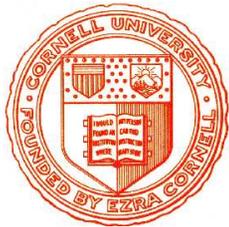


Theoretische Informatik II

Einheit 6.3

\mathcal{NP} -vollständige Probleme



1. Logische Probleme
2. Graphbasierte Probleme
3. Scheduling Probleme

Direkter Beweis ist zu aufwendig

- Würde explizite Codierung beliebiger NTMs erfordern
- Verwende $L \in \mathcal{NPC} \Leftrightarrow L \in \mathcal{NP} \wedge \exists L' \in \mathcal{NPC}. L' \leq_p L$

1. Zeige $L \in \mathcal{NP}$:

- Beschreibe, welchen Lösungsvorschlag die OTM generiert
- Beschreibe, wie Lösungsvorschlag deterministisch überprüft wird
- Zeige, daß das Prüfverfahren polynomiell ist

2. Zeige $\exists L' \in \mathcal{NPC}. L' \leq_p L$:

- Wähle ein ähnliches, bekannt \mathcal{NP} -vollständiges Problem L'
- Beschreibe Transformationsfunktion f , welche Eingaben über dem Alphabet Σ' für L' in Wörter über dem Alphabet für L umwandelt
- Zeige für alle $x \in \Sigma'^*$: $x \in L' \Leftrightarrow f(x) \in L$ (d.h. $L' = f^{-1}(L)$)
- Zeige, daß f in polynomieller Zeit berechnet werden kann

ERFÜLLBARKEIT MIT 3 LITERALEN PRO KLAUSEL

$$\mathbf{3SAT} = \{k_1, \dots, k_m \mid k_i = z_{i1} \vee z_{i2} \vee z_{i3} \text{ mit } z_{ij} \in \{x_1, \overline{x_1}, \dots, x_n, \overline{x_n}\} \\ \wedge \exists a_1, \dots, a_n \in \{0, 1\}. \forall j \leq m. a_1, \dots, a_n \text{ erfüllt } k_j\}$$

1. Zeige $\mathbf{3SAT} \in \mathcal{NP}$:

– Wie $SAT \in \mathcal{NP}$: Rate Belegung der Variablen und werte Klauseln aus

2. Zeige $\mathbf{SAT} \leq_p \mathbf{3SAT}$:

Satz 10.15

e) **Normalisierung** der Klauseln k_1, \dots, k_m über x_1, \dots, x_n .

Ersetze Klausel k_i durch äquivalente Menge von Dreierklauseln

- Ersetze einelementige Klauseln $k_i = z$ durch $z \vee z \vee z$
- Ersetze zweielementige Klauseln $k_i = z \vee z'$ durch $z \vee z \vee z'$
- Übernehme dreielementige Klauseln unverändert
- Ersetze Klauseln $k_i = z_1 \vee z_2 \vee \dots \vee z_j$ durch $j-2$ neue Klauseln mit neuen Variablen $y_{i,l}$: $(z_1 \vee z_2 \vee y_{i,1}) \wedge (\overline{y_{i,1}} \vee z_3 \vee y_{i,2}) \wedge \dots \wedge (\overline{y_{i,j-3}} \vee z_{j-1} \vee z_j)$

f) k_i erfüllbar genau dann wenn normalisierte Klauselmenge erfüllbar

Für die Transformation f gilt: $\forall F. F \in \mathbf{SAT} \Leftrightarrow f(F) \in \mathbf{3SAT}$

g) Normalisierung der Klauseln möglich in **polynomieller Zeit**

DAS CLIQUEN PROBLEM IST \mathcal{NP} -VOLLSTÄNDIG

CLIQUE = $\{ (G, k) \mid G=(V, E) \text{ Graph} \wedge \exists V_c \subseteq V. |V_c| \geq k \wedge V_c \text{ Clique in } G \}$

1. Zeige **CLIQUE** $\in \mathcal{NP}$:

a) Rate eine Knotenmenge $V_c \subseteq V$

b/c) Prüfe $|V_c| \geq k$

maximal $|V_c|$ Schritte

Prüfe: $\forall v \neq v' \in V_c. \{v, v'\} \in E$

*maximal $|V_c|^2 * |E| \leq |V|^4$ Schritte*

2. Zeige **3SAT** \leq_p **CLIQUE**:

– Gegeben $F = (k_1, \dots, k_m)$ mit $k_i = z_{i1} \vee z_{i2} \vee z_{i3}$ und $z_{ij} \in \{x_1, \dots, \overline{x_n}\}$

– Konstruiere Graphen $G_F := (V, E)$ mit

$V := \{v_{ij} \mid 1 \leq i \leq m, 1 \leq j \leq 3\}$ und $E := \{ \{v_{ij}, v_{i'j'}\} \mid i \neq i' \wedge z_{ij} \neq \overline{z_{i'j'}} \}$

e) Setze $f(F) := (G_F, m)$

f) Dann gilt $F \in 3SAT \Leftrightarrow f(F) \in CLIQUE$

(Beweis folgt)

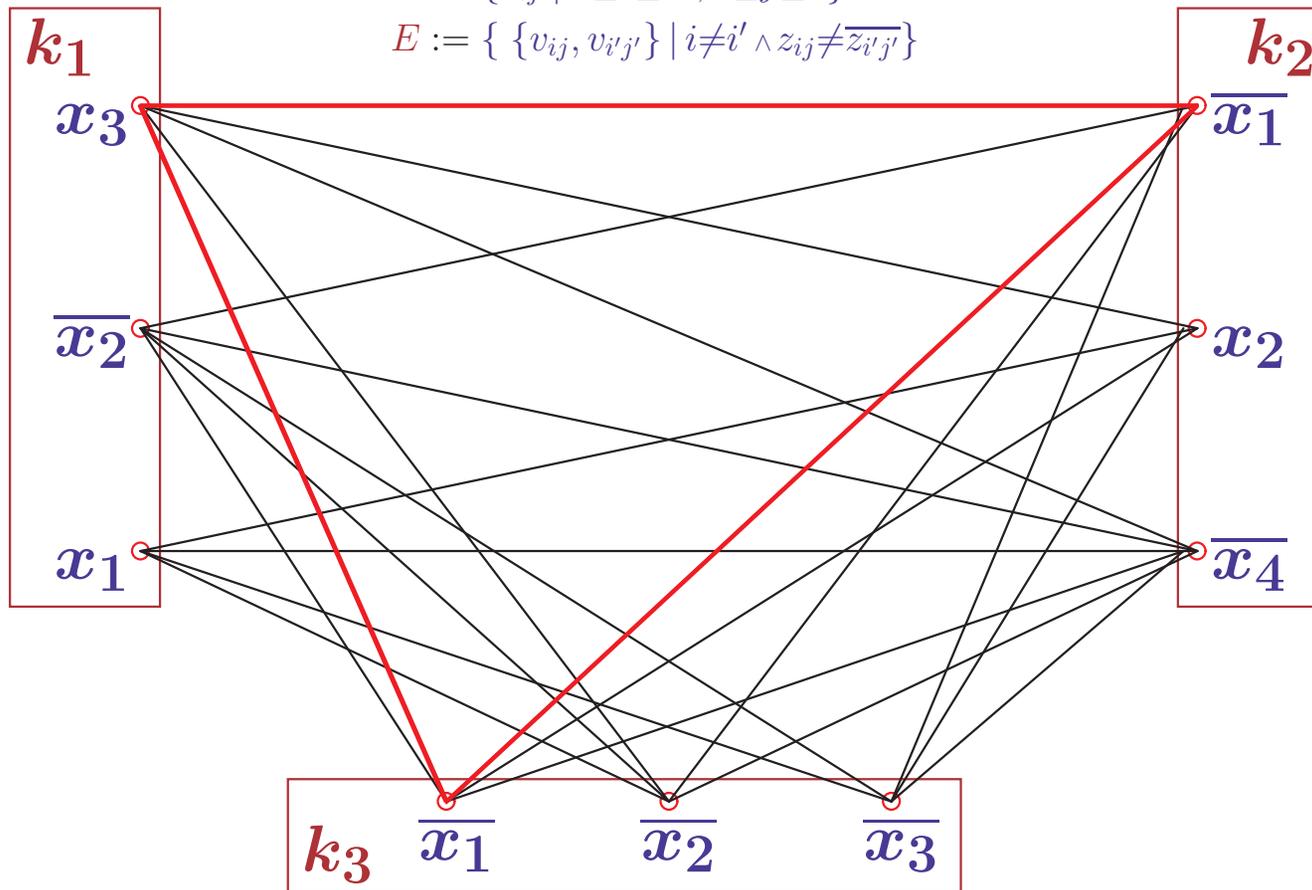
g) f ist in polynomieller Zeit berechenbar

