

# Theoretische Grundlagen der Informatik

## Vorlesung am 27. November 2018

INSTITUT FÜR THEORETISCHE INFORMATIK



- Die Klasse  $\mathcal{P}$  aller Probleme, die von **deterministischen** Turing-Maschinen in **polynomialer Zeit** entschieden werden.
- Die Klasse  $\mathcal{NP}$  aller Probleme, die von **nichtdeterministischen** Turing-Maschinen in **polynomialer Zeit** entschieden werden.

Ist  $\mathcal{P} = \mathcal{NP}$ ?

- **Polynomiale Transformation**  $\Pi' \propto \Pi$ 
  - Instanzen von  $\Pi' \rightarrow$  Instanzen von  $\Pi$
  - in polynomialer Zeit berechenbar
  - Ja-Instanz von  $\Pi' \rightarrow$  Ja-Instanz von  $\Pi$   
Nein-Instanz von  $\Pi' \rightarrow$  Nein-Instanz von  $\Pi$
- Problem  $\Pi$  ist  **$\mathcal{NP}$ -vollständig** wenn
  - $\Pi \in \mathcal{NP}$  und
  - $\Pi' \propto \Pi$  für alle  $\Pi' \in \mathcal{NP}$

- Die Klasse  $\mathcal{P}$  aller Probleme, die von **deterministischen** Turing-Maschinen in **polynomialer Zeit** entschieden werden.
- Die Klasse  $\mathcal{NP}$  aller Probleme, die von **nichtdeterministischen** Turing-Maschinen in **polynomialer Zeit** entschieden werden.

Ist  $\mathcal{P} = \mathcal{NP}$ ?

- **Polynomiale Transformation**  $\Pi' \propto \Pi$ 
  - Instanzen von  $\Pi' \rightarrow$  Instanzen von  $\Pi$
  - in polynomialer Zeit berechenbar
  - Ja-Instanz von  $\Pi' \rightarrow$  Ja-Instanz von  $\Pi$   
Nein-Instanz von  $\Pi' \rightarrow$  Nein-Instanz von  $\Pi$
- Problem  $\Pi$  ist  **$\mathcal{NP}$ -vollständig** wenn
  - $\Pi \in \mathcal{NP}$  und
  - ~~$\Pi' \propto \Pi$  für alle  $\Pi' \in \mathcal{NP}$~~
  - $\Pi' \propto \Pi$  für ein  $\mathcal{NP}$ -vollständiges  $\Pi'$

- Die Klasse  $\mathcal{P}$  aller Probleme, die von **deterministischen** Turing-Maschinen in **polynomialer Zeit** entschieden werden.
- Die Klasse  $\mathcal{NP}$  aller Probleme, die von **nichtdeterministischen** Turing-Maschinen in **polynomialer Zeit** entschieden werden.

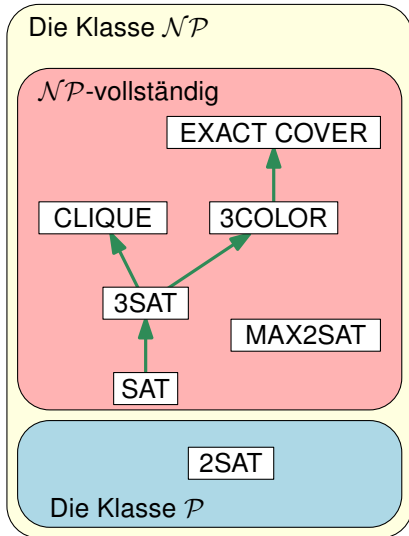
Ist  $\mathcal{P} = \mathcal{NP}$ ?

- **Polynomiale Transformation**  $\Pi' \propto \Pi$ 
  - Instanzen von  $\Pi' \rightarrow$  Instanzen von  $\Pi$
  - in polynomialer Zeit berechenbar
  - Ja-Instanz von  $\Pi' \rightarrow$  Ja-Instanz von  $\Pi$   
Nein-Instanz von  $\Pi' \rightarrow$  Nein-Instanz von  $\Pi$
- Problem  $\Pi$  ist  **$\mathcal{NP}$ -vollständig** wenn
  - $\Pi \in \mathcal{NP}$  und
  - ~~$\Pi' \propto \Pi$  für alle  $\Pi' \in \mathcal{NP}$~~
  - $\Pi' \propto \Pi$  für ein  $\mathcal{NP}$ -vollständiges  $\Pi'$

**Satz von Cook.** SAT ist  $\mathcal{NP}$ -vollständig.

# Plan für heute

- 3SAT ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3SAT \in \mathcal{NP}$
  - $\rightsquigarrow SAT \propto 3SAT$
- 2SAT ist in  $\mathcal{P}$
- MAX2SAT ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow$  Übung
- CLIQUE ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow CLIQUE \in \mathcal{NP}$
  - $\rightsquigarrow 3SAT \propto CLIQUE$
- 3COLOR ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3COLOR \in \mathcal{NP}$
  - $\rightsquigarrow 3SAT \propto 3COLOR$
- EXACT COVER ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3COLOR \in \mathcal{NP}$
  - $\rightsquigarrow 3COLOR \propto EXACT COVER$



## Problem 3SAT

**Gegeben:** Menge  $U$  von Variablen  
Menge  $C$  von Klauseln über  $U$   
jede Klausel enthält genau drei Literale

**Frage:** Existiert eine erfüllende Wahrheitsbelegung für  $C$ ?

## Problem 3SAT

**Gegeben:** Menge  $U$  von Variablen  
Menge  $C$  von Klauseln über  $U$   
jede Klausel enthält genau drei Literale

**Frage:** Existiert eine erfüllende Wahrheitsbelegung für  $C$ ?

## Satz:

Das Problem 3SAT ist  $\mathcal{NP}$ -vollständig.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

3SAT  $\in \mathcal{NP}$ :

- Es existiert eine nichtdeterministische Turing-Maschine mit polynomialer Zeitkomplexitätsfunktion die (immer stoppt und) in  $q_J$  hält bei Ja-Instanzen und in  $q_N$  hält bei Nein-Instanzen.



# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

3SAT  $\in \mathcal{NP}$ :

- Es existiert eine nichtdeterministische Turing-Maschine mit polynomialer Zeitkomplexitätsfunktion die (immer stoppt und) in  $q_J$  hält bei Ja-Instanzen und in  $q_N$  hält bei Nein-Instanzen.
- Wir konstruieren eine Orakel-Turing-Maschine:
  - Das Orakelmodul schreibt eine Wahrheitsbelegung  $t: U \rightarrow \{\text{wahr, falsch}\}$ .
  - Die endliche Kontrolle überprüft, ob jede Klausel in  $C$  durch  $t$  erfüllt ist.
  - Wenn alle Klauseln erfüllt, gehe in  $q_J$ .
  - Wenn (mindestens) eine Klausel nicht erfüllt, gehe in  $q_N$ .
  - Laufzeit ist linear in der Größe der eingegebenen Klauselmenge  $C$ .

