

Geometric Representations of Graphs

prof. RNDr. Jan Kratochvíl, CSc.

October 15, 2021

Written by Kyrylo Karlov based on lectures in WS 2021/22

Contents

1	Introduction	2
1.1	Course overview	2
1.2	Intersection graphs	2
2	Chordal graphs	3
3	Interval graphs	6

1 Introduction

1.1 Course overview

- Finite graphs
- Existence theorems
- Algorithms, NP-completeness
- Graph drawing

How much information do we need to store the drawing?

Upward drawing: each edge directed upwards.

Most of the time we talk about **Intersection graphs**.

1.2 Intersection graphs

Definition 1.1 (Intersection graph). Let \mathcal{A} be a family of sets. Then **the** intersection graph is

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

Let \mathcal{M} be a large family of sets, then G is **an** intersection graph of \mathcal{M} if

$$\exists A \subseteq \mathcal{M} : G \simeq IG(A)$$

Note that in a family an element can be repeated several times.

Observation 1.2. IF \mathcal{M} contains nonempty set, then \forall complete graphs is in $\mathcal{IG}(\mathcal{M})$.

Proof.

$$\begin{aligned} A \in \mathcal{M}, A \neq \emptyset \\ V(K_n) = \{u_1, u_2, \dots, u_m\} \end{aligned}$$

Every vertex u is represented by A . □

Observation 1.3.

$$G \in \mathcal{IG}(\mathcal{M}) \iff \exists f : V(G) \rightarrow \mathcal{M} : \forall u \neq v \in V(G) : uv \in E(G) \iff f(u) \cap f(v) \neq \emptyset$$

Observation 1.4. $\forall \mathcal{M} : \mathcal{IG}(\mathcal{M})$ is **hereditary** if

$$\forall G \in \mathcal{IG}(\mathcal{M}) \forall H \leq G : H \in \mathcal{IG}(\mathcal{M})$$

Induced subgraph is just deleting some edges, which in sets case means forgetting edges that represent sets.

Definition 1.5 (Interval graph). Are intersection graphs of intervals on connected subsets of 1 dimensional Euclidian space.

We are interested in *arc connected* sets of the plain.

Definition 1.6 (Arc connected). Is different from topological connected.

Definition 1.7 (Circle graphs (CA)). Arcs on a circle.

Definition 1.8 (Circular arc graphs (CIR)). Chords on a circle.

Definition 1.9 (Polygon circle graphs (PC)). Polygons that can be inscribed in a circle.

Definition 1.10 (Permutation graphs (PER)). Segments connecting two parallel lines.

Formally:

$$V(G) = \{u_1, u_2, \dots, u_m\}$$

$$\exists \pi \in \text{Sym}(m) : u, v \in E \iff i < j \wedge \pi(i) > \pi(j)$$

Definition 1.11 (Function graphs (FUN)). Curves connecting two parallel lines.

Definition 1.12 (Segments graphs (SEG)). Straight-line segments in the plane.

Definition 1.13 (String graphs (STRING)). Curves in the plane.

For each 2 sets, take a point that lies in the intersection of these sets. Then connect unused dots by branching curve.

Problems we want to solve:

1. Given a graph, does it belong to the class.
2. How can we describe a representation. What size?
3. Given a graph and a representation, can we find max independent set, clique and so on. Such questions that are NP-complete in general.

Q: TODO is graph class recognition decidable?

A: Yes, but not polynomial in all cases.

INT, CA, CIR, PC, PER, FUN can be polynomially recognized. SEG, CONN are considered NP-hard. STRING is NP-complete.

$$\forall n \exists G_N \in \text{SEG}$$

but in $\forall \text{SEG}$ segment representation there is a coordinate that is at least double exponential 2^{2^n} , so even in bits is exponential.

Representation of STRING graph: make each intersection as a vertex. All curves between intersection are edges. Then graph becomes planar. Planarity can be checked in Polynomial time. But also $\forall n \exists G_n \in \text{STRING} \forall$ representation requires $\geq 2^{cn}$ crossing points.

2 Chordal graphs

Definition 2.1 (Chordal graph). G is chordal if $\forall k \geq 4 : C_k \not\subseteq G$. Sometimes called triangulated graphs.

Definition 2.2 (Simplicial). A vertex $u \in V(G)$ is simplicial if $G[N_G(u)]$ (reduction of graph to neighborhood of u) is a complete graph. Definition is independent from taking closed (includes u) or open neighborhood.

