Visualization of Graphs

Lecture 1a:
The Graph Visualization Problem

Part I:
Organizational & Overview

Jonathan Klawitter
Organizational

**Lectures:**  ■ Pre-recorded videos (as you see here)
Organizational

Lectures:  ■ Pre-recorded videos (as you see here)
          ■ Release date: Weekend before
Organizational

**Lectures:**
- Pre-recorded videos (as you see here)
- Release date: Weekend before
- Thursday 10:15 – 11:15: Questions/Discussion in Zoom
Organizational

**Lectures:**
- Pre-recorded videos (as you see here)
- Release date: Weekend before
- Thursday 10:15 – 11:15: Questions/Discussion in Zoom
Organizational

Lectures:  ■ Pre-recorded videos (as you see here)
          ■ Release date: Weekend before
          ■ Thursday 10:15 – 11:15: Questions/Discussion in Zoom
          ■ Questions/Tasks in the Videos
Organizational

Lectures:
- Pre-recorded videos (as you see here)
- Release date: Weekend before
- Thursday 10:15 – 11:15: Questions/Discussion in Zoom
- Questions/Tasks in the Videos

Tutorials:
- One sheet per lecture
Organizational

Lectures: ■ Pre-recorded videos (as you see here)
■ Release date: Weekend before
■ Thursday 10:15 – 11:15: Questions/Discussion in Zoom
■ Questions/Tasks in the Videos

Tutorials: ■ One sheet per lecture
■ 20 Points per sheet
■ Scoring 50% overall ⇒ bonus
Organizational

**Lectures:**
- Pre-recorded videos (as you see here)
- Release date: Weekend before
- Thursday 10:15 – 11:15: Questions/Discussion in Zoom
- Questions/Tasks in the Videos

**Tutorials:**
- One sheet per lecture
- 20 Points per sheet
- Scoring 50% overall ⇒ bonus
- Submit solutions online
- Recommend LaTeX (template provided)
Organizational

**Lectures:**
- Pre-recorded videos (as you see here)
- Release date: Weekend before
- Thursday 10:15 – 11:15: Questions/Discussion in Zoom
- Questions/Tasks in the Videos

**Tutorials:**
- One sheet per lecture
- 20 Points per sheet
- Scoring 50% overall ⇒ bonus
- Submit solutions online
- Recommend LaTeX (template provided)
- Discussion and solutions ..
Books

G. Di Battista, P. Eades, R. Tamassia, I. Tollis:
Graph Drawing: Algorithms for the Visualization of Graphs
Prentice Hall, 1998

M. Kaufmann, D. Wagner:
Drawing Graphs: Methods and Models
Springer, 2001

T. Nishizeki, Md. S. Rahman:
Planar Graph Drawing
World Scientific, 2004

R. Tamassia:
Handbook of Graph Drawing and Visualization
CRC Press, 2013

http://cs.brown.edu/people/rtamassi/gdhandbook/
Books

G. Di Battista, P. Eades, R. Tamassia, I. Tollis:
Graph Drawing: Algorithms for the Visualization of Graphs
Prentice Hall, 1998

M. Kaufmann, D. Wagner:
Drawing Graphs: Methods and Models
Springer, 2001

T. Nishizeki, Md. S. Rahman:
Planar Graph Drawing
World Scientific, 2004

R. Tamassia:
Handbook of Graph Drawing and Visualization
CRC Press, 2013
http://cs.brown.edu/people/rtamassi/gdhandbook/
What is this course about?

Learning objectives
What is this course about?

**Learning objectives**

- Overview of graph visualization
What is this course about?

**Learning objectives**

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms
What is this course about?

**Learning objectives**
- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

**Visualization problem:**
- Given a graph $G$, visualize it with a drawing $\Gamma$
What is this course about?

**Learning objectives**
- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

**Visualization problem:**
- Given a graph $G$, visualize it with a drawing $\Gamma$

**Here:**
- Reducing the visualisation problem to its algorithmic core
What is this course about?

Learning objectives
■ Overview of graph visualization
■ Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:
■ Given a graph $G$, visualize it with a drawing $\Gamma$

Here:
■ Reducing the visualization problem to its algorithmic core

graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis
What is this course about?

**Learning objectives**

- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

**Visualization problem:**

- Given a graph $G$, visualize it with a drawing $\Gamma$

**Here:**

- Reducing the visualisation problem to its algorithmic core

  graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

- modeling
- data structures
What is this course about?