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

3SAT  $\in \mathcal{NP}$ :

- Es existiert eine nichtdeterministische Turing-Maschine mit polynomialer Zeitkomplexitätsfunktion die (immer stoppt und) in  $q_J$  hält bei Ja-Instanzen und in  $q_N$  hält bei Nein-Instanzen.
- Wir konstruieren eine Orakel-Turing-Maschine:
  - Das Orakelmodul schreibt eine Wahrheitsbelegung  $t: U \rightarrow \{\text{wahr}, \text{falsch}\}$ .
  - Die endliche Kontrolle überprüft, ob jede Klausel in  $C$  durch  $t$  erfüllt ist.
  - Wenn alle Klauseln erfüllt, gehe in  $q_J$ .
  - Wenn (mindestens) eine Klausel nicht erfüllt, gehe in  $q_N$ .
  - Laufzeit ist linear in der Größe der eingegebenen Klauselmenge  $C$ .
- Für eine feste Wahrheitsbelegung  $t$  kann in polynomialer Zeit  $\mathcal{O}(|C|)$  überprüft werden, ob  $t$  alle Klauseln aus  $C$  erfüllt.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

SAT  $\propto$  3SAT:

- Wir geben eine polynomiale Transformation  $f$  von SAT zu 3SAT an.
- Gegeben sei eine SAT-Instanz  $I$

Wir konstruieren eine 3SAT-Instanz  $f(I)$  indem wir jede Klausel  $c$  in  $I$  einzeln auf Klausel(n)  $f(c)$  in  $f(I)$  abbilden:

- Besteht die Klausel  $c = x_1$  aus **einem** Literal, so wird  $c$  auf  $x_1 \vee x_1 \vee x_1$  abgebildet.
- Besteht die Klausel  $c = x_1 \vee x_2$  aus **zwei** Literalen, so wird  $c$  auf  $x_1 \vee x_2 \vee x_1$  abgebildet.
- Besteht die Klausel  $c$  aus **drei** Literalen, so wird  $c$  auf sich selbst abgebildet.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

Wir konstruieren eine 3SAT-Instanz  $f(I)$  indem wir jede Klausel  $c$  in  $I$  einzeln auf Klausel(n)  $f(c)$  in  $f(I)$  abbilden:

- Besteht die Klausel  $c = x_1 \vee \dots \vee x_k$  aus  $k > 3$  Literalen, bilde  $c$  wie folgt ab:
  - Führe  $k - 3$  neue Variablen  $y_{c,3}, \dots, y_{c,k-1}$  ein.
  - Bilde  $c$  auf die folgenden  $k - 2$  Klauseln ab:

$x_1 \vee x_2 \vee y_{c,3}$	“Falls $x_1, x_2 = \text{falsch} \rightsquigarrow y_{c,3}$ muss wahr”
$\overline{y_{c,3}} \vee x_3 \vee y_{c,4}$	“ $y_{c,3} = \text{wahr}, x_3 = \text{falsch} \rightsquigarrow y_{c,4} = \text{wahr}$ ”
$\vdots$	$\vdots$
$\overline{y_{c,k-2}} \vee x_{k-2} \vee y_{c,k-1}$	“ $y_{c,k-2} = \text{wahr}, x_{k-2} = \text{falsch} \rightsquigarrow y_{c,k-1} = \text{wahr}$ ”
$\overline{y_{c,k-1}} \vee x_{k-1} \vee x_k$	“ $y_{c,k-1} = \text{wahr} \rightsquigarrow x_{k-1}$ oder $x_k$ muss wahr”

- Diese Klauseln lassen sich in Zeit  $\mathcal{O}(|C| \cdot |U|)$  konstruieren.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

Noch zu zeigen:

- $I$  ist erfüllbar  $\Leftrightarrow f(I)$  ist erfüllbar

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

$I$  ist erfüllbar  $\Rightarrow f(I)$  ist erfüllbar

- Sei die SAT-Instanz  $I$  erfüllbar.
- Wir setzen eine erfüllende Wahrheitsbelegung  $t$  von  $I$  auf  $f(I)$  fort.
  - Also wenn es Literal  $x$  in  $I$  und  $f(I)$  gibt, lasse  $t(x)$  wie gehabt.
- Wir untersuchen jede Klausel  $c = x_1 \vee \dots \vee x_k$  in  $I$  einzeln.
- Da  $c$  von  $t$  erfüllt ist, gilt für mindestens ein  $i$ , dass  $t(x_i) = \text{wahr}$ .
- Fall  $k \leq 3$ : Damit ist auch Klausel  $c$  in  $f(I)$  erfüllt.
- Fall  $k > 3$ : Setze für  $j = 3, \dots, k - 1$

$$t(y_{c,j}) = \begin{cases} \text{wahr} & \text{falls } t(x_i) = \text{wahr für (mind.) ein } i \geq j \\ \text{falsch} & \text{falls } t(x_i) = \text{falsch für alle } i \geq j \end{cases}$$

- Diese Erweiterung erfüllt alle Klauseln in  $f(I)$ , die zu  $c$  gehören.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3SAT

$I$  ist erfüllbar  $\Leftrightarrow f(I)$  ist erfüllbar

- Sei die 3SAT-Instanz  $f(I)$  erfüllbar.
- Wir schränken eine erfüllende Wahrheitsbelegung  $t$  von  $f(I)$  auf  $I$  ein.
  - Also nehme für jedes Literal  $x$  in  $I$  die Belegung  $t(x)$  wie in  $f(I)$ .
- Wir untersuchen jede Klausel  $c = x_1 \vee \dots \vee x_k$  in  $I$  einzeln.
- Fall  $k \leq 3$ : Dann ist Klausel  $c$  auch in  $f(I)$ , also durch  $t$  erfüllt.
- Fall  $k > 3$ : Alle Klauseln in  $f(I)$  zu Klausel  $c$  in  $I$  sind erfüllt:

$$x_1 \vee x_2 \vee y_{c,3}$$

Falls  $t(x_1), t(x_2) = \text{falsch}$ , dann  $t(y_{c,3}) = \text{wahr}$ .

$$\overline{y_{c,j}} \vee x_j \vee y_{c,j+1}$$

Falls  $t(y_{c,j}) = \text{wahr}$ ,  $t(x_j) = \text{falsch}$ , dann  $t(y_{c,j+1}) = \text{wahr}$ .

$$\overline{y_{c,k-1}} \vee x_{k-1} \vee x_k$$

Falls  $t(y_{c,k-1}) = \text{wahr}$ , dann  $t(x_{k-1})$  oder  $t(x_k) = \text{wahr}$ .

- Also ist  $t(x_i) = \text{wahr}$  für (mind.) ein  $i$ , und demnach  $c$  erfüllt durch  $t$ .

## Problem 2SAT

**Gegeben:** Menge  $U$  von Variablen  
Menge  $C$  von Klauseln über  $U$   
wobei jede Klausel genau zwei Literale enthält

**Frage:** Existiert eine erfüllende Wahrheitsbelegung für  $C$ ?

## Satz:

Das Problem 2SAT liegt in  $\mathcal{P}$ .

