intersection graph are adjacent (cf. the blue and red filaments in Fig. 3.2). On the other hand, if one of the base intervals is included in the other one, their filaments may or may not be disjoint (cf. the blue and yellow filaments for the disjoint case, and the red and green filaments for the non-disjoint one, both in Fig. 3.2).*

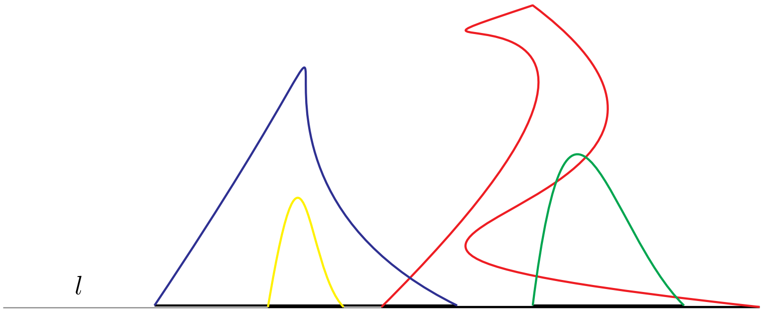


Figure 3.2: An illustration to the possible relative positions of interval-filaments.

**Observation.** *IFA contains INT, CHOR, FUN, CIR, CA, and PC.*

**Definition 11** (class $\mathcal{A}$-mixed)**.** *Let $\mathcal{A}$ be a graph class. A graph $G = (V, E)$ belongs to the class $\mathcal{A}$-mixed if its edge set allows a partition $E = E_1 \cup E_2$ and a transitive orientation $\overrightarrow{E_2}$ of $E_2$ such that for any three vertices $u, v, w \in V(G)$, $uv \in \overrightarrow{E_2}$ and $vw \in E_1$ imply $uw \in E_1$.*

**Theorem 8.** *co-IFA = (co-INT)-mixed.*

*Proof.* "$\subseteq$": Let $G = (V, E) \in$ IFA and let $\{f_u : u \in V\}$ be an interval-filament representation of $G$, with $f_u$ being a filament with its base interval $I_u$ lying on a common line $l$. Then its complement $-G$ is in co-IFA, and our goal is to show that $-G \in$ (co-INT)-mixed. Define

$$E_1 = \{uv : I_u \cap I_v = \emptyset\}$$

and

$$E_2 = \{uv : ((I_u \subseteq I_v) \vee (I_v \subseteq I_u)) \wedge (f_u \cap f_v = \emptyset)\}.$$

Then

$$\overrightarrow{E_2} = \{uv : (I_u \subseteq I_v) \wedge (f_u \cap f_v = \emptyset)\}$$

is a transitive orientation of $E_2$, $E_1 \cup E_2 = \binom{V}{2} \setminus E$, $(V, E_1) \in$ co-INT and $E_1$ and $\overrightarrow{E_2}$ satisfy the mixing property. Hence $-G \in$ (co-INT)-mixed.

"$\supseteq$": Consider a graph in (co-INT)-mixed and denote its complement by $G = (V, E)$. Our goal is to show that $-G \in$ co-IFA, or equivalently that $G \in$ IFA. Since $-G$ is (co-INT)-mixed, the non-edges of $G$ can be partitioned into disjoint sets $E_1$, $E_2$ so that $G_1 = (V, E_1) \in$ co-INT, and $E_2$ allows a transitive orientation $\overrightarrow{E_2}$ such that $E_1$ and $\overrightarrow{E_2}$ satisfy the mixing property. Fix this partition $E_1$, $E_2$ and the transitive orientation $\overrightarrow{E_2}$.

Consider an interval intersection representation $\mathcal{R} = \{I_u = [l_u, r_u] : u \in V\}$ of $-G_1$ . We will only consider representations in which all end-points of the intervals are different points on the base line. For two vertices $u, v \in V$, there are three possibilities for the relative position of the intervals $I_u$, $I_v$ – the intervals are disjoint (when $r_u < l_v$ or $r_v < l_u$), or in inclusion (when $l_u < l_v < r_v < r_u$ or $l_v < l_u < r_u < r_v$), or overlapping (when $l_u < l_v < r_u < r_v$ or $l_v < l_u < r_v < r_u$). Given a representation $\mathcal{R}$, we set $\mathcal{R}_{\text{disjoint}} = \{uv : I_u$ and $I_v$ are disjoint$\}$, $\mathcal{R}_{\text{inclusion}} = \{uv : I_u$ and $I_v$ are in inclusion$\}$ and $\mathcal{R}_{\text{overlap}} = \{uv : I_u$ and $I_v$ are overlapping$\}$. Observe that $\mathcal{R}$ is an interval intersection representation of $-G_1$ if and only if $\mathcal{R}_{\text{disjoint}} = E_1$. In such a case, $\mathcal{R}_{\text{inclusion}} \cup \mathcal{R}_{\text{overlap}} = E \cup E_2$.

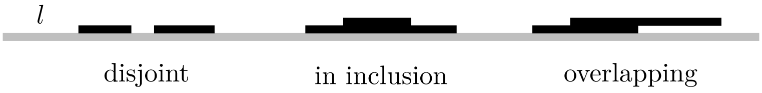


Figure 3.3: An illustration to the possible relative positions of pairs of intervals.

**Claim 9.** *If $E_1$ and $E_2$ satisfy the mixing condition, $-G_1$ has an interval intersection representation such that for every two vertices $u, v \in V$, $uv \in \overrightarrow{E_2}$ implies $I_u \subseteq I_v$.*

*Proof of claim.* Let us call a violation a pair of endpoints of intervals $I_u$, $I_v$ with $uv \in \overrightarrow{E_2}$ such that $l_u < l_v$ or $r_v < r_u$. Observe that overlapping intervals provide one violation, while intervals in inclusion provide either 0 (when $I_u \subseteq I_v$) or 2 (when $I_v \subseteq I_u$) violations.

Consider a representation $\mathcal{R}$ with the smallest possible number of violations. We will prove that this number is zero, i.e., that this $\mathcal{R}$ satisfies the statement of the Claim.

Suppose, for the contrary, that $\mathcal{R}$ has $k > 0$ violations. For every violation, count the number of endpoints of other intervals lying between the two endpoints forming this violation, and let $m$ be the minimum of these counts over all violations. Assume that $\mathcal{R}$ is chosen such that $m$ is minimum possible among all representations with $k$ violations. In the case analysis which follows, we assume that the violation achieving this minimum number of endpoints is formed by the right endpoints of $I_u$ and $I_v$. If it is achieved by their left endpoints, the arguments are similar (and symmetric).

Case $B1$. $m = 0$ Let $r_v < r_u$, with $uv \in \overrightarrow{E_2}$, be a violation such that there are no other endpoints between $r_v$ and $r_u$. Change the representation $\mathcal{R}$ to $\mathcal{R}'$ by changing the interval $I_v$ to $I'_v = [l_v, r_u + \epsilon]$ for a positive $\epsilon$ small enough so that no endpoint of any interval is between $r_u$ and $r_u + \epsilon$ (other intervals remain the same as in $\mathcal{R}$). Then $\mathcal{R}'$ is still an interval intersection representation of $-G_1$, but it has $k - 1 < k$ violations, contradicting the assumption that $k$ was the minimum possible number of violations in an interval representation of $-G_1$.

Case $B2$. $m > 0$ Let $r_v < r_u$, with $uv \in \overrightarrow{E_2}$, be a violation with $m$ endpoints between $r_v$ and $r_u$. Let $P$ be the leftmost of these endpoints.

Subcase $B2\alpha$. $P$ is the right endpoint of an interval. Let $w \in V$ be such that $P = r_w$. Change the representation $\mathcal{R}$ to $\mathcal{R}'$ by changing the interval $I_v$ to $I'_v = [l_v, r_w + \epsilon]$ for a positive $\epsilon$ small enough so that no endpoint of any interval is between $r_w$ and $r_w + \epsilon$. Then $\mathcal{R}'$ is still an interval intersection representation of $-G_1$. The number of endpoints between $r'_v$ and $r_u$ is $m - 1 < m$. If $\mathcal{R}'$ had the same number of violations as $\mathcal{R}$, this would contradict the assumption of minimality of $m$. Thus $\mathcal{R}'$ must have more than $k$ violations. The only new violation can be formed by $r_w < r'_v$, which means that $vw \in \overrightarrow{E_2}$. Then the transitivity of $\overrightarrow{E_2}$ implies that $uw \in \overrightarrow{E_2}$ and $r_w < r_u$ is a violation in $\mathcal{R}$ with $m - 1 < m$ endpoints between $r_w$ and $r_u$, contradicting the assumption on minimality of $m$.

Subcase $B2\beta$. $P$ is the left endpoint of an interval. Let $w \in V$ be such that $P = l_w$. Then $I_v \cap I_w = \emptyset$ and hence $vw \in E_1$. The mixing property applied to $u, v, w$ then implies $uw \in E_1$, which is impossible since $P \in I_u \cap I_w \neq \emptyset$.

Thus both Cases $B2$ and $B1$ lead to contradictions, and hence $k = 0$ and $\mathcal{R}$ satisfies the property described in the Claim. □

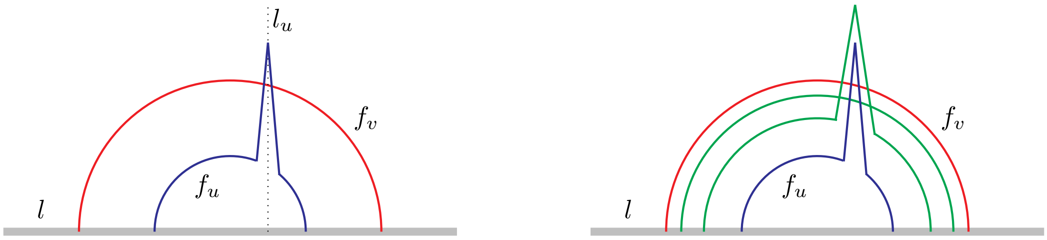
Assume we are having an interval intersection representation $\mathcal{R}$ guaranteed by the Claim. It follows that $E_2 \subseteq \mathcal{R}_{\text{inclusion}}$, and hence $\mathcal{R}_{\text{overlap}} \subseteq E$. In the first step, define, for every vertex $u \in V$, the interval filament $f_u$ as the half-circle with diameter $I_u$. At this point we observe the following

1. if $I_u$ and $I_v$ are disjoint, so are also $f_u$ and $f_v$, which corresponds to the fact that $uv \in E_1$ and thus $uv \notin E$;

2. if $I_u$ and $I_v$ are overlapping, $f_u$ and $f_v$ cross each other, which corresponds to the fact that $uv \in E$ which is observed above; however

3. if $I_u$ and $I_v$ are in inclusion, say $I_u \subseteq I_v$, then either $uv \in E_2$ (and thus $uv \in \overrightarrow{E_2}$) or $uv \in E$, but $f_u \cap f_v = \emptyset$ in both cases.

We will now modify some of the filaments in order to make filaments of case 3 intersect if the corresponding vertices are adjacent in $G$. For every pair of vertices $u, v \in V$ such that $I_u \subseteq I_v$ and $uv \in E$, choose a line $l_u$ perpendicular to the base line $l$ and crossing it in an interior point of $I_u$. Then pull the filament $f_u$ up along $l_u$ to make it cross $f_v$ (cf. an illustrative Fig. 3.4a). To avoid creating undesired intersections with other filaments, we must modify also those filaments which cross the line $l_u$ between its crossings with $f_u$ and $f_v$. Imagine this as a dynamic process, as if $f_u$ would slowly grow a spike upward and this spike would be pushing every filament $f_w$ in front of it, if $w$ is such that $uw \notin E$. Moreover, if there were another filament $f_z$ above $f_w$ such that $wz \notin E$, $f_z$ would be pushed by $f_w$ etc. See Fig. 3.4b. We claim that in this way no undesired intersections arise. Such could be only between one of the pushed filaments, say $f_y$, and the filament $f_v$. If $f_y$ is pushed, there is a sequence of vertices $u_0 = u, u_1, \ldots, u_t = y$ such that $f_{u_{i-1}}$ pushes $f_{u_i}$ for $i = 1, 2, \ldots, t$, i.e., $u_{i-1}u_i \notin E$ and $I_{u_{i-1}} \subseteq I_{u_i}$, hence $u_{i-1}u_i \in \overrightarrow{E_2}$. Transitivity of $\overrightarrow{E_2}$ then implies $uy \in \overrightarrow{E_2}$. Similarly, if $f_y$ should not cross $f_v$, i.e., $yv \notin E$, we would have $yv \in \overrightarrow{E_2}$. But that would imply $uv \in \overrightarrow{E_2}$ contradicting the assumption that $uv \in E$.



(a) Crossing filaments.

(b) Pushing outwards.

Figure 3.4: Final modification of interval-filaments in the representation.

□

# Chapter 4

# Cliques and independent sets

We will show the computational complexity of the following optimization problems:

**Clique**

*Input:* A graph $G$.

*Output:* $\omega(G)$.

and

**Independent Set**

*Input:* A graph $G$.

*Output:* $\alpha(G)$.

and their weighted variants

**Weighted Clique**

*Input:* A graph $G$ and a non-negative weight function $w : V(G) \to \mathbb{R}_0^+$.

*Output:* A clique $C \subseteq V(G)$ which maximizes $w(C) = \sum_{u \in C} w(u)$.

and

**Weighted Independent Set**

*Input:* A graph $G$ and a non-negative weight function $w : V(G) \to \mathbb{R}_0^+$.

*Output:* An independent set $A \subseteq V(G)$ which maximizes $w(A) = \sum_{u \in A} w(u)$.

Our goal is to show that for many of the intersection-defined graph classes that we have seen so far, these problems can be solved in polynomial time. For the sake of brevity, we denote by $\omega_w(G)$ the maximum possible weight $w(C)$ over all cliques $C$ in $G$, and by $\alpha_w(G)$ the maximum possible weight $w(A)$ over all independent sets $A$ in $G$.

## 4.1 Interval graphs

**Theorem 10.** ***Weighted Clique*** *can be solved in polynomial time for interval graphs.*

*Proof.* Interval graphs have only linearly many maximal cliques. We look at all of them and compare their weights. □

**Theorem 11.** ***Weighted Independent Set*** *can be solved in polynomial time for interval graphs.*

*Proof.* Suppose we are given an interval intersection representation $\mathcal{R} = \{I_u = [l_u, r_u] : u \in V\}$ of a graph $G = (V, E)$, equipped with a weight function $w$. We may assume that all endpoints of the intervals are different. Number the endpoints $P_1, P_2, \ldots, P_{2n}$ so that $P_i < P_{i+1}$ for $i = 1, 2, \ldots, 2n - 1$. Use dynamic programming to compute $w_i$ which is defined as the maximum possible weight of an independent set $A$ in $G$ such that $r_u < P_i$ for all $u \in A$. This can be computed as follows:

□

*Corollary.* **Weighted Clique** and **Weighted Independent Set** can both be solved in polynomial time on co-INT graphs.

```
$w_i := 0$
for $i := 1$ to $2n$ do
    if $P_i$ is the left endpoint of an interval then
        $w_{i+1} := w_i$
    else
        let $u \in V$ be such that $r_u = P_i$ and let $j$ be such that $P_j = l_u$;
        set $w_{i+1} = \max\{w_j + w(u), w_i\}$
    end if
end for
return $w_{2n+1}$
```

## 4.2 Comparability graphs

**Theorem 12.** ***Weighted Clique*** *can be solved in polynomial time for comparability graphs.*

*Proof.* Given a transitive orientation $\overrightarrow{E}$ of $G = (V, E)$, order the vertices linearly $V = \{v_1, v_2, \ldots, v_n\}$ in a topological sorting according to $\overrightarrow{E}$ (i.e., $v_iv_j \in \overrightarrow{E}$ implies $i < j$). For each $i$, set $W_i = \{j : v_jv_i \in E\}$ and let $w_i$ be the maximum weight $w(C)$ over all cliques $C \subseteq W_i \cup \{v_i\}$. Note that if $w(v_i) > 0$, each clique attaining the maximum weight contains $v_i$, and we may consider without loss of generality only the cliques that contain $v_i$ even if $w(v_i) = 0$. The values $w_i, i = 1, 2, \ldots, n$ can be computed recursively as follows

```
for $i := 1$ to $n$ do
    $w_i := \max_{j \in W_i} w_j + w(v_i)$
end for
```

and clearly $\omega_w(G) = max_{i=1}^{n} w_i$. □

**Theorem 13.** ***Weighted Independent Set*** *can be solved in polynomial time for comparability graphs.*

*Proof.* Given a transitive orientation $\overrightarrow{E}$ of $G = (V, E)$, consider the partial order $P = (V, \overrightarrow{E})$ determined by this orientation. The weighted version of Dilworth theorem yields that $\alpha_w(P)$ is equal to the minimum cost of a flow in the network $N = (V \cup \{s, t\}, E \cup \{su, ut : u \in V\})$ with vertex demands $f(u) \geq w(u)$, and this can be computed in polynomial time by network flow algorithms. □

*Corollary.* **Weighted Clique** and **Weighted Independent Set** can both be solved in polynomial time on function (= co-comparability) graphs.

## 4.3 Interval-filament graphs

In this section we show the strongest results. Note, however, that we need the input graph to be given with an interval-filament representation (or, equivalently, with a partition of its edge set satisfying the mixing property).

**Theorem 14.** ***Weighted Clique*** *can be solved in polynomial time for interval-filament graphs, if an interval-filament representation is provided on the input.*

*Proof.* Suppose $\mathcal{R} = \{f_u : u \in V\}$ is an interval-filament representation of $G$, with $I_u$ being the base interval of the filament $f_u$ for each $u \in V$, and let $w$ be the input weight function. We may suppose that the endpoints of the base intervals are mutually distinct, and we number them $P_1, P_2, \ldots, P_{2n}$ from left to right. We further introduce points $Q_1, Q_2, \ldots, Q_{2n-1}$ so that $Q_i$ lies between $P_i$ and $P_{i+1}$ for $i = 1, 2, \ldots, 2n-1$. Set $V_i = \{u : Q_i \in I_u\}$. If $C \subseteq V$ is a clique in $G$, the intervals $I_u$, $u \in C$ have a non-empty intersection, and hence $C \subseteq V_i$ for some $i$. Thus $\omega_w(G) = max_{i=1}^{2n-1} \omega_w(G[V_i])$ and each $\omega_w(G[V_i])$ can be computed in polynomial time by Theorem 13 since $G[V_i]$ is a co-comparability graph. □

**Theorem 15.** *Suppose* ***Weighted Clique*** *can be solved in polynomial time for graphs from a hereditary graph class $\mathcal{A}$. Then it can be solved in polynomial time for $\mathcal{A}$-mixed graphs, provided a partition of the edges of the input graph that satisfies the mixing property is a part of the input.*

*Proof.* Given a graph $G = (V, E)$, a weight function $w \to \mathbb{R}_0^+$ and a partition $E = E_1 \cup E_2$ such that $(V, E_1) \in \mathcal{A}$, together with a transitive orientation $\overrightarrow{E_2}$ of $E_2$ which satisfies the mixing property, order the vertices $V = \{v_1, v_2, \ldots, v_n\}$ in a topological sorting with respect to $\overrightarrow{E_2}$. For every vertex $v_i \in V$, set $W_i = \{v_j : v_jv_i \in \overrightarrow{E_2}\}$. Let $w_i = \omega_w(G[W_i \cup \{v_i\}])$ and let $C_i$ be a clique in $G[W_i \cup \{v_i\}]$ of weight $w_i$.

**Claim 16.** *Let $M \subseteq W_i$ be a maximum weighted clique in the graph $(W_i, E_1 \cap \binom{W_i}{2})$ with respect to the weight function $\bar{w}(v_j) = w_j$ for $v_j \in W_i$. Then*

$$C = \bigcup_{v_j \in M} C_j \cup \{v_i\}$$

*is a maximum weighted clique in $G[W_i \cup \{v_i\}]$ with respect to $w$ and its weight is $w_i = \sum_{j:v_j \in M} w_j + w(v_i)$.*

*Proof of claim.* First we show that $C$ is a clique. Due to transitivity of $\overrightarrow{E_2}$, for every $v_j \in M$ and every $x \in C_j$, $xv_i \in \overrightarrow{E_2}$, and hence $xv_i \in E_2 \subseteq E$. Hence $C$ is a clique if $|M| = 1$. If $|M| > 1$, consider $v_j, v_h \in M$ and $x \in C_j$, $y \in C_h$. Suppose $x \neq v_j$. The mixing property, when applied to $x, v_j, v_h$, then implies $xv_h \in E_1$. If $y \neq v_h$, the mixing property, when applied to $y, v_h, x$, implies $yx \in E_1$. These situations cover all pairs of vertices in $C$.

Next we show that any two $C_j$, $C_h$ are disjoint. If there were an $x \in C_j \cap C_h$, this $x$ would be different from both $v_j$ and $v_h$. But then $xv_j \in \overrightarrow{E_2}$ and $v_jv_h \in E_1$ imply (via the mixing property) $xv_h \in E_1$, contradicting the assumption $x \in C_h$. It further follows that $w(C) = \sum_{j:v_j \in M} \sum_{x \in C_j} w(x) + w(v_i) = \sum_{j:v_j \in M} w_j + w(v_i)$.