Learning objectives
- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:
- Given a graph $G$, visualize it with a drawing $\Gamma$

Here:
- Reducing the visualisation problem to its algorithmic core
  
  graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

- modeling
- data structures

- divide & conquer, incremental
- combinatorial optimization (flows, ILPs)
- force-based algorithm
What is this course about?

Learning objectives
- Overview of graph visualization
- Improved knowledge of modeling and solving problems via graph algorithms

Visualization problem:
- Given a graph $G$, visualize it with a drawing $\Gamma$

Here:
- Reducing the visualisation problem to its algorithmic core

graph class $\Rightarrow$ layout style $\Rightarrow$ algorithm $\Rightarrow$ analysis

- modeling
- data structures
- divide & conquer, incremental
- combinatorial optimization (flows, ILPs)
- force-based algorithm
- proofs
What is this course about?

**Topics**

- Drawing Trees and Series-Parallel Graphs
- Tutte Embedding and Force-Based Drawing Algorithms
- Straight-Line Drawings of Planar Graphs
- Orthogonal Grid Drawings
- Octilinear Drawings for Metro Maps
- Upwards Planar Drawings
- Hierarchical Layouts of Directed Graphs
- Contact Representations
- Visibility Representations
- The Crossing Lemma
- Beyond Planarity
Visualization of Graphs

Lecture 1a:
The Graph Visualization Problem

Part II:
The Layout Problem

Jonathan Klawitter
Graphs and their representations

What is a graph?

- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$
Graphs and their representations

What is a graph?

- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$

Representation?
Graphs and their representations

What is a graph?
- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$

Representation?
- Set notation

\[
V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}
\]
\[
E = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\},
\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\},
\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\},
\{v_9, v_{10}\}\} 
\]
Graphs and their representations

What is a graph?
- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$

Representation?
- Set notation
  
  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$
  
  $E = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \{v_5, v_{10}\}, \{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}, \{v_9, v_{10}\}\}$

- Adjacency list
  
  $v_1 : v_2, v_8$
  $v_2 : v_1, v_3$
  $v_3 : v_2, v_5, v_9, v_{10}$
  $v_4 : v_5, v_6, v_9$
  $v_5 : v_3, v_4, v_8$
  $v_6 : v_4, v_8, v_9$
  $v_7 : v_8, v_9$
  $v_8 : v_1, v_5, v_6, v_7, v_9, v_{10}$
  $v_9 : v_3, v_4, v_6, v_7, v_8, v_{10}$
  $v_{10} : v_3, v_8, v_9$
Graphs and their representations

What is a graph?
- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$

Representation?
- Set notation
  
  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$
  
  $E = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\},$
  
  $\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_9\}, \{v_5, v_8\},$
  
  $\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\},$
  
  $\{v_9, v_{10}\}\}$

- Adjacency list
  
  $v_1 : v_2, v_8$
  $v_2 : v_1, v_3$
  $v_3 : v_2, v_5, v_9, v_{10}$
  $v_4 : v_5, v_6, v_9$
  $v_5 : v_3, v_4, v_8$
  $v_6 : v_4, v_8, v_9$
  $v_7 : v_8, v_9$
  $v_8 : v_1, v_5, v_6, v_7, v_9, v_{10}$
  $v_9 : v_3, v_4, v_6, v_7, v_8, v_{10}$
  $v_{10} : v_3, v_8, v_9$

- Adjacency matrix
  
  $\begin{pmatrix}
  0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
  1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
  0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
  0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
  0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
  1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
  0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\
  0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0
  \end{pmatrix}$
Graphs and their representations

What is a graph?

- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$

Representation?

