Gliederung

Polynomzeitreduktion I

Vielleicht das wichtigste Konzept der Komplexitätstheorie!

Σ^* =

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt **reduzierbar auf** eine Sprache $B \subseteq \Pi^*$ (**in Zeichen** $A \leq B$), wenn es eine totale, berechenbare Funktion $f : \Sigma^* \to \Pi^*$ gibt, sodass für alle $x \in \Sigma^*$ gilt

$$\boxed{x \in A \Leftrightarrow f(x) \in B.}$$

„Reduktionseigenschaft"

Wir nennen f eine **Reduktion** von A auf B

(**Beachte**: f muss weder surjektiv noch injektiv sein).

Polynomzeitreduktion I

$\leq_m^p$ → „many one"

Vielleicht das wichtigste Konzept der Komplexitätstheorie!

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt **polynomiell reduzierbar auf** eine Sprache $B \subseteq \Pi^*$ (**in Zeichen** $A \leq_m^p B$), wenn es eine totale, in Polynomzeit berechenbare Funktion $f\colon \Sigma^* \to \Pi^*$ gibt, sodass für alle $x \in \Sigma^*$ gilt

$$\boxed{x \in A \Leftrightarrow f(x) \in B.}$$

Wir nennen f eine **Polynomzeit-Reduktion** von A auf B
(**Beachte**: f muss weder surjektiv noch injektiv sein).

Polynomzeitreduktion I

Vielleicht das wichtigste Konzept der Komplexitätstheorie!

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt **polynomiell reduzierbar auf** eine Sprache $B \subseteq \Pi^*$ (**in Zeichen** $A \leq_m^p B$), wenn es eine totale, in Polynomzeit berechenbare Funktion $f \colon \Sigma^* \to \Pi^*$ gibt, sodass für alle $x \in \Sigma^*$ gilt

$$x \in A \Leftrightarrow f(x) \in B.$$

Wir nennen f eine **Polynomzeit-Reduktion** von A auf B

(**Beachte**: f muss weder surjektiv noch injektiv sein).

Bemerkung: „m“ in $\leq_m^p$ steht für „many-one-Reduktion”.

Polynomzeitreduktion I

Vielleicht das wichtigste Konzept der Komplexitätstheorie!

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt **polynomiell reduzierbar auf** eine Sprache $B \subseteq \Pi^*$ (**in Zeichen** $A \leq_m^p B$), wenn es eine totale, in Polynomzeit berechenbare Funktion $f : \Sigma^* \to \Pi^*$ gibt, sodass für alle $x \in \Sigma^*$ gilt

$$x \in A \Leftrightarrow f(x) \in B.$$

Wir nennen f eine **Polynomzeit-Reduktion** von A auf B

(**Beachte**: f muss weder surjektiv noch injektiv sein).

Bemerkung: „m“ in $\leq_m^p$ steht für „many-one-Reduktion”.

Mitteilungen:

(a) $\underline{A \leq_m^p B} \quad \Rightarrow \quad \underline{A \leq B}$

Polynomzeitreduktion I

Vielleicht das wichtigste Konzept der Komplexitätstheorie!

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt **polynomiell reduzierbar auf** eine Sprache $B \subseteq \Pi^*$ (**in Zeichen** $A \leq_m^p B$), wenn es eine totale, in Polynomzeit berechenbare Funktion $f : \Sigma^* \to \Pi^*$ gibt, sodass für alle $x \in \Sigma^*$ gilt

$$x \in A \Leftrightarrow f(x) \in B.$$

Wir nennen f eine **Polynomzeit-Reduktion** von A auf B (**Beachte**: f muss weder surjektiv noch injektiv sein).

$$\underbrace{f_{BC}(\underbrace{f_{AB}(x)}_{\leq \text{poly}(x)})}_{\text{poly}(\text{poly}(x))} \quad \text{poly}(\text{poly}(x)) \to \text{poly}(x)$$

Bemerkung: „m“ in $\leq_m^p$ steht für „many-one-Reduktion”.