Finally we show that the weight of $C$ is maximum possible. Suppose $D \subseteq W_i \cup \{v_i\}$ is a clique in $G[W_i \cup \{v_i\}]$ of maximum weight with respect to $w$. Obviously $v_i \in D$. Denote $\bar{D} = D \setminus \{v_i\}$ and consider $\bar{G} = (\bar{D}, \overrightarrow{E_2} \cap \bar{D} \times \bar{D})$. This $\bar{G}$ is an acyclic graph. Denote by $\bar{M}$ the sinks of $\bar{G}$. Every other vertex of $\bar{D}$ is connected to some vertex of $\bar{M}$ by a directed path, and hence, due to transitivity of $\overrightarrow{E_2}$, by a directed edge from $\overrightarrow{E_2}$. Set $\bar{D}_j = \{u : uv_j \in \overrightarrow{E_2} \wedge u \in \bar{D}\}$ and $D_j = \bar{D}_j \cup \{v_j\}$, for $v_j \in \bar{M}$. The mixing property implies that for $j \neq h$, $D_j \cap D_h = \emptyset$. Since $D_j$ is a clique in $G[W_j \cup \{v_j\}]$, it is $w(D_j) \leq w_j = \bar{w}(v_j)$. All vertices of $\bar{M}$ are sinks of $\bar{G}$, and thus any two of them are connected by an edge of $E_1$. Hence $M$ is a clique in $(W_i, E_1 \cap \binom{W_i}{2})$, and therefore $\bar{w}(\bar{M}) \leq \bar{w}(M)$. Thus

$$\begin{aligned} w(D) &= w(v_i) + \sum_{j:v_j \in \bar{M}} \sum_{x \in D_j} w(x) = w(v_i) + \sum_{j:v_j \in \bar{M}} w(D_j) \leq w(v_i) + \sum_{j:v_j \in \bar{M}} \bar{w}(v_j) \\ &= w(v_i) + \bar{w}(\bar{M}) \leq w(v_i) + \bar{w}(M) = w(C) \end{aligned}$$

and $C$ is indeed a maximum weighted clique with respect to the weight function $w$. □

Having proved the observations above, we can now summarize the algorithm for finding a maximum weighted clique in an $\mathcal{A}$-mixed graph. Note that the preprocessing trick with adding a dummy vertex $v_{n+1}$ is introduced for the comfort of a more succinct write-up of the algorithm.

**Algorithm 1** Weighted Clique in $\mathcal{A}$-mixed graphs

```
Require: A graph $G = (V, E)$, a partition $E = E_1 \cup E_2$ such that $(V, E_1) \in \mathcal{A}$, a transitive orientation $\overrightarrow{E_2}$ of
    $E_2$ such that $E_1, \overrightarrow{E_2}$ satisfy the mixing property, and a weight function $w : V \to \mathbb{R}_0^+$.
Ensure: The weighted clique number $\omega_w(G)$ of $G$ and a maximum weighted clique $C$.
Order the vertices of $G$ linearly in a topological sorting with respect to $\overrightarrow{E_2}$ as $V = \{v_1, v_2, \ldots, v_n\}$ and add
    a dummy vertex $v_{n+1}$ with weight $w(v_{n+1}) = 0$ and edges $v_iv_{n+1} \in E_2$, all of them being oriented from $v_i$
    to $v_{n+1}$, for $i = 1, 2, \ldots, n$.
for $i = 1 \ldots n+1$ do
    $w_i := 0$, $C_i := \emptyset$
end for
for $i = 1 \ldots n+1$ do
    $W_i := \{v_j : v_jv_i \in \overrightarrow{E_2}\}$
    for $v_j \in W_i$ do
        $\bar{w}(v_j) := w_j$
    end for
    find maximum weighted clique $M$ in $(W_i, E_1 \cap W_i \times W_i)$ w.r.t. $\bar{w}$
    $C_i := \cup_{j:v_j \in M} C_j \cup \{v_i\}$
    $w_i := \sum_{j:v_j \in M} w_j + w(v_i)$
end for
return $C = C_{n+1} \setminus \{v_{n+1}\}, \omega_w(G) = w_{n+1}$
```

The correctness of the algorithm was proven in the Claim above, it may only be added here that the introduction of the dummy vertex $v_{n+1}$ does not break the mixing property of $E_1$ and $\overrightarrow{E_2}$ and that $v_{n+1}$ belongs to every maximal clique, and hence also to every maximum weighted one, but does not affect its weight. If $F(n)$ is the worst-case running time of the algorithm for maximum weighted clique on $\mathcal{A}$ graphs, the running time of our algorithm is majorized by $nF(n)$. □

*Corollary.* **Weighted Independent Set** can be solved in polynomial time for interval-filament graphs, provided a representation is given as part of the input.

*Proof.* We have seen that **Weighted Clique** is polynomial time solvable for co-INT graphs. Theorem 15 implies that it is polynomial-time solvable for (co-INT)-mixed graphs, alas for co-IFA graphs. Thus **Weighted Independent Set** is polynomial-time solvable for IFA graphs. □

# Chapter 5

# Recognition of Chordal graphs

It is not surprising that chordal graphs can be recognized in polynomial time. Just search for a simplicial vertex, delete it if you find one and proceed recursively, or quit and say that the input graph is not chordal if it has no simplicial vertex. Checking a vertex for being simplicial requires checking at most $n^2$ pairs of vertices if they are adjacent, hence a simplicial vertex can be found in $O(n^3)$ steps. Thus chordality can be checked in time $O(n^4)$. However, even if the graph is dense, i.e., if it has $m = \Omega(n^2)$, this naive algorithm still takes $\Omega(m^2)$ time. With more care, one can test chordality in time linear in the number of edges by the following algorithm.

**Algorithm 2** LexBFS

```
Require: A graph $G = (V, E)$ with $n$ vertices.
$T := \emptyset, w(x) := \emptyset \ \forall x \in V$
for $i := n \dots 1$ do
    let $u \in V \setminus T$ be a vertex with lexicographicly maximum $w(u)$
    set $\sigma(i) := u$
    $T := T \cup \{u\}$
    for $x \in N_G(u) \cap (V \setminus T)$ do
        $w(x) := w(x) \cup \{i\}$
    end for
end for
```

**Theorem 17.** *If $G = (V, E)$ is chordal, then $\sigma(n), \sigma(n-1), ..., \sigma(1)$ is a PES* 🐕.

*Proof.* Suppose $G$ is chordal but there exist indices $i < j < h$ such that $\sigma(i)\sigma(j) \in E, \sigma(i)\sigma(h) \in E$ while $\sigma(j)\sigma(h) \notin E$. Let $i_0 < i_1 < i_2$ be such a triple and such that $i_2$ is largest possible.

Let $u_i = \sigma(i), i = 1, 2, \dots, n$ be the ordering output by the Algorithm. Consider the step of the LexBFS algorithm when $i_1$ was processed. At this moment, $i_2 \in w(u_{i_0})$ while $i_2 \notin w(u_{i_1})$. Thus $u_{i_0}$ and $u_{i_1}$ do not have the same weights, and since $u_{i_1}$ was chosen for $\sigma(i_1)$, we must have $w(u_{i_0}) <_{lex} w(u_{i_1})$. Hence there must be $i_3 > i_2$ such that $u_{i_1}u_{i_3} \in E$ and $u_{i_0}u_{i_3} \notin E$. Let us choose $i_3$ as the largest index with these properties. An edge between $u_{i_2}$ and $u_{i_3}$ would imply that $G[\{u_0, u_1, u_2, u_3\}]$ would be isomorphic to $C_4$, which would contradict the assumption that $G$ is chordal, and thus we conclude that $u_{i_2}u_{i_3} \notin E$.

The same argument can now be used for $i_2$ and $i_3$ and further on:

**Claim 18.** *For every $k > 3$, there exists a sequence $i_0 < i_1 < \dots < i_k$ such that $\{u_{i_0}u_{u_1}\} \cup \{u_{i_j}u_{i_{j+2}} : j = 0, 1, 2, \dots, k-2\}$ are the only edges of $G$ among these vertices.*

*Proof of claim.* We proceed by induction on $k$. We have just seen that such a sequence exists for $k = 4$. Since the sequence is constructed recursively, we will further assume that in each step we have chosen $i_k$ largest possible.

For the induction step $k \to k+1$, we observe that at the moment when $i_{k-1}$ was processed, $i_k$ was in $w(u_{k-2})$ but not in $w(u_{k-1})$. Since $u_{i_{k-1}}$ was chosen for $\sigma(i_{k-1})$, it must have been $w(u_{i_{k-1}}) >_{lex} w(u_{i_{k-2}})$, and so there must be an $i_{k+1} > i_k$ such that $u_{i_{k-1}}u_{i_{k+1}} \in E$ while $u_{i_{k-2}}u_{i_{k+1}} \notin E$ (recall that we choose $i_{k+1}$ maximum possible with this property). Suppose there is a $j$, $0 \le j \le k-3$ such that $u_{i_j}u_{i_{k+1}} \in E$, and consider the largest possible such $j$. If $j = k-3$, then both $u_{i_{k+1}}$ and $u_{i_{k-1}}$ are adjacent to $u_{i_{k-3}}$ and non-adjacent to $u_{i_{k-2}}$ and this contradicts the presumed choice of $i_{k-1} < i_{k+1}$ as the largest possible index of this property. If $j < k-3$, then $i_j < i_{j+2} < i_{k+1}$ would be a triple violating the PES condition (since $u_{i_j}u_{i_{j+2}}, u_{i_j}u_{i_{k+1}} \in E$ and $u_{i_{j+2}}u_{i_{k+1}} \notin E$) with $i_{k+1} > i_2$ contradicting the choice of $i_0 < i_1 < i_2$. Thus $u_{i_j}u_{i_{k+1}} \notin E$ for $j = 0, 1, \dots, k-2$. If $u_{i_k}u_{i_{k+1}}$ were an edge, $G[\{u_{i_h} : h = 0, 1, \dots, k+1\}]$ would be isomorphic to $C_{k+2}$, which is impossible as $G$ is assumed to be chordal. This concludes the proof of the Claim. □

Since $G$ is finite, the sequence of the properties guaranteed by the Claim cannot exist, e.g., for $k > n$. So no triple of indices $i < j < h$ violating the PES condition may exist. □

With a suitable data structure, the LexBFS algorithm can be implemented to run in time linear in the number $m$ of edges of the input graph (the point is that this algorithm processes every edge of the graph only once).

However, this is only one half of testing for chordality. The algorithm LexBFS outputs a linear ordering of the vertices of $G$, but we need to test this ordering if it is a PES or not. This can, fortunately, also be achieved in linear time.

**Algorithm 3** TestPES

```
Require: A graph $G = (V, E)$ with $n$ vertices, a linear ordering $\sigma(i) \in V, i = 1, 2, \dots, n$ of its vertices.
Ensure: YES or NO if $\sigma(n), \sigma(n-1), \dots, \sigma(1)$ is a PES.
$A(x) := \emptyset$ for all $x \in V$
for $i := 1 \dots n - 1$ do
    $v := \sigma(i)$
    $X := \{x : x \in N_G(v) \wedge \sigma^{-1}(v) < \sigma^{-1}(x)\}$
    if $X \neq \emptyset$ then
        $u := \min \sigma^{-1}(X)$
        $A(u) := A(u) \cup (X \setminus \{u\})$
    end if
    if $A(v) \setminus N_G(v) \neq \emptyset$ then
        return NO
    end if
end for
return YES
```

**Theorem 19.** *Algorithm TestPES correctly answers if a linear ordering $\sigma(n), \sigma(n-1), \dots, \sigma(1)$ is a PES for $G$ or not and it can be implemented in time linear in $m$.*

*Proof.* For each vertex $u \in V$, the set $A(u)$ contains vertices that should be neighbors of $u$ if the ordering is a PES. The point is that it is not necessary to put in $A(u)$ all neighbors, and thus the running time can be significantly reduced.

**Claim 20.** *If $\sigma(n), \sigma(n-1), \dots, \sigma(1)$ is not a PES, then at least one violation gets detected.*

*Proof of claim.* Let $i < j < h$ be a violation (i.e., $\sigma(i)\sigma(j), \sigma(i)\sigma(h) \in E$, $\sigma(j)\sigma(h) \notin E$) such that the difference $j - i$ is minimum possible. Let $v = \sigma(i)$. Then $\sigma(j), \sigma(h) \in X$ when $v$ is processed by the algorithm. Suppose there is a $j', i < j' < j$ such that $\sigma(j') \in X$. If $\sigma(j')\sigma(j) \notin E$, $i < j' < j$ is a violation with smaller difference $j' - i < j - i$. If $\sigma(j')\sigma(h) \notin E$, $i < j' < h$ is a violation with smaller difference $j' - i < j - i$. If both $\sigma(j')\sigma(j)$ and $\sigma(j')\sigma(h)$ are edges, $j' < j < h$ is a violation with smaller difference $j - j' < j - i$. Since all possibilities lead to contradictions, we conclude that $j = \min\{l : \sigma(l) \in X\}$ and $\sigma(h)$ is put into $A(\sigma(j))$ when $v$ is being processed. Thus the non-edge $\sigma(j)\sigma(h)$ gets detected when $u = \sigma(j)$ is processed (unless the algorithm quits even sooner). □

□

*Remark.* Note that the correctness of Algorithm LexBFS implies that not only one can leave any simplicial vertex as the last one in a PES, but it is also true that any vertex of a chordal graph can be used as the first vertex in a PES.

# Chapter 6

# Recognition of Comparability graphs

Recall that comparability graphs are exactly the transitively orientable graphs. And that a transitive orientation is a binary relation on the vertex set of the graph which is transitive and orients every edge of the graph in exactly one direction. In order to eventually construct such a relation, if it exists, we will examine partial orientations and relations with further useful properties. Throughout this chapter we assume that we are processing a given simple undirected graph $G = (V, E)$.

**Definition 12.** *A relation $M \subseteq V \times V$ is called*

- ***sensitive*** *if for every three vertices $x, y, z \in V$, it holds true that $(x, y) \in M, xz \in E, yz \notin E$ imply $(x, z) \in M$, and $(x, y) \in M$, $zy \in E$, $xz \notin E$ imply $(z, y) \in M$,*
- ***complete*** *if it is sensitive and transitive,*
- ***faithful*** *if for every two vertices $x, y \in V$, it holds true that $(x, y) \in M$ implies $xy \in E$, and*
- ***whole*** *if for every edge $xy \in E$, at least one of $(x, y), (y, x)$ is in $M$.*

**Observation.** *Every faithful, transitive and whole relation is necessarily sensitive.*

*Proof.* Suppose $x, y, z$ are such that $(x, y) \in M$, $xz \in E$ and $yz \notin E$. Since $M$ is whole, we have either $(x, z) \in M$ or $(z, x) \in M$. In the case of $(z, x) \in M$, transitivity of $M$ would imply $(z, y) \in M$, what would be in contradiction with the assumed faithfulness of $M$. Hence it must be $(x, z) \in M$. The symmetric rule is proven analogously. □

**Observation.** *Every transitive and faithful relation is necessarily antisymmetric (because we only consider simple - and henceforth loopless - graphs). Therefore transitive*
*orientations of $G$ are exactly those relations that are transitive, whole and faithful, and these are exactly those relations that are complete, whole and faithful.*

**Observation.** *The intersection of sensitive (transitive, complete) relations is a sensitive (transitive, complete, respectively) relation.*

The last observation implies that closures are defined uniquely:

**Definition 13.** *Let $M \subseteq V \times V$ be a binary relation. The smallest relation which contains $M$ and which is sensitive (transitive, complete) is called the sensitive- (transitive-, complete-, respectively) closure of $M$ and is denoted by $\langle M \rangle_S$ ($\langle M \rangle_T$ , $\langle M \rangle$, respectively).*

**Lemma 3.** *For any binary relation $M \subseteq V \times V$, it is $\langle M \rangle = \langle \langle M \rangle_S \rangle_T$.*

*Proof.* It suffices to show that $\langle \langle M \rangle_S \rangle_T$ is sensitive. Suppose $x, y, z \in V$ are such that $(x, y) \in \langle \langle M \rangle_S \rangle_T, xz \in E, yz \notin E$. Then there is a sequence $x = x_1, x_2, \ldots, x_k = y$ such that $x_i x_{i+1} \in \langle M \rangle_S$ for every $i = 1, 2, \ldots, k-1$. Since $x_1 z \in E$ and $x_k z \notin E$, there is an index $i, 1 \leq i < k$ such that $x_i z \in E$ and $x_{i+1} z \notin E$. Then $(xi, z) \in \langle M \rangle_S$ (this follows from its sensitivity), and hence $(x, z) \in \langle \langle M \rangle_S \rangle_T$ follows from the sequence of arcs $(x_1, x_2), (x_2, x_3), \ldots, (x_{i-1}, x_i), (x_i, z) \in \langle M \rangle_S$. □

## 6.1 Blocks

**Definition 14.** *A* ***block*** *is the sensitive closure $\langle (x,y)\rangle_S$ of a single arc $(x,y)$ such that $xy \in E$. A pathway of length $k-1$ in $V \times V$ from $(a,b)$ to $(c,d)$ is a sequence $(x_i, y_i), i = 1, 2, \ldots, k$ such that*

- $(a,b) = (x_1, y_1), (c,d) = (x_k, y_k)$,
- *for every $i = 1, 2, \ldots, k, x_i y_i \in E$,*
- *for every $i = 1, 2, \ldots, k-1$, either $x_i = x_{i+1}$ and $y_i y_{i+1} \notin E$, or $y_i = y_{i+1}$ and $x_i x_{i+1} \notin E$.*

*The* ***distance*** *of $(a,b)$ and $(c,d)$ (denoted by $dist_\Gamma((a,b),(c,d))$) is the length of a shortest pathway between $(a,b)$ and $(c,d)$.*

**Proposition 21.** *Let $xy \in E$ be an edge of $G$. Then the block $\langle (x,y)\rangle_S$ determined by an orientation $(x,y)$ of $xy$ contains exactly those arcs $(u,v)$ that are connected by pathways from $(x,y)$ to $(u,v)$. For every $(u,v) \in \langle (x,y)\rangle_S$, this arc defines the same block, i.e., $\langle (x,y)\rangle_S = \langle (u,v)\rangle_S$.*

*Proof.* Define the graph $\Gamma_G = ((V \times V) \cap \{(x,y) : xy \in E\}, \{(x,y)(u,v) : (x = u, yv \notin E) \wedge (y = v, xu \notin E)\})$ that captures the sensitive-rule constellations. Then blocks are connected components of $\Gamma_G$. The concatenation of pathways from $(x,y)$ to $(u,v)$ and from $(u,v)$ to $(s,t)$ is a pathway from $(x,y)$ to $(s,t)$. Hence $\langle (u,v)\rangle_S \subseteq \langle (x,y)\rangle_S$ for every $(u,v) \in \langle (x,y)\rangle_S$. Since a pathway from $(x,y)$ to $(u,v)$ traversed in the opposite way is a pathway from $(u,v)$ to $(x,y)$, we get $\langle (x,y)\rangle_S = \langle (u,v)\rangle_S$. □

*Corollary.* For an edge $xy \in E$, it is $\langle (y,x)\rangle_S = \langle (x,y)\rangle_S^{-1}$ and both $\langle (x,y)\rangle_S$ and $\langle (y,x)\rangle_S$ are faithful.

**Lemma 4.** *Let $xy \in E$ be an edge of $G$. If the block $\langle (x,y)\rangle_S$ is antisymmetric, then it is also transitive, and hence $\langle (x,y)\rangle = \langle (x,y)\rangle_S$ is faithful and complete.*

*Proof.* For the sake of brevity, denote $U = \langle (x,y)\rangle_S$. Suppose $U$ is not transitive, i.e., there exist vertices $a, b, c \in V$ such that $(a,b), (b,c) \in U$ and $(a,c) \notin U$. Since $U$ is faithful, $ab, bc \in E$. It follows from the Proposition above that $\langle (a,b)\rangle_S = U$, and hence there is a pathway from $(a,b)$ to $(b,c)$ in $G$. Let the choice of the transitivity violating triple $a, b, c$ be such that the distance of $(a,b)$ and $(b,c)$ is the smallest possible.

If $ac$ were not an edge of $G$, sensitivity of $U$ would imply $(c,b) \in U$ and that would be a contradiction with the assumed antisymmetry of $U$. Thus $ac \in E$.

Let $(a,b) = (x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k) = (b,c)$ be a shortest pathway from $(a,b)$ to $(b,c)$ in $G$ and let $l$ be the largest index such that $y_l \neq c$. Note that $x_l = x_{l+1}$ and $y_i = c$ for all $i = l+1, \ldots, k$. Set $\alpha = x_l$ and $\beta = y_l$.

**Claim 22.** *For every $i = l+1, \ldots, k, ax_i \in E$.*

*Proof of claim.* If for any such $i$ were $ax_i \notin E$, the sensitivity rule applied to $a, x_i, y_i = c$ would imply $ac \in U$ and $a, b, c$ would not violate transitivity. □

**Claim 23.** *For every $i = l+1, \ldots, k, (a, x_i) \in U$.*

*Proof of claim.* We know that $(a, x_k = b) \in U$ and applying the sensitivity rule backwards on triples $x_i, c, x_{i-1}$ for $i = k, k-1, \ldots, l+1$, the claim is proven. □

**Claim 24.** *It is $(a, \beta = y_e) \notin U$.*

*Proof of claim.* If $(a, \beta)$ were in $U$, the sensitivity rule applied to $\beta, a, c$ would imply $(a,c) \in U$, what is assumed not be the case. □

Now $a, \alpha = x_l, \beta = y_l$ is another transitivity violating example (as we have proved that $(a, \alpha) \in U, (a, \beta) \notin U$, and $(\alpha, \beta) \in U$ since it is included in the pathway). The sequence $(a, \alpha), (a, x_{l+2}), \ldots,$
$(a, x_k = b), (x_2, y_2), \ldots, (x_l = \alpha, y_l = \beta)$ is a pathway from $(a, \alpha)$ to $(\alpha, \beta)$ of length $k - l - 1 + l - 1 = k - 2 < \mathrm{dist}_\Gamma((a,b),(b,c))$, contradicting the choice of $a, b, c$. □

## 6.2 Structure of transitive orientations

If $G$ allows a transitive orientation $T$, each edge $xy \in E$ must be oriented one way or the other, and hence $\langle (x,y)\rangle_S \subseteq T$ or $\langle (y,x)\rangle_S \subseteq T$. Thus $\langle (x,y)\rangle_S$ must be antisymmetric for every edge $xy \in E$. The goal of this section is to show that this obvious necessary condition is also sufficient.