CODIERUNG EINER FORMEL ALS CLIQUENPROBLEM

$$F = (k_1, k_2, k_3) \text{ mit } k_1 = x_1 \vee \overline{x_2} \vee x_3 \quad k_2 = \overline{x_1} \vee x_2 \vee \overline{x_4} \quad k_3 = \overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$$

$$V := \{v_{ij} \mid 1 \leq i \leq m, 1 \leq j \leq 3\}$$

$$E := \{ \{v_{ij}, v_{i'j'}\} \mid i \neq i' \wedge z_{ij} \neq \overline{z_{i'j'}} \}$$



Gibt es in dem Graphen eine 3-Clique?

KORREKTHEIT DER TRANSFORMATION $3SAT \mapsto CLIQUE$

Gegeben $F = (k_1, \dots, k_m)$ mit $k_i = z_{i1} \vee z_{i2} \vee z_{i3}$ und $z_{ij} \in \{x_1, \dots, \overline{x_n}\}$

Setze $f(F) := (G_F, m)$ mit $G_F := (V, E)$, wobei

$V := \{v_{ij} \mid 1 \leq i \leq m, 1 \leq j \leq 3\}$ und $E := \{ \{v_{ij}, v_{i'j'}\} \mid i \neq i' \wedge z_{ij} \neq \overline{z_{i'j'}} \}$

Es sei $F \in 3SAT$. Dann gibt es eine erfüllende Belegung der z_{ij}

- Wähle aus jeder Klausel k_i ein Literal mit dem Wert 1
- Dann bilden die zugehörigen Knoten eine m -Clique in G_F

Also gilt $f(F) \in CLIQUE$

Sei umgekehrt $f(F) \in CLIQUE$

- Dann hat G_F eine m -Clique V_c , d.h. $\{v_{ij}, v_{i'j'}\} \in E$ für alle $v_{ij} \neq v_{i'j'} \in V_c$
- Per Konstruktion von E enthält V_c für jedes i genau einen Knoten v_{ij} und für je zwei Knoten $v_{ij}, v_{i'j'} \in V_c$ gilt $z_{ij} \neq \overline{z_{i'j'}}$
- Belegen aller zu $v_{ij} \in V_c$ gehörigen z_{ij} mit 1
ist widerspruchsfrei möglich und erfüllt alle Klauseln k_i

Also gilt $F \in 3SAT$



$3SAT \leq_p CLIQUE$

DAS VERTEX COVER PROBLEM IST \mathcal{NP} -VOLLSTÄNDIG

$$VC = \{(G, k) \mid G \text{ Graph} \wedge \exists V' \subseteq V. |V'| \leq k \wedge V' \text{ Knotenüberdeckung von } G\}$$

1. Zeige $VC \in \mathcal{NP}$:

a) Rate eine Knotenmenge $V' \subseteq V$

b) Prüfe $|V'| \leq k$

Prüfe: $\forall \{v, v'\} \in E. v \in V' \vee v' \in V'$

maximal $|V'|$ Schritte

*maximal $|V'| * |E| \leq |V|^3$ Schritte*

c) Gesamte Anzahl der Schritte ist in $\mathcal{O}(|V|^3)$

2. Zeige $CLIQUE \leq_p VC$:

(vgl. Einheit 6.2)

e) Es ist V_c eine Clique in $G = (V, E)$

$\Leftrightarrow \forall v, v' \in V_c. v \neq v' \Rightarrow \{v, v'\} \in E$

(Definition)

$\Leftrightarrow \forall \{v, v'\} \notin E. v \neq v' \Rightarrow v \notin V_c \vee v' \notin V_c$

(Kontraposition)

$\Leftrightarrow \forall \{v, v'\} \in E^c. v \in V - V_c \vee v' \in V - V_c$

(Positive Formulierung)

$\Leftrightarrow V - V_c$ Knotenüberdeckung des Komplementgraphen $G^c = (V, E^c)$

Setze $f(G, k) := (G^c, |V| - k)$

f) Es folgt $(G, k) \in CLIQUE \Leftrightarrow f(G, k) \in VC$

g) f ist in polynomieller Zeit $\mathcal{O}(|V|^2)$ berechenbar

GERICHTETER HAMILTONSCHER KREIS (DHC)

AUSGANGSPUNKT FÜR BEWEIS DER \mathcal{NP} -VOLLSTÄNDIGKEIT VON TSP

$$DHC = \{ G \mid G=(V, E) \text{ gerichteter Graph} \\ \wedge \exists c=(v_{i_1}, \dots, v_{i_{|V|}}). c \text{ Hamiltonscher Kreis in } G \}$$

Directed Hamiltonian Circuit: Gibt es in einem gerichteten Graphen einen Hamiltonschen Kreis?

- Kreis, der jeden Knoten von G genau einmal berührt
- Beschrieben als Liste disjunkter Knoten $(v_{i_1}, \dots, v_{i_{|V|}})$ mit $(v_{i_j}, v_{i_{j+1}}) \in E$ und $(v_{i_{|V|}}, v_{i_1}) \in E$

1. Zeige $DHC \in \mathcal{NP}$:

- a) Rate Zyklus $(v_{i_1}, \dots, v_{i_n})$
- b) Prüfe ob alle $(v_{i_j}, v_{i_{j+1}})$ und (v_{i_n}, v_{i_1}) Kanten aus E sind
- c) Anzahl der Schritte ist maximal $n * |E| \in \mathcal{O}(n^3)$

2. Zeige $3SAT \leq_p DHC$

↪ Anhang (aufwendig)

- Konstruiere Graphen, der Variablen und Klauseln codiert

HAMILTONSCHER KREIS (HAMILTONIAN CIRCUIT)

$HC = \{ G \mid G = (V, E) \text{ ungerichteter Graph} \wedge G \text{ hat Hamiltonschen Kreis} \}$

Hamiltonian Circuit: Gibt es in einem ungerichteten Graphen einen Hamiltonschen Kreis?

