

Discrete Mathematics in Computer Science

C4. Further Topics in Graph Theory

Malte Helmert, Gabriele Röger

University of Basel

November 17/19, 2025

Discrete Mathematics in Computer Science

November 17/19, 2025 — C4. Further Topics in Graph Theory

C4.1 Subgraphs

C4.2 Isomorphism

C4.3 Planarity and Minors

C4.1 Subgraphs

Overview

- ▶ We conclude our discussion of (di-) graphs by giving a brief tour of some further topics in graph theory that we do not have time to discuss in depth.
- ▶ In the interest of brevity (and hence wider coverage of topics), we do not give proofs for the results in this chapter.

Subgraphs

Definition (subgraph)

A **subgraph** of a graph (V, E) is a graph (V', E') with $V' \subseteq V$ and $E' \subseteq E$.

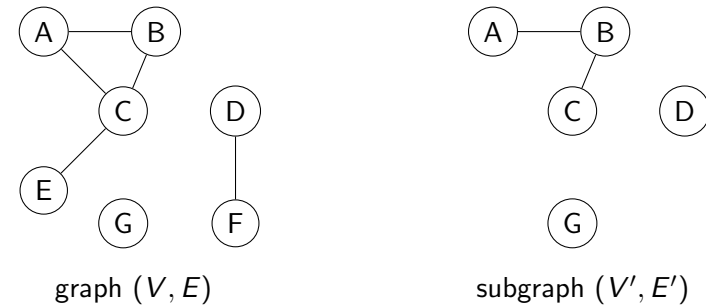
A **subgraph** of a digraph (N, A) is a digraph (N', A') with $N' \subseteq N$ and $A' \subseteq A$.

German: Teilgraph/Untergraph

Question: Can we choose V' and E' arbitrarily?

The subgraph relationship defines a **partial order** on graphs (and on digraphs).

Subgraphs – Example



Induced Subgraphs (1)

Definition (induced subgraph)

Let $G = (V, E)$ be a graph, and let $V' \subseteq V$.

The **subgraph of G induced by V'** is the graph (V', E') with $E' = \{\{u, v\} \in E \mid u, v \in V'\}$.

We say that G' is an **induced subgraph** of $G = (V, E)$ if G' is the subgraph of G induced by V' for any set of vertices $V' \subseteq V$.

German: induzierter Teilgraph (eines Graphen)

Induced Subgraphs (2)

Definition (induced subgraph)

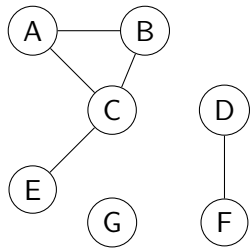
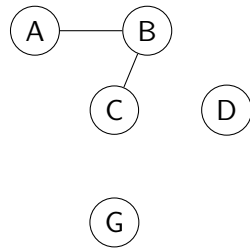
Let $G = (N, A)$ be a digraph, and let $N' \subseteq N$.

The **subgraph of G induced by N'** is the digraph (N', A') with $A' = \{\{u, v\} \in A \mid u, v \in N'\}$.

We say that G' is an **induced subgraph** of $G = (N, A)$ if G' is the subgraph of G induced by N' for any set of nodes $N' \subseteq N$.

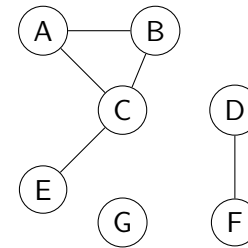
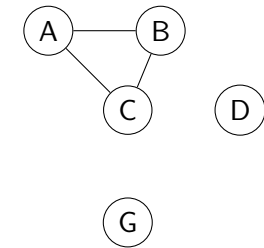
German: induzierter Teilgraph (eines gerichteten Graphen)

Induced Subgraphs – Example

graph (V, E) 

Is this an induced subgraph?

Induced Subgraphs – Example

graph (V, E) 

This is an induced subgraph.

Induced Subgraphs – Discussion

- ▶ Induced subgraphs are subgraphs.
- ▶ They are the **largest** (in terms of the set of edges) subgraphs with any given set of vertices.
- ▶ A typical example is a subgraph induced by one connected component of a graph.
- ▶ The subgraphs induced by the connected components of a forest are trees.

Counting Subgraphs

- ▶ How many subgraphs does a graph (V, E) have?
- ▶ How many induced subgraphs does a graph (V, E) have?

For the second question, the answer is $2^{|V|}$.