**Theorem 25.** *A graph $G$ is transitively orientable if and only if the block $\langle (x,y)\rangle_S$ is antisymmetric for every $xy \in E$.*

The proof of the Theorem will follow from the following Lemma 5, which will also serve as a tool for finding a transitive orientation, if one exists.

**Lemma 5.** *Let $M \subseteq V \times V$ be a faithful complete binary relation and let $xy \in E$ be an edge such that $\langle (x,y)\rangle_S$ is antisymmetric and $M \cap \{(x,y),(y,x)\} = \emptyset$. Then $\langle M \cup \{(x,y)\}\rangle$ is faithful.*

*Proof.* Denote by $U = \langle (x,y)\rangle_S$. Then $U = \langle (x,y)\rangle$ follows from Lemma 4. Since $M \cap \{(x,y),(y,x)\} = \emptyset$, it follows from the structure of blocks that $M \cap (U \cup U^{-1}) = \emptyset$, and hence $\langle M \cup \{(x,y)\}\rangle_S = M \cup U$. Lemma 3 then implies that $\langle M \cup \{(x,y)\}\rangle = \langle \langle M \cup \{(x,y)\}\rangle_S \rangle_T = \langle M \cup U\rangle_T$.

Suppose for the contrary that $\langle M \cup U\rangle_T$ is not faithful. Then there exists a sequence of edges $x_1x_2, x_2x_3, \ldots, x_{k-1}x_k$ such that $(x_i, x_{i+1}) \in M \cup U$ for all $i = 1,2,\ldots,k-1$ and $x_1x_k \notin E$. Since $x_2x_1 \in E$ and $x_kx_1 \notin E$, there is a $j, 2 \le j < k$ such that $x_jx_1 \in E$ and $x_{j+1}x_1 \notin E$. Sensitivity of $M \cup U$ then implies that $x_jx_1 \in M \cup U$ and $M \cup U$ contains a directed cycle.

Let $C$ be a shortest cycle in $M \cup U$. Since both $M$ and $U$ are transitive, the arcs of $C$ alternatively come from $M$ and $U$. The length of $C$ is at least 4, because a cycle of length 2 would contradict the assumption that $M \cap U^{-1} = \emptyset$. No diagonal of $C$ belongs to $M \cup U$, since any such diagonal would create a shorter cycle. Finally, all diagonals of $C$ are edges of $G$, since otherwise the sensitivity of $M \cup U$ would imply that either some diagonal belongs to $M \cup U$, or some arc of the cycle belongs to $M \cap U^{-1}$.

Let now $a,b,c,d$ be consecutive vertices of $C$ such that $(a,b),(c,d) \in U$ and $(b,c) \in M$. Let the choice of $C$ and of $a,b,c,d$ be such that $\mathrm{dist}_\Gamma((a,b),(c,d))$ is minimum possible among all such choices. Note again that we have already observed that $ac, bd, ad \in E$, $\{(a,c),(c,a),(b,d),(d,b),(a,d)\} \cap (M \cup U) = \emptyset$ and $(d,a) \notin U$.

Consider a shortest pathway $(a,b) = (x_1,y_1),(x_2,y_2),\ldots,(x_k,y_k) = (c,d)$. Let $l$ be the smallest index such that $y_l \neq b$ (i.e., $y_1 = y_2 = \cdots = y_{l-1} = b, x_{l-1} = x_l and x_1, x_2, \ldots, x_{l-1}$ are pair-wise different). Set $\alpha = x_l$ and $\beta = y_l$ . We proceed with a sequence of observations.

**Claim 26.** *For every $i = 1,2,\ldots,l, x_id \in E$.*

*Proof of claim.* If $x_id$ were a non-edge for some $i$, sensitivity would imply $(d,b) \in U$. □

**Claim 27.** *For every $i = 1,2,\ldots,l, x_ic \in E$.*

*Proof of claim.* If $x_jc$ were a non-edge for some $j$, sensitivity of $U$ would imply $(x_j,d) \in U$, and hence $(x_i,d) \in U$ for all $i = 1,2,\ldots,l$. Thus $(a,d) \in U$, contradicting the assumption. □

**Claim 28.** *It is $c\beta \in E$.*

*Proof of claim.* Otherwise sensitivity applied to $c, \alpha = x_l, \beta = y_l$ would imply $(\alpha,c) \in U$, and hence $(x_i,c) \in U$ for all $i = 1,2,\ldots,l$, and thus $(a,c) \in U$. □

**Claim 29.** *Now $(\beta,c) \in M$ follows from sensitivity of $M$ applied to $b,c,\beta$.*

**Claim 30.** *For every $i = 1,2,\ldots,l, x_i\beta \in E$.*

*Proof of claim.* If $x_j\beta$ were a non-edge for some $j$, sensitivity of $M$ would imply $(x_j,c) \in M$, hence we would get $(x_i,c) \in M$ for all $i = 1,2,\ldots,l$, and also $(a,c) \in M$. □

**Claim 31.** *Sensitivity of $U$ implies that $(x_i,\beta) \in U$ for all $i = 1,2,\ldots,l$, and hence also $(a,\beta) \in U$.*

Now we see that $a,\beta,c,d,\ldots$ is a directed cycle in $M \cup U$ of the same length as $C$, while $\mathrm{dist}_\Gamma((a,\beta),(c,d)) \le l-2+k-l = k-2 < k-1 = \mathrm{dist}_\Gamma((a,b),(c,d))$ since $(x_1,\beta),(x_2,\beta),\ldots,(x_{l-1},\beta),(x_{l+1},y_{l+1}),\ldots,(x_k,y_k)$ is a pathway from $(a,\beta)$ to $(c,d)$. This is the desired contradiction. □

*Proof of Theorem 25.* We have already seen that if $G$ is transitively orientable, each block must be asymmetric.

For the opposite implication, suppose that $G$ is such that each block $\langle (x,y)\rangle_S$, $xy \in E$, is asymmetric and suppose that $M \subseteq V \times V$ is a largest possible complete and faithful binary relation on $V$. If $M$ is not whole, there is an edge $xy \in E$ which is not oriented by $M$, i.e., $M \cap \{(x,y),(y,x)\} = \emptyset$. Lemma 5 implies that $\langle M \cup \{(x,y)\}\rangle$ is a faithful complete relation, and it is clearly a strict superset of $M$. That would contradict the choice of $M$. Hence $M$ is whole, and thus a transitive orientation of $G$. □

## 6.3 Algorithmic aspects

Lemmas 2 and 3 also provide the correctness argument for a polynomial time algorithm for deciding if a graph is transitively orientable, and constructing such an orientation, if one exists.

**Algorithm 4** Transitive Orientation

```
Require: A graph $G = (V, E)$.
M := ∅
while $M$ is not whole do
    let $xy \in E$ be such that $M \cap \{(x, y), (y, x)\} = \emptyset$
    construct $U = \langle (x, y) \rangle_S$
    if $U$ is not antisymmetric then
        return "$G$ is not transitively orientable"
    else
        $M := \langle M \cup U \rangle_T$
    end if
end while
return "$G$ is transitively orientable by $M$"
```

# Chapter 7

# Visibility and contact representations of planar graphs

## 7.1 $s - t$ numbering

**Definition 15.** *An $s - t$ numbering (also called a bipolar orientation) of a graph is an acyclic orientation of its edges which has exactly one source (a vertex with no in-coming arcs) and exactly one sink (a vertex with no out-going arcs).*

**Proposition 32.** *Every vertex 2-connected graph has an $s - t$ numbering.*

*Proof.* By induction on adding ears, using the Ear Decomposition Lemma. Given the starting cycle, choose two distinct vertices on it (to be the source and the sink) and orient the cycle as two directed paths from the source to the sink. The loop invariant will be that every vertex lies on a directed path from the source to the sink. When adding an ear, orient it in the direction from the source to the sink if both end-vertices of the ear lie on the same path from the source to the sink. If the end-vertices of the ear lie on different paths from the source to the sink, the ear may be oriented either way (but all of its edges in the same direction). □

**Theorem 33.** *Every vertex 2-connected plane graph has an $s - t$ numbering and a noncrossing planar drawing such that*

*i) every edge is drawn as a $y$-monotone curve (i.e., every horizontal line crosses the drawing of the edge in at most one point), and*

*ii) the drawing of every edge is oriented upward in the $s - t$ numbering.*

This $s - t$ numbering and a corresponding planar drawing can be constructed in polynomial time.

**Comment.** *By saying a plane graph it is meant a planar graph with a given noncrossing drawing in the plane, and it is understood that only drawings which are homeomorphic to the given one are considered, including the choice of the outerface.*

*Proof.* By Ear Decomposition Lemma, the given graph $G$ can be constructed from the cycle bounding its outerface by adding ears. Choose two distinct vertices, say $a$ and $b$, on this cycle, place them in the plane so that they have different $y$-coordinates, the coordinate of $a$ being smaller than the coordinate of $b$, orient the two $a - b$ paths forming the cycle from $a$ to $b$ and draw them as $y$-monotone paths from (the drawing of) $a$ to (the drawing of) $b$.

Then continue adding the ears, orienting their edges and adding them to the drawing constructed while keeping the following loop invariant − the drawing is homeomorphic to the so far constructed part of $G$, it satisfies i) and ii), all vertices have distinct $y$-coordinates, and each face is bounded by two upward oriented paths connecting its vertices with the lowest and the highest $y$-coordinates (we will refer to these paths as the *left* one and the *right* one). Note that i) and ii) together imply that the drawing of any directed path is $y$-monotone.

When an ear is added, it is added inside a face of the so far constructed part of $G$. Direct the edges of the ear in the direction from the vertex with the lower $y$-coordinate to the vertex with the higher one (the end-vertices of the ear belong to the so far constructed part of $G$ and so they have already been drawn). If the end-vertices belong to the same bounding path (the left one or the right one) of the face they should be draw in, draw the ear as a $y$-monotone curve contouring the bounding path. If the end-vertices belong to different bounding paths, draw the ear as a $y$-monotone curve that contours the path which contains the lower endpoint and traverse the face to the other endpoint almost horizontally close to the $y$-coordinate of this other endpoint, in order to avoid crossings with other edges. It is easy to check that the loop invariant of the construction is fulfilled. □

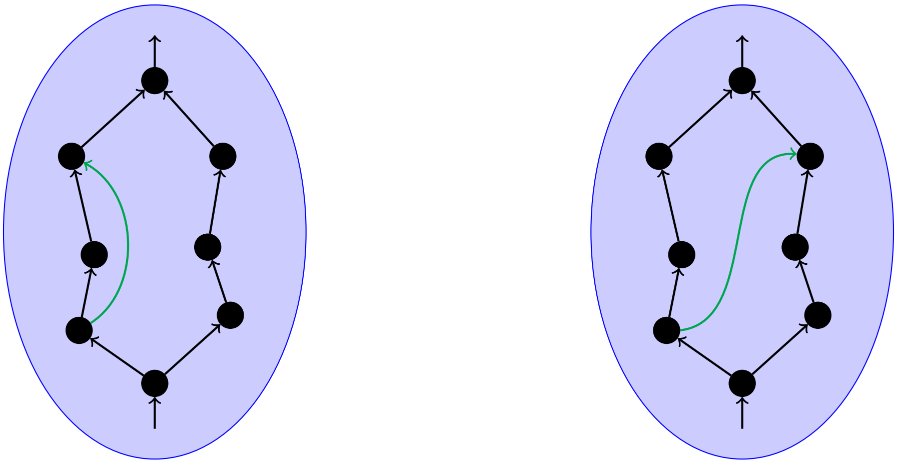


Figure 7.1: An illustration to adding an ear in the construction of an $s-t$ numbering and a corresponding upward drawing.

## 7.2 Rectangle visibility representations

**Definition 16.** *A* ***rectangle visibility representation*** *of a plane graph $G$ is an arrangement of disjoint axes-aligned rectangles in the plane such that the (unions of) horizontal sides of the rectangles correspond to the vertices of $G$, the (unions of) vertical sides to the faces of $G$, and each rectangle corresponds to the edge joining the vertices containing the upper and lower sides of the rectangle, and at the same time to the dual edge joining the faces corresponding to the left and right sides of the rectangle.*

**Comment.** *In this section we allow multiple edges without explicitly talking about multigraphs. Moreover, we artificially choose two vertices on the boundary of the outerface and divide the outerface into two faces by "infinite" dummy edges starting in these points.*

**Theorem 34.** *Every planar vertex 2-connected graph has a rectangle visibility representation.*

*Proof.* We in fact describe an algorithm how such a representation can be constructed. The bonus is that the construction runs in polynomial time. □

It is, however, necessary to prove that this Algorithm really outputs a rectangle visibility representation of $G$. This is done via a series of claims. The first ones talk about the upward drawing of $G$ constructed in Step 1 and the dual graph $G^*$ and its drawing inherited from the drawing of $G$.

**Claim 35.** *Vertex number 1 and vertex number $n$ are both on the boundary of the outerface of $G$ (and also of $G'$). The face that gets the name $A$ is the unbounded face and the face with the highest name, say $Z$, is the other face incident with the dummy edge $1n$.*

**Claim 36.** *The orientation of $G^*$ described in Step 4 is acyclic and $A$ is its only source and $Z$ is its only sink. Hence it is indeed an $s-t$ numbering of $G^*$. To see this, note first that the edge $AZ$ which crosses the dummy edge $1n$ cannot be involved in any directed cycle, as $A$ is a source and $Z$ is a sink in $G^*$. Next observe that a clock-wise oriented cycle in $G^*$ would bound a region with at least one vertex of $G$ inside and all edges of $G$ crossing this cycle would be oriented from inside towards the outside of this region, hence, there would necessarily be a source of $G$ in this region, and this would be different from the vertex 1. Similarly, a counter-clock-wise oriented cycle in $G^*$ would bound a region that would contain a sink different from $n$. This would be a contradiction with the assumption that we were working with an $s-t$-numbering of $G$. Finally, a source different from $A$ (a sink different from $Z$) in $G^*$ would yield a directed cycle on the boundary of this face of $G$. A contradiction again.*

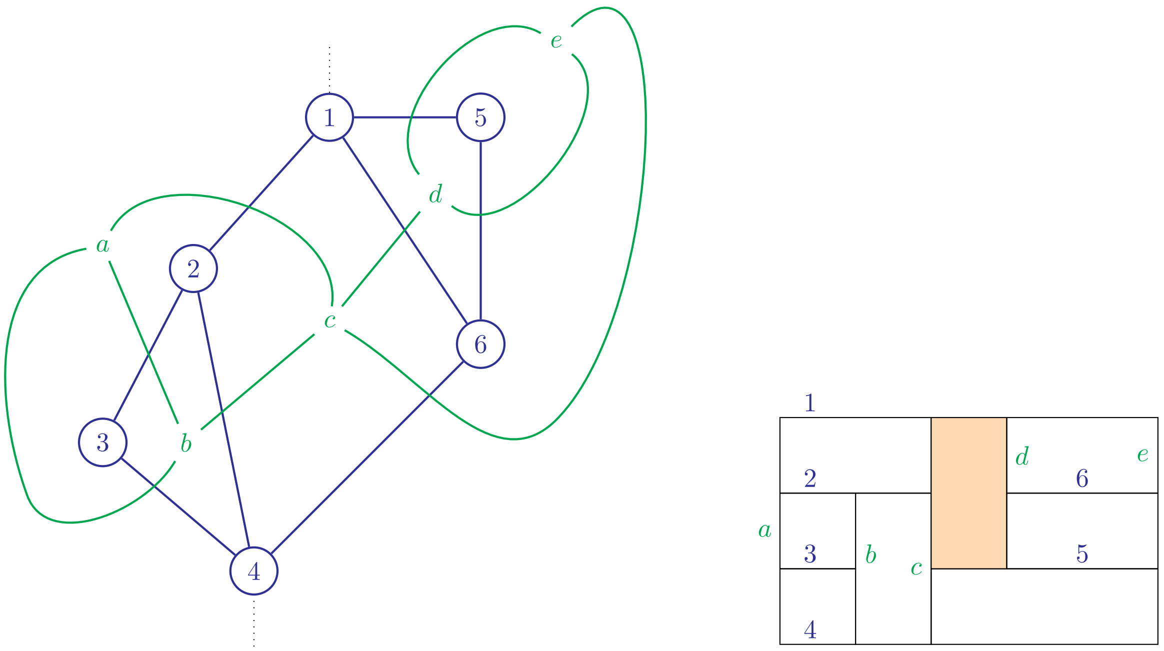


Figure 7.2: An example of a rectangle visibility representation of a planar graph. The highlighted rectangle in the representation corresponds to the primal edge 13 and dual edge *cd*.

**Algorithm 5** RectangleVisibilityRep

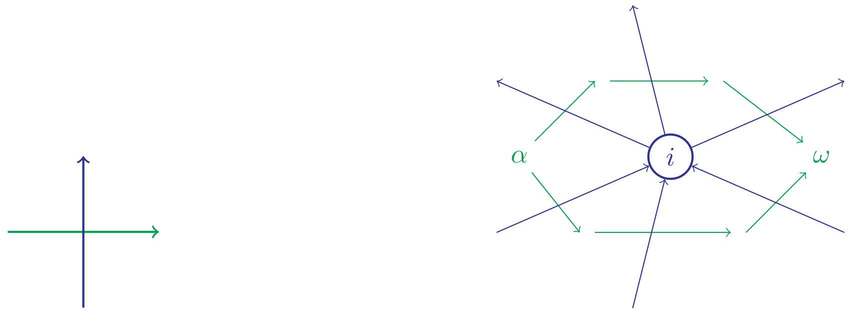
```
Require: A planar graph $G$.
1: Construct an $s-t$ numbering and a corresponding upward planar drawing as in Theorem 33.
2: Number the vertices as $1,\dots,n$ according to their $y$-coordinates (1 is the lowest vertex, $n$ is the highest one).
3: Add a dummy arc from 1 to $n$ drawn to the right of the drawing of $G$, denote by $G'$ the resulting graph.
4: Construct the dual graph $G^*$ to $G'$, and orient its edges so that each edge of $G$ is crossed by its dual edge from left to right, while the added dummy edge is crossed from right to left. Let $n^*$ be the number of vertices of $G^*$.
5: Consider a topological sorting of the vertices of $G^*$ and name them $A, B, \dots$ according to this sorting.
6: Take a grid of size $n \times n^*$. For a vertex $i$ of $G$, let $\alpha(\beta)$ be the face incident with $i$ with the lowest (highest, respectively) name in the topological sorting of $G^*$. Represent $i$ by a horizontal segment on the $i$-th line, starting at the vertical line $\alpha$ and ending on the vertical line $\beta$. For a face $\alpha$ of the drawing of $G'$ (i.e., a vertex of the dual graph $G^*$), let $i$ be its lowest vertex and $j$ its highest vertex (with respect to the topological sorting of $G$). Represent $\alpha$ by a segment on the vertical line $\alpha$, starting on the $i$-th horizontal line and ending on the $j$-th one.
7: return this representation.
```

**Claim 37.** *The boundary of every face $\alpha$ of $G$ consists of two directed paths, the left one and the right one, both connecting the vertex of the lowest number to the vertex of the highest number among the vertices of this face. We have seen this in the proof of Theorem 33.*

**Claim 38.** *For every vertex $i$ of $G, i \neq 1, n$, the faces incident with $i$ induce two directed paths in $G^*$, both connecting the face to the left of $i$ to the face lying to the right of $i$, one of the paths passing through the faces for which $i$ is their topmost vertex, the other one passing through the faces for which $i$ is their lowest vertex. See Fig. 7.3 right.*

In the next claims we prove that the collection of segments constructed in Step 6 defines a rectangle visibility representation of $G$.

**Claim 39.** *Let $i$ be a vertex of $G, i \neq 1, n$. Let $\alpha$ and $\omega$ be the faces to the left and to the right of $i$, respectively. Then the (horizontal) segment $i$ touches the vertical segment for $\alpha$ from the right, it touches the vertical segment for $\beta$ from the left, it is touched by the segments representing the faces lying on the upper path from $\alpha$ to $\beta$ in $G^*$ from above, and it is touched by the faces lying on the lower path from $\alpha$ to $\beta$ in $G^*$ from below. The*



(a) Orientation of the dual edges.

(b) An example to the statement of Claim 38.

Figure 7.3: Examples for claims.