- **Set notation**
  
  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$
  
  $E = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\}, \{v_3, v_{10}\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\}, \{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\}\}$

- **Adjacency list**
  
  $v_1: v_2, v_8$
  
  $v_2: v_1, v_3$
  
  $v_3: v_2, v_5, v_9, v_{10}$
  
  $v_4: v_5, v_6, v_9$
  
  $v_5: v_3, v_4, v_8$
  
  $v_6: v_4, v_8, v_9$
  
  $v_7: v_8, v_9$
  
  $v_8: v_1, v_5, v_6, v_7, v_9, v_{10}$
  
  $v_9: v_3, v_4, v_6, v_7, v_8, v_{10}$
  
  $v_{10}: v_3, v_8, v_9$

- **Adjacency matrix**

\[
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0
\end{bmatrix}
\]

- **Drawing**
Graphs and their representations

What is a graph?
- graph $G = (V, E)$
- vertices $V = \{v_1, v_2, \ldots, v_n\}$
- edge $E = \{e_1, e_2, \ldots, e_m\}$

Representation?
- Set notation
  
  $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}\}$
  $E = \{\{v_1, v_2\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_3, v_5\}, \{v_3, v_9\},$
  $\{v_3, v_{10}\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_4, v_9\}, \{v_5, v_8\},$
  $\{v_6, v_8\}, \{v_6, v_9\}, \{v_7, v_8\}, \{v_7, v_9\}, \{v_8, v_{10}\},$
  $\{v_9, v_{10}\}\}$

- Adjacency list
  
  $v_1: v_2, v_8$
  $v_2: v_1, v_3$
  $v_3: v_2, v_5, v_9, v_{10}$
  $v_4: v_5, v_6, v_9$
  $v_5: v_3, v_4, v_8$
  $v_6: v_4, v_8, v_9$
  $v_7: v_8, v_9$
  $v_8: v_1, v_5, v_6, v_7, v_9, v_{10}$
  $v_9: v_3, v_4, v_6, v_7, v_8, v_{10}$
  $v_{10}: v_3, v_8, v_9$

- Adjacency matrix

  $\begin{pmatrix}
  0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
  1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
  0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\
  0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
  0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
  0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
  1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
  0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\
  0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
  \end{pmatrix}$

- Drawing
Why draw graphs?
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.

Abstract networks
- Social networks
- Communication networks
- Phylogenetic networks
- Metabolic networks
- Class/Object Relation Digraphs (UML)
- …
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.

**Abstract networks**
- Social networks
- Communication networks
- Phylogenetic networks
- Metabolic networks
- Class/Object Relation Digraphs (UML)
- ...

**Physical networks**
- Metro systems
- Road networks
- Power grids
- Telecommunication networks
- Integrated circuits
- ...
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualisations!
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualisations!

- Visualisations help with the **communication** and **exploration** of networks.
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.

- **People think visually** — complex graphs are hard to grasp without good visualisations!

- Visualisations help with the communication and exploration of networks.

- Some graphs are too big to draw them by hand.
Why draw graphs?

Graphs are a mathematical representation of real physical and abstract networks.

- **People think visually** – complex graphs are hard to grasp without good visualisations!
- Visualisations help with the *communication* and *exploration* of networks.
- Some graphs are too big to draw them by hand.

We need algorithms that draw graphs automatically to make networks more accessible to humans.
What are we interested in?
What are we interested in?

- Jacques Bertin defined visualising variables (1967)
What are we interested in?

- Jacques Bertin defined visualising variables (1967)
What are we interested in?

- Jacques Bertin defined visualising variables (1967)

Diagram shows variables:
- Position
- Orientation
- Shape
- Size
- Colour
- Texture
- Shading

→ Layout problem
What are we interested in?

- Jacques Bertin defined visualising variables (1967)
The layout problem

- Here restricted to the standard representation, so-called node-link diagrams.
The layout problem

- *Here* restricted to the **standard representation**, so-called node-link diagrams.

**Graph Visualization Problem**

**in:** Graph $G = (V, E)$

**out:**
The layout problem

- Here restricted to the **standard representation**, so-called node-link diagrams.

**Graph Visualization Problem**

**in:** Graph $G = (V, E)$

**out:** nice drawing $\Gamma$ of $G$

- $\Gamma: V \rightarrow \mathbb{R}^2$, vertex $v \mapsto$ point $\Gamma(v)$
- $\Gamma: E \rightarrow$ curves in $\mathbb{R}^2$, edge $\{u, v\} \mapsto$ simple, open curve $\Gamma(\{u, v\})$ with endpoints $\Gamma(u)$ und $\Gamma(v)$
The layout problem?

Here restricted to the standard representation, so-called node-link diagrams.