– Liste disjunkter Knoten $(v_{i_1}, \dots, v_{i_{|V|}})$ mit $\{v_{i_j}, v_{i_{j+1}}\} \in E$ und $\{v_{i_{|V|}}, v_{i_1}\} \in E$

1. Zeige $HC \in \mathcal{NP}$:

(Beweis wie bei DHC)

2. Zeige $DHC \leq_p HC$

– Gegeben gerichteter Graph $G_d = (V_d, E_d)$

e) Konstruiere ungerichteten Graphen $f(G_d) \equiv G = (V, E)$

mit $v^{ein}, v, v^{aus} \in V$ für $v \in V_d$

und $\{v^{ein}, v\}, \{v, v^{aus}\} \in E$ für $v \in V_d$ und $\{v_i^{aus}, v_j^{ein}\} \in E$ für $(v_i, v_j) \in E_d$

f) Dann gilt $G_d \in DHC \Leftrightarrow f(G_d) \in HC$

(Beweis folgt)

g) f ist in polynomieller Zeit berechenbar

KORREKTHEIT: $G_d \in DHC \Leftrightarrow f(G_d) \in HC$

$v^{ein}, v, v^{aus} \in V$ für $v \in V_d$, $\{v^{ein}, v\}, \{v, v^{aus}\} \in E$ für $v \in V_d$, $\{v_i^{aus}, v_j^{ein}\} \in E$ für $(v_i, v_j) \in E_d$

Es sei $G_d \in DHC$.

- Dann gibt es einen DHC $(v_{i_1}, \dots, v_{i_n})$ in G_d
- In diesem Fall ist $(v_{i_1}^{ein}, v_{i_1}, v_{i_1}^{aus}, \dots, v_{i_n}^{aus})$ ein Hamiltonscher Kreis in G

Also $f(G_d) \in HC$

Es sei $f(G_d) \in HC$. Dann gibt es einen Hamiltonkreis $(u_{j_1}, \dots, u_{j_{3n}})$ in G

- Da jeder Knoten v_i nur v_i^{ein} und v_i^{aus} als Nachbarn hat und sonst nur Kanten zwischen v_i^{aus} und einem v_j^{ein} verlaufen muß der Kreis die Gestalt $(v_{i_1}^{ein}, v_{i_1}, v_{i_1}^{aus}, \dots, v_{i_n}^{ein}, v_{i_n}, v_{i_n}^{aus})$ oder $(v_{i_1}^{aus}, v_{i_1}, v_{i_1}^{ein}, \dots, v_{i_n}^{aus}, v_{i_n}, v_{i_n}^{ein})$ haben
- Da G ungerichtet ist, sind beide Kreise identisch, und wir können einen gerichteten Hamiltonschen Kreis $(v_{i_1}, \dots, v_{i_n})$ für G_d extrahieren

Also gilt $G_d \in DHC$

DAS TRAVELLING SALESMAN PROBLEM IST \mathcal{NP} -VOLLSTÄNDIG

$$TSP = \{ (c_{1,2}, \dots, c_{n-1,n}), B \mid \exists \pi: \text{perm}(\{1..n\}) . \sum_{i=1}^{n-1} c_{\pi(i), \pi(i+1)} + c_{\pi(n), \pi(1)} \leq B \}$$

Gegeben n Städte, Reisekostentabelle $c_{i,j}$, Kostenlimit B

Gibt es eine Rundreise durch alle Städte mit Gesamtkosten B ?

1. Zeige $TSP \in \mathcal{NP}$:

a) Rate eine Rundreise $\pi: \{1..n\} \rightarrow \{1..n\}$ *darstellbar als Liste $(\pi(1).. \pi(n))$*

b/c) Prüfe $\sum_{i=1}^{n-1} c_{\pi(i), \pi(i+1)} + c_{\pi(n), \pi(1)} \leq B$ *maximal n Schritte*

2. Zeige $HC \leq_p TSP$:

e) Es ist $(v_{i_1}, \dots, v_{i_n})$ ein Hamiltonscher Kreis in $G = (V, E)$ mit $n = |V|$

$\Leftrightarrow (i_1, \dots, i_n)$ Rundreise durch n Städte mit Kosten n ,
wobei $c_{ij} = 1$ genau dann, wenn $\{v_i, v_j\} \in E$ (sonst größer)

$\Leftrightarrow (i_1, \dots, i_n)$ Lösung des entsprechenden TSP $((c_{1,2}, \dots, c_{n-1,n}), n)$

Setze $f(G) := ((c_{1,2}, \dots, c_{n-1,n}), |V|)$ mit $c_{i,j} = \begin{cases} 1 & \text{falls } \{v_i, v_j\} \in E \\ 2 & \text{sonst} \end{cases}$

f) Es folgt $G \in HC \Leftrightarrow f(G) \in TSP$

g) f ist in polynomieller Zeit $\mathcal{O}(|V|^2)$ berechenbar

DAS FÄRBBARKEITSPROBLEM (GRAPH COLORING)

$$GC = \{ (G, k) \mid G=(V, E) \text{ Graph} \\ \wedge \exists f_V: V \rightarrow \{1..k\}. \forall \{u, v\} \in E. f_V(u) \neq f_V(v) \}$$

Gegeben ein Graph $G = (V, E)$ und eine Zahl $k \leq |n|$

Gibt es eine Färbung von V mit k verschiedenen Farben, so daß verbundene Knoten verschiedene Farben haben?

1. Zeige $GC \in \mathcal{NP}$:

- Rate Färbung $f_V: V \rightarrow \{1..k\}$ des Graphen
- Prüfe $\forall \{u, v\} \in E. f_V(u) \neq f_V(v)$
- Anzahl der Schritte ist maximal $|E| \in \mathcal{O}(|V|^2)$

2. Zeige $3SAT \leq_p GC$:

- Konstruiere Färbungsproblem aus einer Klauselmeng (Details folgen)

DAS FÄRBBARKEITSPROBLEMS IST \mathcal{NP} -VOLLSTÄNDIG

2. Zeige $3SAT \leq_p GC$:

– Gegeben $F = (k_1, \dots, k_m)$ mit $k_i = z_{i1} \vee z_{i2} \vee z_{i3}$ und $z_{ij} \in \{x_1, \dots, \overline{x_n}\}$

e) Konstruiere Färbungsproblem $f(F) \equiv (G, 3)$ wie folgt

- Teilgraph für **Codierung der Variablenbelegung**

Wähle $V_{var} = \{u, x_1, \dots, \overline{x_n}\}$

und $E_{var} = \{\{u, x_1\}, \{u, \overline{x_1}\}, \{x_1, \overline{x_1}\}, \dots, \{u, x_n\}, \{u, \overline{x_n}\}, \{x_n, \overline{x_n}\}\}$

Bei 3-Färbbarkeit erhalten x_i und $\overline{x_i}$ verschiedene Farben aus 0 oder 1

- Teilgraph für **Codierung der Klauseln**

Wähle $V_k = \{v, a_1, b_1, c_1, y_1, z_1, \dots, a_m, b_m, c_m, y_m, z_m\}$

und $E_k = \{\{v, y_1\}, \{v, z_1\}, \{a_1, y_1\}, \{a_1, z_1\}, \{b_1, y_1\}, \{c_1, z_1\}, \{b_1, c_1\}, \dots, \{v, y_m\}, \dots, \{b_m, c_m\}, \{u, v\}\}$