*segment representing vertex* 1 *spans the whole range from* $A$ *to* $Z$ *and is only touched by vertical segments from above, while the segment representing* $n$ *is only touched by vertical segments from below, and also spans the whole range from* $A$ *to* $Z$*.*

**Claim 40.** *Let* $\alpha$ *be an inner face of* $G$*, i.e., a face not incident with the dummy edge* $1n$*. The vertical segment representing* $\alpha$ *touches the horizontal segment representing its lowest vertex from above, it touches the segment representing its topmost vertex from below, it is touched by the segments representing the vertices on the left boundary path of* $\alpha$ *from the left and it is touched by the segments representing the vertices on the right boundary path of* $\alpha$ *from the right. The segment representing* $A$ *is touched only from right, while the segment representing* $Z$ *is touched only from left, always by the appropriate horizontal segments.*

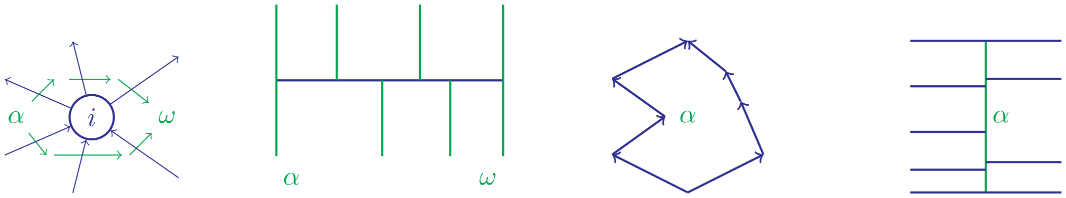


Figure 7.4: Illustration to Claims 39 and 40.

**Claim 41.** *For every edge* $ij$ *of* $G$*, let* $\alpha\beta$ *be its dual edge. Then the segments representing* $i, j, \alpha$ *and* $\beta$ *bound a rectangle in the representation.*

**Claim 42.** *No two segments constructed in Step 6 cross each other. For suppose segment* $j$ *crosses segment* $\alpha$*. By the way the segments are constructed, this means that there exist vertices* $i, k, i < j < k$*, and* $\beta, \gamma, \beta < \alpha < \gamma$*, such that* $i$ *is the lowest and* $k$ *the topmost vertex of face* $\alpha$ *and* $\beta$ *is the left and* $\gamma$ *the right face incident with* $j$*. Consider the upward drawing of* $G$ *and the horizontal stripe of vertices with their y-coordinates between* $i$ *and* $k$*. The face* $\alpha$ *spans this stripe from its bottom to its top lines, and thus it lies either to the left or to the right of vertex* $i$*. Suppose it is to the left. The horizontal ray starting in vertex* $i$ *and pointing to the left passes first through the face* $\beta$ *and then crosses several faces until it finally crosses* $\alpha$*. Since all edges of* $G$ *it crosses on this way are directed upward, these faces form a path directed from* $\alpha$ *to* $\beta$ *in* $G^*$*, which means that* $\alpha < \beta$ *in the topological sorting of* $G^*$*. Which is a contradiction.*

**Claim 43.** *The rectangles from Claim 41 are disjoint and fill in the base rectangle formed by segments* $A, Z, 1, n$*. This now follows from the previous claims. And this means that we have indeed constructed a rectangle visibility representation of* $G$*.*

## 7.3 Grid Contact graphs

**Definition 17.** *A graph is a* ***Grid Intersection graph*** *if it has an intersection representation by vertical and horizontal segments in which no two segments of the same direction share a point (in other words, all vertical, as well as all horizontal, segments are pairwise disjoint). A graph is a* ***Grid Contact*** *graph if it has a Grid Intersection representation in which no two segments cross, i.e., any two non-disjoint segments only touch each other.*

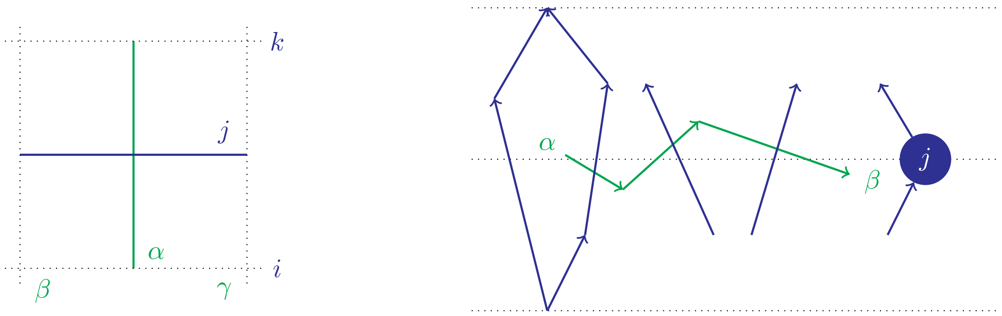


Figure 7.5: An illustration to Claim 42.

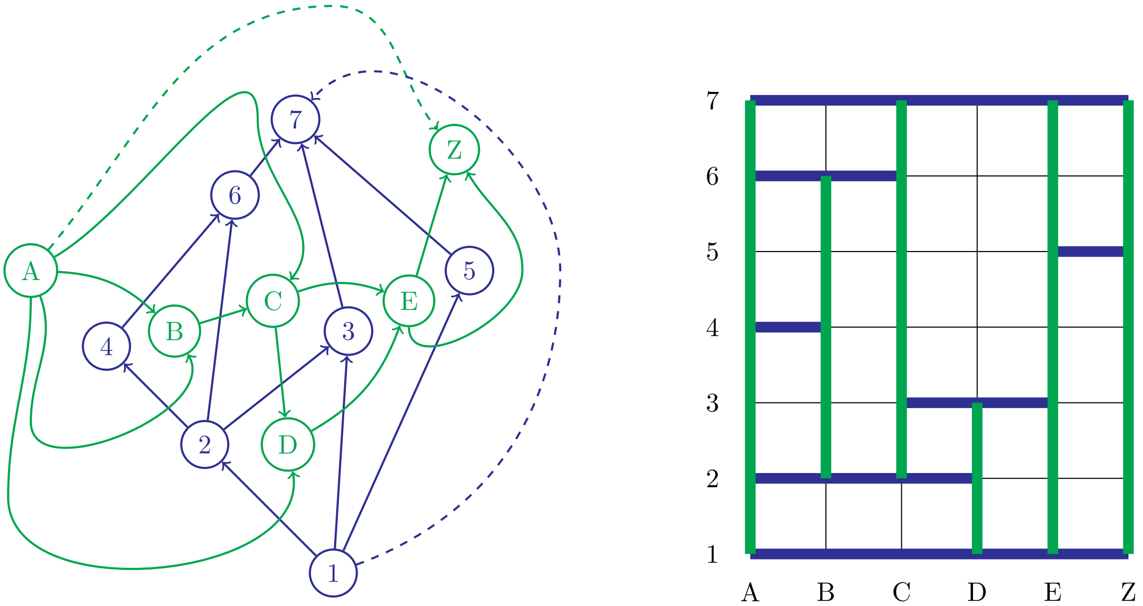


Figure 7.6: An overview of the construction by Algorithm RectangleVisibilityRep.

**Proposition 44.** *Every grid intersection graph is bipartite. Moreover, every grid contact graph is planar.*

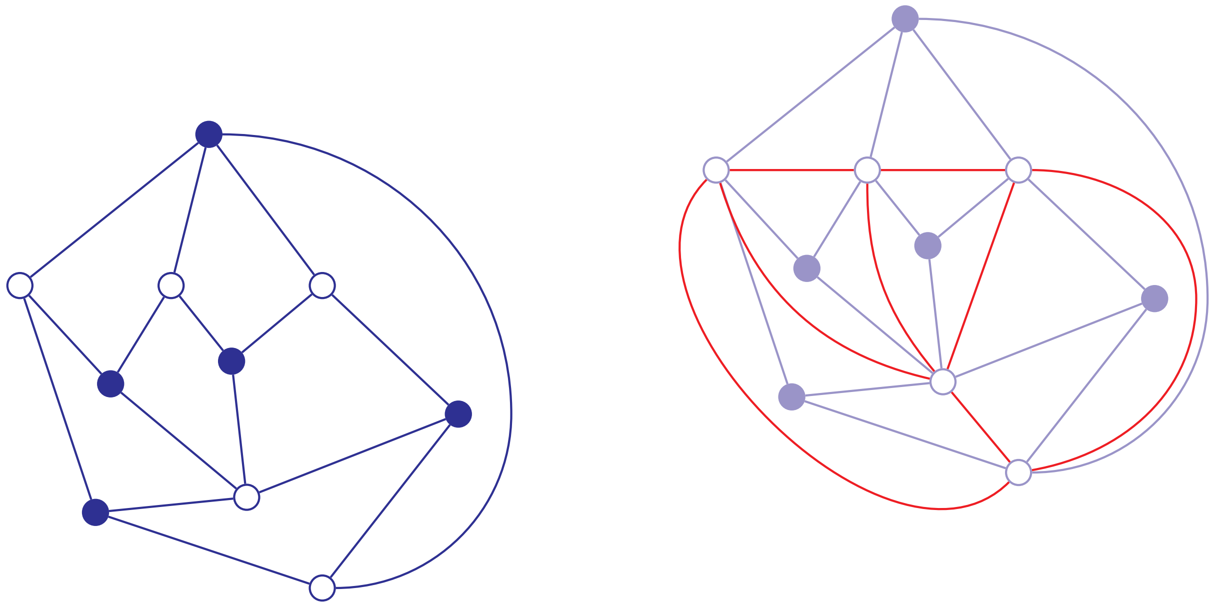
*Proof.* The first claim is a simple observation. For the second claim, note that grid contact graphs form a subclass of triangle-free contact graphs of arc-connected regions in the plane. All such graphs are planar, since a non-crossing drawing can be constructed from a contact representation by selecting a point inside each region to represent its vertex, and connecting it to the contact points with the adjacent regions by curves inside the region. □

**Theorem 45.** *Every planar bipartite graph is a grid contact graph.*

*Proof.* Given a planar bipartite graph $G = (A \cup B, E)$, consider a non-crossing drawing and extend it to a non-crossing drawing of a supergraph $G' = (A' \cup B', E')$ such that

- $G$ is an induced subgraph of $G'$,
- every face of the drawing of $G'$ is bounded by a cycle of length 4 (i.e., $G'$ is a so called quadrangulation),
- every vertex of $G'$ has degree greater than 2, and
- no vertex of $B$ is on the boundary of the outerface of $G'$.

Then construct the graph $\bar{G} = (A', \bar{E})$ by putting $\bar{E}$ the diagonals of the faces of $G'$ connecting their $A'$-vertices. It can be easily seen that $G$ is vertex 2-connected and that the faces of $\bar{G}$ are in 1-1 correspondence with the vertices of $B'$. Thus the segments of a rectangle visibility representation of $\bar{G}$ constructed as in the proof of Theorem 34 form a grid contact representation of $G'$ . Note the technical detail that the vertex (of $B'$) that corresponds to the outerface of $\bar{G}$ is represented by two vertical segments, not one. But this vertex does not belong to $B$, and so the segments corresponding to the vertices of $G$ form a grid contact representation of $G$. □



(a) The graph $G'$.

(b) The graph $G$.

Figure 7.7: An illustration to the proof of Theorem 45.

# Chapter 8

# Schnyder woods

## 8.1 Canonical ordering

**Definition 18.** *Given a plane triangulation (i.e., a non-crossing embedding of a triangulation in the plane, which means that the outerface is fixed) $G = (V, E)$, a canonical ordering of it is a numbering $1, 2, \dots, n$ of its vertices such that*

*i) the vertices of the outerface are $1, 2$ and $n = |V|$,*

*ii) for every $i = 3, 4, \dots, n-1$,*

*a) the graph $G_i = G[1, 2, \dots, i]$ is vertex 2-connected,*

*b) all vertices $j \leq i$ are drawn inside (or on the boundary) of the embedding of $G_i$ inherited from the embedding of $G$,*

*c) all vertices $j > i$ are drawn outside the embedding of $G_i$ inherited from the embedding of $G$,*

*d) the neighbors of vertex $i + 1$ in $G_i$ are lying consecutively on the boundary of $G_i$.*

**Theorem 46.** *Every triangulation has a canonical ordering.*

*Proof.* By induction from $n$ downto 3, assign the numbers to the vertices. Once $n, n-1, \dots, i+1$ are assigned, choose as vertex $i$ such a vertex on the boundary of $G_i$ whose deletion from $G_i$ leaves $G_{i-1}$ vertex 2-connected. The only obstacle to preserving 2-connectedness is if $i$ would be incident to a diagonal edge (an edge with both end-vertices on the boundary of the outerface, but which itself is not a part of the boundary). But there is always a vertex which is not incident with any diagonal (to observe this, consider a vertex which is incident with a shortest possible diagonal, with the length of a diagonal being measured by the number of vertices of the boundary that it cuts off of $G_i$). So assign $i$ to a vertex (there may be more options) which is not incident to any diagonal. Then a), b) and c) are fulfilled for $G_{i-1}$. Note that d) follows from b) and c). □

## 8.2 Schnyder woods

**Algorithm 6** Schnyder

```
Require: A plane triangulation $G = (V, E)$ and a canonical ordering of it.
for $i := 3 \dots n-1$ do
    set $b(i)$ to be the leftmost neighbor of $i$ on the boundary of $G_{i-1}$, direct the edge $ib(i)$ in this direction and color it blue;
    set $g(i)$ to be the rightmost neighbor of $i$ on the boundary of $G_{i-1}$, direct the edge $ig(i)$ in this direction and color it green;
    set $r(i)$ to be the neighbor of $i$ with the highest number, direct the edge $ig(i)$ in this direction and color it red
end for
return the orientation of $G$, the coloring of its edges and the mappings $b, g$ and $r$.
```

**Theorem 47.** *The blue edges form a tree rooted in vertex 1 and spanning the vertices $1, 3, \dots, n-1$, the green edges form a tree rooted in vertex 2 and spanning the vertices $2, 3, \dots, n-1$, and the red edges form a tree rooted in vertex $n$ and spanning the vertices $3, \dots, n$. Every edge of $G$, except of $12, 1n, 2n$, belongs to exactly one of these trees.*

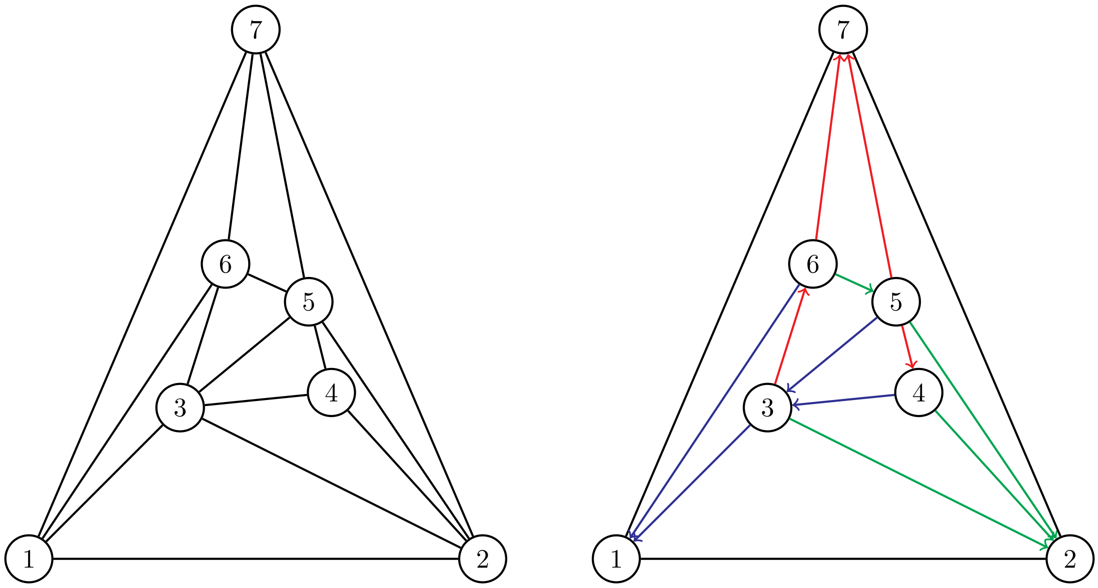


Figure 8.1: An illustration to canonical orderings and Schnyder woods.

*Proof.* Clear from the construction and properties of canonical orderings. □

*Corollary.* The edges of a planar triangulation can be partitioned into edge sets of 3 trees and a triangle. Such a collection of three trees is called a **Schnyder wood** of $G$.

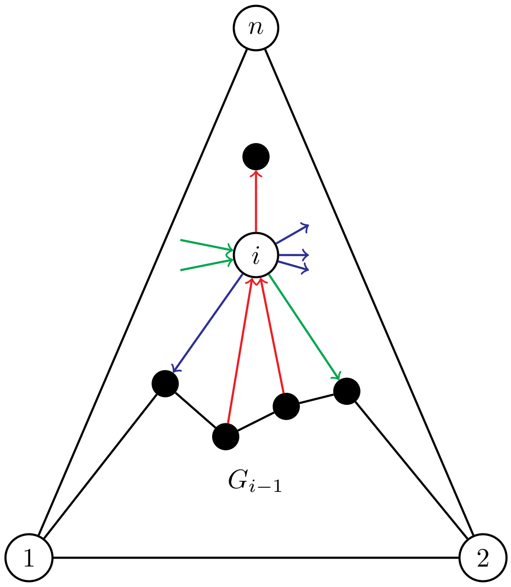


Figure 8.2: The rotation scheme of incoming and outgoing edges around a vertex in the Schnyder wood.

**Proposition 48.** *Locally around every inner vertex of $G$, we see (in the counter-clock-wise order) one outgoing blue edge, several (or none) incoming red edges, one outgoing green edge, several (or none) incoming blue edges, one outgoing red edge, and several (or none) incoming green edges.*

## 8.3 Triangle contact representations

**Theorem 49** (de Fraysseix, Ossona de Mendez, Rosenstiehl)**.** *Every planar graph is a contact graph of isosceles triangles with horizontal bases.*

*Proof.* It suffices to prove the theorem for triangulations, since every planar graph is an induced subgraph of a triangulation. Given a triangulation $G = (V, E)$, fix an embedding and consider a canonical ordering with respect to this embedding, and run algorithm Schnyder on this canonical ordering. Draw $n+1 = |V|+1$ parallel horizontal lines and build triangles as follows:

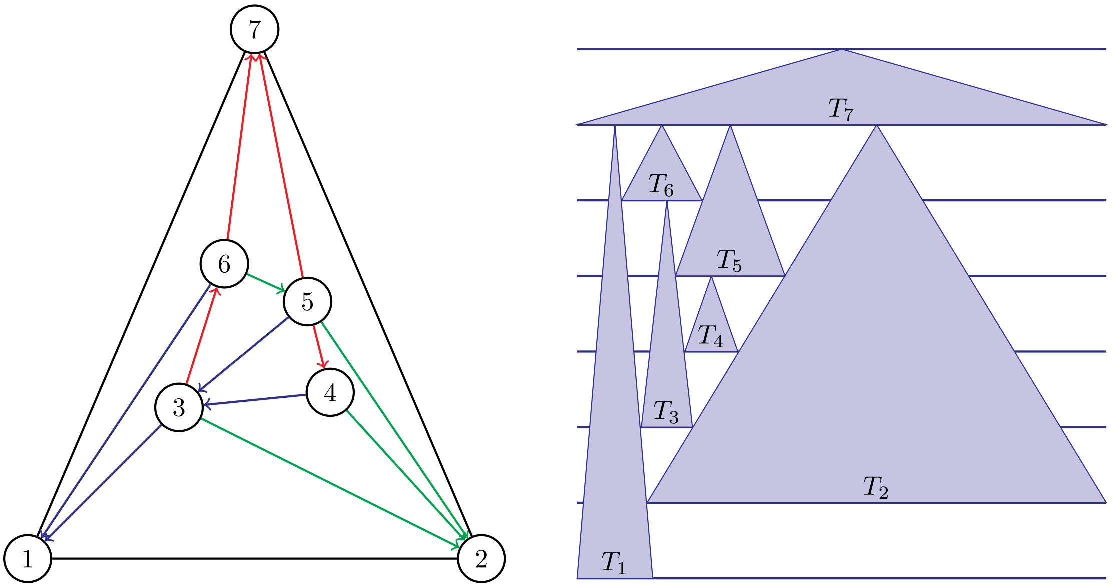


Figure 8.3: An illustration to contact representations by isosceles triangles.

1. Triangle $T_i$ is isosceles and its base lies on the $i$-th line,
2. The peaks of $T_1$ and $T_2$ are on the $(n+1)$-st line, the left corner of $T_2$ touches the right side of $T_1$,
3. For every $i = 1, 2, \ldots, n$, the left corner of $T_i$ lies on the right side of $T_{b(i)}$, the right corner of $T_i$ lies on the left side of $T_{g(i)}$ and the peak of $T_i$ lies on the $r(i)$-th line (on the $(n+1)$-st line for $i = n$).