Lemma 2.3 (1). *Every inclusion-wise minimal vertex cut in a chordal graph induces a clique.*

Proof. $G \setminus A$ has components $V_1, V_2, \dots, V_k, k \geq 2$. Then

$$\forall i \forall u \in A \exists w \in V_i : uw \in E(G)$$

Pick some component V_i and some edge in A then there is an edge between them. On the contrary, if there is no edge, u can be removed from A . Which contradicts with minimality of A .

Now we take $u, v \in A$, by previous observation

$$\exists w_1, w_2 \in V_i : uw_1, vw_2 \in E(G)$$

Then take P_1 shortest path between w_1, w_2 . Similarly $w_3, w_4 \in V_j$ and the shortest path P_2 between w_3, w_4 .

$P_1 \cup P_2$ is an induced cycle unless $uv \in E(G)$.

Also, there is no edge between V_i and $V_j, i \neq j$ as otherwise A is not a cut.

As P_1 is shortest path $vw_1, uw_2 \notin E(G)$.

To sum up, $uv \in E(G)$. Since u, v were arbitrary, A is a complete subgraph. \square

Lemma 2.4 (2). *A chordal graph is complete or it contains 2 non-adjacent simplicial vertices.*

Proof. By induction on $|V(G)|$. The first step is G is a complete graph.

Inductive step: G is not complete. Take a minimal vertex cut A . Let B be a connected component of $G \setminus A$ and $C = (V(G) \setminus A) \setminus B$.

$$G_1 = G[B \cup A]$$

$$G_2 = G[C \cup A]$$

As $|V(G_1)| < |V(G)|$ we can apply induction on it. Note that induced subgraph of chordal graph is also chordal. By induction hypothesis G_1, G_2 are either complete or have 2 simplicial vertices.

One of the simplicial vertices cannot be in A because A is complete graph and simplicial vertices are not adjacent. No edges can connect B, C therefore both of the vertices are simplicial in G . \square

Corollary 2.5. *Every nonempty chordal graphs has a simplicial vertex.*

Sometimes it is easier to proof stronger statement, because we have more power in inductive step.

Definition 2.6 (PES). Perfect elimination scheme - for graph G is a *linear ordering* of its vertices.

$$V(G) = u_1, \dots, u_n$$

Such that $\forall i : u_i$ is simplicial for $G[\{u_1, \dots, u_i\}]$

Lemma 2.7. *Every chordal graph allows a PES.*

Proof. Take any simplicial vertices and move it to the right. Then delete vertex picked in previous step and repeat.

Formally: by induction on n using corollary. \square

Definition 2.8 (Perfect graph). G is perfect if for every subgraph chromatic number is equal to clique number.

$$\forall H \leq G : \chi(H) = \omega(H)$$

Theorem 2.9 (Chordal is Perfect). *A chordal graph is perfect.*

Proof. Take a PES for a graph, color from left to right by colors which are numbers

$$\{1, 2, 3, \dots\}$$

by **first fit method**.

Take smallest number that was not taken by the neighbors.

If we are forced to use color k then neighbors of the vertex used $(k-1)$ colors. Meaning that on the left there is complete graph on $(k-1)$ vertices. \square

Definition 2.10 (Clique-tree decomposition). Clique-tree decomposition of a graph G is a tree T

$$T = (\mathcal{Q}, F) : V(T) = \mathcal{Q} = \{\text{maximal cliques of } G\}$$

and

$$\forall u \in V(G) : T[\{Q : u \in Q \in \mathcal{Q}\}] \text{ is connected}$$

Theorem 2.11 (Chordal equivalent statements). *For any graph G the following are equivalent:*

1. G is chordal
2. G has a PES
3. G allows a Clique-tree decomposition
4. G is an intersection graph of subtrees of a tree

Proof. $1 \Rightarrow 2$ By induction on the number of vertices, using lemma 2.4. Pick simplicial vertex, put it at the end of PES. Remove vertex from graph and continue.

$2 \Rightarrow 3$ let we have a PES $\{u_1, \dots, u_n\}$. G has maximal cliques: $\mathcal{Q} = \{Q_1, \dots, Q_k\}$, $T = (\mathcal{Q}, E(T))$. Remove last vertex in PES from graph

$$G' = G \setminus u_n$$

It has cliques: $\mathcal{Q}' = \{Q'_1, \dots, Q'_k\} \Rightarrow$ by i.h

$$\exists T' = (\mathcal{Q}', E(T'))$$

Consider 2 cases: $N_G(u_n) \cup \{u_n\}$ is a maximal clique. Then is a maximal clique in G' .

O/w $N_G(u_n)$ is not a maximal clique of G' . $\Rightarrow \exists Q'_i \in \mathcal{Q}' : N_G(u_n) \subsetneq Q'_i$. We connect $N_G(u_n)$ to Q'_i .