**Graph Visualization Problem**

<table>
<thead>
<tr>
<th>in:</th>
<th>Graph $G = (V, E)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>out:</td>
<td>nice drawing $\Gamma$ of $G$</td>
</tr>
</tbody>
</table>

- $\Gamma : V \to \mathbb{R}^2$, vertex $v \mapsto$ point $\Gamma(v)$
- $\Gamma : E \to$ curves in $\mathbb{R}^2$, edge $\{u, v\} \mapsto$ simple, open curve $\Gamma(\{u, v\})$ with endpoints $\Gamma(u)$ und $\Gamma(v)$

But what is a nice drawing?
Tree of virtues and tree of vices
ca. 1200
Social networks - family trees

Ahnentafel Herzog Ludwig von Württemberg, 1585

J. Klawitter, T. Mchedlidze, Link: go.uniwue.de/myth-poster
Social network – citation graph

Da Ye, Link: https://go.uniwue.de/citation-graph
Social network - organisational chart
Social network - world finance corporation
Transportation network – European high speed railroads
Transportation network – London Underground

Source: Wiki Commons: London Underground full map - CC BY-SA 3.0
Transportation network – London Underground

Source: Wiki Commons: London Underground Overground DLR Crossrail map - CC BY-SA 4.0
Transportation network – London Underground
Bioinformatics – disease interaction

Source: Wiki Commons: Human disease network - CC BY-SA 4.0
Bioinformatics – molecular metabolic network

Source: Wiki Commons: Citric acid cycle withaconitate 2 - CC BY-SA 3.0

Bioinformatics – phylogenetic trees & networks

Source: Wiki Commons: Phylogenetic network of HVS-I variation - CC BY 4.0
Technical network – very large-scale integration (VLSI)

Source: Wiki Commons: Diopsis - CC BY-SA 3.0

Source: Pixabay
Technical network – transistor diagram, wiring
Technical networks – offshore wind farms

Source: Wiki Commons: Alpha Ventus Windmills - CC BY-SA 3.0
Technical network – UML diagram
Temporal graph layout – storylines

These charts show movie character interactions. The horizontal axis is time. The vertical grouping of the lines indicates which characters are together at a given time.

Source and more: xkcd Comic 657 – xkcd.com/657/
Large graphs – object mesh
General graphs – micro-macro layout

Source: Angori et al., ChordLink: A New Hybrid Visualization Model, GD’19 (2019)
Alternative representations – treemap
Alternative representations – contact graphs
Alternative representations – contact graphs

For more examples see visualcomplexity.com
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma( uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e., with bends)
   - grid drawings
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(\{u,v\}) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   ■ straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   ■ orthogonal edges (i.e. with bends)
   ■ grid drawings
   ■ without crossing
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(\{uv\}) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(\mathit{uv}) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(\bar{uv}) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure

→ such criteria are often inversely related
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(\{uv\}) = \Gamma(u) \Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure

→ such criteria are often inversely related
→ lead to NP-hard optimization problems
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure

3. Local Constraints, e.g.

→ such criteria are often inversely related
→ lead to NP-hard optimization problems
Requirements of a graph layout

1. Drawing conventions and requirements, e.g.,
   - straight edges with $\Gamma(uv) = \Gamma(u)\Gamma(v)$
   - orthogonal edges (i.e. with bends)
   - grid drawings
   - without crossing

2. Aesthetics to be optimized, e.g.
   - crossing/bend minimization
   - edge length uniformity
   - minimizing total edge length/drawing area
   - angular resolution
   - symmetry/structure

3. Local Constraints, e.g.
   - restrictions on neighboring vertices (e.g., “upward”).
   - restrictions on groups of vertices/edges (e.g., “clustered”).

→ such criteria are often inversely related
→ lead to NP-hard optimization problems
The layout problem

Graph visualisation problem

\textbf{in:} Graph $G = (V, E)$

\textbf{out:} Drawing $\Gamma$ of $G$ such that...
The layout problem

Graph visualisation problem

in: Graph $G = (V, E)$

out: Drawing $\Gamma$ of $G$ such that
    ■ drawing conventions are met,
The layout problem

**Graph visualisation problem**

**in:** Graph $G = (V, E)$

**out:** Drawing $\Gamma$ of $G$ such that

- **drawing conventions** are met,
- **aesthetic criteria** are optimised, and
The layout problem

Graph visualisation problem

in: Graph $G = (V, E)$

out: Drawing $\Gamma$ of $G$ such that

- drawing conventions are met,
- aesthetic criteria are optimised, and
- some additional constraints are satisfied.