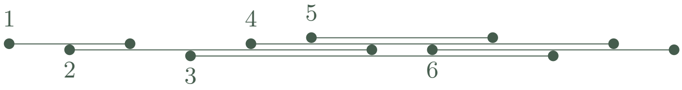
9 About Intersection Graphs

Since this lecture we focus on selected detailed topics in Graph theory that are "close to your teacher's heart"...

The first selected topic is that of *intersection graphs*, i.e. of graphs that are defined by the intersecting pairs of certain – often geometric – objects. This area of graphs is motivated both by its illustartive nature and by its practical applicability (e.g. *interval graphs*).



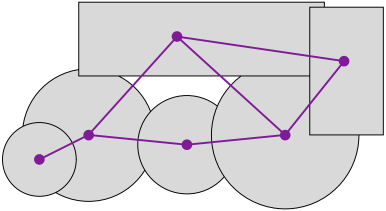
□

Brief outline of this lecture

- What are intersection graphs; the interval graphs as an example.
- Chordal graphs and their properties.
- Several more commonly studied intersection and geometric classes.
- String and segment representations of graphs as another example.

9.1 Intersection graphs; Interval graphs

Definition 9.1. **The intersection graph** of a set family $\mathcal{M}$ is the graph $I_{\mathcal{M}}$ on the vertices $V = \mathcal{M}$ and edges $E = \big\{\{A,B\} \subset \mathcal{M} : A \cap B \neq \emptyset\big\}$.



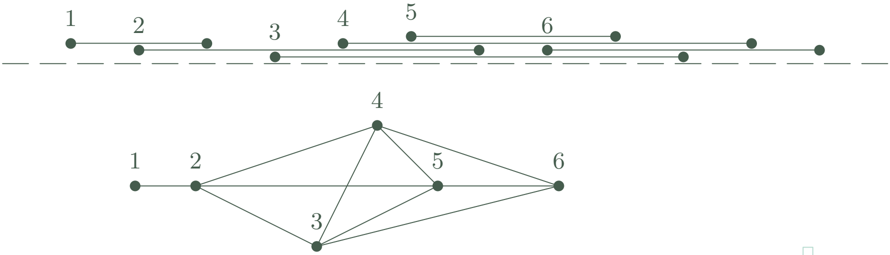
We remark that typical examples of intersection graphs are of geometric nature. □

Fact: Specific intersection graph classes are always closed on induced subgraphs. □

Proposition 9.2. *Every simple graph is isomorphic to the intersection graph of a suitable set system.*

Interval graphs

One of the oldest studied examples of intersection graphs are the *interval graphs* (shortly *INT*) – the intersection graphs of intervals on a line.



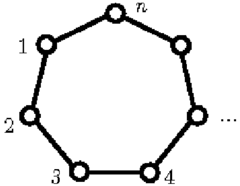
□

Recall that these graphs have already been implicitly used in connection with the single job assignment problem, which actually was the colouring problem for interval graphs.

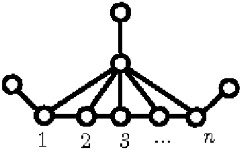
Lemma 9.3. *Every cycle of length more than three in an interval graph has a chord.*

Theorem 9.4. *The class of interval graphs has the following characterizations:*

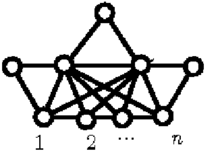
- *A graph is INT if and only if it has no induced subgraph isomorphic to one of*



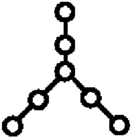
C_n; $n \geq 4$



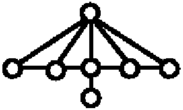
K_n; $n \geq 2$



L_n; $n \geq 1$



B_1



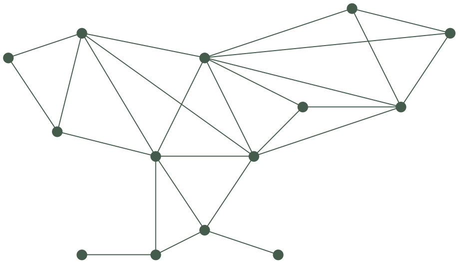
B_2

- *A simple graph G is INT if and only if G has no induced C_4, and the complement of G has a transitive orientation.*

9.2 Chordal graphs

Definition 9.5. **Chordal graph G** (also called triangulated) is such a graph G with no induced cycle (i.e. no chordless cycle) of length > 3 in G.