Construct the triangles from $T_1$ to $T_n$. For $T_1$, only the lines supporting its base and peak are prescribed, the triangle is free otherwise. For $T_2$, the freedom is restricted only to the position of the right corner (which then determines the position of the peak). For $i > 2$, the triangles are then determined uniquely. The loop invariant of this inductive construction is that for $i > 1$, the upper boundary of the union of triangles $T_1, T_2, \ldots, T_i$ is connected and the order in which the triangles appear on this boundary is the same as the order of the corresponding vertices appearing on the upper boundary of $G_i$. This implies that $T_i$ is always placed in the way that it is touching the respective neighbors, but not crossing any triangle of the representation. □

## 8.4 Drawing planar graphs on small grids

**Theorem 50.** *Every planar $n$-vertex graph allows a straight-line non-crossing embedding on a grid of size $n \times n$.*

*Proof.* It suffices to prove the theorem for planar triangulations. Given a triangulation $G = (V, E)$, fix an embedding and consider a canonical ordering with respect to this embedding, and run algorithm Schnyder on this canonical ordering. Assign barycentric coordinates $(x_i, y_i, z_i)$ to every vertex $i = 3, 4, \ldots, n-1$ as follows (see Fig. 8.4 for illustration): The triangle $12n$ is divided into three regions by the blue, green and red directed paths from $i$ to the roots of the trees in the Schnyder wood. Let $x_i$ be the number of vertices in the region bounded by the blue and green paths and the side 12, with the vertices on the green path being counted in, but not the vertices of the blue path. Similarly, $y_i$ is the number of vertices in the region bounded by the blue and red paths and the side $1n$, with the vertices on the blue path being counted in, but not the vertices of the red path, and $z_i$ is the number of vertices in the region bounded by the red and green paths and the side $n2$, with the vertices on the red path being counted in, but not the vertices of the green one. Vertex $i$ itself is not counted in neither of the regions.

Then every vertex except of $i$ belongs to exactly one of the regions, end hence

$$x_i + y_i + z_i = n - 1$$

for every $i$. For $i = 1, 2$, and $n$, we set

$$\begin{array}{lll} x_1 = 0, & y_1 = 0, & z_1 = n-1 \\ x_2 = 0, & y_2 = n-1, & z_2 = 0 \\ x_n = n-1, & y_n = 0, & z_n = 0 \end{array}$$

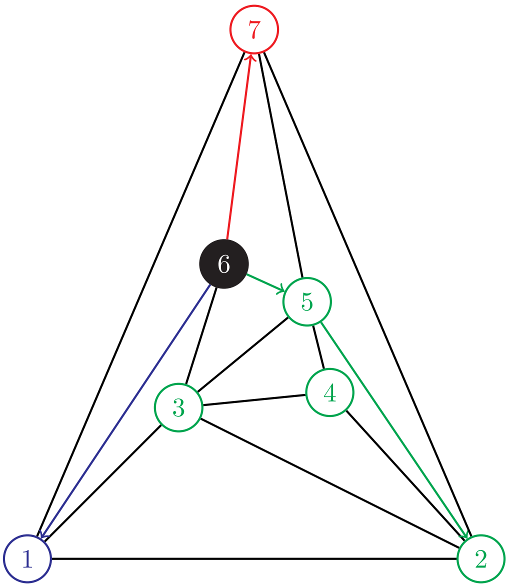


Figure 8.4: An illustration to the definition of barycentric coordinates from Schnyder woods. The green vertices are counted for the definition of $x_6$, the blue one for the definition of $y_6$ and the red one for the definition of $z_6$. Vertex 6 itself does not contribute to any of the coordinates.

Place vertex $i$ in the point with barycentric coordinates $(x_i, y_i, z_i)$ in a triangular $n \times n \times n$ grid (the lines of the grid have coordinates $0, 1, \ldots, n-1$). Draw the edges of $G$ as straight-line segments connecting their vertices.

**Claim 51.** *Consider a vertex $i \in \{3, 4, \ldots, n-1\}$. In the drawing constructed as above, the blue edge $ib(i)$ is directed into the bottom-left sextant of $i$, the edge $ig(i)$ is directed into the bottom-right sextant of $i$, and the edge $ir(i)$ is directed into the top sextant of $i$.*

*Proof of claim.* From the definition of the coordinates, it follows that $xb(i) \leq x_i$, $yb(i) < y_i$ and $zb(i) \geq z_i$, and this implies the direction of the edge $ib(i)$. Similarly for the others. □

**Claim 52.** *The rotation scheme of the edges around each vertex $i$ in the barycentric drawing is the same as the rotation scheme of the edges incident to the same vertex in $G$.*

*Proof of claim.* This follows by application of Claim 51 to the other end-vertices of the edges directed into $i$. □

**Claim 53.** *The barycentric drawing is non-crossing and topologically equivalent to the plane embedding of $G$ we started with.*

*Proof of claim.* If the rotation schemes for all vertices of drawings of two 3-connected graphs are the same and one of the drawings is non-crossing, then so is the other one, and the drawings are topologically equivalent. □

□

## 8.5 Boxicity of graphs

**Definition 19.** *The **boxicity** of a graph $G$, denoted by $box(G)$, is the smallest integer $d$ such that $G$ is an intersection graph of boxes in $\mathbb{R}^d$ (i.e., of $d$-dimensional intervals in the $d$-dimensional Euclidean space).*

**Proposition 54.** *For every graph $G = (V, E)$, its boxicity is a correctly defined finite number. It equals the minimum number of interval graphs whose intersection is equal to $G$, i.e., the minimum number $d$ for which sets $E_i \subseteq \binom{V}{2}, i = 1, 2, \ldots, d$ exist, such that $E = \cup_{i=1}^{d} E_i$ and $(V, E_i)$ is an interval graph for every $i = 1, 2, \ldots, d$.*

*Proof.* An exercise. Proof the claim for graphs $G$ which are not complete. For a complete graph, the boxicity is 0, which corresponds to the fact that the intersection of an empty set of subsets of a ground set ($E$, in this case) is by default set to be equal to the ground set itself. □

## 8.6 Grid intersection graphs

**Definition 20.** *A graph is a* ***Grid Intersection graph*** *if it has an intersection representation by vertical and horizontal segments in which no two segments of the same direction share a point (in other words, all vertical, as well as all horizontal, segments are pairwise disjoint).*

It is easy to observe that every grid intersection graph has a grid intersection representation in which no two segments lie on the same line. Moreover, the exact $x$-coordinates of the vertical segments, nor the exact $y$-coordinates of the horizontal ones, are important, only their linear orders. Thus we assume that our bipartite graph $G = (A \cup B, E)$ has its vertices linearly ordered within the classes of bipartition, $A = \{a_1, a_2, \dots, a_n\}, B = \{b_1, b_2 \dots, b_m\}$. We further assume that the vertices of $A$ are to be represented by vertical segments and the vertices of $B$ by horizontal ones. We say that a grid intersection representation respects the orders of $A$ and $B$ if for every $i < j$, the $x$-coordinate of $a_i$ is smaller than the $x$-coordinate of $a_j$, and the $y$-coordinate of $b_i$ is smaller than the $y$-coordinate of $b_j$.

**Theorem 55.** *The graph $G = (A \cup B, E)$ has a grid intersection representation that respects the linear orders of $A$ and $B$ if and only if there are no 6 indices $i < j < k, \alpha < \beta < \gamma$ such that $a_i b_\beta, a_k b_\beta, a_j b_\alpha, a_j b_\gamma$ are edges of $G$ and $a_j b_\beta$ is not. Such a configuration of 6 vertices is called a* ***volswagen*** *in $G$.*

*Proof.* Exercise. □

It is easy (i.e., decidable in polynomial time) to check if a bipartite graph contains a volkswagen with respect to given linear orderings of the vertices in its classes of bipartition. We will see in the last class that it is NP-complete to decide if a given bipartite graph is a grid intersection graph, which means that it is NP-complete to decide if a given bipartite graph allows linear orderings of its classes of bipartition with respect to which there is no volkswagen. The following problem is thus quite interesting in this context.

**Open problem:** Given a bipartite graph with one class of bipartition linearly ordered. How difficult is to decide if the other class of bipartition can be linearly ordered so that there is no volkswagen with respect to these orderings?

**Theorem 56** (Bellantoni, Hartman, Przytycka, Whitesides)**.** *Every bipartite graph of boxicity 2 is a grid intersection graph.*

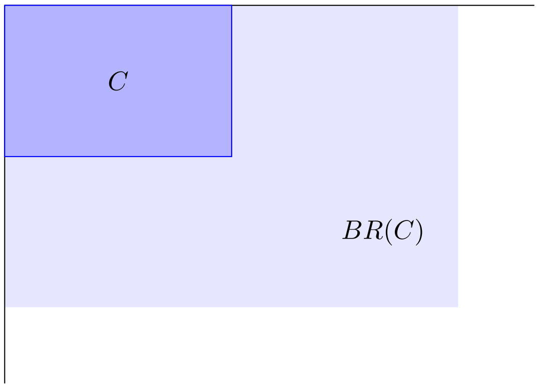


Figure 8.5: An illustration to the definition of region $BR(C)$.

*Sketch of proof.* Suppose $G = (A \cup B, E)$ is a bipartite graph and let $\mathcal{R} = \{R(u) : u \in A \cup B\}$ be an intersection representation of $G$ by axes-parallel rectangles in the plane. For a rectangle $C$ in the plane, we define regions $BL(C)$, $BR(C)$, $TL(C)$, $TR(C)$ as follows: $BR(C)$ contains all points with $x$-coordinate greater than the $x$-coordinate of the left side of $C$, with $y$-coordinate smaller than the $y$-coordinate of the top side of $C$, but which do not lie inside the rectangle $C$. The other regions ($BL$ standing for Bottom-Left, $TL$ standing for Top-Left, and $TR$ standing for Top-Right) are defined in a similar way. Then define two binary relations, one on $A$, the other one on $B$, as follows

$$\begin{array}{lcll} a_1 <_R a_2 & \Leftrightarrow & R(a_1) \cap BR(R(a_2)) \neq \emptyset & \text{for } a_1, a_2 \in A, \\ b_1 <_R b_2 & \Leftrightarrow & R(b_1) \cap BL(R(b_2)) \neq \emptyset & \text{for } b_1, b_2 \in B. \end{array}$$

**Claim 57.** *Both relations $<_R$ and $<_L$ are antireflexive, antisymmetric and acyclic. It follows that the transitive closures $<_R^T$ and $<_L^T$ of $<_R$ and $<_L$, respectively, are partial orders.*

Let $<_R^*$ and $<_L^*$ be topological sortings of $<_R^T$ and $<_L^T$, respectively.

**Claim 58.** *With respect to the linear orderings $<^*_R$ and $<^*_L$ of its classes of bipartition, $G$ has no volkswagens. Hence $G$ has a grid intersection representation respecting these orders by Theorem 55.*

□

# Chapter 9

# String graphs

**Definition 21.** *String graphs are intersection graphs on arbitrary lines in plane.*

**Definition 22.** ***OuterString** class is a class of graphs which are represented by intersection graphs of lines starting from a circle boundary and growing arbitrarily inside the circle. Then **constrained outer strings** is class containing graphs that are outer strings but also it is determined the order of lines appearance on the circle.*

[TODO] *Create some pictures.*

**Theorem 59.** *For $n \geq 4$ the complement of $C_n$, $-C_n \notin$ constrained-outerstring.*

*Proof.* We proceed by induction on $n$. First we start by $n = 4$. Where if we draw the graph and because on the circle the two opposite nodes have to connect somewhere, but not cross any other. That is impossible.

For greater $n > 4$ suppose for contrary that $-C_n$ has a representation. Take such with smallest number of crossing number. Then we create two more graphs $-C_{i+1} = -C_n[1, 2, \dots, i]$ and $-C_{n-1+1} = -C_n[i, i+1, \dots, n]$ since $n \geq 5$ both have size at least 4 but less than the $-C_n$ but we know that one of them don't have a representation. □

**Theorem 60.** *Recognition of STRING is NP-hard.*

*Proof.* The proof is done by creating a transformation of Planar 3-SAT to RECOG (STRING). We won't go deeper into the proof, but shortly as usual you create some gadgets for variables and for clauses. Then you need to bind them. After that you use the planarity of 3-SAT to create a graph that is representable as a STRING graph only if the SAT is satisfied. That is it abuses the previously mentioned theorem.

Also it can be done by straight lines. Therefore also recognition of $GI$ is NP-complete. □

**Theorem 61.** *RECOG(SEG), RECOG(CONV) are in $\exists\mathbb{R} \subseteq$ PSPACE.*

This can be proven by the following $\exists\mathbb{R}$ problem. We ar egiven $n$ polynoms $p_1, \dots, p_n$ for $n$ variables $x_1, \dots, x_n$. The question is then whether there exists such assignment to $x_1, \dots, x_n \in \mathbb{R}$ that all polynoms are non-negative.

*Proof.* We will represent $G$ for each $u \in V(G)$ as $M_u$ convex set. Then we set points $\forall uv \in E(G) : P_{uv} = (x_{uv}, y_{uv})$. After that we take a convex hull of all inner points. Therefore we denote $M'_u = \text{conv}(P_{uv} | v \in N_G(v))$. Now $(M'_u | u \in V(G))$ is convex reprezentation of $G$.

If $M_u \cap M_v \neq \emptyset \Rightarrow P_u v \in M'_u \cap M'_v \neq \emptyset$ if $uv \notin E(G)$ there exists a line separating these two convex hulls. This line can be prescribed as a polynom. Therefore these polynoms must be negative for all non-edges.

For the RECOG(SEG) we may use similiar trick. □

**Comment.** *If $G \in$ STRING then mwe must find such lines instead of nodes, but if we pick points representing edges then the graph is planar. So we must guess such planar graph $H$ and then the paths. But this is not enough for NP since the polynomial is dependent on the size of the unknown $H$.*

**Definition 23.** *$STR(n)$ is min $k$ such that $\forall G \in$ STRING $|V(G)| = n$ there exists STRING representation with at most $k$ crossing points.*

**Definition 24.** *AT-graf is $(G = (V, E), R \subset \binom{E}{2})$ and it is (weakly) realisable if there exists drawing $D$ of a graph $G$ in the plane s.t. $\forall e, f \in E(G) : D_e \cap D_f \neq \emptyset \Rightarrow \{e, f\} \in R$.*

**Observation.** *$R = \emptyset$ then $(G, \emptyset)$ is realisable iff $G$ is planar.*

**Definition 25.** *$AT(n)$ is min $k$ s.t. $\forall$ realisable $AT$-graf with $n = |E(G)|$ is realisable with at most $k$ crossing points.*

**Theorem 62.** *$STR(n) \sim AT(n)$.*

*Proof.* "$\Rightarrow$" Let $G \in$ STRING and $|V(G)| = n, G$ needs STR($n$) crossing points. Lets take such representation $\forall u, v \in V(G)$ where $u \neq v$ and $uv \in E(G)$ we select point $P_{uv} \in S_u \cap S_v$ and say that these are nodes of the new graph $G'$. $S_u$ is a line representing $U \in V(G)$.

[TODO] *finish the proof*

□

**Theorem 63.** *$AT(n) \geq 2^{cn}$ for constant $c$.*

*Proof.* We will create a certain type of a graph depicted on the picture.

[TODO] *Finsih the proof and create a picture.*

□

**Theorem 64.** *RECOG(STRING) $\in$ NP.*

This will be shown in the second part of this course.

**Theorem 65.** *$AT(n) \leq n \cdot 2^n$.*

[TODO] *Finish the proof and also state supporting lemma.*

# Part II

# Geometric Representations of Graphs II

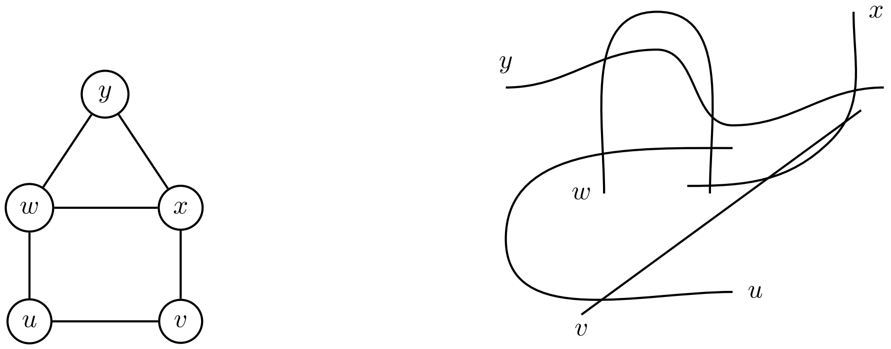
# Chapter 10

# STRING $\in$ NP

Now we will take a look at recognition of STRING graphs. Mainly that this problem is indeed in NP. We have already shown in the first part that STRING-recognition is NP-hard. The main problem is that some string graphs require representation with exponentially many crossings. Which is not possible to guess in an NP algorithm.

Also we have shown (Schaeffer and Štěfanovič) result that any string graph has a representation with at most exponentially many crossings.

Before we continue lets give us an example of a graph and its STRING representation which can be seen on Fig. 10.1.



(a) Graph $G$.

(b) String representation.

Figure 10.1: Example of a STRING graph.

Firstly lets define another problem which can be converted from STRING-recognition.

## 10.1 Weak AT-realization

The **INPUT** is a graph $G = (V, E)$ and $R \subseteq \binom{E}{2}$. The **GOAL** is to find a drawing of $G$ where only the pairs of edges from $R$ are allowed to cross.

Note that AT stands for abstract topological and weak means that the pairs do **not hove** to cross, only they can.

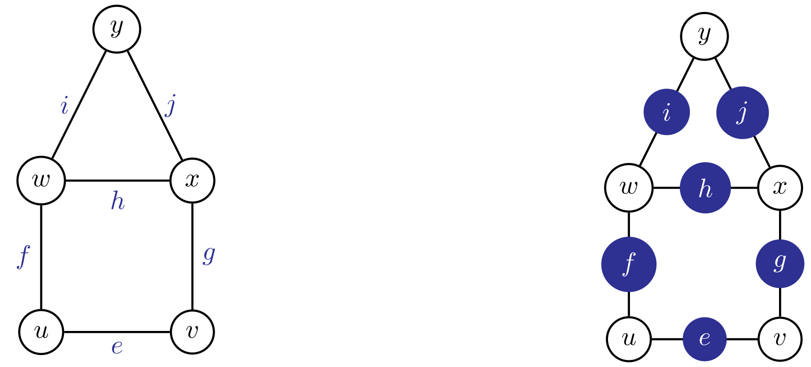
For now the drawing is somewhat basic. That is No edges crosses through a vertex. Vertices are points and edges are curves.

## 10.2 Reduction of STRING-recognition to Weak AT-realization

For a given graph $G = (V_G, E_G)$ we define the following graph $H = (V_H, E_H)$ and $R$. $V_H = V_G \cup E_G$ and $E_H = \{\{v, e\}, v \in V_G, e \in E_G, v \in e\}$. Lets see an example on Fig. 10.2. We may see that the graph is obtained by subdividing all edges.

Lastly we define $R = \{\{\{v, e\}, \{w, f\}\}, \{v, w\} \in E_G\}$.

**Claim 66.** *G is a STRING graph* $\Leftrightarrow$ $(H, R)$ *has Weak AT-realization.*



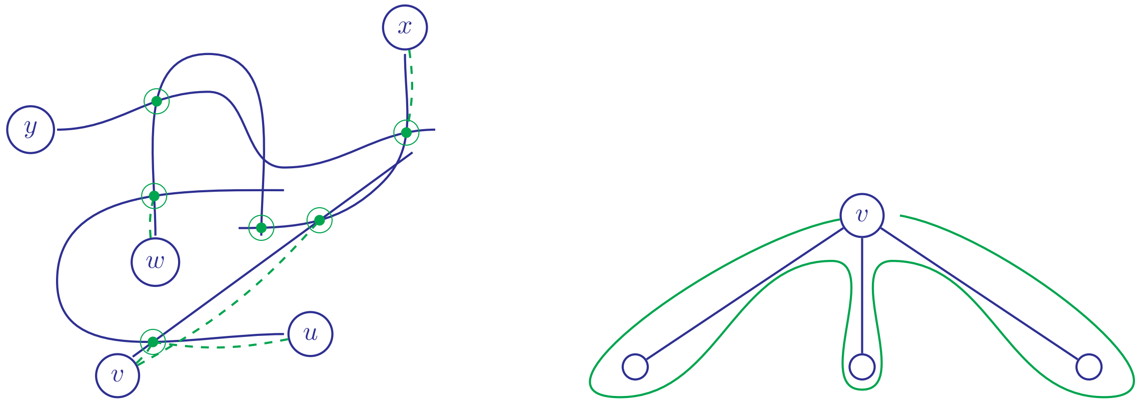
(a) Original graph $G$.

(b) Graph $H$.

Figure 10.2: Creating $H$ from $G$.

*Proof.* "$\Rightarrow$" Suppose $G$ is a STRING graph. Pick endpoints as vertices and crossings as edge-vertices of $H$. Choose these arbitrary (see Fig. 10.3a for an example). The edges will follow the lines in a STRING representation.

"$\Leftarrow$" Suppose $(H, R)$ has a Weak AT-realization. See that $H$ is bipartite. Now represent it by going from the vertex alongside every edge to the edge-vertices of $H$ and almost return to the vertex. See Fig. 10.3b for an example. It is easily observable that when lines cross then indeed they need be connected by an edge.



(a) $\Rightarrow$ (Only some edges are present.)

(b) $\Leftarrow$

Figure 10.3: Examples for the proof.

□

### 10.2.1 Test Weak AT-realization in NP

Firstly we will write down all our assumption about the drawing we will be using.

1. We are on the sphere.
2. Vertex $v$ is drawn as an open disc $D_v$ (also we will denote $\partial D_v$ as the boundary of $D_v$ and $\overline{D_v}$ as the closure of $D_v$, i.e. $\overline{D_v} = D_v \cup \partial D_v$).
3. An edge $e = \{v, w\}$ is represented by a curve $\gamma_e$ connecting a point from $\partial D_v$ to a point from $\partial D_w$ otherwise $\gamma_e$ is disjoint from $\bigcup_{x \in V} \overline{D_x}$.
4. For $e \neq f$ $\gamma_e$ and $\gamma_f$ have distinct endpoints.