Mitteilungen:

(a) $A \leq_m^p B \quad \Rightarrow \quad A \leq B$

(b) $\leq_m^p$ ist transitiv, d.h. wenn $\overset{f_{AB}}{A \leq_m^p B}$ und $\overset{f_{BC}}{B \leq_m^p C}$, dann auch $\overset{f_{BC} \circ f_{AB}}{A \leq_m^p C}$
(Konkatenation der Reduktionen ist Polynomzeitreduktion von A auf C)

$$x \in A \Leftrightarrow f_{AB}(x) \in B \Leftrightarrow f_{BC}(f_{AB}(x)) \in C$$

y

Polynomzeitreduktion II

Lemma

Gilt $A \leq B$ und ist B (semi-)entscheidbar, so ist auch A (semi-)entscheidbar.

Polynomzeitreduktion II

„leicht"

Lemma

Gilt $A \leq_m^p B$ und ist $B \in \mathrm{P}$ (bzw. $B \in \mathrm{NP}$), so ist auch $A \in \mathrm{P}$ (bzw. $A \in \mathrm{NP}$) .

Polynomzeitreduktion II

Lemma

Gilt $A \leq_m^p B$ und ist $B \in \mathrm{P}$ (bzw. $B \in \mathrm{NP}$), so ist auch $A \in \mathrm{P}$ (bzw. $A \in \mathrm{NP}$) .

1 2

Beweis

1. $A \leq_m^p B \rightsquigarrow$ „Reduktionsfunktion“ f in $p(n)$ Schritten berechenbar durch TM M_f

Polynomzeitreduktion II

Lemma

Gilt $A \leq_m^p B$ und ist $B \in \mathrm{P}$ (bzw. $B \in \mathrm{NP}$), so ist auch $A \in \mathrm{P}$ (bzw. $A \in \mathrm{NP}$) .

Beweis

1. $A \leq_m^p B \leadsto$ „Reduktionsfunktion“ f in $p(n)$ Schritten berechenbar durch TM M_f
2. $B \in \mathrm{P}$ (bzw. $B \in \mathrm{NP}$) $\leadsto B$ in $q(n)$ Schritten entscheidbar durch TM M_B

(wobei p und q Polynome)

Polynomzeitreduktion II

Lemma

Gilt $A \leq_m^p B$ und ist $B \in \mathrm{P}$ (bzw. $B \in \mathrm{NP}$), so ist auch $A \in \mathrm{P}$ (bzw. $A \in \mathrm{NP}$) .

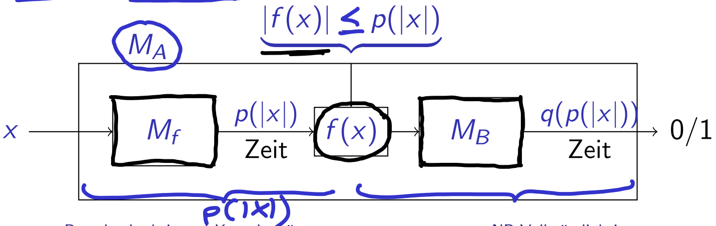
Beweis

1. $A \leq_m^p B \rightsquigarrow$ „Reduktionsfunktion“ f in $p(n)$ Schritten berechenbar durch TM M_f
2. $B \in \mathrm{P}$ (bzw. $B \in \mathrm{NP}$) $\rightsquigarrow$ B in $q(n)$ Schritten entscheidbar durch TM M_B

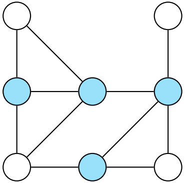
(wobei p und q Polynome)

Wie zuvor gilt $\chi_A = \chi_B \circ f$

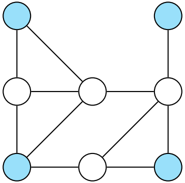
$\rightsquigarrow$ χ_A berechnet in $p(|x|) + q(p(|x|))$ (also polynomiell viele) Schritten.