For example:



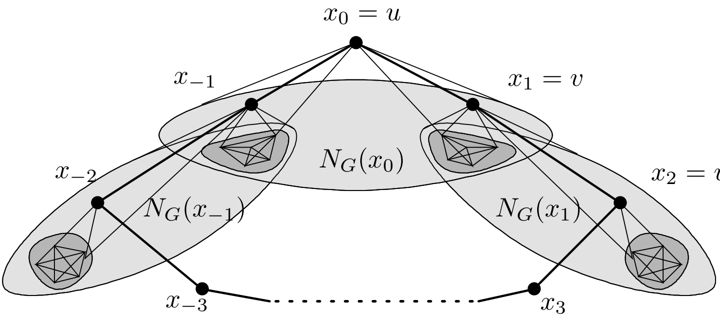
□

Theorem 9.6. *Every chordal graph G contains a simplicial vertex, which is a vertex s such that the neighbours of s in G form a clique.* □

Proof: We moreover say that a graph H is *bisimplicial* if H is complete, or H contains two nonadjacent simplicial vertices. Then...

The proof is acomplished in the following tricky sequence of three relatively straightforward claims:

1. It holds that for every cycle C in any chordal graph G, and an edge e there exists an edge f in G such that $E(C) \setminus \{e\} \cup \{f\}$ contains a triangle. □
2. Let $e = uv$ be an edge of G and let $N_G(v)$ – the neighbours of v – induce a bisimplicial subgr. of G. If v is simplicial in $N_G(u)$ but not in whole G, then there is another w adjacent to v but not to u, such that w is simplicial in $N_G(v)$.



□

3. Hence if G is not bisimplicial, but the neighbourhoods of its vertices all induce bisimplicial subgraphs, then G contains a cycle C contradicting (1).
4. Therefore, G is bisimplicial. □

Simplicial decomposition

From Theorem 9.6, one can easily conclude:

Corollary 9.7. *Every chordal graphs has a simplicial decomposition, i.e. a vertex ordering* $V(G) = (v_1, v_2, \ldots, v_n)$ *such that each* v_i, $i = 2, \ldots, n$, *is simplicial in the subgraph induced on the vertex subset* $\{v_1, \ldots, v_{i-1}\}$. □

Fact: Simplicial decompositions can be used to build efficient recognition algorithms for chordal and interval graphs.

Another intersection characterization

The following shows that chordal graphs actually present a natural generalization of interval graphs...

Theorem 9.8. *A graph G is chordal if and only if there exists a tree T such that G is the intersection graph of a collection of subtrees in T.* □

Proof (only a sketch of $\Longrightarrow$); by induction on the number of vertices of G:

- This is trivial for one vertex. □
- Let v be a simplicial vertex of G, and let $G_0 = G - v$. Then G_0 has an intersection representation by subtrees in a tree T_0. □

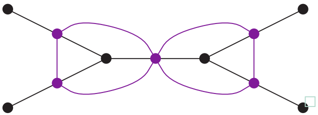
  The neighbours of v in G form a clique $K \subseteq G$, and all the trees representing vertices of K must intersect in a joint node $x \in V(T_0)$. We construct T by adding a new leaf y in T_0 adjacent to x, and represent the vertex v by a tree $\{y\}$.

□

9.3 More intersection graph examples

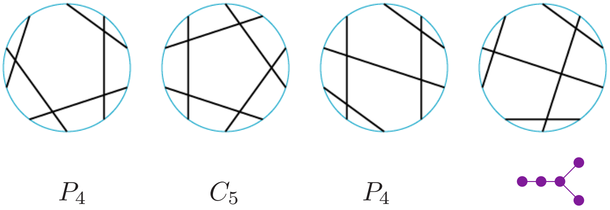
We briefly and informally introduce few more commonly studied types (classes) of intersection graphs, mostly of geometric nature.

- A *line graph* $L(G)$ of a graph G is the intersection graph of the edges $E(G)$.

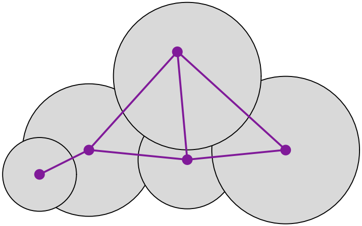


- *Circular-interval graphs* (CA) are the intersection graphs of intervals on a circle.

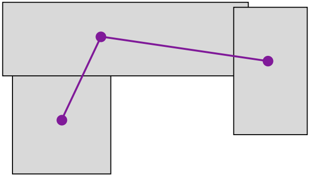
- *Circle graphs* (CIR) are the intersection graphs of straight chords of a circle.



- *Disc graphs* (DISC) are the intersection graphs of closed discs in the plane. Furthermore, unit-disc graphs are such that all the discs have unit size.



- *Box graphs* (BOX) are the intersection graphs of axis-parallel "boxes" (from rectangles to higher dimensional bodies).



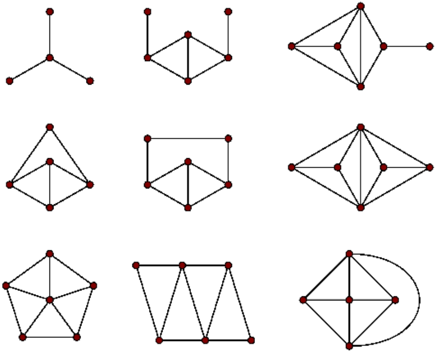
Notice that these classes can be considered as generalizations of interval graphs...