We can easily see that these assumption are not restricting since it can be switch to "normal" drawing we are using by shrinking the discs to single points and switch back by expanding the single points.

Now we will proceed with the algorithm.

Before we continue with the algorithm we will take a short topological detour to some definitions and facts.

**Algorithm 7** NP algorithm for testing Weak AT-realization.

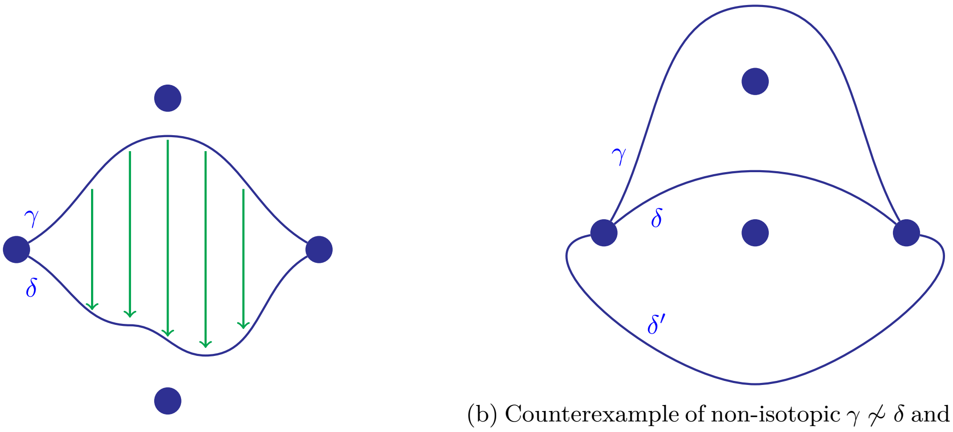
```
Require: $G = (V, E), R \subseteq \binom{E}{2}$
$\forall v \in V$ : choose $D_v$ (so that $v \neq w$ have $\overline{D_v} \cap \overline{D_w} = \emptyset$).
For any edge $e \in E$ incident to $v \in V$ : choose an endpoint of $\gamma_e$ on $\partial D_v$.
...
```

### 10.2.2 Topological detour

**Definition 26.** *$\mathcal{S} :=$ (sphere) $\setminus \bigcup_{v \in V} D_v$. Which is a compact surface with boundary.*

**Definition 27.** *$\mathcal{S} -$ curve is a curve in $\mathcal{S}$ with endpoints in the $\partial\mathcal{S}$ and no other point in $\partial\mathcal{S}$.*

**Definition 28.** *Two $\mathcal{S}$-curves $\gamma, \delta$ are **isotopic** (written as $\gamma \sim \delta$) if $\gamma$ and $\delta$ have the same endpoints and $\gamma$ can be transformed into $\delta$ by a continuous deformations of $\mathcal{S}$ which fixes the boundary.*



(a) Example of $\mathcal{S}$-curves $\gamma, \delta$ that are isotopic.

(b) Counterexample of non-isotopic $\gamma \nsim \delta$ and example of isotopic $\gamma \sim \delta'$.

Figure 10.4: Example of isotopic and non-isotopic $\mathcal{S}$-curves.

**Definition 29.** *For two $\mathcal{S}$-curves $\gamma_1, \gamma_2$ the intersection number $i(\gamma_1, \gamma_2)$ is*

$$\min_{\delta_1 \sim \gamma_1, \delta_2 \sim \gamma_2} |\delta_1 \cap \delta_2|.$$

**Fact.** *Let $\gamma_1, \dots, \gamma_n$ be $\mathcal{S}$-curves. Then there are $\mathcal{S}$-curves $\delta_1, \dots, \delta_n$ s.t.*

1. *$\delta_i \sim \gamma_i$ for $i = 1, \dots, n$,*
2. *$|\delta_i \cap \delta_j| = i(\gamma_i, \gamma_j)$ for $i \neg j \in \{1, \dots, n\}$.*

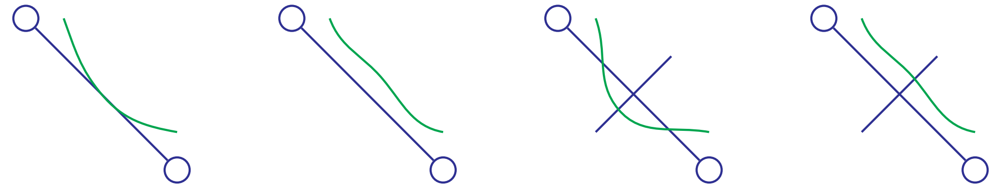
With all these definitions we will choose the following triangulation $T = (V_T, E_T)$ of $\mathcal{S}$:

1. $\forall v \in V : \partial D_v$ has three vertices and three edges of $T$.
2. There are no other vertices of $T$, the remaining edges are chosen arbitrarily as $\mathcal{S}$-curves, so that $\mathcal{S}$ is partitioned into triangle faces.
3. Vertices of $T$ do not coincide with endpoints of $\gamma_e, e \in E$ from algorithm.

**Definition 30.** *An $\mathcal{S}$-curve $\gamma$ is **normal** w.r.t. $T$ if*

1. *The endpoints of $\gamma$ are distinct from $V_T$.*
2. *Any point where an edge of $T$ meets $\gamma$ is either an endpoint of $\gamma$ or a proper crossing (no touching is allowed).*
3. *$\gamma$ does not have two consecutive crossings with any edge of $T$.*
4. *$\gamma$ does not cross the same edge of $T$ no more than $2^{|E_T|+|E|}$-times.*

**Claim 67.** *If $G$ has a weak AT-realization, then it has a realization where each $\gamma_e$ is normal w.r.t. $T$.*



(a) Second before. (b) Second after. (c) Third before. (d) Third after.

Figure 10.5: Examples for the proof.

*Proof.* Take AT with minimum number of crossings. 1. is satisfied by the definition. 2. otherwise minimality is broken, 3. again the minimality is broken and lastly 4. follows from GRG1. □

**Definition 31.** *Let $\gamma$ be an $\mathcal{S}$-curve, normal w.r.t $T$ for $e \in E_T : c_e(\gamma) := |e \cap \gamma|$.*

Sometimes when we will be talking about $\gamma_e$ for some $e \in E$ then for $f \in E_T$ we will denote $c_f(e)$ as $c_f(\gamma_e)$. Also in the context we will sometimes leave out $\gamma_e$ and only have $c_f$ or $c_{uv}$ if $f = \{u, v\}$.

**Claim 68.** *The numbers $\left(c_f(e)\right)_{f \in E_T}$ determine $\gamma_e$ up to isotopy.*

### 10.2.3 Word equations

Before we showcase the proof we will show us a quick introduction to word algebra and its equations. We will be considering an alphabet $\mathcal{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \dots\}$ finite and variables $\mathcal{V} = \{X, Y, \dots\}$. Each variable $X \in V$ has prescribed length $l(X) \in \mathbb{N}_0$ and represents a word over $A$ of length $l(X)$.

*Example.* $\mathcal{A} = \{\mathtt{a}, \mathtt{b}\}, \mathcal{V} = \{X, Y\}, l(X) = 2, l(Y) = 3$ and the equation is:

$$XX\mathtt{a} = XY,$$

where the operation is concatenation. A simple one solution os $X := \mathtt{ba}$ and $Y := \mathtt{baa}$.

We may see that having one equation or a whole system of equation is the same. As we can see here:

$$\begin{array}{l} L_1 = R_1 \\ L_2 = R_2 \\ \quad\vdots \end{array} \quad \Leftrightarrow \quad L_1 L_2 \dots = R_1 R_2 \dots$$

Also it may be that the size od a solution can be even exponential. But there is usually a repeating pattern. We can examine the following example to see that it is indeed true.

$$\begin{array}{rcl} X_1 & = & \mathtt{a} \\ X_2 & = & X_1 X_1 \\ X_3 & = & X_2 X_2 \\ & \vdots & \end{array}$$

**Fact.** *There is an encoding ("LZ-encoding") such that for any system of word equations we can in* ***polynomial time*** *(in $|\mathcal{A}| + |\mathcal{V}| + |equations| + \left|\sum_{X \in \mathcal{V}} \log |l(X)|\right|$) determine if a solution exists and compute the LZ-encoding of the lexicographicaly smallest solution.*

**Fact.** *From the LZ-encoding of a word $X \in \mathcal{A}^*$ we can compute the number of occurrences of any symbol $\mathtt{a} \in \mathcal{A}$ in $X$.*

#### LZ-encoding

Shortly we will talk about the exact LZ-encoding. Firstly for a word $\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3 \dots \mathtt{a}_n \in \mathcal{A}^*$, the **LZ-factorization** is an expresion of the form $\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3 \dots \mathtt{a}_n = \mathtt{b}_1 X_1 \mathtt{b}_2 X_2 \mathtt{b}_3 X_3 \dots \mathtt{b}_n X_n$ where $X_i$ is the longest possible sub-word that also occurs in $\mathtt{b}_1 X_1 \mathtt{b}_2 X_2 \mathtt{b}_3 X_3 \dots \mathtt{b}_i$.

Then for LZ-encoding we replace each $X_i$ with a point to previous occurrence and the length of $X_i$.

*Example.* Here is a simple example of converting a word to LZ-factorization and then to LZ-encoding.

$$\begin{array}{cccccccccc} \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{b} & \mathtt{a} & \mathtt{a} & \mathtt{b} & \mathtt{b} & \mathtt{b} \\ \mathtt{a} & X_1 & \mathtt{b} & X_2 & & \mathtt{a} & X_3 & & \mathtt{b} & X_4 \\ \mathtt{a} & (1,1) & \mathtt{b} & (2,2) & & \mathtt{a} & (2,2) & & \mathtt{b} & (3,1) \end{array}$$

Now we will return back to the claim.

*Proof of claim 68.* To clarify this we must showcase the following two points.

1. Verify that for every $e \in E$, the numbers $(c_f(e), f \in E_T)$ actually describe a curve $\gamma_e$ normal to $T$, with correct endpoints.

   - Choose $e \in E$, check that for $f \in E_T$ belonging to $\partial\mathcal{S}$ we have

   $$c_f(e) = \begin{cases} 1 & \text{if } f \text{ contains an endpoint of } \partial\mathcal{S} \\ 0 & \text{otherwise} \end{cases}.$$

   - For every face $t = \{x, y, z\}$ of $T$ we can see that the number of connections from $xy$ to $xz$ is equal to $\frac{c_{xy}+c_{xz}-c_{xz}}{2}$, therefore we must check that the number is in $\mathbb{N}_0$ for every face $t$ and every pair od edges $xy, xz$.

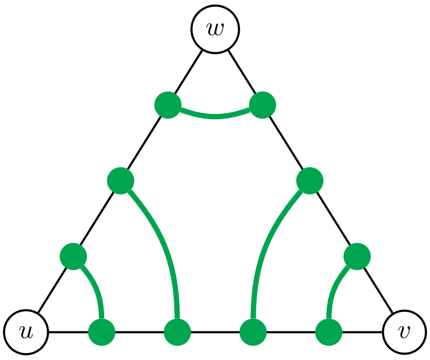


Figure 10.6: Example of connecting the triangle face edges.

   - Also we need to verify that this gives a correct curve $\gamma_e$, therefore no loops are present. For which we will use word equations $\mathcal{A} = \{\mathtt{a}, \mathtt{b}\}$. For every $f = \{u, v\} \in E_T$ we will introduce two variables $X_{uv}$ and $X_{vu}$ for which $l(X_{uv}) = l(X_{vu}) = c_{uv}(e)$. For every face $t = \{u, v, w\}$ and vertex $u \in t$ we will introduce variables $Y_{ut}$ and $Y_{ut}$, where $l(Y_{ut}) = l(Y_{tu}) = \frac{c_{xy}+c_{xz}-c_{xz}}{2}$. The equations will be as follows. For every $\{u, v\} = f \in E_T, f \in \partial\mathcal{S}$:

   $$X_{uv} = X_{vu} = \begin{cases} \mathtt{b} & \text{if } f \text{ contains an endpoint of } \partial\mathcal{S} \\ \emptyset & \text{otherwise} \end{cases}$$

   and for every face $t = \{u, v, w\}$ and edge $\{u, v\}$ we will have:

   $$X_{uv} = Y_{ut}Y_{tv}, X_{vu} = Y_{vt}Y_{tu}.$$

   **Observation.** *If $\gamma_e$ is a single connected curve, the system has a unique solution (all* b*'s). Otherwise, there is a solution containing* a*, we can find it efficiently.*

2. Verify that for each $e_1, e_2 \in E, e_1 \neq e_2, \{e_1, e_2\} \notin R$ have $i(\gamma_{e_1}, \gamma_{e_1}) = 0$.

   - Now we will denote $c_{uv} := c_{uv}(e_1) + c_{uv}(e_2)$ and then we will check that correctness as we did before.
   - Next we also introduce similiar word equations. For alphabet $\mathcal{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ and variables $X_{uv}, X_{vu}$, $Y_{ut}, Y_{tu}$ as was defined earlier. Only the length is different, for example $l(X_{uv}) = c_{uv}(e_1) + c_{vu}(e_2)$ and similarly for $l(Y_{ut}) = \dots$.

     For the first equations it will be more tricky since we can have both $e_1$ and $e_2$ intersecting $f \in E_T, f \subseteq \partial\mathcal{S}$. Otherwise it is straightforward how it is defined.

     For the next equations it will remain the same, which is $X_{uv} = Y_{ut}Y_{tv}, X_{vu} = Y_{vt}Y_{tu}$.
   - Lastly we need to verify $X_{uv}$ has $c_{uv}(e_1)$ occurrences of b and also $c_{uv}(e_2)$ occurrences of c.

□

By all these steps we have finished the proof that Weak AT-realization is in NP, because we only need to guess the numbers $c_{uv}(e)$ and add it to the NP algorithm, because with that we can in polynomial time check it is correct.

**Algorithm 8** NP algorithm for testing Weak AT-realization.

```
Require: $G = (V, E), R \subseteq \binom{E}{2}$
$\forall v \in V$ : choose $D_v$ (so that $v \neq w$ have $\overline{D_v} \cap \overline{D_w} = \emptyset$).
For any edge $e \in E$ incident to $v \in V$ : choose an endpoint of $\gamma_e$ on $\partial D_v$.
Create a triangulation $T$.
Guess all the numbers $c_f(e)$.
```

# Chapter 11

# $\chi$-boundedness

**Notation.** *$\chi(G) =$ chromatic number $G$ and $\omega(G) =$ is the maximal clique in $G$.*

**Observation.** *$\forall G : \chi(G) \geq \omega(G)$.*

**Definition 32.** *Graph class $\mathcal{C}$ is $\chi$-bounded if $\exists f : \mathbb{N} \to \mathbb{N}$ s.t. $\forall G \in \mathcal{C} : \chi(G) \leq f(\omega(G))$.*

*Example.* $\chi$-omezené třídy: úplné grafy, rovinné grafy (funkce $f$ je konstantní), perfektní grafy.

**Observation.** *Intervalové, permutační, chordální, comparability, … jsou perfektní a tudíž i $\chi$-omezené.*

## 11.1 $\chi$-boundedness of CIRC graphs

**Observation.** *Circular-Arc grafy splňují: $\chi(G) \leq 2\omega(G)$.*

*Proof.* Na kružnici máme intervaly. V jednom bodě kružnici rozstřihneme a obarvíme $\omega(G)$ barvami. Potom už máme jen intervalový graf, který je perfektní. □

**Definition 33.** *Následující definice jsou si ekvivalentní:*

- *$G \in$ CIR.*
- *$G$ je průnikový graf sečen v kružnici.*
- *$G$ je průnikový graf půlkružnic nad $x$-ovou osou.*
- *$G$ se dá reprezentovat posloupností čísel, kde se každé číslo $1, \ldots, n$ vyskytuje právě dvakrát a platí, že vrcholy $v_i, v_j$ spolu sousedí právě když ta posloupnost obsahuje podposloupnost $i, j, i, j$ nebo $j, i, j, i$.*

**Theorem 69.** *CIR je $\chi$-omezená.*

**Notation.** *Pro $k \in \mathbb{N}$ definujeme CIR$(k) := \{G \in$ CIR $: \omega(G) \leq k\}$.*

**Observation.** *CIR$(k) \subseteq$ CIR$(k+1) \subseteq \cdots \subseteq$ CIR.*

Chceme dokázat, že $(\forall k \in \mathbb{N})$ $(\exists f(k) \in \mathbb{N})$ $: \forall G \in \mathrm{CIR}(k)$ platí, že $\chi(G) \leq f(k)$.

**Definition 34.** *Mějme $G \in$ CIR reprezentovaný jako průnikový graf půlkružnic. Pro $p, q \in \mathbb{N}_0$ $(p, q)$-**konfigurace** je množina $p + q$ vrcholů $x_1, \ldots, x_p, y_1, \ldots, y_q$ takové, že půlkružnice mají tuto vzájemnou polohu zobrazenou na obrázku 11.1.*

**Notation.** *CIR$(k, p, q) = \{G \in$ CIR$, \omega(G) \leq k, G$ má reprezentaci bez $(p, q)$-konfigurace$\}$.*

**Observation.** *Pro $p > k$ a libovolné $q \in \mathbb{N}_0$: CIR$(k, p, q) =$ CIR$(k)$.*

**Claim 70.** *$\forall k \in \mathbb{N}$ $\forall p, q \in \mathbb{N}_0$ $\exists g(k, p, q) \in \mathbb{N}$ t.ž. $\forall G \in$ CIR$(k, p, q) : \chi(G) \leq g(k, p, q)$.*

**Observation.** *Důkazem tohoto tvrzení platí věta.*

*Proof.* Dané tvrzení dokážeme pomocí dvojité indukce: Indukcí podle $p \in \mathbb{N}_0$:

**Lemma 6.** *$\forall G \in$ CIR$(k, 0, q) : \chi(G) \leq k(q - 1)$ pro $q \geq 2$.*

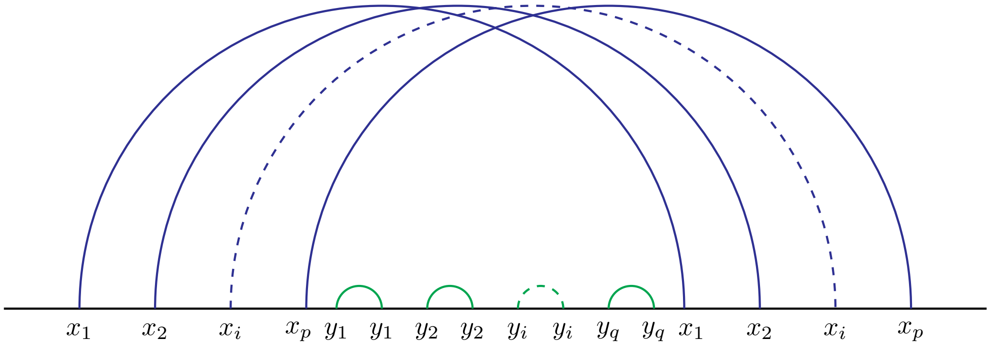


Figure 11.1: Definice $(p-q)$-konfigurace.

*Proof of lemma.* Indukcí podle $q \in \mathbb{N}_0$. Pokud $q = 2$ tak můžeme pozorovat, že všechny obloučky protínají společnou souvislou přímku. Tím pádem $\forall k : \mathrm{CIR}(k, 0, 2) \subseteq \mathrm{PER}$ které jsou perfektní.

Nyní nechť $q > 2$. Nechť $G \in \mathrm{CIR}(k, 0, q)$, vrcholy $G$ rozdělíme na 2 části $V_1, V_2$ tak, že

- $V_1$ bude indukovat graf z $\mathrm{CIR}(k, 0, 2)$ a
- $V_2$ bude indukovat graf z $\mathrm{CIR}(k, 0, q-1)$.

Označme $\pi$ jakožto nejpravější levý konec obloučku. Potom $V_1$ jsou vrcholy, které mají levý konec vlevo od $\pi$. A $V_2$ jsou vrcholy, které mají levý konec napravo od $\pi$. Pak už použijeme indukci. □

**Lemma 7.** *Nechť $p \geq 1, q \geq 1$. Potom $\forall G \in CIR(k, p, q)$ : vrcholy $G$ lze rozdělit na 2 části $V_A, V_B$ tak, že každá komponenta $G[V_A]$ i $G[V_B]$ patří do $CIR(k, p-1, 2q+1)$.*

*Corollary.* Lze vzít $g(k, p, q) = g(k, p-1, 2q+1)$ tedy tvrzení pak platí.

*Proof of lemma.* Mějme $G \in \mathrm{CIR}(k, p, q)$. BÚNO: $G$ je souvislý a máme danou reprezentaci. Nechť $x_0$ je vrchol $G$ jehož oblouček má nejlevější levý konec. Definujeme $V_i := \{x \in V(G) :$ nejkratší cesta v $G$ od $x_0$ do $x$ má délku $i\}$.

**Observation.** *Pokud vede hrana z $V_i$ do $V_j$ tak $|i-j| \leq 1$.*

**Observation.** *Z každého vrcholu $V_{i+1}$ vede aspoň 1 hrana do $V_i$.*

Nyní tvrdíme, že žádná komponenta $G[V_i]$ neobsahuje $(p-1, 2q+1)$-konfiguraci.

Nechť $C$ je komponenta $G[V_i]$ pro spor obsahující $(p-1, 2q+1)$-konfiguraci. Označme $y_{q+1}$ nějaký oblouček $w$ musí protnout $y_{q+1}$, tedy $w \in V_{i-1}$. Aspoň jeden konec je mimo $C$. $V_{i-1} = V_0$ tedy hotovo. Kdyby měl oba konce v $C$, tak nelze protnout oblouček z $V_{i-2}$.