Knoten v erhält Farbe 0 oder 1

- Verbindungskanten zur **Codierung der Klauselliterale**

$E_{lit} = \{\{a_1, z_{11}\}, \{b_1, z_{12}\}, \{c_1, z_{13}\}, \dots, \{a_m, z_{m1}\}, \{b_m, z_{m2}\}, \{c_m, z_{m3}\}\}$

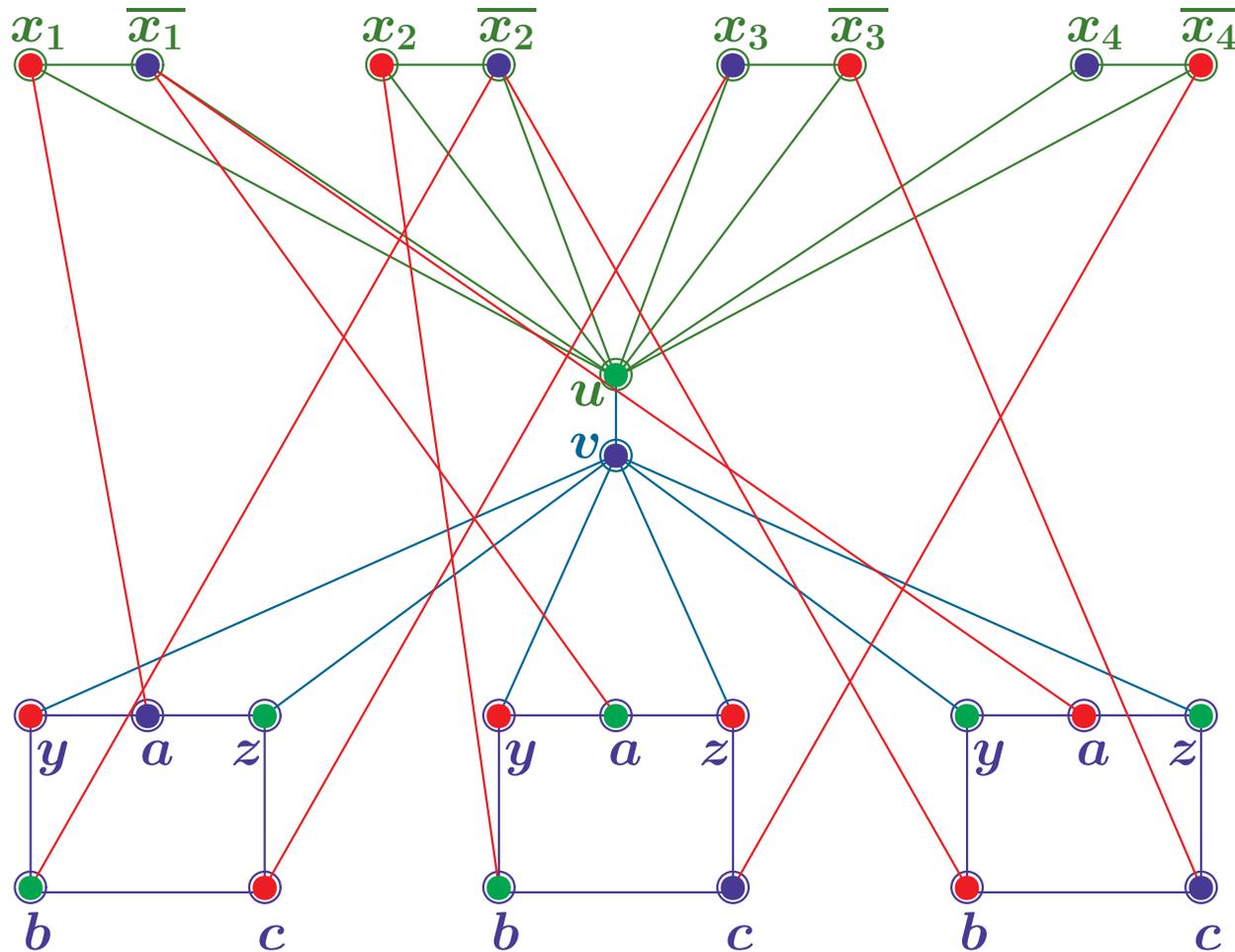
g) $G := (V_{var} \cup V_k, E_{var} \cup E_k \cup E_{lit})$ ist in polynomieller Zeit berechenbar

f) Dann gilt $F \in 3SAT \Leftrightarrow f(F) \in GC$

(Beweis folgt)

CODIERUNG EINER FORMEL ALS FÄRBUNGSPROBLEM

$$F = (k_1, k_2, k_3) \text{ mit } k_1 = x_1 \vee \overline{x_2} \vee x_3 \quad k_2 = \overline{x_1} \vee x_2 \vee \overline{x_4} \quad k_3 = \overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$$



Gibt es für den Graphen eine 3-Färbung?

KORREKTHEIT: $F \in 3SAT \Leftrightarrow f(F) \in GC$

- Sei $F \in 3SAT$

Dann gibt es eine erfüllende Belegung der x_j

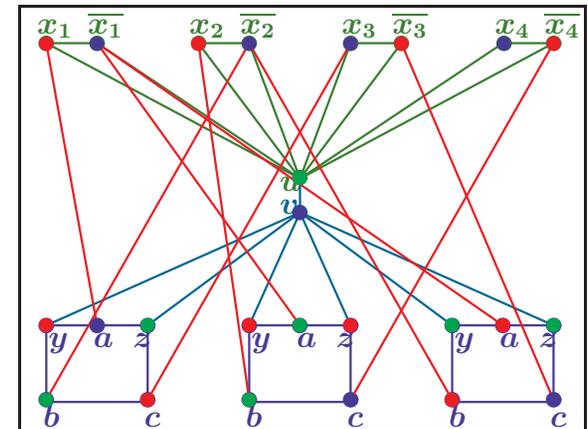
- Wähle $f_v(x_i), f_v(\overline{x_i}) \in \{0,1\}$ entsprechend,
 $f_v(u)=2$ und $f_v(v)=0$.

- Da jedes k_i erfüllbar ist,

kann eines der a_i, b_i, c_i die Farbe 0 erhalten

- Die anderen 4 Knoten bilden eine Kette und werden abwechselnd gefärbt

- Also gibt es eine 3-Färbung des Graphen und somit $f(F) \in GC$



- Ist $f(F) \in GC$ dann ist o.B.d.A. $f_v(u)=2$ und $f_v(v)=0$

- Wähle Belegung der x_i entsprechend der Färbung von x_i

- Wäre Klausel k_i nicht erfüllt, so wäre die Farbe der a_i, b_i, c_i 1 oder 2

- Wegen $f_v(b_i) \neq f_v(c_i)$ und $f_v(v)=0$ wäre dann $f_v(y_i) \neq f_v(z_i) \in \{1,2\}$

- Dies widerspricht der Färbbarkeit, da a_i ebenfalls mit 1 oder 2 gefärbt ist.