Beweis: Übung



## Problem MAX2SAT

**Gegeben:** Menge  $U$  von Variablen  
Menge  $C$  von Klauseln über  $U$   
wobei jede Klausel genau zwei Literale enthält  
Zahl  $K \in \mathbb{N}$

**Frage:** Existiert eine Wahrheitsbelegung, die mindestens  $K$  Klauseln erfüllt?

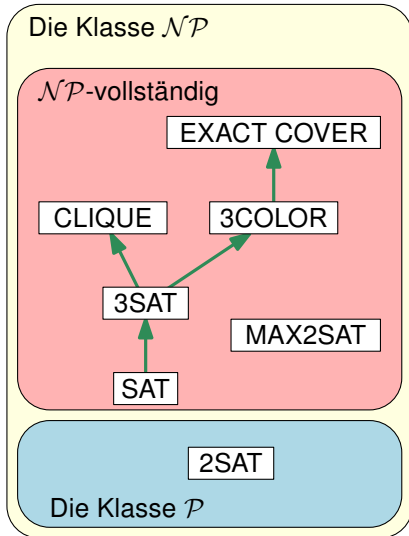
## Satz:

Das Problem MAX2SAT ist  $\mathcal{NP}$ -vollständig.

Beweis: Übung

# Plan für heute

- 3SAT ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3SAT \in \mathcal{NP}$
  - $\rightsquigarrow SAT \propto 3SAT$
- 2SAT ist in  $\mathcal{P}$
- MAX2SAT ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow$  Übung
- CLIQUE ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow CLIQUE \in \mathcal{NP}$
  - $\rightsquigarrow 3SAT \propto CLIQUE$
- 3COLOR ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3COLOR \in \mathcal{NP}$
  - $\rightsquigarrow 3SAT \propto 3COLOR$
- EXACT COVER ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3COLOR \in \mathcal{NP}$
  - $\rightsquigarrow 3COLOR \propto EXACT COVER$

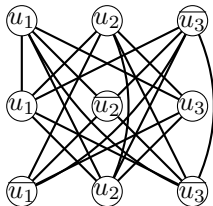


Eine **Clique** in einem Graphen  $G = (V, E)$  ist eine Menge  $V' \subseteq V$  so, dass für alle  $i, j \in V', i \neq j$ , gilt:  $\{i, j\} \in E$ .

## Problem CLIQUE

**Gegeben:** Graph  $G = (V, E)$  und ein Parameter  $K \leq |V|$

**Frage:** Gibt es in  $G$  eine Clique der Größe mindestens  $K$ ?



## Satz:

Das Problem CLIQUE ist  $\mathcal{NP}$ -vollständig.

CLIQUE  $\in \mathcal{NP}$ :

- Für eine gegebene Menge  $V' \subseteq V$  kann in polynomieller Zeit überprüft werden, ob
  - für alle  $i, j \in V', i \neq j$  gilt:  $\{i, j\} \in E$
  - $|V'| \geq K$

# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

## 3SAT $\propto$ CLIQUE

- Sei  $C = \{c_1, \dots, c_n\}$  eine 3SAT-Instanz mit  
 $c_i = x_{i1} \vee x_{i2} \vee x_{i3}$  und  $x_{ij} \in \{u_1, \dots, u_m, \overline{u_1}, \dots, \overline{u_m}\}$ .

Wir transformieren  $C$  in eine CLIQUE-Instanz  $(G = (V, E), K)$ .

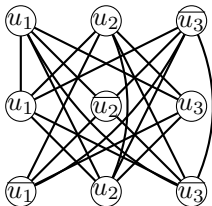
- $V$  enthält  $3n$  Knoten  $v_{ij}$  für  $1 \leq i \leq n$ ,  $1 \leq j \leq 3$ .
- $v_{ij}$  und  $v_{k\ell}$  sind durch Kanten aus  $E$  verbunden genau dann, wenn:
  - $i \neq k$  (Literale sind in verschiedenen Klauseln)
  - $x_{ij} \neq \overline{x_{k\ell}}$  (Literale sind gleichzeitig erfüllbar)
- Wir setzen  $K := n$

# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

- $V$  enthält  $3n$  Knoten  $v_{ij}$  für  $1 \leq i \leq n$ ,  $1 \leq j \leq 3$ .
- $v_{ij}$  und  $v_{kl}$  sind durch Kanten aus  $E$  verbunden genau dann, wenn:
  - $i \neq k$  (Literale sind in verschiedenen Klauseln)
  - $x_{ij} \neq \overline{x_{kl}}$  (Literale sind gleichzeitig erfüllbar)

**Beispiel:** Sei  $C = \{u_1 \vee u_2 \vee \overline{u_3}, u_1 \vee \overline{u_2} \vee u_3, \overline{u_1} \vee u_2 \vee \overline{u_3}\}$ .

Knotennummer	$v_{11}$	$v_{12}$	$v_{13}$	$v_{21}$	$v_{22}$	$v_{23}$	$v_{31}$	$v_{32}$	$v_{33}$
Literal	$u_1$	$u_2$	$\overline{u_3}$	$u_1$	$\overline{u_2}$	$u_3$	$\overline{u_1}$	$u_2$	$\overline{u_3}$



# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

- Die Transformation kann in polynomieller Zeit berechnet werden.

Noch zu zeigen:

- 3SAT-Instanz  $C$  ist Ja-Instanz  $\Leftrightarrow$  CLIQUE-Instanz  $(G, K)$  ist Ja-Instanz
  - $C$  ist Ja-Instanz:  
Es existiert Wahrheitsbelegung  $t: U \rightarrow \{\text{wahr}, \text{falsch}\}$ , so dass alle Klauseln in  $C$  unter  $t$  erfüllt sind.
  - $(G, K)$  ist Ja-Instanz:  
Es existiert Knotenmenge  $V' \subseteq V$ , so dass  $|V'| \geq K$  und alle Knoten in  $V'$  paarweise miteinander durch Kanten in  $G$  verbunden sind.

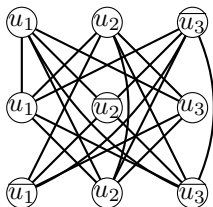
# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

3SAT-Instanz  $C$  ist Ja-Instanz  $\Rightarrow$  CLIQUE-Instanz  $(G, K)$  ist Ja-Instanz:

- Wähle eine beliebige erfüllende Wahrheitsbelegung  $t$  von  $C$ .
- Wähle in jeder Klausel ein wahres Literal.
- Diese Knoten in  $G$  bilden eine Clique  $V'$  der Größe  $K = n$ .

**Beispiel:** Sei  $C = \{u_1 \vee u_2 \vee \overline{u_3}, u_1 \vee \overline{u_2} \vee u_3, \overline{u_1} \vee u_2 \vee \overline{u_3}\}$ .

Knotennummer	$v_{11}$	$v_{12}$	$v_{13}$	$v_{21}$	$v_{22}$	$v_{23}$	$v_{31}$	$v_{32}$	$v_{33}$
Literal	$u_1$	$u_2$	$\overline{u_3}$	$u_1$	$\overline{u_2}$	$u_3$	$\overline{u_1}$	$u_2$	$\overline{u_3}$





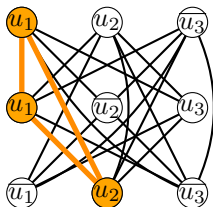
# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

3SAT-Instanz  $C$  ist Ja-Instanz  $\Rightarrow$  CLIQUE-Instanz  $(G, K)$  ist Ja-Instanz:

- Wähle eine beliebige erfüllende Wahrheitsbelegung  $t$  von  $C$ .
- Wähle in jeder Klausel ein wahres Literal.
- Diese Knoten in  $G$  bilden eine Clique  $V'$  der Größe  $K = n$ .