Nyní nechť $x_1, \ldots, x_{p-1}, y_1, \ldots, y_{2q+1}$ je $(p-1, 2q+1)$-konfigurace. Nechť $I$ je interval mezi nejlevějším a nejpravějším koncem obloučku v $C$.

**Observation.** *Každý soused $w \in V_{i-1}$ vrcholu $y_{q+1}$ musí mít jeden konec mimo $I$.*

Potom $w, x_1, \ldots, x_{p-1}, y_1, \ldots, y_q$ nebo $w, x_1, \ldots, x_{p-1}, y_{q+2}, \ldots, y_{2q+1}$ je $(p, q)$-konfigurace v $G$, což je spor. Závěrem $V_A := \cup_{i \text{ sudé}} V_i$ a $V_B := \cup_{i \text{ liché}} V_i$. □

□

## 11.2 $\chi$-unboundedness of SEG graphs

**Definition 35.** *SEG is the class of graphs of intersection graphs of segments in the plane.*

Now the question might be if SEG graphs are $\chi$-bounded or not. This was firstly stated by Erdos. The answer came later by the following theorem.

**Theorem 71** (Pawlik, Kozik, Krawczyk, Lagos, Micek, Trotter, Wolezak, 2012)**.** *$\forall k \in \mathbb{N}$ there exists triangle-free graph $G_k \in SEG$ with $\chi(G_k) \geq k$.*

**Definition 36.** ***L-curve** is a union of a horizontal and vertical segment sharing a common bottom-left endpoint. **L-graph** is an intersection graph of L-curves.*

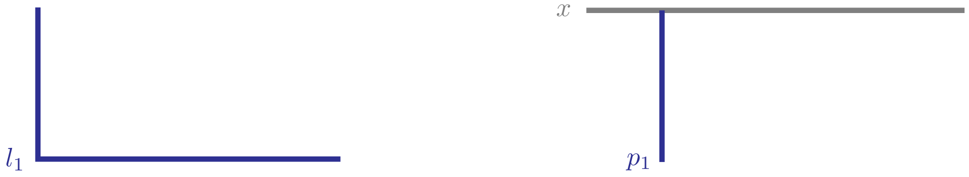
**Theorem 72.** *Any L-graph is also SEG-graph.*

*Proof.* Suppose $\mathcal{L} = \{l_1, l_2, \dots, l_n\}$ is a set of L-curves, we will show by induction on $|\mathcal{L}| = n$ that there is a set $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$ of segments s.t. $l_i \cap l_j \neq \emptyset \Leftrightarrow s_i \cap s_j \neq \emptyset$, and moreover:

1. all $s_1, \dots, s_n$ lie in the half plane below the $x$-axis;
2. $s_i$ touches the $x$-axis if and only if $l_i$ can be extended upwards to infinity without crossing any other $l_j \in \mathcal{L}$;
3. left-to-right order of $s_i$'s touching the $x$-axis is the same as for $l_i$'s.

Now for $n = 1$ it is simple, see Fig. 11.2, and for $n > 1$ WLOG $l_n$ has topmost horizontal part. By induction we represent $\mathcal{L}' = \{l_1, \dots, l_{n-1}\}$ by $\mathcal{S}' = \{s_1, \dots, s_{n-1}\}$ then

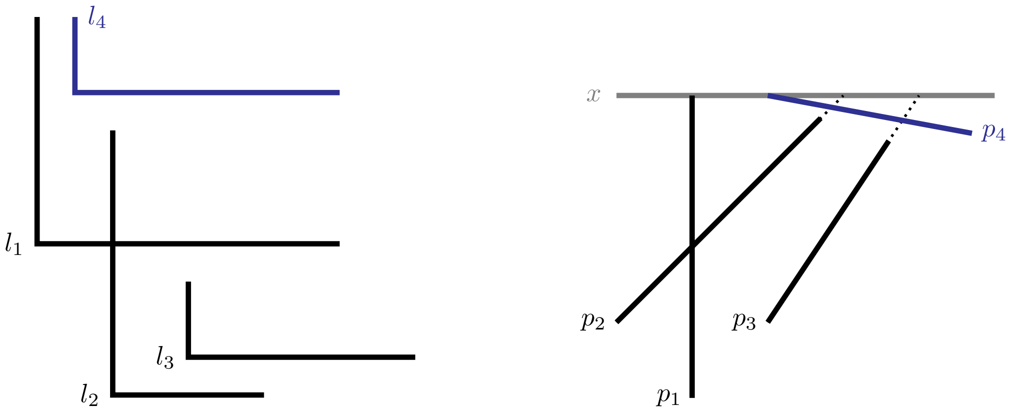
1. shorten the $s_i$ if $l_i$ no longer can extend upwards and
2. insert $s_n$ touching $x$-axis, nearly horizontal. See Fig. 11.3.



(a) L-curves.

(b) Segments.

Figure 11.2: Simple of start for an induction.



(a) L-curves.

(b) Segments.

Figure 11.3: Simple example for conversion.

□

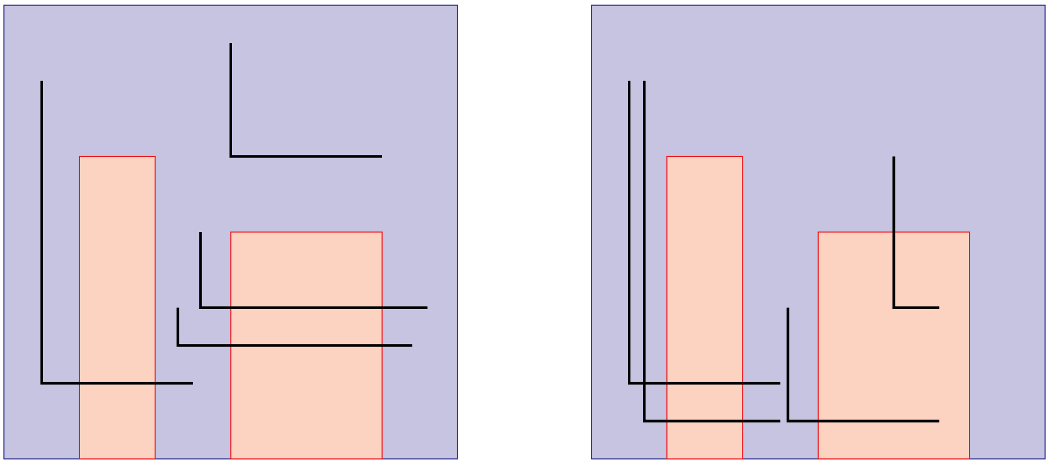
*Proof of theorem 71.* In fact we show that $\forall k \in \mathbb{N}$ there exists triangle-free L-graph $G_k$ with $\chi(G_k) \geq k$.

**Definition 37.** *A configuration is*

1. *a collection of L-curves inside the unit square* $[0,1] \times [0,1]$*, whose intersection graph os triangle-free*
2. *a set* $\{P_1, \dots, P_m\}$ *of ”probes” which are pairwise disjoint rectangles inside* $[0,1] \times [0,1]$ *touching its bottom boundary*
3. *any L-shape from the collection intersecting a probe* $P_i$ *must cross it from left to right*
4. *the L-shape crossing a given probe are pairwise disjoint.*

We will construct two sequences of $A_1, A_2, A_3, \dots$ and $B_1, B_2, B_3, \dots$ of configurations s.t.:

1. $\forall k \in \mathbb{N}$ in any proper coloring of the L-curves in $A_k$, the L-curves seen inside the probes use at least $k$-different colors.



(a) Example of configuration where we have two probes in a [0,1] box with four L-curves.

(b) Counterexample of violating the definitions 3 and 4.

Figure 11.4: Example and counterexamples of the definition.

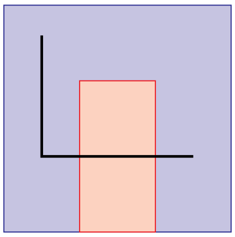


Figure 11.5: Start of the induction with one probe and one L-curve.

2. $\forall k \in \mathbb{N}$ in any proper coloring of the L-curves in $B_k$ there exist a probe of $B_k$ which is crossed by L-curves of at least $k$ different colors.

By induction for $k = 1$ it is straightforward, see Fig. 11.5. Now we show two parts.

1. From $B_k$ to $A_k + 1$. See Fig. 11.6.
   (a) Insert one new L-shape inside every probe of $B_k$ and
   (b) replace each probe by 2 new probes.
2. From $B_k$ and $A_{k+1}$ to $B_{k+1}$. See Fig. 11.7.
   (a) Insert a small copy of $A_{k+1}$ near the top of every probe of $B_k$ and
   (b) extend the probes of these small copies all the way down to obtain probes of $B_{k+1}$.

□

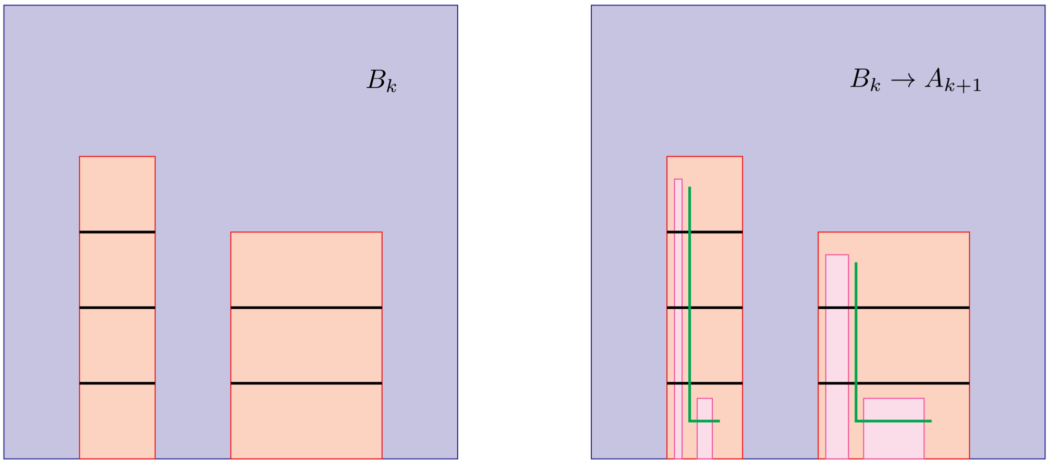
Now we may ask ourselves if this prove can be extended to some other type of intersection graphs. Well indeed it can be done. Note that *arc-connected* set means that all pairs from the set can be connected via an arc.

**Fact.** *For any compact arc-connected set $X \subseteq \mathbb{R}^2$ other then an axis parallel rectangle there is a triangle-free graph $G_k$, $\chi(G_k) \geq k$, which is the intersection graph of horizontally and vertically scaled copies of $X$.*

**Definition 38.** *A graph $G$ is $d$-degenerate if every non-empty subgraph of $G$ contains a vertex of degree at most $d$.*

**Observation.** *$G$ is $d$-degenrate then $\chi(G) \leq d + 1$.*

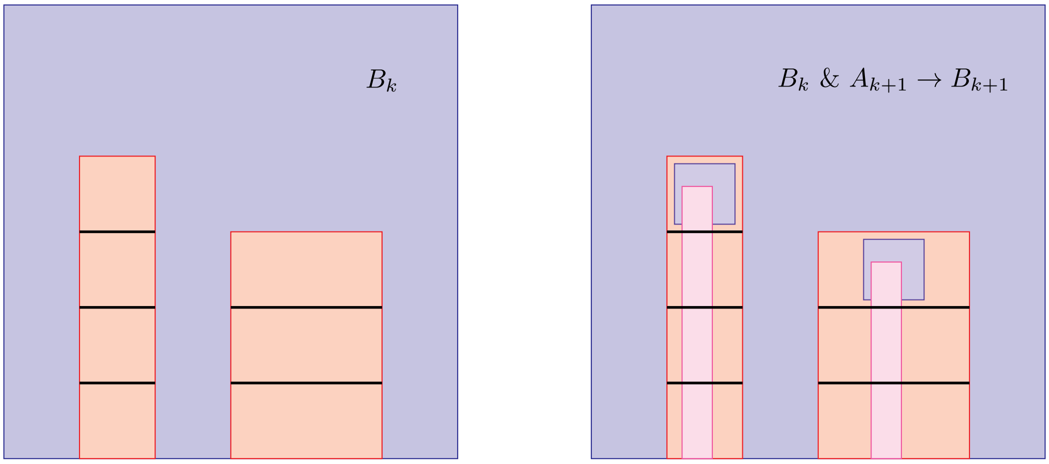
*Proof.* This observation can be seen by induction. Lets take $u$ which has $\deg(u) \leq d$ now take $G - u$ which is also $d$-degenrate and therefore by I.H. colorable by $d + 1$ colors. Lets take such coloring and now give $u$ a possible color. Because it only has $d$ neighbors we will have at least 1 possibility. □



(a) Simplification with two probes.

(b) Two new L-curves and four new probes.

Figure 11.6: Conversion from $B_k$ to $A_{k+1}$.



(a) Simplification with two probes.

(b) Copies of $A_{k+1}$ and prolonged probes.

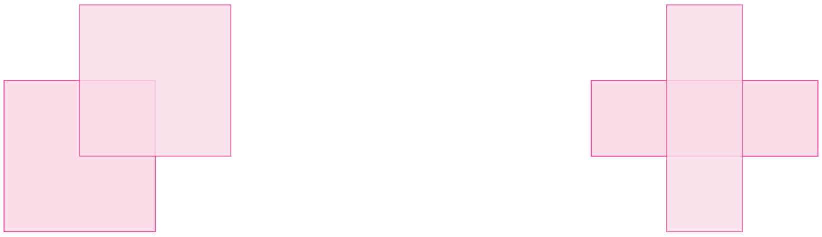
Figure 11.7: Conversion from $A_{k+1}$ and $B_k$ to $B_{k+1}$.

## 11.3 Axis-parallel rectangles intersection graphs

**Theorem 73** (Asblund, Grunbaum)**.** *If $G$ is an intersection graph of axis parallel rectangles in the plane then $\chi(G) \leq O(\omega^2(G))$.*

*Proof.* Suppose we have $G = (V, E)$ as above: each vertex $u \in V$ is represented by a rectangle $R_u$. Let $E = E_1 \dot\cup E_2$ as follows

1. $\{u, v\} \in E_1$ if a vertex of $R_u$ is inside $R_v$ or vice versa
2. $E_2 = E \setminus E_1$ if it is cross-like.



(a) One vertex is inside the other rectangle.

(b) Cross-like intersection.

Figure 11.8: Two possibilities.

Lets denote $G_1 = (V, E_1)$, $G_2 = (V, E_2)$ and $k = \omega(G)$. We will show $\chi(G_1) = O(k)$ and $\chi(G_2) = O(k)$.

- For $G_1$ we claim that $|E_1| \leq 8k \cdot |V|$ (and any subgraph of $G_1$ induced by $W \subseteq V$ has at most $8k \cdot |W|$ edges). This is because each vertex of a rectangle is inside at most $k-1$ other rectangles, due to the maximal clique. So this means one for each vertex of a rectangle is in total $4(k-1)$. Now we direct an edge $uv$ if vertex of $R_u$ is inside $R_v$. Thus the outdegree of any vertex is $\leq 4(k-1)$ so $|E_1| \leq 4(k-1) \cdot |V|$. Hence $G_1$ (and any of its induced subgraphs) has average degree $\frac{2|E_1|}{|V|} = 8(k-1)$. So the min degree is $\leq 8(k-1)$ therefore it is $8(k-1)$-degenrate and $\Rightarrow \chi(G_1) \leq 8(k-1)+1$.
- For $G_2$ we claim that $\chi(G_2) = \omega(G_2) = O(k)$ because $G_2$ is a comparability graph (for example sort the boxes from tallest to shortest).
- For $G$ we have $\chi(G) \leq \chi(G_1)\chi(G_2) = O(k^2)$. For this lets have $E_1 \dot{\cup} E_2$ and see Fig. 11.9. That is we have coloring $1, 2, \dots, p$ colors for $E_1$ and $1, 2, \dots, q$ colors for $E_2$ and we create a coloring by tuples $(x, y)$ where $x \in [p]$ and $y \in [q]$.

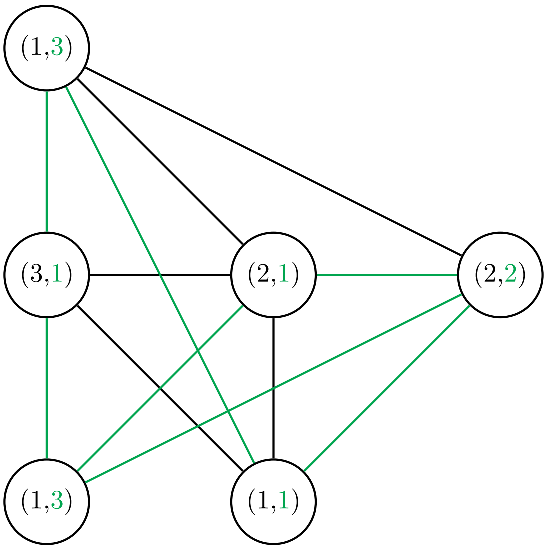


Figure 11.9: Disjoint union of edges for coloring.

□

## 11.4 1-String with big girth

Now we will continue further and see that L-graphs $\subseteq$ (proper, i.e. not overlapping segments) SEG $\subseteq$ 1-String $\subseteq$ String. For this we need to see the definitions first.

**Definition 39.** *$G \in$ 1-String if $G$ has a String representation in which every two strings intersect at most once.*

**Definition 40.** *The **girth** of a graph $G$ is the length of the shortest cycle in $G$, if $G$ has no cycle then girth is $+\infty$. We will denote girth of $G$ as $\gamma(G)$.*

With the newly established terminology we have previously shown that L-graphs of girth $\geq 4$ have unbounded $\chi$, but what about girth $\geq 5$ (or $6, \dots$). Also we will give a remark which is that for arbitrary large girth and chromatic number can be created. This can be shown by probabilistic techniques.

**Theorem 74** (Kostochka & Nešetřil)**.** *Let $G$ be a 1-String graph, then*

$$\begin{array}{lcll} \text{if } \gamma(G) \geq 5 & \Rightarrow & \chi(G) \leq 6 & (\text{degeneracy } \leq 5), \\ \text{if } \gamma(G) \geq 6 & \Rightarrow & \chi(G) \leq 4 & (\text{degeneracy } \leq 3), \\ \text{if } \gamma(G) \geq 8 & \Rightarrow & \chi(G) \leq 3 & (\text{degeneracy } \leq 2). \end{array}$$

*Proof.* Let $G = (V, E)$ be a connected 1-String graph, with a given 1-String representation. $n = |V|, m = |E|$. We will show the minimal degree of this graph where the degeneracy will follow, since its subgraphs can only have larger girth. Let $H = (V_H, E_H)$ be a plane graph whose drawing is obtained by placing a vertex on every crossing of strings of $G$, pieces of string between adjacent crossing are the edges of $H$. Recall Euler's formula $v - e + f = 2 \geq 0$.

$$\begin{aligned}
v &= |V_H| = \frac{1}{2}\sum_{v\in V}\deg(v) = m\\
e &= |E_H| = \sum_{w\in V}(\deg(w)-1) = \left(\sum_{w\in V}\deg(w)\right) - n = 2m - n\\
f &= \#\text{ of faces of } H
\end{aligned}$$

**Observation.** *If $H$ has a face bounded by a cycle of length $l$, then $G$ has a closed walk of length $l$ in which each edge appears at most once. Hence $\gamma(G) \leq l$.*

And consequently every face of $H$ is incident to at least $\gamma(G)$ edges of $H$. Hence $f \cdot \gamma(G) \leq 2e$.

$$f \leq \frac{2e}{\gamma(G)} = \frac{4m-2n}{\gamma(G)}$$

So we can use the following computation.

$$0 < v - e + f \leq m - 2m + n + \frac{4m-2n}{\gamma(G)} = n\left(1 - \frac{2}{\gamma(G)}\right) - m\left(1 - \frac{4}{\gamma(G)}\right)$$

This enforces the following.

$$m\left(1 - \frac{4}{\gamma(G)}\right) < n\left(1 - \frac{2}{\gamma(G)}\right)$$

Thus the min. degree $\leq$ average degree which is

$$= \frac{2m}{n} \leq \frac{2\left(1 - \frac{2}{\gamma(G)}\right)}{\left(1 - \frac{4}{\gamma(G)}\right)} = \frac{2(\gamma(G)-2)}{\gamma(G)-4}$$

Lastly we compute the exact values. If $\gamma(G) \geq 5$ then min degree $< 6$ so $\leq 5$ hence $\chi(G) \leq 6$. If $\gamma(G) \geq 6$ then min degree $< 4$ so $\leq 3$ hence $\chi(G) \leq 4$. If $\gamma(G) \geq 8$ then min degree $< 3$ so $\leq 2$ hence $\chi(G) \leq 3$. □

Now we will further generalize to String graphs, which will use completely new topic which is called Game of robber and cops. Note that there exists more versions of this game.

## 11.5 Game of robber and cops

We have a given graph $G$. Then 1 robber (🔴) and $c$ cops (🟩). This game is for 2 players. Both players take turns. First one plays with cops and the second with robber.

**Start of the game:** First player places cops on vertices of graph $G$. Then second player places the robber on some vertex.

**Single turn:** First player can with each cop either move to adjacent vertex or stay in the same one. Then the robber can move to adjacent or stay in the same vertex.