$3 \Rightarrow 4$ we want to find an intersection graph of subtrees in tree.

$$G \simeq IF(\{T_u : u \in V(G)\})$$

such that

$$V(T_u) = \{Q_i : u \in Q_i\} \subseteq \mathcal{Q}$$

Proving the equivalence:

$$uv \in E(G) \Rightarrow \exists Q_i \in \mathcal{Q} : u, v \in Q_i \Rightarrow Q_i \in V(T_u) \cap V(T_v) \neq \emptyset$$

On the other hand

$$V(T_u) \cap V(T_v) \neq \emptyset \Rightarrow \exists Q_i \in V(T_u) \cap V(T_v) \Rightarrow u, v \in Q_i \Rightarrow uv \in E(G)$$

$4 \Rightarrow 1$ Let we have a tree T with a collection of subtrees.

$$V_u \subseteq V(T), u \in V(G) : T[V_u] \text{ is connected} : \forall u \neq v \in V(G) : uv \in E(G) \iff V_u \cap V_v \neq \emptyset$$

Assume by contradiction G is not chordal. By definition $\exists k \geq 4 \in \mathbb{N} : C_k \leq_{ind} G$. Let the cycle be $\{u_1, u_2, \dots, u_k\}$. $u_k u_1 \in E(G)$ by the assumption, take

$$T_1 = T[V_{u_1}], T_2 = T[V_{u_2}], T_3 = T[V_{u_3}]$$

T_1 should cross T_2 , also T_3 should cross T_2 but not T_1 .

$$V_{u_1} \cap V_{u_2} \neq \emptyset \wedge V_{u_1} \cap V_{u_3} = \emptyset$$

Therefore

$$\exists e \in E(T_2) : e \notin E(T_1), E(T_3)$$

Removing 1 edge makes tree disconnect:

$$T \setminus e = T_a \cup T_b, V_{u_1} \subseteq V(T_a), V_{u_3} \subseteq V(T_b)$$

Proceed by induction

$$\forall j, j = 3 \dots k : V(T_j) \subseteq T_b$$

inductive step

$$V_{u_j} \subseteq V(T_b), V_{u_{j+1}} \cap V(T_b) \neq \emptyset, V_{u_{j+1}} \cap V_{u_2} \neq \emptyset \Rightarrow V_{u_{j+1}} \subseteq V(T_b)$$

Therefore

$$V_{u_k} \cap V_{u_1} = \emptyset$$

which contradicts $u_k u_1 \in E(g)$.

□

3 Interval graphs

If there is an interval that is *open* on one side, we can replace it by smaller one but closed. This does not change the graph. Therefore WLOG we may assume that all intervals are closed.

Works because we have finite number of intervals.

Definition 3.1 (Interval graph(INT)).

$$INT = IG(\{\text{interval on a line}\}) = IG(\{\text{closed intervals on a line}\})$$

Theorem 3.2 (Chordal equivalent statements). *For any graph G the following are equivalent:*

1. $G \in INT$
2. G allows a Clique-path decomposition

3. $G \in IG(\{\text{subpaths of a path}\})$

Proof. $1 \iff 3$

Let Q be a maximal clique of G . All intervals that are represented by Q intersect.

$$\forall u, v \in Q : I_u \cap I_v \neq \emptyset$$

Intervals on a line have a Helly 2-property. E.g. convex sets in the plane have Helly 3-property.

Therefore

$$\bigcap_{u \in Q} I_u \neq \emptyset$$

Let x be the rightmost of the endpoints of $I_u, u \in Q$. Then $x \in \bigcap_{u \in Q} I_u$ As o/w

$$\exists v \in Q : x \notin I_v$$

Also

$$\exists w \in Q : w = [x, \dots] \Rightarrow I_v \cap I_w = \emptyset$$

and o/w x was not the rightmost point.

G has maximal cliques: $Q = \{Q_1, \dots, Q_k\}$. Let p_1, \dots, p_k be path s.t. $p_i \in \bigcap_{u \in Q_i} I_u$. It cannot happen that $p_i = p_{i+1}$ as it means $Q_i \cap Q_{i+1} = p_i$. Combining the two will make a larger clique. Therefore

$$p_1 < \dots < p_k$$

Then

$$P = (Q, \{Q_i, Q_{i+1} | i = 1, 2, \dots, k-1\})$$

is a clique-path decomposition.

Having

$$u \in Q_i, Q_h : i < h$$

We want for $i < j < h \Rightarrow p_i, p_g \in I_u \Rightarrow p_j \in I_u \Rightarrow q \in Q_j$.

$1 \Rightarrow 2$. $2 \Rightarrow 1$.