**Beispiel:** Sei  $C = \{u_1 \vee u_2 \vee \overline{u_3}, u_1 \vee \overline{u_2} \vee u_3, \overline{u_1} \vee u_2 \vee \overline{u_3}\}$ .

Knotennummer	$v_{11}$	$v_{12}$	$v_{13}$	$v_{21}$	$v_{22}$	$v_{23}$	$v_{31}$	$v_{32}$	$v_{33}$
Literal	$u_1$	$u_2$	$\overline{u_3}$	$u_1$	$\overline{u_2}$	$u_3$	$\overline{u_1}$	$u_2$	$\overline{u_3}$



$t(u_1) = \text{wahr}$

$t(u_2) = \text{wahr}$

$t(u_3) = \text{falsch}$

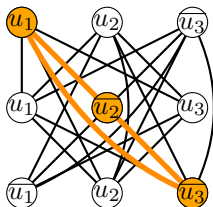
# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

3SAT-Instanz  $C$  ist Ja-Instanz  $\Rightarrow$  CLIQUE-Instanz  $(G, K)$  ist Ja-Instanz:

- Wähle eine beliebige erfüllende Wahrheitsbelegung  $t$  von  $C$ .
- Wähle in jeder Klausel ein wahres Literal.
- Diese Knoten in  $G$  bilden eine Clique  $V'$  der Größe  $K = n$ .

**Beispiel:** Sei  $C = \{u_1 \vee u_2 \vee \overline{u_3}, u_1 \vee \overline{u_2} \vee u_3, \overline{u_1} \vee u_2 \vee \overline{u_3}\}$ .

Knotennummer	$v_{11}$	$v_{12}$	$v_{13}$	$v_{21}$	$v_{22}$	$v_{23}$	$v_{31}$	$v_{32}$	$v_{33}$
Literal	$u_1$	$u_2$	$\overline{u_3}$	$u_1$	$\overline{u_2}$	$u_3$	$\overline{u_1}$	$u_2$	$\overline{u_3}$



$$t(u_1) = \text{wahr}$$

$$t(u_2) = \text{falsch}$$

$$t(u_3) = \text{falsch}$$

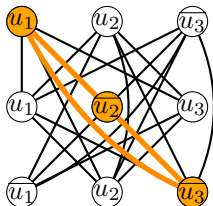
# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

3SAT-Instanz  $C$  ist Ja-Instanz  $\Leftrightarrow$  CLIQUE-Instanz  $(G, K)$  ist Ja-Instanz:

- Wähle eine Clique  $V'$  der Größe  $K = n$  in  $G$ .
- Dies ist ein Knoten pro Klausel. Wir setzen dieses Literal in  $t$  auf wahr.
- Dann ist  $t(u) \neq t(\bar{u})$  und alle Klauseln in  $C$  sind erfüllt.

**Beispiel:** Sei  $C = \{u_1 \vee u_2 \vee \bar{u}_3, u_1 \vee \bar{u}_2 \vee u_3, \bar{u}_1 \vee u_2 \vee \bar{u}_3\}$ .

Knotennummer	$v_{11}$	$v_{12}$	$v_{13}$	$v_{21}$	$v_{22}$	$v_{23}$	$v_{31}$	$v_{32}$	$v_{33}$
Literal	$u_1$	$u_2$	$\bar{u}_3$	$u_1$	$\bar{u}_2$	$u_3$	$\bar{u}_1$	$u_2$	$\bar{u}_3$ .



$$t(u_1) = \text{wahr}$$

$$t(u_2) = \text{falsch}$$

$$t(u_3) = \text{falsch}$$

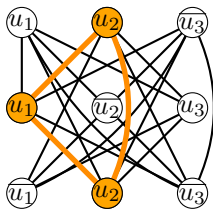
# Beweis: $\mathcal{NP}$ -Vollständigkeit von CLIQUE

3SAT-Instanz  $C$  ist Ja-Instanz  $\Leftrightarrow$  CLIQUE-Instanz  $(G, K)$  ist Ja-Instanz:

- Wähle eine Clique  $V'$  der Größe  $K = n$  in  $G$ .
- Dies ist ein Knoten pro Klausel. Wir setzen dieses Literal in  $t$  auf wahr.
- Dann ist  $t(u) \neq t(\bar{u})$  und alle Klauseln in  $C$  sind erfüllt.

**Beispiel:** Sei  $C = \{u_1 \vee u_2 \vee \bar{u}_3, u_1 \vee \bar{u}_2 \vee u_3, \bar{u}_1 \vee u_2 \vee \bar{u}_3\}$ .

Knotennummer	$v_{11}$	$v_{12}$	$v_{13}$	$v_{21}$	$v_{22}$	$v_{23}$	$v_{31}$	$v_{32}$	$v_{33}$
Literal	$u_1$	$u_2$	$\bar{u}_3$	$u_1$	$\bar{u}_2$	$u_3$	$\bar{u}_1$	$u_2$	$\bar{u}_3$



$$t(u_1) = \text{wahr}$$

$$t(u_2) = \text{wahr}$$

$$t(u_3) = \text{beliebig}$$

## Problem COLOR

**Gegeben:** Graph  $G = (V, E)$  und ein Parameter  $K \in \mathbb{N}$ .

**Frage:** Gibt es eine Knotenfärbung von  $G$  mit höchstens  $K$  Farben, so dass je zwei adjazente Knoten verschiedene Farben besitzen?

3COLOR bezeichnet das Problem COLOR mit festem Parameter  $K = 3$ .

## Satz:

Das Problem 3COLOR ist  $\mathcal{NP}$ -vollständig.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3COLOR

3COLOR  $\in \mathcal{NP}$

- Es kann in Zeit  $\mathcal{O}(|E|)$  überprüft werden, ob eine Färbung von Graph  $G = (V, E)$  mit drei Farben zulässig ist.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von 3COLOR

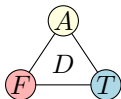
## 3SAT $\propto$ 3COLOR

- Sei  $I$  eine 3SAT-Instanz mit Variablen  $U = \{u_1, \dots, u_m\}$  und Klauseln  $\{c_1, \dots, c_n\}$ .
- Wir konstruieren in Polynomialzeit eine 3COLOR-Instanz  $G$ .
- Es soll gelten:  $I$  ist erfüllbar  $\Leftrightarrow G$  ist 3-färbbar.