- Also $F \in 3SAT$



$$3SAT \leq_p GC$$

DAS RUCKSACKPROBLEM (KNAPSACK)

$$KP = \{(g_1..g_n, a_1..a_n, G, A) \mid \exists J \subseteq \{1..n\}. \sum_{i \in J} g_i \leq G \wedge \sum_{i \in J} a_i \geq A\}$$

Gegeben n Objekte mit Gewichten g_i und Nutzenwerten a_i
ein Rucksack mit Gewichtslimit G , Minimalnutzwert A

Gibt es eine Bepackung des Rucksacks mit Gegenständen,
die mindestens Nutzen A und maximal Gewicht G haben?

1. Zeige $KP \in \mathcal{NP}$:

- Rate Menge von Gegenständen $J \subseteq \{1..n\}$
- Prüfe $\sum_{i \in J} g_i \leq G$ und $\sum_{i \in J} a_i \geq A$
- Anzahl der Schritte ist maximal $2|J| \in \mathcal{O}(n)$

2. Zeige $3SAT \leq_p KP$:

(Details folgen)

- Codiere Anzahl der Vorkommen von Literalen in den Klauseln
in Dezimalstellen der Gewichte und Nutzenwerte

DAS RUCKSACKPROBLEM IST \mathcal{NP} -VOLLSTÄNDIG

2. Zeige $3SAT \leq_p KP$:

- Gegeben $F = (k_1, \dots, k_m)$ mit $k_i = z_{i1} \vee z_{i2} \vee z_{i3}$ und $z_{ij} \in \{x_1, \dots, \overline{x_n}\}$
- Codiere Anzahl der Vorkommen von Literalen in den Klauseln in Dezimalstellen der Gewichte und Nutzenwerte

e) Konstruiere Rucksackproblem $f(F) \equiv (g_1, \dots, g_{2m+2n}, a_1, \dots, a_{2m+2n}, G, A)$

wobei alle a_j und g_j $m + n$ -stellige Zahlen sind mit $g_j = a_j$ und

- a_j : Anzahl der x_j in k_i an Stelle $i \leq m$, 1 an Stelle $m+j$, sonst 0 ($j \leq n$)
- $a_{n+j} \equiv b_j$: Zahl der $\overline{x_j}$ in k_i an Stelle $i \leq m$, 1 an Stelle $m+j$, sonst 0 ($j \leq n$)
- $a_{2n+i} \equiv c_i$: 1 an Stelle i , sonst 0 ($i \leq m$)
- $a_{2n+m+i} \equiv d_i$: 2 an Stelle i , sonst 0 ($i \leq m$)

Setze $A \equiv G = \underbrace{4 \dots 4}_{m\text{-mal}} \underbrace{1 \dots 1}_{n\text{-mal}}$

f) Dann gilt $F \in 3SAT \Leftrightarrow f(F) \in KP$

(Beweis folgt)

g) f ist in polynomieller Zeit berechenbar

CODIERUNG EINER FORMEL ALS RUCKSACKPROBLEM

$$F = (k_1, k_2, k_3) \text{ mit } k_1 = x_1 \vee \overline{x_2} \vee x_3 \quad k_2 = \overline{x_1} \vee x_2 \vee \overline{x_4} \quad k_3 = \overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$$

$$A = 444\ 1111$$

$$a_1 = 100\ 1000 \quad b_1 = 011\ 1000 \quad c_1 = 100\ 0000 \quad d_1 = 200\ 0000$$

$$a_2 = 010\ 0100 \quad b_2 = 101\ 0100 \quad c_2 = 010\ 0000 \quad d_2 = 020\ 0000$$

$$a_3 = 100\ 0010 \quad b_3 = 001\ 0010 \quad c_3 = 001\ 0000 \quad d_3 = 002\ 0000$$

$$a_4 = 000\ 0001 \quad b_4 = 010\ 0001$$

$(1, 1, 0, 0)$ ist erfüllende Belegung

$$a_1 + a_2 + b_3 + b_4 + c_1 + c_3 + d_1 + d_2 + d_3 = A$$

KORREKTHEIT: $F \in 3SAT \Leftrightarrow f(F) \in KP$

- a_j : Anzahl der x_j in k_i an Stelle $i \leq m$, 1 an Stelle $m+j$, sonst 0 $(j \leq n)$
- b_j : Anzahl der \bar{x}_j in k_i an Stelle $i \leq m$, 1 an Stelle $m+j$, sonst 0 $(j \leq n)$
- c_i : 1 an Stelle i , sonst 0 d_i : 2 an Stelle i , sonst 0 $(i \leq m)$
- $g_j = a_j$ für alle j $A \equiv G = \underbrace{4 \dots 4}_{m\text{-mal}} \underbrace{1 \dots 1}_{n\text{-mal}}$

Ist $F \in 3SAT$, so gibt es eine erfüllende Belegung der x_j

– Für $j \leq n$ wähle a_j falls $x_j=1$ und b_j sonst

Dann haben in der Summe alle Stellen $m+j$ den Wert 1

und die Stellen $i \leq n$ einen Wert aus $\{1..3\}$, da jedes k_i erfüllt wird

– Ergänzung der Stellen $i \leq n$ mit c_i und d_i zu 4 liefert A und G

Also $f(F) \in KP$

Gilt $f(F) \in KP$, so gibt es eine Bepackung die genau den Wert A ergibt

– Diese enthält für Stelle $m+j$ entweder a_j (setze $x_j:=1$) oder b_j ($x_j:=0$)

– Wegen $c_i+d_i=3$ muß die Summe der a_j und b_j in der Bepackung an jeder Stelle $i \leq m$ mindestens den Wert 1 haben

– Also kommt in Klausel k_i mindestens ein Literal mit dem Wert 1 vor

Damit erfüllt die gewählte Belegung die Formel F , d.h. $F \in 3SAT$

ANMERKUNGEN ZUM BEWEIS $3SAT \leq_p KP$

- **Reduktion nutzt Spezialfall des Rucksackproblems**

- Gewichte g_i und Nutzen a_i sind jeweils gleich
- Gesamtnutzen G und Gesamtgewicht A sind gleich
- Wegen $\sum_{i \in J} g_i \leq G \wedge \sum_{i \in J} a_i \geq A$ gilt also $\sum_{i \in J} a_i = A$

$$KP^* = \{(a_1..a_n, A) \mid \exists J \subseteq \{1..n\}. \sum_{i \in J} a_i = A\}$$

