$\boxed{\mathbf{P = NP?}}$

Ralf Schindler

Turingmaschinen

		$\triangledown$						
0	1	1	0	1	0	1	$\sqcup$	$\sqcup$ ...

- $Q =$ endliche Menge der *Zustände*
- $\Sigma =$ endliches *Alphabet*
- $\delta : Q \times \Sigma \to Q \times \Sigma \times \{L, R\}$
- $q_0 \in Q$, der *Anfangszustand*
- $q_+ \in Q$, der *positive Endzustand*
- $q_- \in Q$, der *negative Endzustand*

- *Anfangskonfiguration*: Kopf ganz links, *Eingabe* $\in \Sigma^* = \{f\colon n \to \Sigma \mid n \in \mathbb{N}\}$ auf dem Band
- *Rechenschritt* gegeben durch δ
- *Endkonfiguration*: wenn Zustand q_+ oder q_- erreicht ist

$\mathcal{L} \subset \Sigma^*$ heißt eine *Sprache*

$\mathcal{L}$ heißt *rekursiv aufzählbar* gdw. es eine Turingmaschine T gibt, so daß

- $\mathcal{L}$ die Menge aller Eingaben ist, die T akzeptiert

$\mathcal{L}$ heißt *rekursiv* (oder *entscheidbar*) gdw. es eine Turingmaschine T gibt, so daß

- $\mathcal{L}$ die Menge aller Eingaben ist, die T akzeptiert, und
- T bei allen Eingaben hält

$\mathcal{L} \subset \Sigma^*$ ist rekursiv gdw. $\mathcal{L}$ und $\Sigma^* \setminus \mathcal{L}$ beide rekursiv aufzählbar sind

- $\{(m,n) \mid m$ und n sind relativ prim $\}$ ist entscheidbar
- $\{p \in \mathbb{N} \mid p$ ist prim $\}$ ist entscheidbar
- die Menge aller aussagenlogischen Tautologien ist entscheidbar

- das Halteproblem ist rekursiv aufzählbar, aber nicht entscheidbar:

`Beweis:` Angenommen, H(#T, e) $\downarrow$ 1 , falls T Eingabe e akzeptiert, $\downarrow$ 0 sonst. Sei H′(#T) $\downarrow$ 1 gdw. H(#T, #T) $\downarrow$ 0. Dann H′(#H′) $\downarrow$ 1 gdw. H′(#H′) $\downarrow$ 0. Widerspruch!

- die Menge der Theoreme von PA ist rekursiv aufzählbar, aber nicht entscheidbar (Church 1930)
- die Menge der Polynome mit ganzzahligen Lösungen ist rekursiv aufzählbar, aber nicht entscheidbar (Matijasevič 1970)

Die Frage, ob $P = NP$, kann als “finite Version” des Entscheidungsproblems angesehen werden: gibt es eine schnelle Entscheidung, ob ein Satz einen kurzen Beweis hat?

- Euklidischer Algorithmus zeigt, daß “m und n sind relativ prim” schnell entscheidbar ist

Definition. $\mathcal{L} \subset \Sigma^*$ ist in **p**olynomieller Zeit entscheidbar ($\mathcal{L} \in P$) gdw. es eine Turingmaschine T und ein $m \in \mathbb{N}$ gibt, so daß
- $\mathcal{L}$ die Menge aller Eingaben ist, die T akzeptiert, und
- T bei allen Eingaben w nach $lh(w)^m$ vielen Schritten hält

Satz. $\{(m, n) \mid m$ und n sind relativ prim $\} \in P$

Frage. $\{p \in \mathbb{N} \mid p$ ist prim $\} \in P$?

Definition. $\mathcal{L} \subset \Sigma^*$ ist **n**ichtdeterministisch in **p**olynomieller Zeit entscheidbar gdw. es eine Turingmaschine T und ein $m \in \mathbb{N}$ gibt, so daß

- $\mathcal{L}$ die Menge aller w ist mit $\exists z\ T$ akzeptiert $w \oplus z$, und
- T bei allen Eingaben $w \oplus z$ nach $lh(w)^m$ vielen Schritten hält

Satz. $\{p \in \mathbb{N} \mid p$ ist prim $\} \in NP \cap coNP$

`Hinweis:` für "... $\in NP$" benutze: p ist prim gdw. $\mathbb{Z}_p^\star$ ist eine zyklische Gruppe der Ordung $p - 1$

$\boxed{\text{P=NP?}}$

Frage. Gilt $NP \subset P$?

Bedeutung (Beispiel):

Kodierung mittels “öffentlicher Schlüssel” (RSA-Algorithmus).

S will Nachricht n an E verschicken.
E schickt (pq, e) an S (e rel. prim zu $\phi(pq)$)
S schickt $c \equiv n^e (mod\ pq)$ an E
E dekodiert n: für $d \equiv e^{\phi(\phi(pq))-1} (mod\ \phi(pq))$ gilt $n \equiv c^d (mod\ pq)$
(Berechnung von d in polynomieller Zeit möglich!)