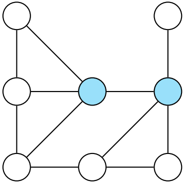
Independent Set, Vertex Cover und Dominating Set



Vertex Cover



Independent Set

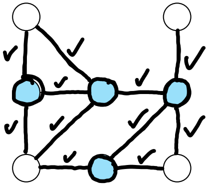


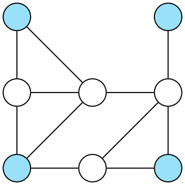
Dominating Set

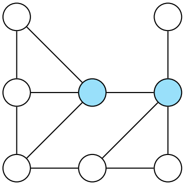
Eingabe: ungerichteter Graph G, Zahl $k > 0$ $\langle G, k \rangle$

Independent Set, Vertex Cover und Dominating Set

$k=4$



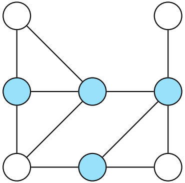




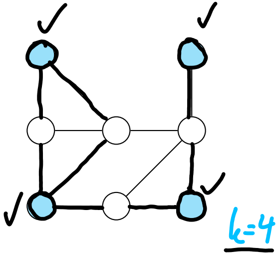
Eingabe: ungerichteter Graph G, Zahl $k > 0$
Frage: gibt es k Knoten in G, sodass ...

Vertex Cover: ... jede Kante in G mindestens einen dieser k Knoten als Endpunkt hat?

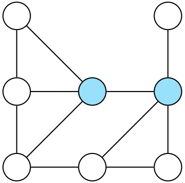
Independent Set, Vertex Cover und Dominating Set



Vertex Cover



Independent Set



Dominating Set

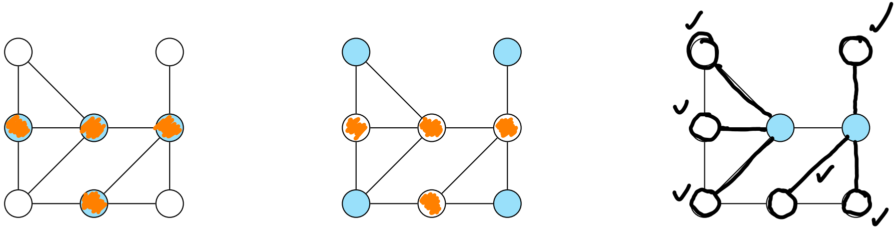
Eingabe: ungerichteter Graph G, Zahl $k > 0$

Frage: gibt es k Knoten in G, sodass ...

Vertex Cover: ...jede Kante in G mindestens einen dieser k Knoten als Endpunkt hat?

Independent Set: ...keine 2 dieser k Knoten mit einer Kante verbunden sind?

Independent Set, Vertex Cover und Dominating Set



$k=2$

Vertex Cover $\leq^p_m$ Independent Set

Dominating Set

Eingabe: ungerichteter Graph G, Zahl $k > 0$
Frage: gibt es k Knoten in G, sodass ...

Vertex Cover: ... jede Kante in G mindestens einen dieser k Knoten als Endpunkt hat?

Independent Set: ... keine 2 dieser k Knoten mit einer Kante verbunden sind?

Dominating Set: ... jeder andere Knoten eine Kante zu mindestens einem dieser Knoten hat?

Vertex Cover und Independent Set

Theorem

Vertex Cover $\leq_m^p$ Independent Set.

Vertex Cover und Independent Set

Theorem

Vertex Cover $\leq_m^p$ Independent Set.

$V(G)$ = Knotenmenge von G

Beweis

Definiere Reduktionsfunktion f vermöge $f(\langle G, k\rangle) := \langle G, |V(G)| - k\rangle$.
(offensichtlich ist f in polynomieller Zeit berechenbar)

Vertex Cover und Independent Set

$\neg\neg \forall_{e} \; \exists_{\text{EP } u \text{ v. } e} \; u \in X$

$\neg \exists_{e} \; \forall_{\text{EP } u \text{ v. } e} \; u \notin X$

Theorem

Vertex Cover $\leq_m^p$ Independent Set.