Complexity of recognition

Theorem 9.9. *The class recognition problem – to decide whether a given abstract graph belongs to the specified class, is*

- *polynomial time solvable for INT, line graphs, CA, and CIR;* □
- *and NP-hard for DISC, unit-DISC, BOX (in any dimension ≥ 2).* □

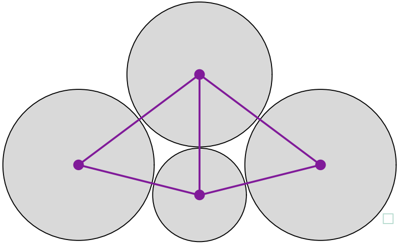
Theorem 9.10. *A graph G is a line graph of a simple graph if, and only if, G does not contain any of the following induced subgraphs:*



Contact (touching) graphs

Considering intersection graphs of geometric objects, it is sometimes natural to define the following restriction:

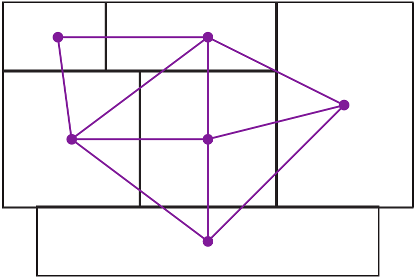
- *Contact graphs* are the graphs having an intersection representation such that the objects "do not overlap" (formally, their topol. interiors are pairwise disjoint).



A particularly beautiful result of Koebe reads:

Theorem 9.11. *A graph G is planar if, and only if, G is a contact graph of discs in the plane (a coin graph).* □

- *Rectangular duals* – another example: Those are the graphs having a contact representation by a collection of (non-overlap.) axis-parallel rectangles in the plane.



  – Note; meeting of four rectangles in one point is disallowed! □

Fact: Only planar graphs can have rectangular duals, but not all planar ones have. □

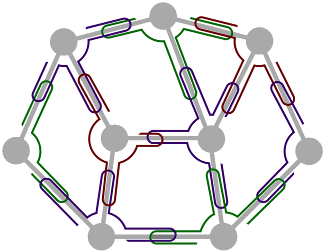
  – In a strict sense, rectangual dual repres. must "fill the plane without holes".

Fact: Strict rectangular duals always represent planar *quasi-triangulations* (all faces except the outer one are triangles).

9.4 Curve and line segment intersection graphs

- *String graphs* are the intersection graphs of simple curves in the plane.

For example, every planar graph is a string graph, see:



□

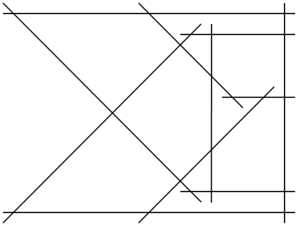
On the other hand, not all graphs are string graphs (the smallest non-string graphs have 12 vertices.) □

Moreover, the structure of string representations can be very complicated.

Proposition 9.13. *There exist string graphs such that every their representation contains a pair of curves having exponentially many intersections.*

- *Segment intersection graphs* are the intersection graphs of straight line segments in the plane.

  For example, see:



  This is a proper subclass of string graphs, and the structure is again quite complicated. For instance, there exist segment intersection graphs such that every their representation requires double-exponential precision of segment coordinates. □

Theorem 9.14. *The class recognition problems are NP-hard for both string and segment intersection graphs.*

Some more difficult findings

On the other hand, the following two statements are highly nontrivial and their proofs have been searched for many years.

Theorem 9.15. *The recognition problem of string graphs is in NP.*

[Schaeffer and Štefankovič]
Recall that a string intersection representation may be exponentially complex. . . □

Theorem 9.16. *Every planar graph is a segment intersection graph.*

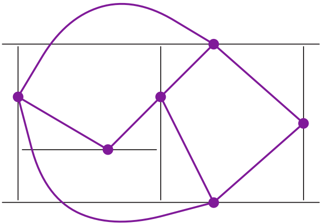
[Chalopin, Goncalves] □

Question. How many "segment slopes" one needs to represent every planar graph as a segment intersection graph?

“Match” graphs

Similarly to coin graphs, one can straightforwardly define a special subclass of segment graphs in which the line segments in an intersection representation are not allowed to cross in their interior points (i.e., having pairwise disjoint interiors).

- The aforementioned class is called the class of *segment contact graphs*. □



Theorem 9.17. *A graph G is a segment contact graph of only vertical and horizontal segments if, and only if, G is a planar bipartite graph.* □

Though...

Theorem 9.18. *The recognition problem is NP-complete for (general) segment contact graphs.*