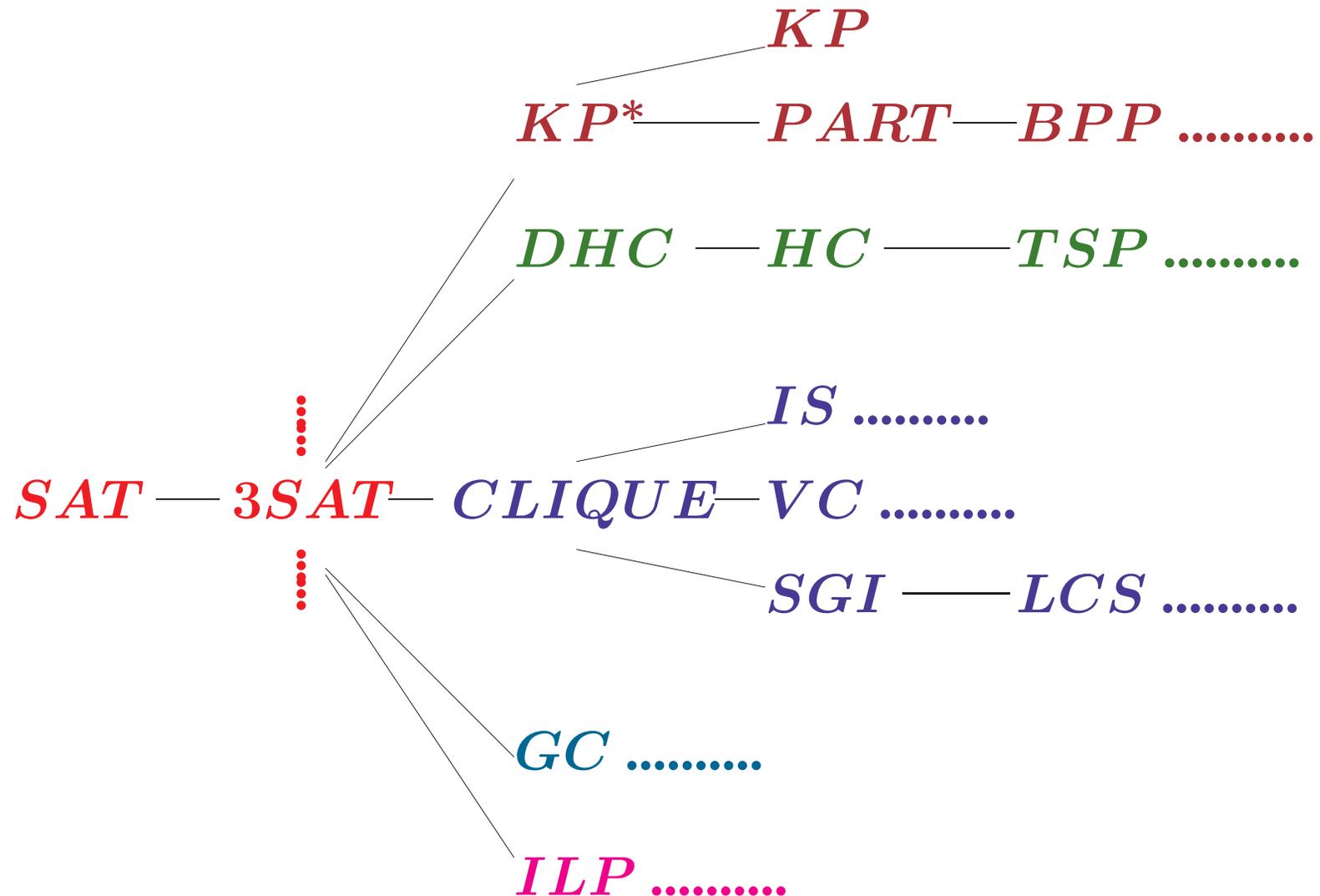
- KP^* ist Ausgangsproblem für viele Reduktionen

Manche Lehrbücher bezeichnen dies als “das Rucksackproblem”

- **Beweis zeigt eigentlich $3SAT \leq_p KP^* \leq_p KP$**

- Reduktion $3SAT \leq_p KP^*$ konstruiert die a_i und A wie gezeigt
- Reduktion $KP^* \leq_p KP$ konstruiert $g_i = a_i$ und $G = A$
- Damit sind beide Versionen des Rucksackproblems \mathcal{NP} -vollständig

\mathcal{NP} -VOLLSTÄNDIGE PROBLEME – REDUKTIONSTRUKTUR



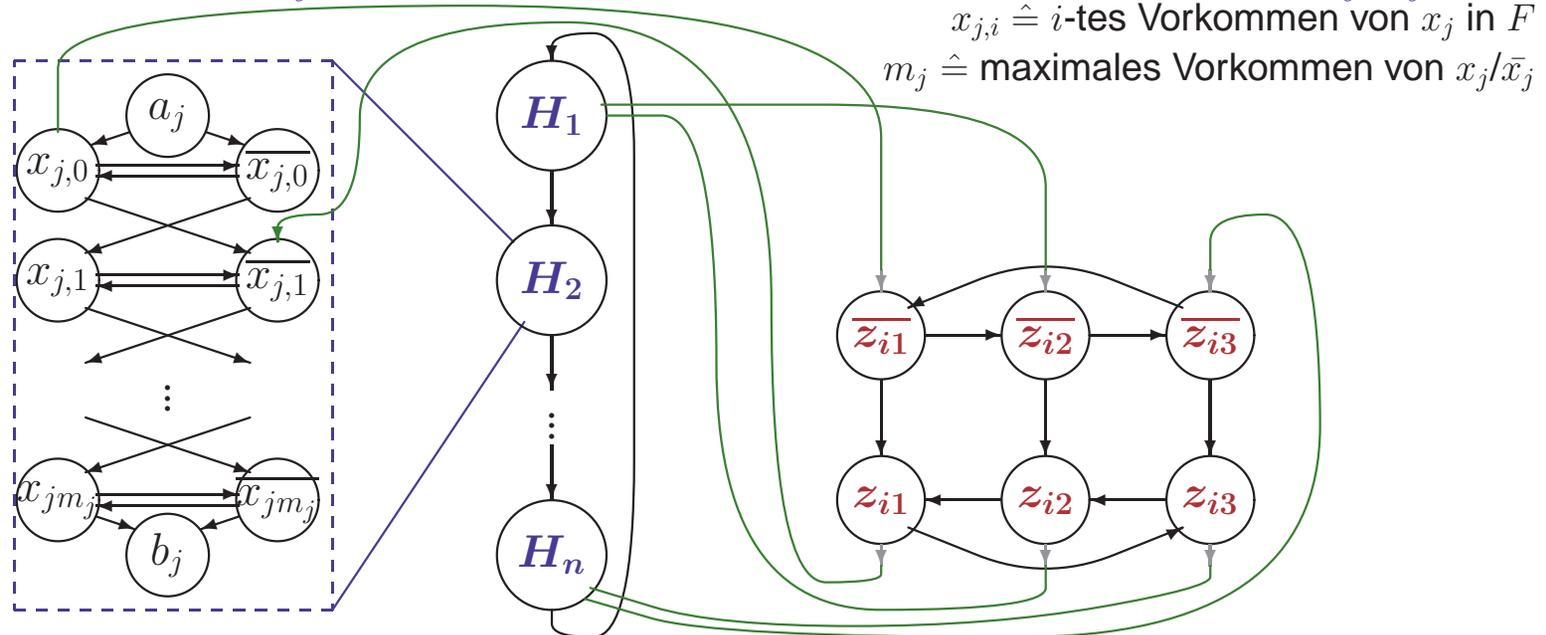
ANHANG

DHC IST \mathcal{NP} -VOLLSTÄNDIG: $3SAT \leq_p DHC$

Gegeben $F = (k_1, \dots, k_m)$ mit $k_i = z_{i1} \vee z_{i2} \vee z_{i3}$ und $z_{ij} \in \{x_1, \dots, \bar{x}_n\}$

e) **Konstruiere Graphen $G_F \equiv f(F)$ aus drei Komponenten**

- Teilgraphen H_j für Codierung der Variablenvorkommen von x_j/\bar{x}_j



- Teilgraph für **Codierung der Klauseln**
- Verbindungskanten **zwischen Variablen und Klauselliteralen**