The first question is in general not easy to answer because vertices and edges of a subgraph cannot be chosen independently.

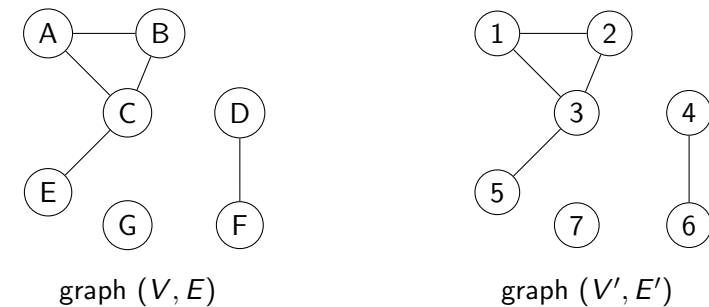
Example (subgraphs of a complete graph)

A **complete** graph with n vertices (i.e., with all possible $\binom{n}{2}$ edges) has $\sum_{k=0}^n \binom{n}{k} 2^{\binom{k}{2}}$ subgraphs. (Why?)

for $n = 10$: 1024 induced subgraphs, 35883905263781 subgraphs

C4.2 Isomorphism

Motivation



What is the difference between these graphs?

Isomorphism

- ▶ In many cases, the “names” of the vertices of a graph do not have any particular semantic meaning.
- ▶ Often, we care about the **structure** of the graph, i.e., the relationship between the vertices and edges, but not what we **call** the different vertices.
- ▶ This is captured by the concept of **isomorphism**.

Isomorphism – Definition

Definition (Isomorphism)

Let $G = (V, E)$ and $G' = (V', E')$ be graphs.

An **isomorphism** from G to G' is a **bijective** function $\sigma : V \rightarrow V'$ such that for all $u, v \in V$:

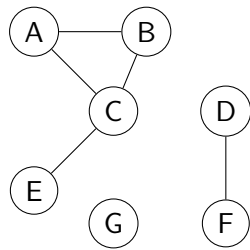
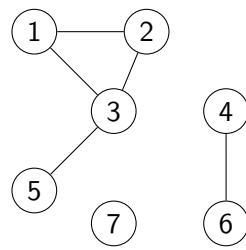
$$\{u, v\} \in E \quad \text{iff} \quad \{\sigma(u), \sigma(v)\} \in E'.$$

If there exists an isomorphism from G to G' , we say that they are **isomorphic**, in symbols $G \cong G'$.

German: Isomorphismus, isomorph

- ▶ derives from Ancient Greek for “equally shaped/formed”
- ▶ analogous definition for digraphs omitted

Isomorphism – Example

graph (V, E) graph (V', E')

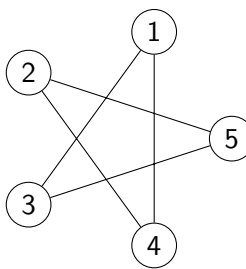
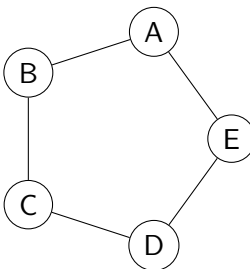
- ▶ $\sigma = \{A \mapsto 1, B \mapsto 2, C \mapsto 3, D \mapsto 4, E \mapsto 5, F \mapsto 6, G \mapsto 7\}$
- ▶ for example: $\{A, B\} \in E$ and $\{\sigma(A), \sigma(B)\} = \{1, 2\} \in E'$
- ▶ for example: $\{A, D\} \notin E$ and $\{\sigma(A), \sigma(D)\} = \{1, 4\} \notin E'$

Isomorphism – Discussion

- ▶ The **identity function** is an isomorphism.
- ▶ The **inverse** of an isomorphism is an isomorphism.
- ▶ The **composition** of two isomorphisms is an isomorphism (when defined over matching sets of vertices)

It follows that being isomorphic is an **equivalence relation**.