**Goal:** If both cop and robber end up in the same vertex cops have won. Alternatively robber wants to stay as long as possible.

*Example.* We have an easy example of the given graph and robber and cops. See Fig. 11.10

**Definition 41.** *The **cop-number** $\mathsf{cn}(G)$ := smallest number of cops for which the cops have a winning strategy on $G$. And for class $\mathcal{C}$ of graphs we define $\mathsf{cn}(\mathcal{C}) := \sup\{\mathsf{cn}(G), G \in \mathcal{C}, G$ connected$\}$.*

*Example* (Cycles). For cycles $C_n$ with $n \geq 4$ we can easily see that $\mathsf{cn}(G) = 2$ because we may put two cops in one vertex and then enclose the circle by two pats. Alternatively one is not enough since the robber may only wait and if cop is in the neighboring vertex the robber moves away.

*Example* (Int). In Int which is class of interval graphs we may find out that $\mathsf{cn}(\text{Int}) = 1$. Firstly we sort the intervals by its left endpoints, then place cop in the leftmost interval. The strategy is either catch a robber or move right to the next interval. If robber could move away the robber must stay way apart to the right or left. But note that if he would be in the left we didn't follow our strategy.

Alternatively we can say the Int has a clique-path decomposition and in each step we are either in the same clique so we can catch the robber or we may move to the next clique.

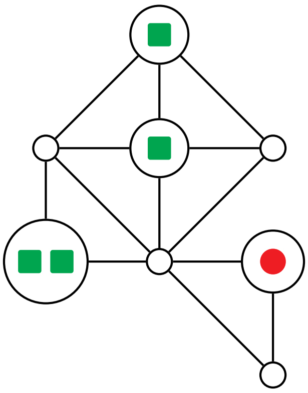


Figure 11.10: Example for cops and robber game.

**Proposition 75.** *Let $G$ be a graph with $\gamma(G) \geq 5$. If every vertex of $G$ has degree $\geq d$, then $\mathsf{cn}(G) \geq d$.*

*Proof.* If $v$ is the robbers current vertex and $v_1, v_2, \ldots, v_k$ are the neighbours of $v$, then each cop can be in the closed neighbourhood of at most one of $v_1, v_2, \ldots, v_k$. If $k >$ # of cops, robber may move to a "safe" neighbour $v_i$ if needed. □

*Consequence.* If $\mathcal{C}$ is a class of graphs closed under induced subgraphs with $\mathsf{cn}(\mathcal{C}) = c < +\infty$, then every graph $G \in \mathcal{C}$ of girth $\geq 5$ is $c$-degenrate and henc $\chi(G) \leq c + 1$.

**Theorem 76** (Das and Gahlawat, 2022)**.** $\mathsf{cn}(String) \leq 13$.

We won't show this theorem. Also the lower bound is known to be 3. Also String graphs of girth $\geq 5$ have $\chi \leq 14$ which follows from the theorem and proposition 75.

**Theorem 77.** $\mathsf{cn}(Outer\ planar) \in \{3, 4\}$.

*Proof of the lower bound.* We may construct a so called "$3 \times 5$ grid" see Fig. 11.11 for the graph itself and a representation. Then if 2 cops are adjacent to the robber then there is always one vertex for "lazy strategy" robber.

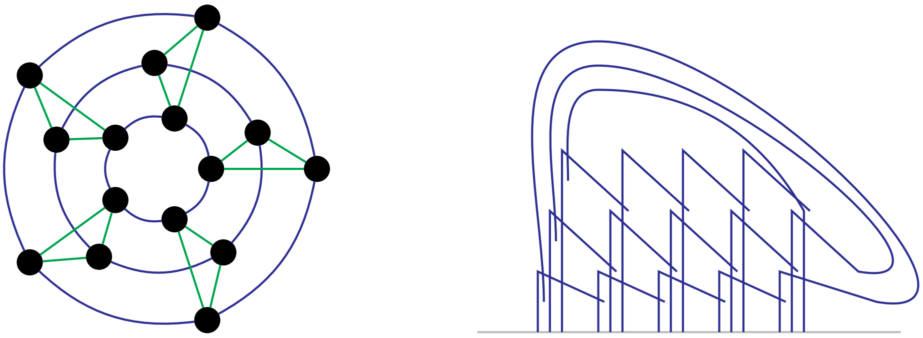


Figure 11.11: Graph on the left and on the right its representation.

□

**Theorem 78.** $\mathsf{cn}(Planar) = 3$.

*Proof.* For the lower bound we can use the proposition 75 and draw a planar graph which has girth $\gamma = 5$, one such graph is a dodecahedron, see Fig. 11.12.

Now for the upper bound we will show a strategy for 3 cops on connected planar graph $G$. Firstly one lemma.

**Lemma 8** (cop guards path $P$)**.** *Let $G$ be a graph, let $P$ be a shortest path from $x$ to $y$ in $G$, $x, y \in V(G)$. Then 1 cop after finitely many moves can take position on $P$ and play a strategy that catches the robber if the robber takes any vertexof $P$.*

*Proof.* For simplicity denote $d(u, v)$ as the distance from $u \to v$, $c$ for cop and $r$ for robber. Strategy: If there is a vertex $w \in P$ s.t. $d(w, c) > d(w, r)$ on cops turn, then $c$ moves towards $w$, otherwise $c$ stays in place.

If there is such $w$, then for $w' \in P$ on the other side of $c$ than $w$, we have $d(r, w') > d(w', c)$. Eventually, the game reaches a situation, where after every cop move $\forall w \in P : d(c, w) \leq d(r, w)$. □

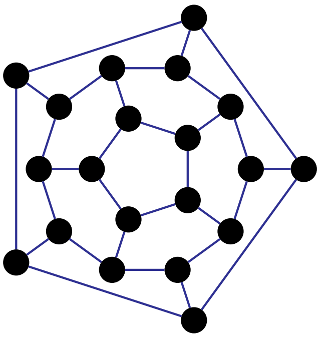


Figure 11.12: Dodecahedron as a planar graph.

The overview of the whole strategy is to constrain the robber by two paths and use the third cop to shrinken the graph even more. We will restrict the robber to smaller and smaller subgraphs $G = G_1 \supsetneq G_2 \subsetneq G_3 \subsetneq \cdots \subsetneq \emptyset$ s.t. for $G_i$:

I) There are two paths $P, Q$ each guarded by 1 cop, $G_i$ is the connected component of $G \setminus (P \cup Q)$ containing the robber. $P, Q$ share the same endpoints $x_i, y_i$ and $P$ is the shortest $x_i \to y_i$ path in $G_i \cup P \cup Q$ and $Q$ is the shortest $x_i \to y_i$ path in $G_i \cup Q$. Where $G_i \cup P \cup Q$ is the same as $G_i \cup G[P] \cup G[Q]$.

II) There is a vertex $x$ s.t. $G_i$ is a component of $G - x$ containing the robber, $x$ is occupied by a cop.

Now for the strategy itself. Cops will take any vertex $x$ and robber takes any vertex $y \neq x$, let $G_2$ be the component of $G - x$ containing $y$, therefore we get type II. Now consider subcases of the starting type.

Type II: consider few subcases.

- $x$ has only one neighbour $z$ in $G_i$, so put a cop there, and let $G_{i+1}$ be the component of $G_i - z$ containing the robber. Therefore we obtain again type II.
- $x$ has more neighbours, choose $z \neq y$ neighbour of $x_i$ in $G_i$, $P :=$ shortest $xz$-path in $G_i$. Guard $P$ with a cop and let $G_{i+1}$ be the component of $G_i \setminus P$, so we obtain type I.

Type I: again some subcases.

- If $G_i$ is adjacent to one vertex of $P \cup Q$ then we have type II. So see above case.
- Otherwise there is a path $R$ from $x$ to $y$ in $P \cup Q \cup G_i$, containing a vertex from $G_i$. Let $R$ be shortest such path, put cop to guard it. Afterwards either $P$ or $Q$ is no longer needed to constraint the robber so we will have free cop and smaller graph. This can be seen by dividing the face $G_i$ by the path $R$, after the cops guard all three paths then the orbber is left out in one face which is constrained by $R$ and one of $P$ or $Q$. Also we won't need to guard whole $R$ and the second path, since the face may be way smaller, only the common endpoints.

□

# Chapter 12

# PQ trees

**Notation.** *A set $\{1, \dots, n\}$ will be denoted as $[n]$. Then **permutation** of $[n]$ is a sequence $\pi = \pi_1, \pi_2, \dots, \pi_n$ in which each $i \in [n]$ appears exactly once. **Interval** in a permutation $\pi$ of $[n]$ is a set $S = \{\pi_i, \pi_{i+1}, \dots, \pi_j\}$ for some $1 \leq i \leq j \leq n$. **Cyclic shift** of $\pi_1, \pi_2, \dots, \pi_n$ is a permutation of the form $\pi_i, \pi_{i+1}, \dots, \pi_n, \pi_1, \pi_2, \dots, \pi_{n-1}$ for some $i \in [n]$. Lastly **cyclic interval** of $\pi$ is interval in a cyclic shift of $\pi$.*

*Example.* Lets see an example for a permutation $\pi = 31524$ of $[5]$. Then $\{1, 2, 5\}$ is an interval and $\{2, 3, 5\}$ is **not** an interval. Furthermore its cyclic shift can be 52431. Where one cyclic interval of the original permutation can be $\{2, 3, 4\}$.

Now we will introduce two problems.

*Consecutivity*

**Input:** $n \in \mathbb{N}$, sets $S_1, S_2, \dots, S_k \subseteq [n]$.

**Question:** is there a permutation $\pi$ of $[n]$ in which $S_1, S_2, \dots, S_k$ are all intervals?

*Cyclic consecutivity*

**Input:** $n \in \mathbb{N}$, sets $S_1, S_2, \dots, S_k \subseteq [n]$.

**Question:** is there a permutation $\pi$ of $[n]$ in which $S_1, S_2, \dots, S_k$ are all cyclic intervals?

**Lemma 9.** *Consecutivity can be reduced to cyclic consecutivity.*

*Proof.* We are given $n \in \mathbb{N}$, and sets $S_1, S_2, \dots, S_k \subseteq [n]$. Now see the following: $\exists \pi$ of $[n]$ in which $S_1, \dots, S_k$ are intervals $\iff$ $\exists \pi^+$ of $[n+1]$ in which $S_1, \dots, S_k$ are cyclic intervals. For one way see that if we have $\pi$ which has intervals $S_1, \dots, S_k$ and create $(\pi, n+1) = \pi^+$ permutation which has cyclic intervals. For the other way lets have $\pi^+$ of $[n+1]$ that has cyclic intervals in $S_1, \dots, S_k$, choose the cyclic shift of $\pi^+$ with $n+1$ at the end. Then $\pi^+ = (\pi, n+1)$ where $\pi$ will be the permutation of $[n]$ containing $S_1, \dots, S_k$ intervals. □

**Cyclic permutation** is determined by a permutation $\pi$ and it is the set of all cyclic shifts of $\pi$. (Unformally we may draw a circle with the elements on the boundary and all cyclic permutations are when going clockwise around the circle.) We also denote $\mathsf{Cyc}_n(S_1, \dots, S_k)$ as the set of cyclic permutations of $[n]$ in which all the sets $S_1, \dots, S_k$ are cyclic intervals.

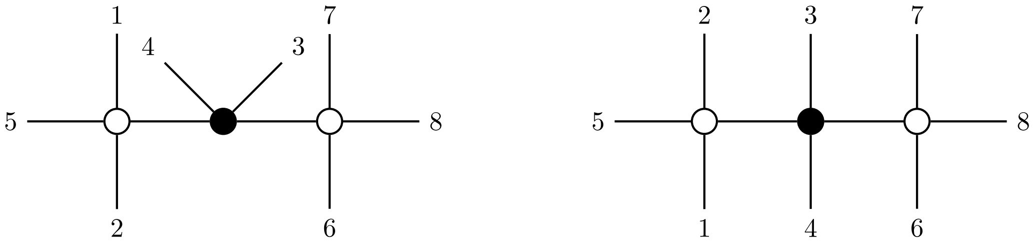
**Definition 42.** *A **PQ-tree** of order $n$ is an (unrooted, undirected0 tree with $n$ leaves labeled $1, 2, \dots, n$ and two types of interval nodes: P-nodes ●, Q-nodes ○, every internal node has a prescribed cyclic permutation of its neighbours.*

**Definition 43.** *$\pi_T$ for PQ-tree $T$ is the cyclic permutation of $[n]$ induced by the clockwise order of the leaves of $T$.*

**Definition 44.** *Two PQ-trees $T, T'$ are equivalent if $T'$ can be obtained from $T$ by a sequence of the following operations:*

1. *change the cyclic order of neighbours of P-node arbitrarily, and*
2. *reverse the order of neighbours of Q-node.*

**Definition 45.** *The set of cyclic permutations represented by $T$, denoted by $R_T$ is $\{\pi_{T'} | T' \text{ equivalent to } T\}$.*



(a) PQ-tree $T$ with permutation $\pi_T = 14378625$.

(b) Equivalent PQ-tree $T'$ with $\pi_{T'} = 15237864$.

Figure 12.1: Example of PQ-tree, its permutation and equivalent tree.

**Theorem 79.** *For any $n \in \mathbb{N}$, sets $S_1, \ldots, S_k \subseteq [n]$, we can, in time $O(n + \sum_{i=1}^{k} |S_i|)$ determine whether $\mathsf{Cyc}_n(S_1, \ldots, S_k)$ is non-empty, and if it is, construct a PQ-tree $T$ such that $R_T = \mathsf{Cyc}_n(S_1, \ldots, S_k)$.*

*Construction of the PQ-tree.* We won't show the whole proof, but only the construction of $T$ which will be shown by an induction on $k$. For $k = 0$ we create one internal P-node which has all $[n]$ leaves ordered as $1, 2, \ldots, n$ clockwise. Suppose for $k > 0$ we have constructed PQ-tree $T_{k-1}$ with $R_{T_{k-1}} = \mathsf{Cyc}_n(S_1, \ldots, S_{k-1})$. The goal is to fing $T$ with $R_T = \mathsf{Cyc}_n(S_1, \ldots, S_k)$.

Let $e$ be an edge in $T_{k-1}$, then $T_{k-1} - e$ has two components ”substrees determined by $e$”. Then subtrees is **full** if each of its leaves is in $S_k$, **empty** if none of its leaves are from $S_k$ and **mixed** otherwise. Then an edge $e$ of $T_{k-1}$ is **mixed** if both subtrees of $T_{k-1} - e$ are mixed.

**Observation.** *Mixed edges form a connected subgraph of $T_{k-1}$.*

*Proof.* If it is not true then there is a path connecting two mixed edges, but the edges on the path has to be also mixed. □

**Observation.** *If there is a certex of $T_{k-1}$ incident to three or more mixed edges, then $\mathsf{Cyc}_n(S_1, \ldots, S_k) = \emptyset$.*

Suppose mixed edges form a path $P$. Now we will show steps to create new PQ-tree.

1. Replace $T_{k-1}$ by an equivalent tree in which around every vertex of $P$ the edges towards full subtrees are above $P$, the edges towards empty subtrees are below. If this is not possible, then $\mathsf{Cyc}_n(S_1, \ldots, S_k) = \emptyset$.
2. Replace every node $v_i$ of $P$ by two nodes $v_i^+$ connected to the full subtrees only and $v_i^-$ connected to the empty subtrees only.
3. Insert a new Q-node adjacent to $v_1^+, v_2^+, \ldots, v_m^+, v_m^-, v_{m-1}^-, \ldots, v_1^-$ in this order ($m$ is for the number of nodes of $P$), call the new node $w$.
4. If $v_i^+$ or $v_i^-$ is a Q-node, then contract the edge $wv_i^+$ (or $wv_i^-$). But keep the order.
5. If there is a node of degree 2, suppress it, if $v_i^+$ or $v_i^-$ has degree 1, delete it. (Where suppressing is swapping the 2-edge path by a single edge.)

The correctness of this process involves more checking if all representations are still preserved and that all present in the new one was already there.

For time complexity one must use clever data structure and use amortization arguments to obtain such result. □

## 12.1 Applications

### 12.1.1 Recognition of INT in linear time

Recall that we have already shown that INT = Chordal ∩ co-Co nad also $G \in$ INT $\iff$ the maximal cluques of $G$ can be arranged into a sequence $Q_1, Q_2, \ldots, Q_l$ so that for every vertex $v$, the cliques containing $v$ form an interval (in the permutation of maximal cliques).

Note that from PQ-trees this can be solved in linear time $O(|V| + \sum_{v \in V} |S_v|)$. Now we will dig deeper into the details of each step.

**Algorithm 9** Idea of the algorithm

```
Require: Graph $G = (V, E)$.
Ensure: Is $G \in$ INT?
Create a set $\{Q_i; i$ is maximal clique of $G\} = \{Q_1, Q_2, \ldots, Q_l\}$.
$\forall v \in V(G)$ : create a set $S_v = \{i; v \in Q_i\}$.
Solve consecutivity for $\{S_v; v \in V\}$ in $[l]$.
```

1. We know that $G$ is chordal $\iff$ $G$ has PES 🐕. Also we have shown how to compute PES in linear time. Therefore create PES $v_1, \ldots, v_n$. If it does not exists, return that $G$ is not INT. From now on assume it exists. Create $\forall i = 1, \ldots, n$ cliques $Q_i := \{v_i\} \cup \{\text{left neighbours of } v_i\}$. Firstly observe that all maximal cliques are among these cliques (simply because all cliques have their rightmost vertex in PES, therefore its $Q_i$ since it is maximal). Next we will discard $Q_i$ if $\exists j : Q_i \subsetneq Q_j$. Note that this happens if $j > i$ and $v_i v_j \in E$ and any more to the left neighbour of $v_j$ is also a neighbour of $v_i$ so we only need to compute sizes $|Q_i| = S_i$ and denote $k$ as the number of left neighbours starting by $v_i$.

    - If $k = S_i$ then $Q_i \subseteq Q_j$.
    - If $k < S_i$ then $Q_i \nsubseteq Q_j$.
    - Also $k > S_i$ cannot happen.

2. We will also use PES.

$$S_v = (\{i\} \cup \{j | v_j \text{ is a right neighbour of } v_i\}) \cap \{k; Q_k \text{ is maximal}\}$$

    Which also implies that $|S_v| \leq$ right degree of $v$ + 1. Therefore $\sum_{v \in V} |S_v| \leq |E| + |V|$. So all together it is $O(|E| + |V|)$.

### 12.1.2 Planarity testing

**Input:** Graph $G = (V, E)$.

**Output:** Planar embedding of $G$ or a $K_5$ or $K_{3,3}$ minor of $G$.

For our usecase we will denote "planar embedding" as the rotation scheme, which is for any $v \in V$, cyclic order of edges incident to $v$ in a planar drawing.

**Definition 46.** *Fragment of a graph $G = (V, E)$ induced by a set $X \subseteq V$ contains*

1. *The subgraph of $G$ induced by $X$.*
2. *For any edge $e = \{u, v\} \in E$ s.t. $u \in X$ and $v \notin X$ create new vertex $s_e$ ("stump of $e$") and an edge $\{u, s_e\}$.*

We say that fragment induced by $X$ in $G$ is **good** if both $X$ and $V \setminus X$ induces a connected subgraph of $G$. Also a planar embedding of a fragment is **good** if all the stumps are embedded on the boundary of a single face ("outer face").

**Observation.** *If $G$ is planar, then also any good fragment has a good embedding.*

From now on assume that $G$ is 2-connected.

**Fact.** *Any good embedding of a good fragment induces a cyclic order of the stumps. Moreover for any good fragment with at least one good embedding, there is a PQ-tree whose leaves correspond to the stumps and which represents precisely the cyclic orders of stupms in the good embeddings of the fragments.*

*Sketch of proof.* Firstly generally for a graph $H = (V_H, E_H)$ equivalence on $E_H$ $e, f \in E_h : e \sim f$ if either $e = f$ or $e, f$ belongs to common cycle. Then classes of $\sim$ are 1. bridges and 2. biconnected components.

Now the constructions is roughly that we will replace all biconected components by Q nodes. Then cutvertices to P nodes, stumps to leaves, bridge to edge and incidence between cutvertex and biconnected component will also form an edge.

Every single operation with PQ-trees can be also done to change the embedding of the graph. It is also good to show the other way and that if PQ-tree wouldn't be able to keep pace with the embedding then there is a $K_5$ or $K_{3,3}$ minor. □

**Algorithm 10** Planarity testing

```
Require: Graph $G = (V, E)$.
Ensure: Planar embedding of $G$ or a $K_5$ or $K_{3,3}$ minor of $G$.
Create $T$ as a DFS tree of $G$.
Number its vertices in postorder (in order left-right-root).
$T_i :=$ subtree rooted in vertex numbered $i$ and $F_i :=$ good fragment induced by vertices of $T_i$.
Proceed bottom to top and construct PQ-trees for $F_1, F_2, \ldots, F_n$.
```

The exact procedure of building fragments is that in the leaf we just create a single P node and all its stumps. Then by induction let al childrens have their PQ-trees. Lets take one of these PQ-trees and it has stumps which has to be connected to the current vertex and other vertices. The former vertices has to be arranged consecutive, if it is not possilbe then end. That is similar to the PQ-trees where we tried to put all free subtrees to the above part and others to the below, now it is pretty much the same. Also in this step we remember the orientation of the stumps that will be conneted to the root. This will merge to one big Q node and connect to the root which will be new P node. Also we will create such Q nodes for all childrens of the current root.