# Konstruktion von 3COLOR-Instanz $G$

Der Graph  $G$  enthält:

- Hauptdreieck  $D$  aus Knoten  $\{T, F, A\}$  und Kanten  $\{\{T, F\}, \{F, A\}, \{T, A\}\}$
- Interpretation:  $T, F, A$  sind die drei Farben mit denen  $G$  gefärbt wird.
- Interpretation:  $T \longleftrightarrow$  wahr,  $F \longleftrightarrow$  falsch

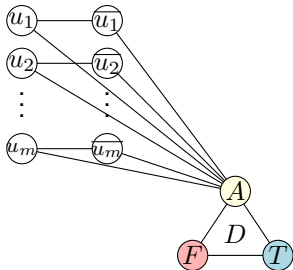




# Konstruktion von 3COLOR-Instanz $G$

Der Graph  $G$  enthält:

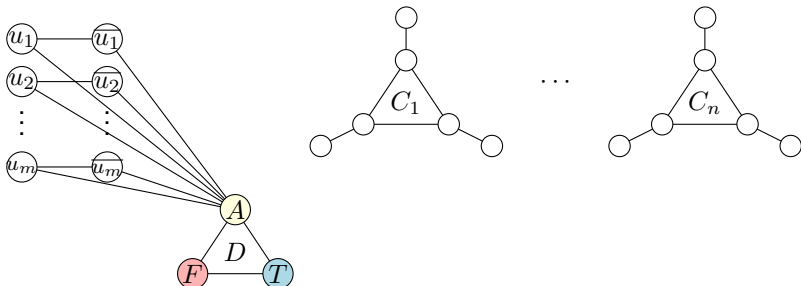
- Hauptdreieck  $D$  aus Knoten  $\{T, F, A\}$  und Kanten  $\{\{T, F\}, \{F, A\}, \{T, A\}\}$
- Interpretation:  $T, F, A$  sind die drei Farben mit denen  $G$  gefärbt wird.
- Interpretation:  $T \longleftrightarrow$  wahr,  $F \longleftrightarrow$  falsch
- Für jede Variable  $u_i \in U$  zwei Knoten  $u_i, \bar{u}_i$  und ein Dreieck  $\{u_i, \bar{u}_i, A\}$ .
- Interpretation: Falls  $u_i$  in Farbe  $T$ , muss  $\bar{u}_i$  in Farbe  $F$ .



# Konstruktion von 3COLOR-Instanz $G$

Der Graph  $G$  enthält:

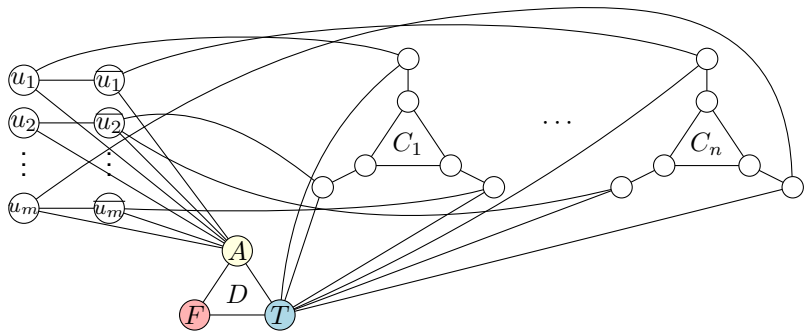
- Für jede Klausel  $c_j = x \vee y \vee z$  eine Komponente  $C_j$  wie folgt:
  - $C_j$  besteht aus sechs Knoten, einem "inneren Dreieck" und drei "Satelliten".



# Konstruktion von 3COLOR-Instanz $G$

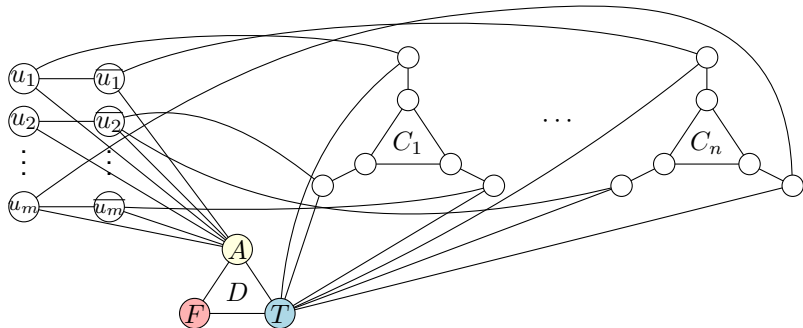
Der Graph  $G$  enthält:

- Für jede Klausel  $c_j = x \vee y \vee z$  eine Komponente  $C_j$  wie folgt:
  - $C_j$  besteht aus sechs Knoten, einem "inneren Dreieck" und drei "Satelliten".
- Jeder der drei Satelliten wird mit einem der Literale  $x, y, z$  verbunden.
- Alle drei Satelliten werden mit dem Knoten  $T$  in  $D$  verbunden.



# Polynomialität der Reduktion

- Die Knotenanzahl von  $G$  liegt in  $\mathcal{O}(n + m)$ .
- Deswegen ist die Transformation polynomial.

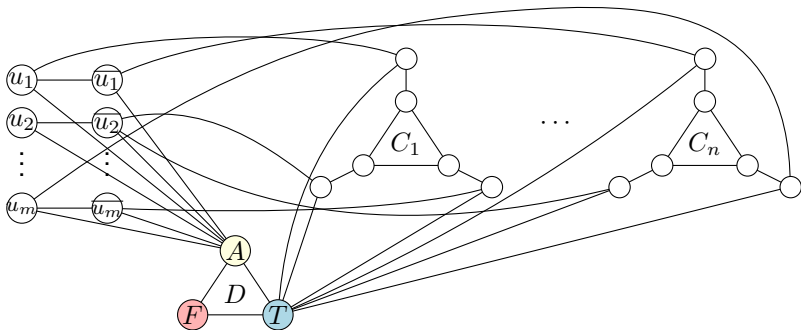


**Zu zeigen:**

3SAT-Instanz  $I$  ist Ja-Instanz  $\Leftrightarrow$  3COLOR-Instanz  $G$  ist Ja-Instanz.

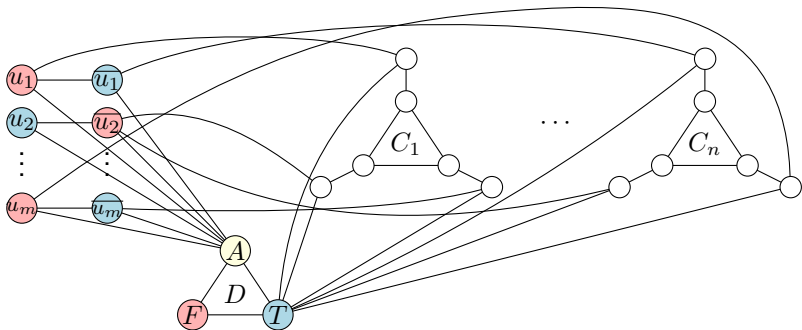
# Instanz / erfüllbar $\Rightarrow$ Instanz $G$ 3-färbbar

- Betrachte zulässige Wahrheitsbelegung  $t$  für  $I$ .
- Färbe wahre Literale wie Knoten  $T$ , falsche Literale wie Knoten  $F$ .



