

# Algorithmen und Datenstrukturen

## C7. Graphen: Ausblick

Gabriele Röger

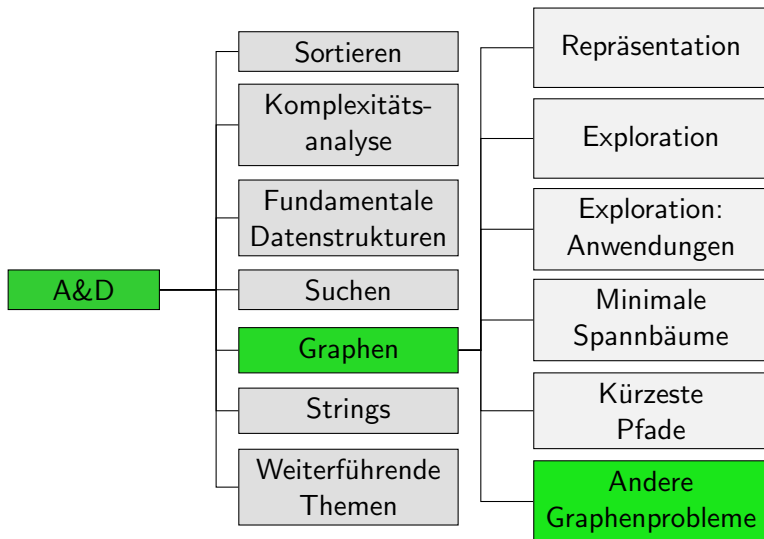
Universität Basel

20. Mai 2020

## C7.1 Andere Graphenprobleme

## C7.1 Andere Graphenprobleme

# Inhalt dieser Veranstaltung



# Crashkurs Komplexitätstheorie

- ▶ **Entscheidungsprobleme:** Ja/Nein-Antwort gesucht  
Gegeben gewichteter Graph, Knoten  $s$ ,  $t$  und Zahl  $K$ .  
Gibt es einen Pfad von  $s$  nach  $t$  mit Kosten höchstens  $K$ ?
- ▶ **Suchprobleme:** tatsächliche Lösung gesucht  
Gegeben gewichteter Graph und Knoten  $s$ ,  $t$ .  
Finde einen kürzesten Pfad von  $s$  nach  $t$ .

# Crashkurs Komplexitätstheorie

Wir unterscheiden verschiedene Klassen von Problemen:

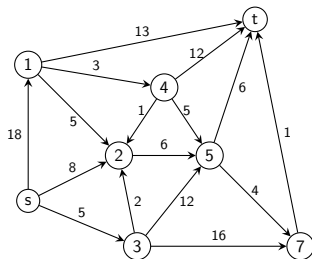
- ▶ **P**: alle Probleme, die man mit einem **polynomiellen Algorithmus** (in  $O(p)$  für irgendein Polynom  $p$ ) **lösen** kann.
- ▶ **NP**: alle Probleme, bei denen man einen Beweis für eine Ja-Antwort des Entscheidungsproblems **in polynomieller Zeit verifizieren kann**.  
Beweis: z.B. konkreter Pfad mit Kosten  $\leq K$
- ▶  **$P \neq NP$ ?** Wir wissen es nicht.
- ▶ **NP-schwere Probleme**: Probleme, die mindestens so schwierig sind, wie die schwierigsten Probleme in NP.  
→ keine **polynomiellen Verfahren** bekannt.
- ▶ **NP-vollständige** Entscheidungsprobleme: NP-schwer & in NP
- ▶ **NP-äquivalente** Suchprobleme: zugehöriges Entscheidungsproblem NP-vollständig

# Flüsse in Graphen I

## Definition (Flussnetzwerk)

Ein **Flussnetzwerk**  $N = (G, s, t, k)$  ist gegeben durch

- ▶ einen **gerichteten Graphen**  $G = (V, E)$ ,
- ▶ einer **Quelle** (source)  $s \in V$ ,
- ▶ einer **Senke** (target)  $t \in V$ , und
- ▶ einer **Kapazitätsfunktion**  $k : E \rightarrow \mathbb{R}_+^\infty$ .



# Flüsse in Graphen II

## Definition (Fluss)

Ein **s-t-Fluss**  $f$  weist jeder Kante einen Wert aus  $\mathbb{R}_{\geq 0}$  zu, wobei

- ▶ der **Flusswert** die **Kapazität** der Kante **nicht übersteigt**:

$$f(e) \leq k(e) \text{ für alle } e \in E$$

- ▶ bei allen Knoten ausser der Quelle und der Senke **genauso viel hinein wie hinaus** fliesst:

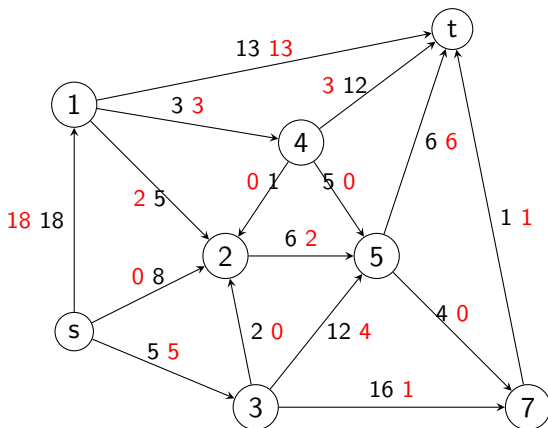
$$\sum_{\substack{(u,w) \in E \\ w=v}} f((u,w)) = \sum_{\substack{(u,w) \in E \\ u=v}} f((u,w)) \text{ für alle } v \in V \setminus \{s, t\}$$