Beweis

Definiere Reduktionsfunktion f vermöge $f(\langle G, k\rangle) := \langle G, |V(G)| - k\rangle$.
(offensichtlich ist f in polynomieller Zeit berechenbar)
Dann gilt:

$$\begin{aligned}
\langle G, k\rangle \in \text{Vertex Cover} &\overset{\text{df}}{\Leftrightarrow} G \text{ hat eine Knotenmenge } X \subseteq V(G) \text{ mit } |X| \leq k \text{, so dass} \\
&\quad\;\; \text{jede Kante mindestens einen Endpunkt in } X \text{ hat} \\
&\Leftrightarrow G \text{ hat eine Knotenmenge } X \subseteq V(G) \text{ mit } |X| \leq k \text{, so dass} \\
&\quad\;\; \text{keine Kante beide Endpunkte in } V(G) \setminus X \text{ hat} \\
&\Leftrightarrow \langle G, |V(G)| - k\rangle \in \text{Independent Set.}
\end{aligned}$$

NP-Vollständigkeit

Halteproblem?

Definition

Problem

Eine Sprache $A \subseteq \Sigma^*$ heißt...

a) ... **NP-schwer**, falls $\forall_{L \in \text{NP}}\ L \leq_m^p A$.

Frage: Können Sie zeigen, dass das Halteproblem NP-schwer ist? **

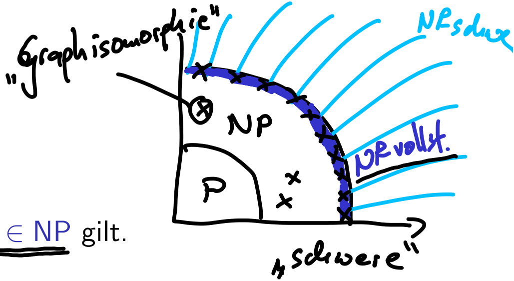
NP-Vollständigkeit

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt...

a) ... **NP-schwer**, falls $\forall_{L \in \mathrm{NP}}\ L \leq_m^p A$.

b) ... **NP-vollständig**, wenn A NP-schwer ist und $A \in \mathrm{NP}$ gilt.



NP-Vollständigkeit

SAT $\leq_m^p$ VC $\leq_m^p$ SAT

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt...

a) ... **NP-schwer**, falls $\forall_{L \in \mathrm{NP}}\ L \leq_m^p A$.

b) ... **NP-vollständig**, wenn A NP-schwer ist und $A \in \mathrm{NP}$ gilt.

Anschaulich: (mit „polynomieller Unschärfe")

1. NP-schwere Sprachen sind „mindestens so schwer" zu entscheiden wie jede Sprache in NP
2. NP-vollständige Sprachen sind „genau so schwer" wie jede NP-vollständige Sprache

NP-Vollständigkeit

Definition

Eine Sprache $A \subseteq \Sigma^*$ heißt...

a) ... **NP-schwer**, falls $\forall_{L \in \mathrm{NP}}\ L \leq_m^p A$.

b) ... **NP-vollständig**, wenn A NP-schwer ist und $A \in \mathrm{NP}$ gilt.

Anschaulich: (mit „polynomieller Unschärfe“)

1. NP-schwere Sprachen sind „mindestens so schwer“ zu entscheiden wie jede Sprache in NP
2. NP-vollständige Sprachen sind „genau so schwer“ wie jede NP-vollständige Sprache

Lemma

Ist A NP-schwer und $A \leq_m^p B$, so ist auch B NP-schwer

1 2

Beweis

Für jede Sprache $L \in \mathrm{NP}$ gilt $L \leq_m^p A \leq_m^p B$.
Somit gilt wegen Transitivität auch $L \leq_m^p B$. Also ist B auch NP-schwer.

NP-Vollständigkeit II

Theorem

Für jede NP-vollständige Sprache A gilt: $A \in \mathsf{P} \Leftrightarrow \mathsf{P} = \mathsf{NP}$.

NP-Vollständigkeit II

"$\leq_m^p$" $\equiv$ "leichter"

Theorem

Für jede NP-vollständige Sprache A gilt: $A \in \mathrm{P} \Leftrightarrow \mathrm{P} = \mathrm{NP}$.