Boolesche Formel: Resultat der Verknüpfung von Variablen x, y, z, ... mittels logischer Junktoren, $\neg$, $\wedge$, $\vee$

Beispiele: $(\neg x \wedge z) \vee y$, $\neg z \vee z$

Belegung der Variablen durch 0 bzw. 1 liefert einen Formelwert gemäß der Regeln

- $x \wedge y = min\{x, y\}$
- $x \vee y = max\{x, y\}$
- $\neg x = 1 - x$

Eine Boolesche Formel Φ heißt *erfüllbar* gdw. es eine Variablenbelegung gibt, die den Formelwert 1 liefert

$SAT = \{\Phi \mid \Phi \text{ ist erfüllbar }\}$

Satz. $SAT \in NP$.

$\boxed{NP\text{-Vollständigkeit}}$

Definition. A heißt *in polynomieller Zeit auf* B *reduzierbar*, kurz $A \leq_P B$, gdw. eine Funktion $f\colon \Sigma^* \to \Sigma^*$ existiert mit

- $w \in A \Leftrightarrow f(w) \in B$, und
- f ist in polynomieller Zeit berechenbar.

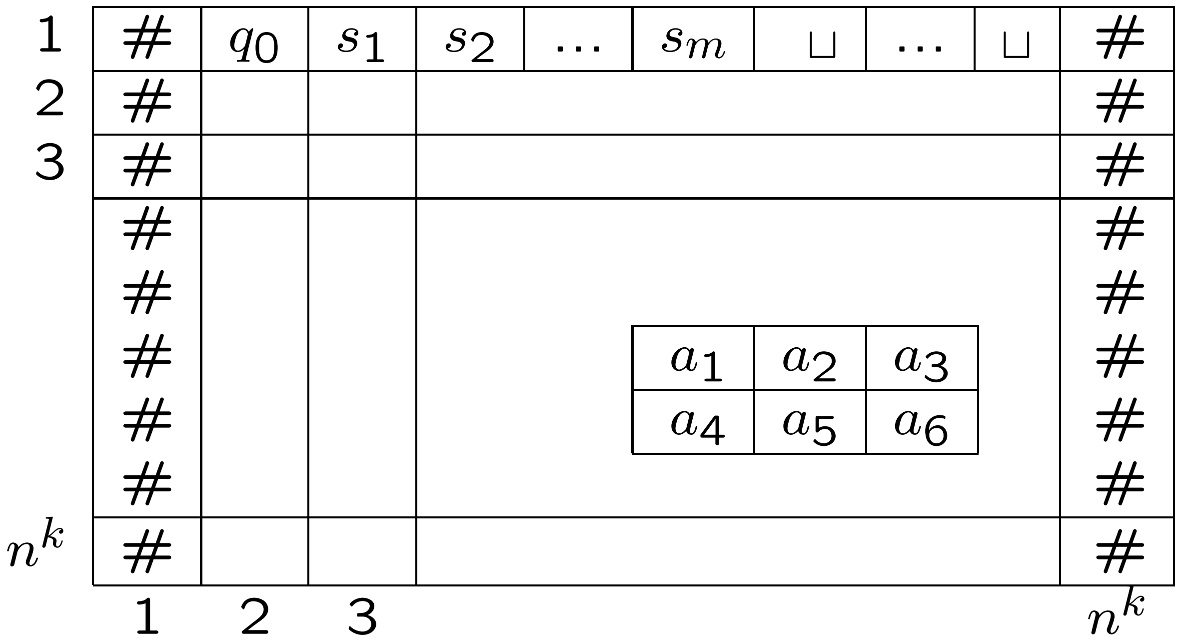
Definition. B heißt NP-*vollständig* gdw. $B \in NP$ und $A \leq_P B$ für alle $A \in NP$.

Satz (Cook, Levin 1971). SAT ist NP-vollständig.

Korollar. $P = NP$ gdw. $SAT \in P$.

`Beweis` des Satzes von Cook-Levin:

Formuliere Boolesche Formel, die erfüllbar ist gdw. Maschine Eingabe $s_1 s_2 ... s_m = s_1 s_2 ... s_n \oplus z$ für ein z akzeptiert



$x_{i,j,s} = 1$: Zelle (i, j) enthält das Symbol s

Jede Zelle enthält ein Symbol $s \in Q \cup \Sigma \cup \{\#\}$:

$$\bigwedge_{1 \le i,j \le n^k} [\bigvee_s x_{i,j,s} \wedge \bigwedge_{s \ne t} (\neg x_{i,j,s} \vee \neg x_{i,j,t})]$$

Startkonfiguration bei Eingabe $s_1 s_2 ... s_n \oplus z$:

$$x_{1,1,\#} \wedge x_{1,2,q_0} \wedge x_{1,3,s_1} \wedge ... \wedge x_{1,n+2,s_n} \wedge x_{1,n^k,\#}$$