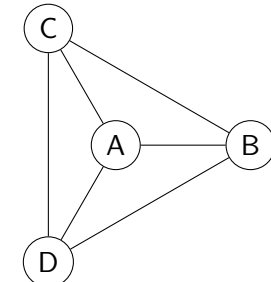
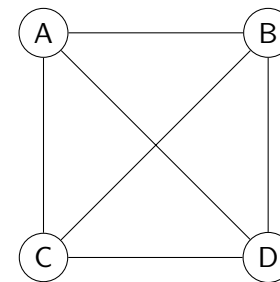
Isomorphic or Not? (1)



isomorphic

$$\sigma = \{A \mapsto 1, B \mapsto 3, C \mapsto 5, D \mapsto 2, E \mapsto 4\}$$

Isomorphic or Not? (2)

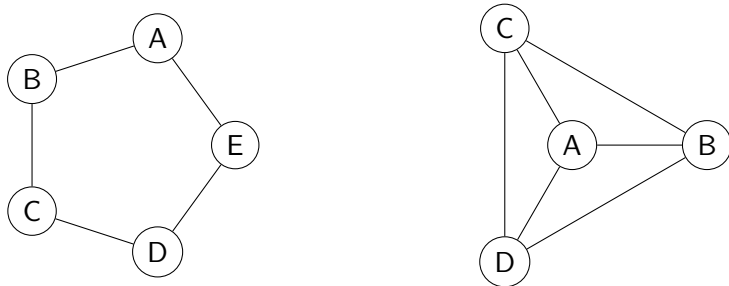


isomorphic

$\rightsquigarrow$ in fact, **red same graph!**

$$\sigma = \{A \mapsto A, B \mapsto B, C \mapsto C, D \mapsto D\}$$

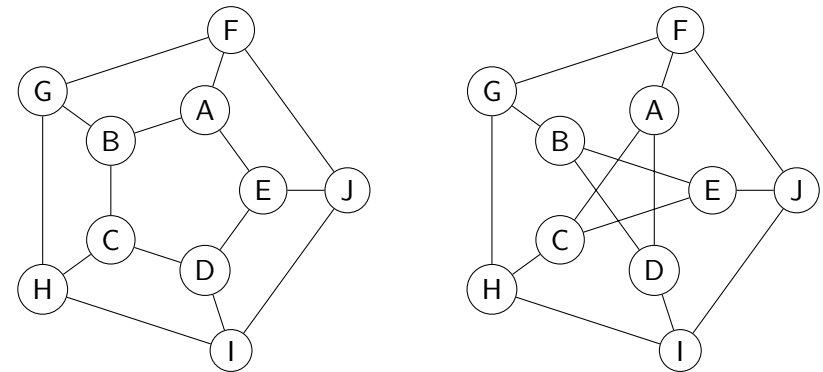
Isomorphic or Not? (3)



not isomorphic

There does not even exist a bijection between the vertices.

Isomorphic or Not? (4)

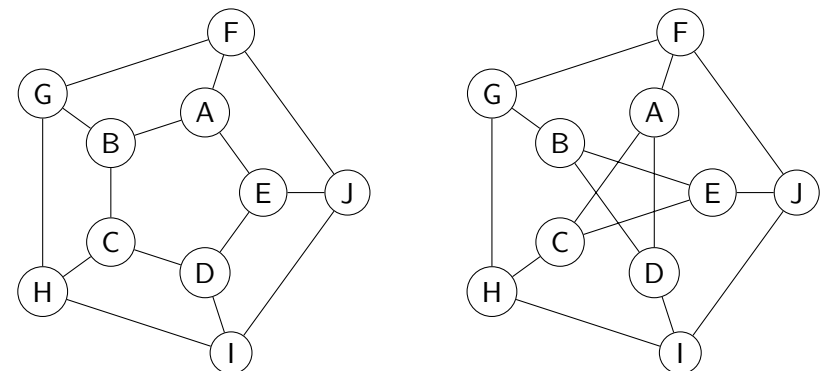


isomorphic or not?

Proving and Disproving Isomorphism

- ▶ To prove that two graphs **are** isomorphic, it suffices to state an isomorphism and verify that it has the required properties.
- ▶ To prove that two graphs are **not** isomorphic, we must rule out all possible bijections.
 - ▶ With n vertices, there are $n!$ bijections.
 - ▶ example $n = 10$: $10! = 3628800$
- ▶ A common disproof idea is to identify a **graph invariant**, i.e., a property of a graph that must be **the same** in isomorphic graphs, and show that it differs.
 - ▶ examples: number of vertices, number of edges, maximum/minimum degree, sorted sequence of all degrees, number of connected components

Isomorphic or Not? (5)



not isomorphic

- ▶ The left graph has cycles of length 4 (e.g., $\langle A, B, G, F, A \rangle$).
- ▶ The right graph does not.
- ▶ Having a cycle of a given length is an invariant.