1 2 3

$NP \supseteq P$

$NP \subseteq P$

Beweis

„$\Rightarrow$“: $(\forall_{L \in \mathrm{NP}}\ L \leq_m^p A) \wedge (A \in \mathrm{P}) \overset{\text{Lem}}{\Rightarrow} \forall_{L \in \mathrm{NP}}\ L \in \mathrm{P} \Rightarrow \boxed{\mathrm{NP} = \mathrm{P}}$

„$\Leftarrow$“: $(A \in \mathrm{NP}) \wedge (\mathrm{P} = \mathrm{NP}) \Rightarrow A \in \mathrm{P}$

1 3

NP-Vollständigkeit II

Theorem

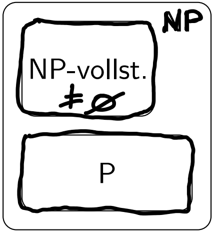
Für jede NP-vollständige Sprache A gilt: $A \in \mathrm{P} \Leftrightarrow \mathrm{P} = \mathrm{NP}$.

Beweis

„$\Rightarrow$“: $(\forall_{L \in \mathrm{NP}}\ L \leq_m^p A) \wedge (A \in \mathrm{P}) \;\Rightarrow\; \forall_{L \in \mathrm{NP}}\ L \in \mathrm{P} \;\Rightarrow\; \mathrm{NP} = \mathrm{P}$

„$\Leftarrow$“: $(A \in \mathrm{NP}) \wedge (\mathrm{P} = \mathrm{NP}) \;\Rightarrow\; A \in \mathrm{P}$

„**Geglaubte**“ (d.h. Annahme $\mathrm{P} \neq \mathrm{NP}$) **Situation:**



Erfüllbarkeitsproblem I

Cook & Levin

SAT

Eingabe: aussagenlogische Formel F

Frage: Ist F **erfüllbar**, d.h. gibt es eine $\{0,1\}$-wertige Belegung der in F verwendeten Booleschen Variablen derart, dass F zu **wahr** (d.h. 1) ausgewertet wird?

Erfüllbarkeitsproblem I

SAT

Eingabe: aussagenlogische Formel F

Frage: Ist F **erfüllbar**, d.h. gibt es eine $\{0,1\}$-wertige Belegung der in F verwendeten Booleschen Variablen derart, dass F zu **wahr** (d.h. 1) ausgewertet wird?

Beispiele

$$\underline{0}, \underline{1}, \qquad \underline{x_1}, \underline{x_2}, \underline{\overline{x_3}}, \qquad \underline{(x_1 \wedge \overline{x_2})}, \qquad \underline{(\overline{(x_1 \wedge \overline{x_2})} \vee x_2 \vee \overline{x_3})}$$

Erfüllbarkeitsproblem I

SAT

Eingabe: aussagenlogische Formel F

Frage: Ist F **erfüllbar**, d.h. gibt es eine $\{0,1\}$-wertige Belegung der in F verwendeten Booleschen Variablen derart, dass F zu **wahr** (d.h. 1) ausgewertet wird?

Beispiele

$0, 1,\qquad x_1, x_2, \overline{x_3},\qquad (x_1 \wedge \overline{x_2}),\qquad (\overline{(x_1 \wedge \overline{x_2})} \vee x_2 \vee \overline{x_3})$

Theorem (Satz von Cook und Levin)

SAT ist NP-vollständig.

Erfüllbarkeitsproblem I

SAT

Eingabe: aussagenlogische Formel F

Frage: Ist F **erfüllbar**, d.h. gibt es eine $\{0,1\}$-wertige Belegung der in F verwendeten Booleschen Variablen derart, dass F zu **wahr** (d.h. 1) ausgewertet wird?

Beispiele

$0, 1$, $x_1, x_2, \overline{x_3}$, $(x_1 \wedge \overline{x_2})$, $(\overline{(x_1 \wedge \overline{x_2})} \vee x_2 \vee \overline{x_3})$

Theorem (Satz von Cook und Levin)

SAT ist NP-vollständig.

guess & check

Beweis (Idee, Details später)

Teil 1: „SAT $\in$ NP“: rate erfüllende Belegung (Zertifikat) und verifiziere sie.

Teil 2: „SAT ist NP-schwer“: mit $L \in \mathrm{NP}$ beliebig,
transformiere NTM N mit $T(N) = L$ in Formel $\varphi(x)$ sodass $x \in L \Leftrightarrow \varphi(x) \in \mathrm{SAT}$.