" \Rightarrow "

$$uv \in E(G) \Rightarrow \exists i : u, v \in Q_i \Rightarrow Q_i \in V_u \cap V_v \neq \emptyset$$

" \Leftarrow "

$$V_u \cap V_v \in Q_i \Rightarrow u, v \in Q_i \Rightarrow uv \in E(G)$$

□

Definition 3.3 (Comparability graphs (CO)). POSET is $\mathcal{P} = (P, \leq)$, \leq is

1. antireflexive $x \not\leq x$ - no loops
2. antisymmetric $x \leq y \Rightarrow y \not\leq x$ - no oriented 2 cycles
3. transitive $x \leq y \wedge y \leq z \Rightarrow x \leq z$ - edge between 2 connected

G is comparability graph $\iff \exists \mathcal{P} = (P, \leq)$. s.t.

$$G \simeq (P, \{xy : x \leq y \vee y \leq x\}) := C_G(\mathcal{P})$$

Comparing to Hasse diagram, comparability graph has more edges as in Hasse diagram we do not depict edges that come from transitivity.

Observation 3.4. How to check if graph is comparability? Orient every edge from the smaller to larger element. $G \in CO \iff$ edges of G can be transitively oriented. Every edge gets only 1 orientation. Note that such orientation preserves antireflexive and antisymmetric property.

Notation 3.5 (Graph Complement).

$$-G = \left(V(G), \binom{V(G)}{2} - E(G) \right)$$

If \mathcal{A} is a class of graphs, then $co - \mathcal{A} = \{-G : G \in \mathcal{A}\}$.

Theorem 3.6 (Complements). .

1. $FUN = co - CO$
2. $PER = CO \cap co - CO$
3. $INT = CHOR \cap co - CO$

Proof. $FUN \subseteq co - CO$

Let $G \in FUN$ and $\{c_u, u \in V(G)\}$ is a FUN representation. Having $u \neq v : uv \neq E(G) \Rightarrow c_u \cap c_v = \emptyset$

From the Jordan path theorem, plane is divided into 2 parts by the line. $-G$ orient s.t. $u \rightarrow v \iff c_u$ lies below c_v .

Therefore either $u \rightarrow v \vee v \rightarrow u$.

Orientation is transitive from

$co - CO \subseteq FUN^* \subseteq FUN$ Having $G \in co - CO$ we can transitively orient non edges $\binom{V(G)}{2} - E(G)$. Which corresponds to some partial order.

Definition 3.7 (PO dimension). Dimension of partial order is minimal number of linear orders k s.t

$$\exists L_1, \dots, L_k : \mathcal{P} = \bigcap L_i$$

Take k vertical lines and make n points on each of the line. Points on the line correspond to $V(G)$. Put vertices in the order of linear order L_i for line i . Connect the occurrences of vertex on each line.

Take $u, v \in V(G)$ and check their lines

$$uv \in E(G) \iff uv \notin E(-G) \Rightarrow uv \notin P \wedge vu \notin P \rightarrow \exists i \neq j Luv \in L_i \wedge vu \in L_j \Rightarrow c_u \cap c_v \neq \emptyset$$

On the other hand

$$wv \notin E(G) \Rightarrow uv \in E(-G) \Rightarrow uv \in P \vee vu \notin P \Rightarrow \forall i : uv \in L_i \Rightarrow c_u \cap c_w = \emptyset$$

Such piecewise continuous lines is the representation of the graph as a FUN graph. \square

List of Theorems

1.1	Definition (Intersection graph)	2
1.5	Definition (Interval graph)	2
1.6	Definition (Arc connected)	2
1.7	Definition (Circle graphs (CA))	3
1.8	Definition (Circular arc graphs (CIR))	3
1.9	Definition (Polygon circle graphs (PC))	3
1.10	Definition (Permutation graphs (PER))	3
1.11	Definition (Function graphs (FUN))	3
1.12	Definition (Segments graphs (SEG))	3
1.13	Definition (String graphs (STRING))	3
2.1	Definition (Chordal graph)	3
2.2	Definition (Simplicial)	3
2.6	Definition (PES)	4
2.8	Definition (Perfect graph)	4
2.10	Definition (Clique-tree decompositon)	5
3.1	Definition (Interval graph(INT))	6
3.3	Definition (Comparability graphs (CO))	7
3.5	Notation (Graph Complement)	8
3.7	Definition (PO dimension)	8

List of Theorems

2.3	Lemma (1)	3
2.4	Lemma (2)	4
2.7	Lemma	4
2.9	Theorem (Chordal is Perfect)	5
2.11	Theorem (Chordal equivalent statements)	5
3.2	Theorem (Chordal equivalent statements)	6
3.6	Theorem (Complements)	8