Scientific Pop Culture

- ▶ Determining if two graphs are isomorphic is an algorithmic problem that has been famously resistant to studying its complexity.
- ▶ For more than 40 years, we have not known if polynomial algorithms exist, and we also do not know if it belongs to the famous class of **NP-complete** problems.
- ▶ In 2015, László Babai announced an algorithm with **quasi-polynomial** (worse than polynomial, better than exponential) runtime.

Further Reading

Martin Grohe, Pascal Schweitzer.

[The Graph Isomorphism Problem.](#)

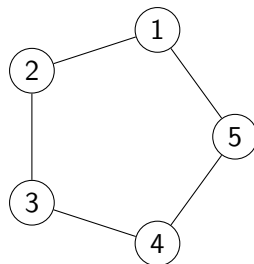
Communications of the ACM 63(11):128–134, November 2020.

<https://dl.acm.org/doi/10.1145/3372123>

Symmetries, Automorphisms and Group Theory

- ▶ An isomorphism σ between a graph G and itself is called an **automorphism** or **symmetry** of G .
- ▶ For every graph, its symmetries are permutations of its vertices that form a mathematical structure called a **group**:
 - ▶ the identity function is a symmetry
 - ▶ the composition of two symmetries is a symmetry
 - ▶ the inverse of a symmetry is a symmetry

Automorphism Group of a Graph



What are the symmetries?

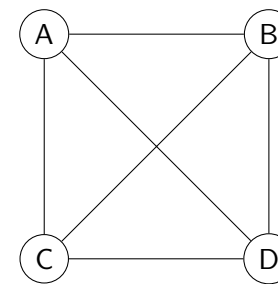
- ▶ one example is the **rotation**
 $\sigma_1 = \{1 \mapsto 2, 2 \mapsto 3, 3 \mapsto 4, 4 \mapsto 5, 5 \mapsto 1\}$
- ▶ another example is the **reflection**
 $\sigma_2 = \{1 \mapsto 5, 2 \mapsto 4, 3 \mapsto 3, 4 \mapsto 2, 5 \mapsto 1\}$
- ▶ There are 10 symmetries in total, and they are all **generated** by (= can be composed from) σ_1 and σ_2 .

C4.3 Planarity and Minors

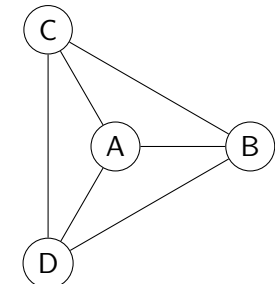
Planarity

- ▶ We often draw graphs as 2-dimensional pictures.
- ▶ When we do so, we usually try to draw them in such a way that different edges do not cross.
- ▶ This often makes the picture neater and the edges easier to visualize.
- ▶ A picture of a graph with no edge crossings is called a **planar embedding**.
- ▶ A graph for which a planar embedding exists is called **planar**.

Planar Embeddings – Example



not a planar embedding



planar embedding

The complete graph over 4 vertices is planar.

Planar Graphs

Definition (planar)

A graph $G = (V, E)$ is called **planar** if there exists a **planar embedding** of G , i.e., a picture of G in the Euclidean plane in which no two edges intersect.

German: planar

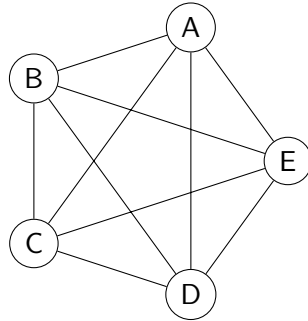
Notes:

- ▶ We do not formally define planar embeddings, as this is nontrivial and not necessary for our discussion.
- ▶ In general, we may draw edges as arbitrary curves.
- ▶ However, it is possible to show that a graph has a planar embedding iff it has a planar embedding where all edges are **straight lines**.

Planar Graphs – Discussion

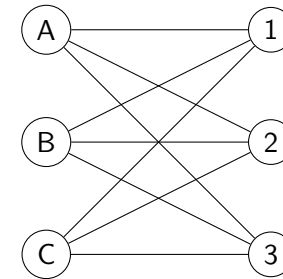
- ▶ Planar graphs arise in many practical applications.
- ▶ Many computational problems are **easier** for planar graphs.
 - ▶ For example, every planar graph can be **coloured** with at most 4 colours (i.e., we can assign one of four colours to each vertex such that two neighbours always have different colours).
- ▶ For this reason, planarity is of great practical interest.
- ▶ How can we **recognize** that a graph is planar?
- ▶ How can we prove that a graph is **not** planar?