Der **Wert** des Flusses ist der Überschuss in der Senke:

$$|f| = \sum_{\substack{(u,w) \in E \\ w=t}} f((u,w)) - \sum_{\substack{(u,w) \in E \\ u=t}} f((u,w))$$



# Beispiel

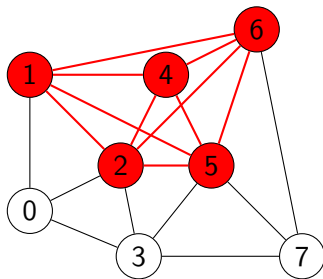


Wie schwer ist es, einen maximalen Fluss zu finden?  
 z.B. mit Edmonds-Karp-Algorithmus in  $O(|E|^2|V|)$

# Cliquen

## Definition (Clique)

Eine **Clique** in einem ungerichteten Graphen  $(V, E)$  ist eine Teilmenge  $C \subseteq V$  der Knoten, bei der jedes Knotenpaar durch eine Kante verbunden ist: für  $u, v \in C$  mit  $u \neq v$  gilt  $\{u, v\} \in E$ .



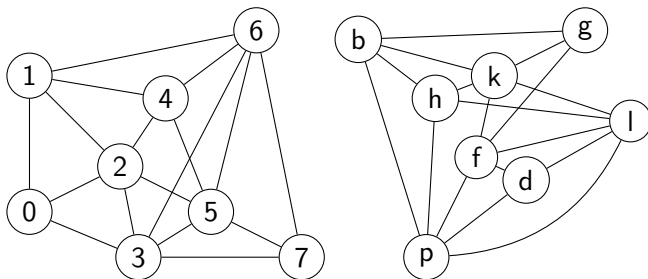
Wie schwer ist es, eine grösste Clique in einem Graphen zu finden?

**NP-äquivalent**

# Graphenisomorphie

## Definition (Graphenisomorphie)

Zwei Graphen sind **isomorph**, wenn sie bis auf die Namen der Knoten gleich sind.



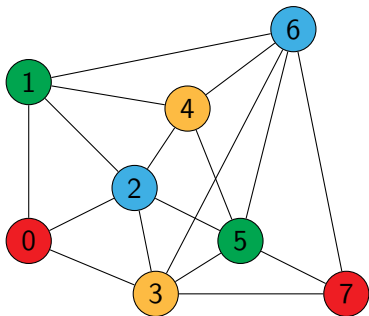
Wie schwer ist es zu entscheiden, ob zwei Graphen isomorph sind?

In NP, aber unbekannt ob in P und/oder NP-vollständig

# Färbbarkeit

## Definition ( $k$ -Färbbarkeit)

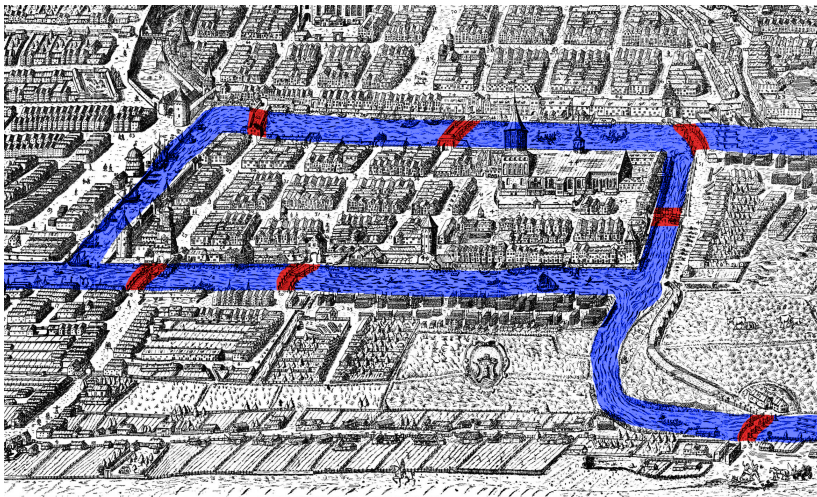
Ein ungerichteter Graph  $G = (V, E)$  ist  **$k$ -färbbar** ( $k \in \mathbb{N}$ ), falls es eine Färbung  $f : V \rightarrow \{1, \dots, k\}$  gibt, so dass für alle  $\{v, w\} \in E$  gilt:  $f(v) \neq f(w)$ .



Wie schwer ist es zu entscheiden,  
ob ein gegebener Graph  $k$ -färbbar ist?

**NP-vollständig**

# Königsberger Brückenproblem

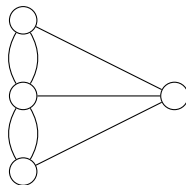
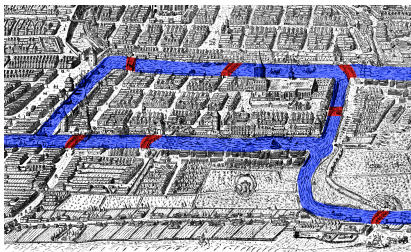


Gibt es einen Rundweg, der jede Brücke exakt einmal verwendet?

# Eulerkreis

## Definition (Eulerkreis)

Ein **Eulerkreis** in einem Graphen ist ein Zyklus, der jede Kante genau einmal enthält.



Wie schwer ist es zu entscheiden, ob ein Graph einen Eulerkreis hat?

Hat Eulerkreis gdw. jeder Knoten geraden Grad hat und Graph verbunden ist.

# Inhalt dieser Veranstaltung