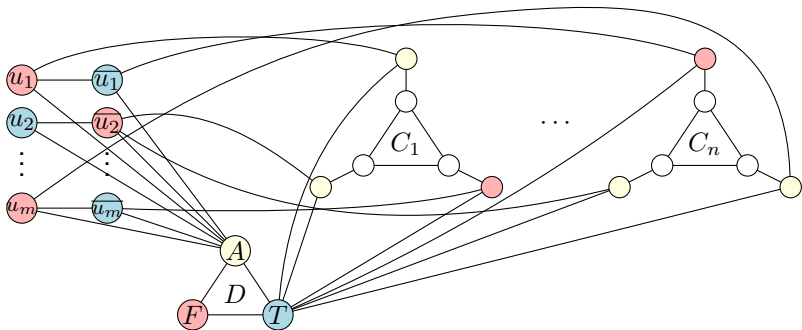
# Instanz / erfüllbar $\Rightarrow$ Instanz $G$ 3-färbbar

- Betrachte zulässige Wahrheitsbelegung  $t$  für  $I$ .
- Färbe wahre Literale wie Knoten  $T$ , falsche Literale wie Knoten  $F$ .
- Färbe Satelliten zu einem beliebigen wahren Literal mit  $F$ .



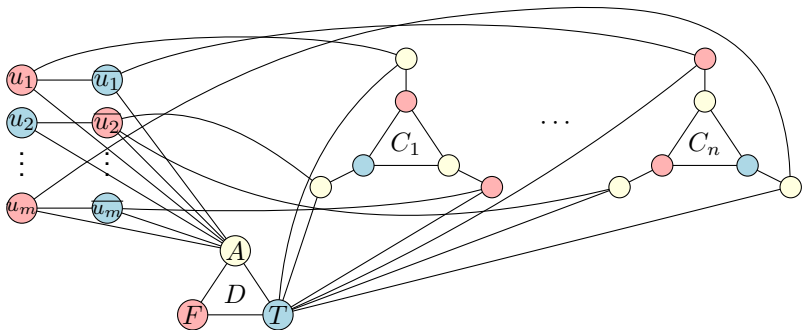
# Instanz / erfüllbar $\Rightarrow$ Instanz $G$ 3-färbbar

- Betrachte zulässige Wahrheitsbelegung  $t$  für  $I$ .
- Färbe wahre Literale wie Knoten  $T$ , falsche Literale wie Knoten  $F$ .
- Färbe Satelliten zu einem beliebigen wahren Literal mit  $F$ .
- Färbe die beiden anderen Satelliten mit  $A$ .



# Instanz / erfüllbar $\Rightarrow$ Instanz $G$ 3-färbbar

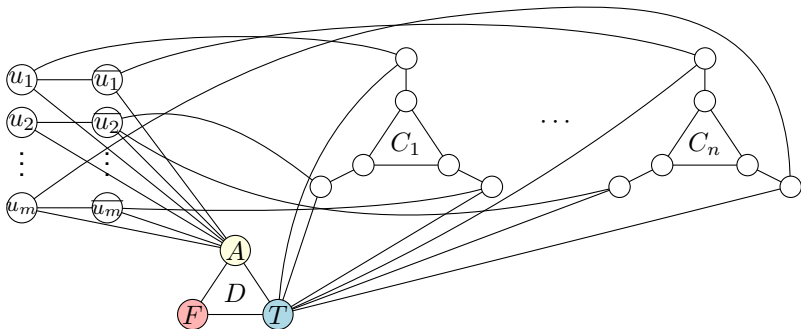
- Betrachte zulässige Wahrheitsbelegung  $t$  für  $I$ .
- Färbe wahre Literale wie Knoten  $T$ , falsche Literale wie Knoten  $F$ .
- Färbe Satelliten zu einem beliebigen wahren Literal mit  $F$ .
- Färbe die beiden anderen Satelliten mit  $A$ .
- Inneres Dreieck kann dann zulässig gefärbt werden.





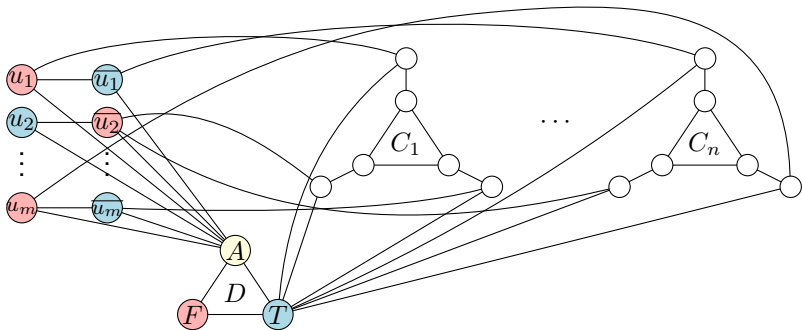
# Instanz / erfüllbar $\Leftrightarrow$ Instanz $G$ 3-färbbar

- Betrachte 3-Färbung von  $G$ . Interpretiere  $T \leftrightarrow$  wahr,  $F \leftrightarrow$  falsch.



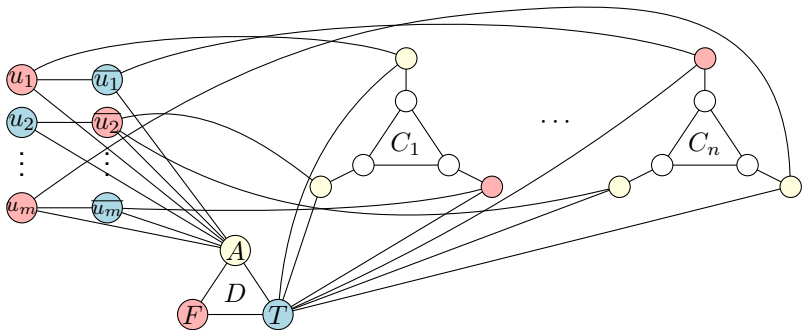
# Instanz / erfüllbar $\Leftrightarrow$ Instanz $G$ 3-färbbar

- Betrachte 3-Färbung von  $G$ . Interpretiere  $T \leftrightarrow$  wahr,  $F \leftrightarrow$  falsch.
- Färbung von Literal-Knoten induziert eine Wahrheitsbelegung  $t$  von  $I$ .



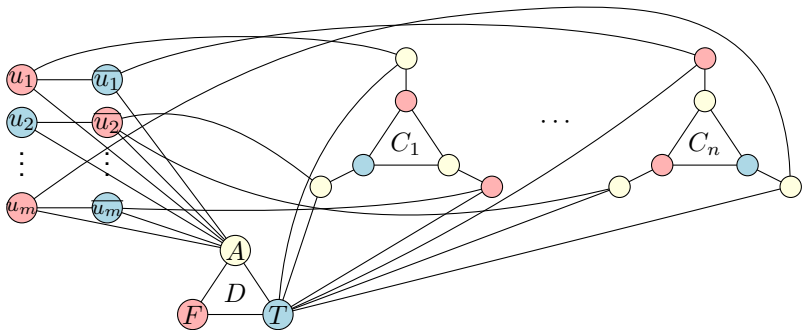
# Instanz / erfüllbar $\Leftrightarrow$ Instanz $G$ 3-färbbar

- Betrachte 3-Färbung von  $G$ . Interpretiere  $T \leftrightarrow$  wahr,  $F \leftrightarrow$  falsch.
- Färbung von Literal-Knoten induziert eine Wahrheitsbelegung  $t$  von  $I$ .
- Kein Satellit ist in Farbe von  $T$ .