Planar Graphs – Counterexample (1)



The complete graph K_5 over 5 vertices is not planar.
(We do not prove this result.)

Planar Graphs – Counterexample (2)



The complete bipartite graph $K_{3,3}$ over 3 + 3 vertices is not planar.
(We do not prove this result.)

Non-Planarity in General

- ▶ The two non-planar graphs K_5 and $K_{3,3}$ are special: they are the **smallest** non-planar graphs.
- ▶ In fact, something much more powerful holds: a graph is planar **iff** it does not **contain** K_5 or $K_{3,3}$.
- ▶ The notion of **containment** we need here is related to the notion of subgraphs that we introduced, but a bit more complex. We will discuss it next.

Edge Contraction

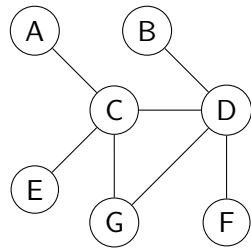
We say that $G' = (V', E')$ can be obtained from graph $G = (V, E)$ by **contracting the edge** $\{u, v\} \in E$ if

- ▶ $V' = (V \setminus \{u, v\}) \cup \{uv\}$, where $uv \notin V$ is a new vertex
- ▶ $E' = \{e \in E \mid e \cap \{u, v\} = \emptyset\} \cup \{\{uv, w\} \mid w \in V \setminus \{u, v\}, \{u, w\} \in E \text{ or } \{v, w\} \in E\}$.

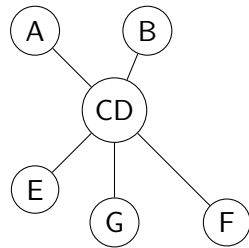
In words, we **combine** the vertices u and v (which must be connected by an edge) into a single vertex uv .

The neighbours of uv are the union of the neighbours of u and the neighbours of v (excluding u and v themselves).

Edge Contraction – Example



graph (V, E)



after contracting $\{C, D\}$

Minor

Definition (minor)

We say that a graph G' is a **minor** of a graph G if it can be obtained from G through a sequence of transformations of the following kind:

- ➊ remove a vertex (of degree 0) from the graph
- ➋ remove an edge from the graph
- ➌ contract an edge in the graph

German: Minor (plural: Minoren)

Notes:

- ▶ If we only allowed the first two transformations, we would obtain the regular subgraph relationship.
- ▶ It follows that every subgraph is a minor, but the opposite is not true in general.

Wagner's Theorem

Theorem (Wagner's Theorem)

A graph is planar iff it does not contain K_5 or $K_{3,3}$ as a minor.

German: Satz von Wagner

Note: There exist linear algorithms for testing planarity.

Minor-Hereditary Properties

- ▶ Being planar is what is called a **minor-hereditary** property: if G is planar, then all its minors are also planar.
- ▶ There exist many other important such properties.
- ▶ One example is acyclicity.

How could one prove that a property is minor-hereditary?

The Graph Minor Theorem

Theorem (Graph minor theorem)

Let Π be a minor-hereditary property of graphs.

Then there exists a finite set of **forbidden minors** $F(\Pi)$ such that the following result holds:

A graph has property Π iff it does not have any graph from $F(\Pi)$ as a minor.

German: Minorentheorem

Examples:

- ▶ the forbidden minors for **planarity** are K_5 and $K_{3,3}$
- ▶ the (only) forbidden minor for **acyclicity** is K_3 , the complete graph with 3 vertices (a.k.a. the 3-cycle graph)

Remarks on the Graph Minor Theorem (1)

- ▶ The graph minor theorem is also known as the **Robertson-Seymour theorem**.
- ▶ It was proved by Robertson and Seymour in a series of 20 papers between 1983–2004, totalling 500+ pages.
- ▶ It is one of the most important results in graph theory.

Remarks on the Graph Minor Theorem (2)

- ▶ In principle, for every **fixed** graph H , we can test if H is a minor of a graph G in polynomial time in the size of G .
- ▶ This implies that every minor-hereditary property can be tested in polynomial time.
- ▶ However, the constant factors involved in the known general algorithms for testing minors (which depend on $|H|$) are so astronomically huge as to make them infeasible in practice.