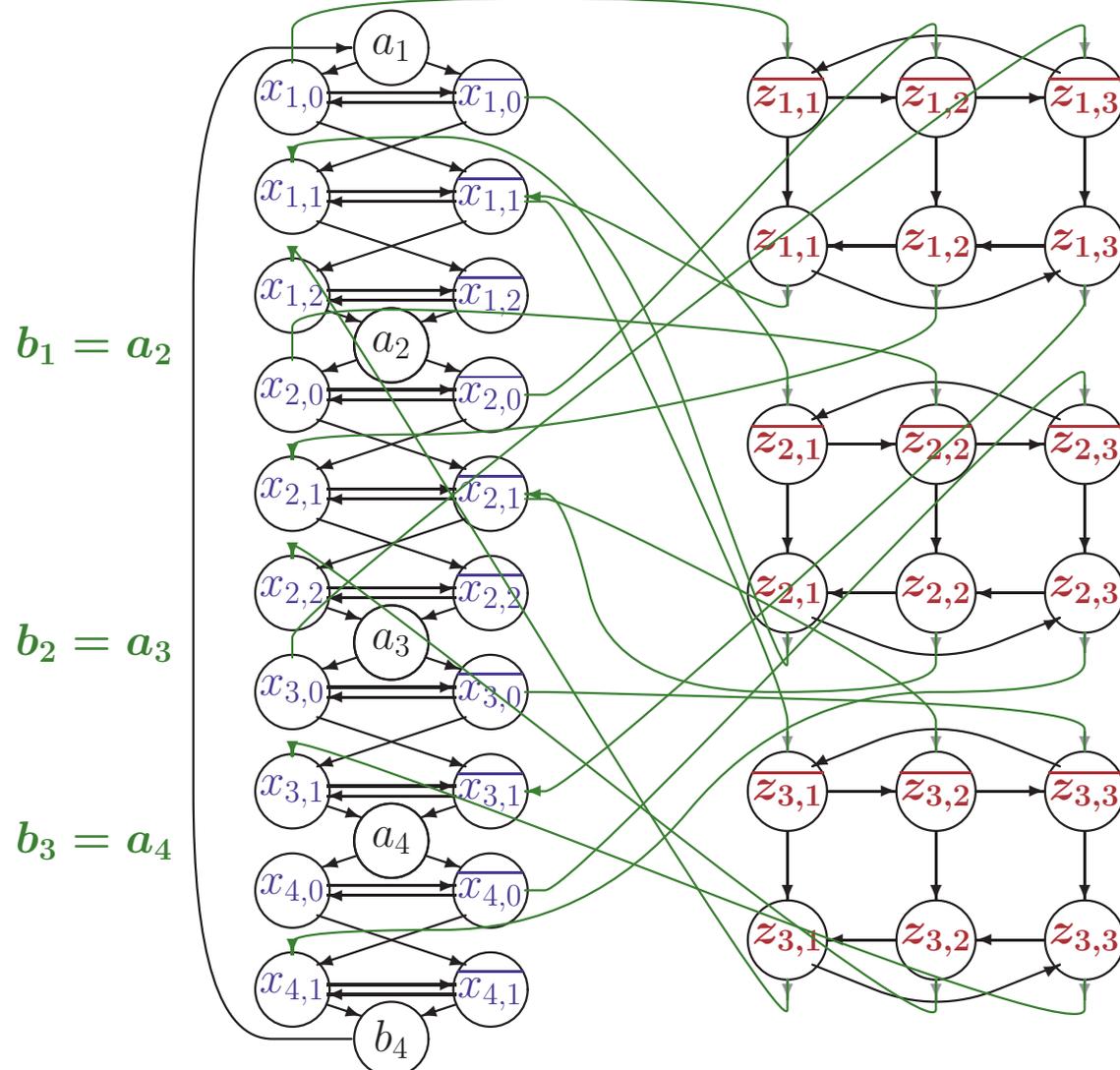
Für $z_{ik}=x_j$ verbinde erstes ungenutzte Vorkommen $x_{j,p}$ mit \bar{z}_{ik} und $z_{i,k}$ mit $\bar{x}_{j,p+1}$

Für $z_{ik}=\bar{x}_j$ verbinde erstes ungenutzte Vorkommen $\bar{x}_{j,p}$ mit \bar{z}_{ik} und z_{ik} mit $x_{j,p+1}$

g) **f ist in polynomieller Zeit berechenbar**

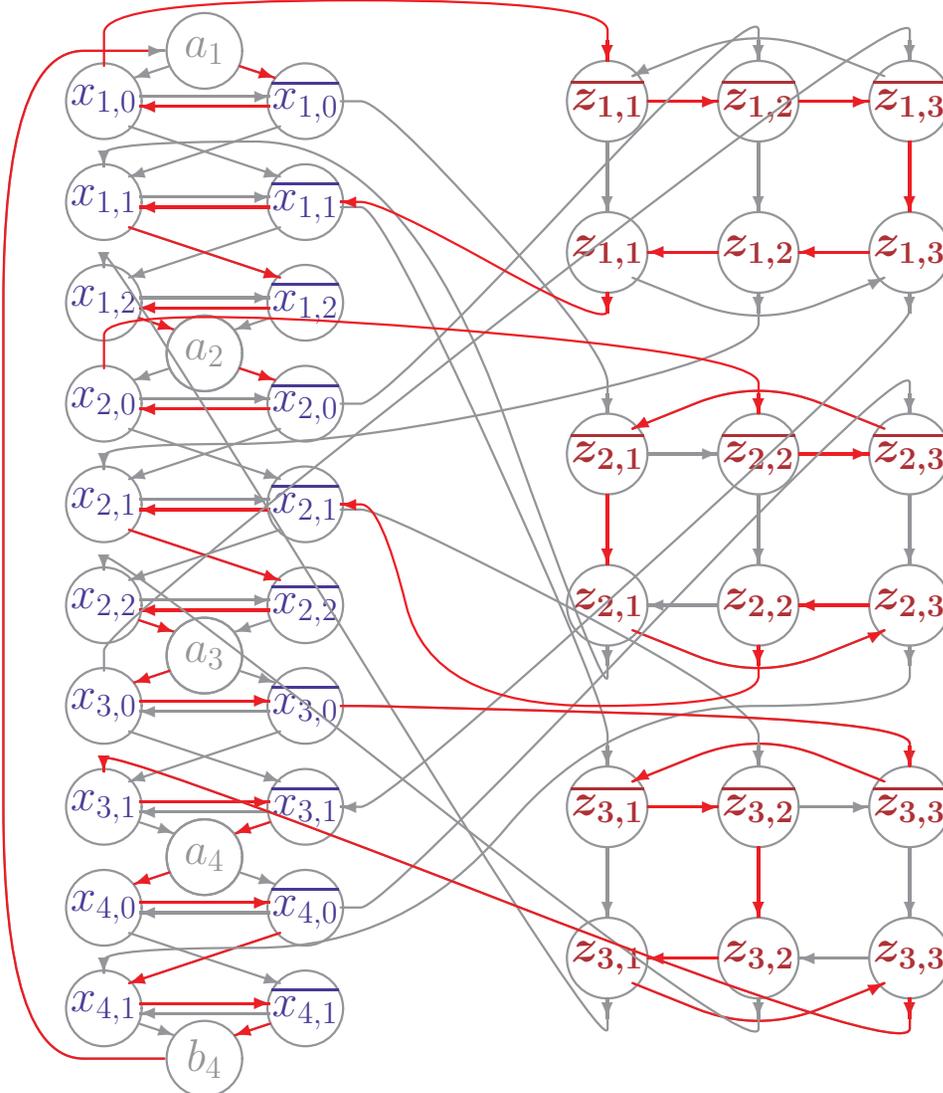
CODIERUNG EINER FORMEL ALS DHC PROBLEM

$$F = (k_1, k_2, k_3) \text{ mit } k_1 = x_1 \vee \overline{x_2} \vee x_3 \quad k_2 = \overline{x_1} \vee x_2 \vee \overline{x_4} \quad k_3 = \overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$$