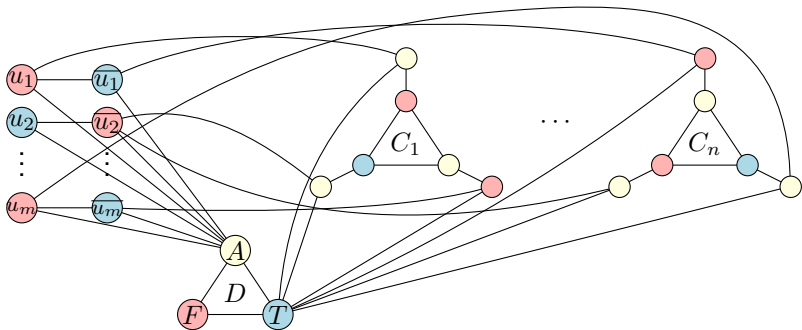
# Instanz / erfüllbar $\Leftrightarrow$ Instanz $G$ 3-färbbar

- Betrachte 3-Färbung von  $G$ . Interpretiere  $T \leftrightarrow$  wahr,  $F \leftrightarrow$  falsch.
- Färbung von Literal-Knoten induziert eine Wahrheitsbelegung  $t$  von  $I$ .
- Kein Satellit ist in Farbe von  $T$ .
- Nicht alle Satelliten sind in Farbe von  $A$  wegen des inneren Dreiecks.



# Instanz $I$ erfüllbar $\Leftrightarrow$ Instanz $G$ 3-färbbar

- Betrachte 3-Färbung von  $G$ . Interpretiere  $T \leftrightarrow$  wahr,  $F \leftrightarrow$  falsch.
- Färbung von Literal-Knoten induziert eine Wahrheitsbelegung  $t$  von  $I$ .
- Kein Satellit ist in Farbe von  $T$ .
- Nicht alle Satelliten sind in Farbe von  $A$  wegen des inneren Dreiecks.  
 $\Rightarrow$  Mindestens ein Literal pro Klausel in Farbe  $T$ , also  $I$  erfüllbar.



# Zwischenstand Polynomiale Reduktion

$\Pi'$	$\propto$	$\Pi$
bekannt $\mathcal{NP}$ -vollständig (wähle ähnlich zu $\Pi$ )		$\mathcal{NP}$ -vollständig zu zeigen
beliebige Instanzen	$\xrightarrow{f}$	spezielle Instanzen (größer, aber noch polynomial)
Ja-Instanzen	$\longleftrightarrow$	Ja-Instanzen

$\Pi'$	$\propto$	$\Pi$
bekannt $\mathcal{NP}$ -vollständig (wähle ähnlich zu $\Pi$ )		$\mathcal{NP}$ -vollständig zu zeigen
beliebige Instanzen	$\xrightarrow{f}$	spezielle Instanzen (größer, aber noch polynomial)
Ja-Instanzen	$\longleftrightarrow$	Ja-Instanzen

## Testen Sie sich:

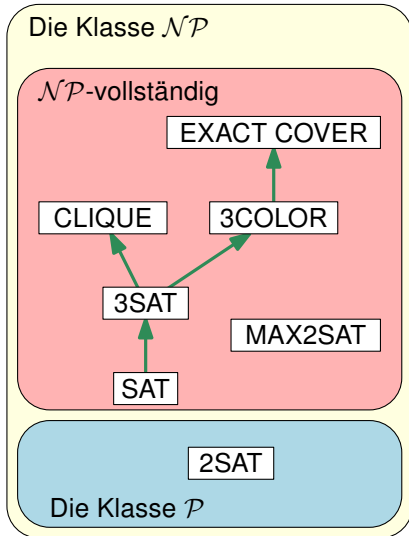
Können Sie zeigen, dass folgende Probleme  $\mathcal{NP}$ -vollständig sind?

**Geg.:** Graph  $G$ , Zahl  $K$   
**Frage:**  $\exists V' \subseteq V, |V'| = k$ ,  
keine zwei Knoten  
in  $V'$  verbunden?

**Geg.:** Graph  $G = (V, E)$   
**Frage:**  $\exists$  4-Färbung von  $V$ ,  
keine zwei Knoten  
gleichfarbig verbunden?

# Plan für heute

- 3SAT ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3SAT \in \mathcal{NP}$
  - $\rightsquigarrow SAT \propto 3SAT$
- 2SAT ist in  $\mathcal{P}$
- MAX2SAT ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow$  Übung
- CLIQUE ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow CLIQUE \in \mathcal{NP}$
  - $\rightsquigarrow 3SAT \propto CLIQUE$
- 3COLOR ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3COLOR \in \mathcal{NP}$
  - $\rightsquigarrow 3SAT \propto 3COLOR$
- EXACT COVER ist  $\mathcal{NP}$ -vollständig
  - $\rightsquigarrow 3COLOR \in \mathcal{NP}$
  - $\rightsquigarrow 3COLOR \propto EXACT COVER$





## Problem EXACT COVER

**Gegeben:** Eine endliche Menge  $X$  und eine Familie  $\mathcal{S}$  von Teilmengen von  $X$ .

**Frage:** Existiert eine Menge  $\mathcal{S}' \subseteq \mathcal{S}$ , so dass jedes Element aus  $X$  in genau einer Menge aus  $\mathcal{S}'$  liegt?

## Problem EXACT COVER

**Gegeben:** Eine endliche Menge  $X$  und eine Familie  $\mathcal{S}$  von Teilmengen von  $X$ .

**Frage:** Existiert eine Menge  $\mathcal{S}' \subseteq \mathcal{S}$ , so dass jedes Element aus  $X$  in genau einer Menge aus  $\mathcal{S}'$  liegt?

## Beispiel:

$$X = \{1, 2, \dots, 7\}$$

$$\mathcal{S} = \{\{1, 2, 3\}, \{1, 2, 4\}, \{2, 3, 4\}, \{1, 3, 4\}, \{1, 5\}, \{3, 5\}, \{1, 3\}, \\ \{5, 6, 7\}, \{4, 5, 6\}, \{4, 5, 7\}, \{4, 6, 7\}, \{5, 6\}, \{5, 7\}, \{6, 7\}\}$$

Ist  $(X, \mathcal{S})$  eine Ja-Instanz?

## Problem EXACT COVER

**Gegeben:** Eine endliche Menge  $X$  und eine Familie  $\mathcal{S}$  von Teilmengen von  $X$ .

**Frage:** Existiert eine Menge  $\mathcal{S}' \subseteq \mathcal{S}$ , so dass jedes Element aus  $X$  in genau einer Menge aus  $\mathcal{S}'$  liegt?

## Beispiel:

$$X = \{1, 2, \dots, 7\}$$

$$\mathcal{S} = \{\{1, 2, 3\}, \{1, 2, 4\}, \{2, 3, 4\}, \{1, 3, 4\}, \{1, 5\}, \{3, 5\}, \{1, 3\}, \\ \{5, 6, 7\}, \{4, 5, 6\}, \{4, 5, 7\}, \{4, 6, 7\}, \{5, 6\}, \{5, 7\}, \{6, 7\}\}$$

$$\mathcal{S}' = \{\{1, 5\}, \{2, 3, 4\}, \{6, 7\}\}$$

Ja.

## Problem EXACT COVER

**Gegeben:** Eine endliche Menge  $X$  und eine Familie  $\mathcal{S}$  von Teilmengen von  $X$ .

**Frage:** Existiert eine Menge  $\mathcal{S}' \subseteq \mathcal{S}$ , so dass jedes Element aus  $X$  in genau einer Menge aus  $\mathcal{S}'$  liegt?