ERFÜLLENDE BELEGUNG ALS DHC

$$F = (k_1, k_2, k_3) \text{ mit } k_1 = x_1 \vee \overline{x_2} \vee x_3 \quad k_2 = \overline{x_1} \vee x_2 \vee \overline{x_4} \quad k_3 = \overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$$



Erfüllende Belegung: (1, 1, 0, 0)

Beginne mit Teilpfad $a_1 \rightarrow \overline{x_{1,0}} \rightarrow x_{1,0}$

Verbinde $x_{1,0}$ mit Klausel 1

Laufe von $\overline{x_{1,1}}$ durch H_1 bis a_2

Weiter mit Teilpfad $a_2 \rightarrow \overline{x_{2,0}} \rightarrow x_{2,0}$

Verbinde $x_{2,0}$ mit Klausel 2

Laufe von $\overline{x_{2,1}}$ durch H_2 bis a_3

Weiter mit Teilpfad $a_3 \rightarrow x_{3,0} \rightarrow \overline{x_{3,0}}$

Verbinde $\overline{x_{3,0}}$ mit Klausel 3

Laufe von $x_{3,1}$ durch H_3 bis a_4

Beliebig weiter durch H_3 bis b_4 und a_1

KORREKTHEIT (SKIZZE): $F \in 3SAT \Leftrightarrow f(F) \in DHC$

- **Es gelte $F \in 3SAT$**

Sei c_1, \dots, c_n eine erfüllende Belegung von F .

Konstruiere einen Hamiltonschen Kreis wie folgt

- In H_j beginne mit a_j und $\overline{x_{j,0}}$ falls $c_j = 1$, sonst mit a_j und $x_{j,0}$
- Verbinde $\overline{x_{j,p}}$ mit $x_{j,p}$ und dann mit $\overline{x_{j,p+1}}$.

Wenn möglich, gehe dabei über die Knoten einer verbundenen Klausel z_i

- Analog verbinde $x_{j,p}$ mit $\overline{x_{j,p}}$ und dann mit $x_{j,p+1}$, evtl. mit Umweg
- Verlasse H_j in b_j und verbinde mit H_{j+1}

Da c_1, \dots, c_n die Formel F erfüllt, **wird jedes H_j und z_i durchlaufen.**

- **Es gelte $G_F \in DHC$**

Verbindet der Kreis a_j mit $x_{j,0}$ wähle $c_j = 0$, sonst $c_j = 1$

- Betritt der Kreis Klausel z_i bei $\overline{z_{i,k}}$, so muß er sie bei $z_{i,k}$ verlassen
- Damit verbindet der Kreis immer ein $\overline{x_{j,p}}$ mit $x_{j,p}$ mit $\overline{x_{j,p+1}}$ ($c_j = 1$) oder $x_{j,p}$ mit $\overline{x_{j,p}}$ mit $x_{j,p+1}$ ($c_j = 0$), bis er H_j verläßt.
- Bei Umweg über z_i muß das verbundene Literal $z_{ik} \in \{x_j, \overline{x_j}\}$ erfüllt sein
- Da alle Klauseln durchlaufen werden, **sind alle Klauseln erfüllt.**

WEITERE \mathcal{NP} -VOLLSTÄNDIGE GRAPHENPROBLEME

● Independent Set

(HMU §10.4.2) $3SAT \leq_p IS$

- Gegeben ein Graph $G = (V, E)$ der Größe n und eine Zahl $k \leq |V|$.
- Gibt es in G eine unabhängige Knotenmenge der Größe k ?

$$IS = \{ (G, k) \mid G = (V, E) \text{ Graph} \wedge \exists V_i \subseteq V. |V_i| \geq k \wedge \forall u, v \in V_i. \{u, v\} \notin E \}$$

● Subgraph Isomorphism

$CLIQUE \leq_p SGI$

- Gegeben zwei Graphen $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$.
- Gibt es einen Subgraphen H von G_1 , der isomorph zu G_2 ist?

$$SGI = \{ (G_1, G_2) \mid G_1, G_2 \text{ Graphen} \wedge \exists H \text{ Graph. } H \subseteq G_1 \wedge H \cong G_2 \}$$

● Largest Common Subgraph

$SGI \leq_p LCS$

- Gegeben Graphen $G_1 = (V_1, E_1)$ und $G_2 = (V_2, E_2)$ und eine Zahl $k \leq |G_1|$
- Gibt es isomorphe Subgraphen H_1 von G_1 und H_2 von G_2 der Größe k ?

$$LCS = \{ (G_1, G_2, k) \mid G_1, G_2 \text{ Graphen} \wedge k \leq |G_1| \wedge \exists H_1, H_2 \text{ Graphen.} \\ H_1 \subseteq G_1 \wedge H_2 \subseteq G_2 \wedge H_1 \cong H_2 \wedge |H_1| \geq k \}$$

WEITERE \mathcal{NP} -VOLLSTÄNDIGE PROBLEME

● **Partitionsproblem**

$KP \leq_p PART$

- Gegeben n Objekte mit Wert b_1, \dots, b_n .
- Gibt es eine Aufteilung der Objekte in zwei gleichwertige Stapel?

$$PART = \{ b_1, \dots, b_n \mid b_i \in \mathbb{N} \wedge \exists I \subseteq \{1..n\}. \sum_{i \in I} b_i = \sum_{i \in \bar{I}} b_i \}$$

● **Binpacking**

$PART \leq_p BPP$

- Gegeben n Objekte der Größe a_1, \dots, a_n und k Behälter der Größe b
- Kann man alle Objekte in den Behältern unterbringen?

$$BPP = \{ (a_1, \dots, a_n, b, k) \mid \exists f: \{1..n\} \rightarrow \{1..k\}. \forall j \leq k. \sum_{i \in \{i \mid f(i)=j\}} a_i \leq b \}$$

● **Multiprozessor-Scheduling**

$MPS \hat{=} BPP$

- Gegeben n Prozesse j_i mit Laufzeit $t(j_i)$, m Prozessoren, Deadline t_D .
- Gibt es eine Verteilung der Prozesse auf die Prozessoren, so daß bei Startzeit t_0 alle Prozesse vor der Zeit t_D beendet sind?

● **Integer Linear Programming**

$3SAT \leq_p ILP$

- Gegeben eine $k \times k$ Matrix A und einen Vektor $\vec{b} \in \mathbb{Z}^k$
- Gibt es ein $\vec{x} \in \mathbb{Z}^k$, welches das lineare Ungleichungssystem $A * \vec{x} \geq \vec{b}$ löst?