## Satz:

Problem EXACT COVER ist  $\mathcal{NP}$ -vollständig.

# Beweis: $\mathcal{NP}$ -Vollständigkeit von EXACT COVER

EXACT COVER  $\in \mathcal{NP}$

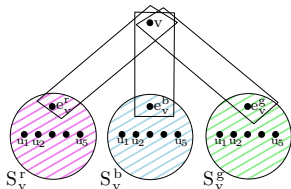
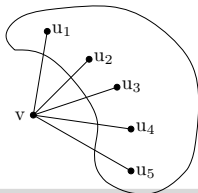
- Es kann in Polynomialzeit überprüft werden, ob eine Teilmenge  $S' \subseteq S$  aus disjunkten Mengen besteht und  $X$  überdeckt.

## 3COLOR $\propto$ EXACT COVER

- Sei  $G = (V, E)$  eine 3COLOR-Instanz.
- Wir konstruieren in Polynomialzeit eine EXACT COVER-Instanz  $(X, \mathcal{S})$ .
- Es soll gelten:  $G$  ist 3-färbbar  $\Leftrightarrow (X, \mathcal{S})$  hat exakte Überdeckung

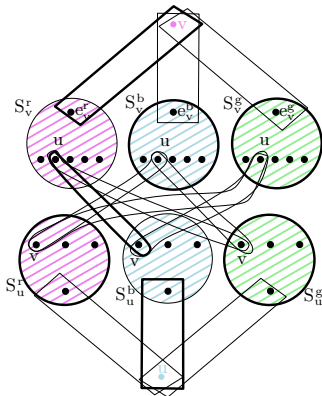
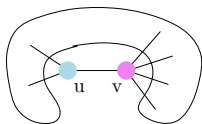
# Konstruktion von $(X, S)$

- Sei  $C = \{r, b, g\}$  (steht für rot, blau, grün).
- Sei  $N(v) := \{u \in V : \{u, v\} \in E\}$  die Nachbarschaft von  $v$ .
- Für jedes  $v \in V$  enthalte  $X$  ein "Element"  $v$  und jeweils  $3 \cdot |N(v)| + 3$  zusätzliche Elemente.
- Zu jedem  $v \in V$  gebe es in  $S$  drei disjunkte Mengen  $S_v^r, S_v^b, S_v^g$  mit jeweils  $|N(v)| + 1$  Elementen.
- Außerdem enthalte  $S$  für jedes  $v$  drei zweielementige Mengen  $\{v, e_v^r\}, \{v, e_v^b\}$  und  $\{v, e_v^g\}$  mit  $e_v^r \in S_v^r, e_v^b \in S_v^b$  und  $e_v^g \in S_v^g$ .
- **Interpretation:**  $S_v^r$  entspricht der "Farbe"  $r$ , enthält für jeden Knoten aus  $N(v)$  eine Kopie und einen zusätzlichen Knoten  $e_v^r$ .



# Konstruktion von $(X, S)$

- Außerdem enthält  $S$  für jede Kante  $\{u, v\} \in E$  und je zwei  $c, c' \in C$ ,  $c \neq c'$ , die zweielementigen Mengen  $\{u_v^c, v_u^{c'}\}$ ,  $u_v^c \in S_v^c$  "Kopie" von  $u$ ,  $v_u^{c'} \in S_u^{c'}$  "Kopie" von  $v$ .





# Konstruktion von $(X, S)$

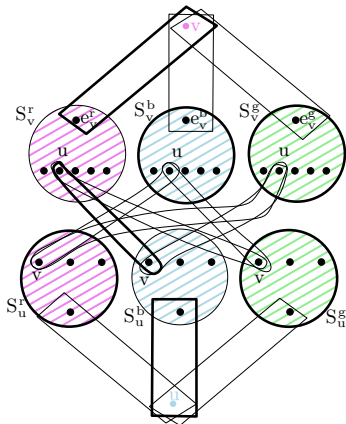
- Die Konstruktion ist polynomial.

Noch zu zeigen:

- $G$  ist 3-färbbar  $\Leftrightarrow (X, S)$  hat exakte Überdeckung

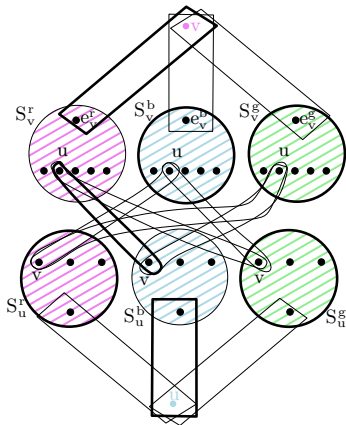
## $G$ 3-färbbar $\Rightarrow (X, S)$ hat exakte Überdeckung

- Sei  $\chi : V \rightarrow C$  eine zulässige Dreifärbung.
- $S'$  enthalte für jedes  $v \in V$  die Mengen  $\{v, e_v^{\chi(v)}\}$  und  $S_v^c$  mit  $c \neq \chi(v)$ .
- Diese Mengen überdecken alle Elemente exakt, außer den Elementen der Form  $u_v^{\chi(v)}, v_u^{\chi(u)}$  für  $\{u, v\} \in E$ .
- Daher enthalte  $S'$  für jede Kante  $\{u, v\} \in E$  die Menge  $\{u_v^{\chi(v)}, v_u^{\chi(u)}\}$ .
- Diese Menge existiert, da  $\chi(u) \neq \chi(v)$ , und damit überdeckt  $S'$  jedes Element aus  $X$  genau einmal.



# $G$ 3-färbbar $\Leftrightarrow (X, S)$ hat exakte Überdeckung

- Sei also  $S'$  eine exakte Überdeckung.
- Jedes Element  $v$  muss von genau einer Menge der Form  $\{v, e_v^c\}$  überdeckt sein.
- Dies induziert eine Färbung  $\chi$  von  $G$  mit den Farben  $r, b$  und  $g$ .
- Wir müssen beweisen, dass diese Färbung zulässig ist
- Da für jedes  $v$  bereits  $\{v, e_v^{\chi(v)}\} \in S'$ , kann  $e_v^c$  mit  $c \neq \chi(v)$  nur durch die Menge  $S_v^c$  überdeckt werden.



# $G$ 3-färbbar $\Leftrightarrow (X, S)$ hat exakte Überdeckung

- Da für jedes  $v$  bereits  $\{v, e_v^{\chi(v)}\} \in S'$ , kann  $e_v^c$  mit  $c \neq \chi(v)$  nur durch die Menge  $S_v^c$  überdeckt werden.
- Da die Mengen der Form  $\{v, e_v^{\chi(v)}\}$  und  $S_v^c$ ,  $c \neq \chi(v)$ , alle Elemente außer den  $u^{\chi(v)}$  mit  $\{u, v\} \in E$  überdecken, müssen auch die Mengen  $\{u^{\chi(v)}, v_u^{\chi(u)}\}$  für  $\{u, v\} \in E$  in  $S'$  enthalten sein.
- Für diese gilt per Konstruktion  $\chi(v) \neq \chi(u)$ .