Eingabe wird akzeptiert:

$$\bigvee_{1 \le i,j \le n^k} x_{i,j,q_+}$$

Schritt von einer Zeile zur nächsten gibt Rechenschritt wieder:

$$\bigwedge_{1 \le i,j \le n^k} \bigvee_{\substack{a_1,\ldots,a_6 \\ \text{ist legal}}} (x_{i-1,j,a_1} \wedge x_{i,j,a_2}$$

$$\wedge x_{i+1,j,a_3} \wedge x_{i-1,j+1,a_4} \wedge x_{i,j+1,a_5} \wedge x_{i+1,j+1,a_6})$$

Beobachtung: $SAT \leq_P B$ und $B \in NP \Rightarrow B$ ist NP-vollständig

Weitere Beispiele für NP-vollständige Probleme:

- $CLIQUE$
- $HAMPATH$
- $SUBSET\text{-}SUM = \{(x_1, ..., x_n, y) \mid \exists X \subset \{1, ..., n\} \sum_{i \in X} x_i = y\}$
- $TRAVELING\ SALESMAN$

Offen: ist $\{x \mid \exists n \exists m\ x = n \cdot m\}$ NP-vollständig?

Satz (Miller 1976). Die Erweiterte Riemann-Hypothese beweist, daß $\{x \mid \exists n \exists m\ x = n \cdot m\} \in P$.

$\boxed{\text{Lösungsansätze}}$

1. <u>Diagonalisierung.</u>

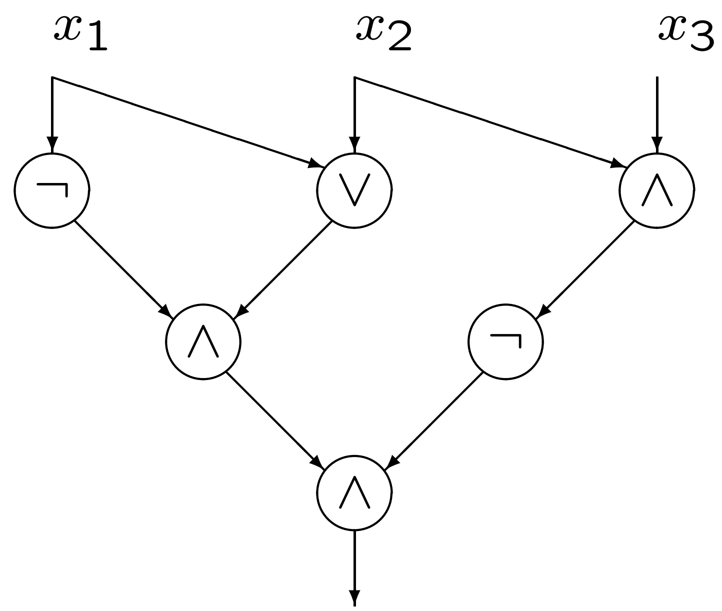
Orakel: $\mathcal{L} \subset \Sigma^*$

Eine *Orakel-Turingmaschine* darf während der Berechnung Anfragen an das Orakel stellen.

Satz (Baker, Gill, Solovay 1975). Es existiert ein Orakel $\mathcal{L}$ mit $P^{\mathcal{L}} \neq NP^{\mathcal{L}}$.

Satz (Baker, Gill, Solovay 1975). Es existiert ein Orakel $\mathcal{L}$ mit $P^{\mathcal{L}} = NP^{\mathcal{L}}$.

2. Boolesche Schaltkreise.



Definition. Eine *Familie von Schaltkreisen* ist eine Liste $\mathcal{C} = (C_0, C_1, C_2, ...)$. $\mathcal{C}$ entscheidet $\mathcal{L}$ gdw. für alle w,

$$w \in \mathcal{L} \Leftrightarrow C_{lh(w)}(w) = 1.$$

Definition. Die Größe eines Schaltkreises ist die Anzahl seiner Knotenpunkte.

Satz (Savage 1972). Jedes Problem $\mathcal{L} \in P$ hat eine Familie von Schaltkreisen von polynomieller Größe, die $\mathcal{L}$ entscheidet.

Satz. $\{C \mid C$ ist ein erfüllbarer Boolescher Schaltkreis $\}$ ist NP-vollständig.

Hoffnung: Analyse von Booleschen Schaltkreisen liefert untere Schranken für die Komplexität von Problemen.

Satz (Blum 1984). Es existiert $\mathcal{L} \in NP$, so daß $\mathcal{L}$ eine Netzwerkgröße von $3n$ benötigt.

3. Unabhängigkeit.

Satz (Hartmanis, Hopcroft 1976). Es existiert ein $\mathcal{L}$, so daß "$P^{\mathcal{L}} = NP^{\mathcal{L}}$" unabhängig von den Axiomen der Mengenlehre ist.

Es gibt auch Resultate, die zeigen, daß "$P = NP$" unabhängig von schwachen Fragmenten der Zahlentheorie ist.

- Jede unabhängige Π^0_1 Aussage ist wahr
- $P \neq NP$ ist eine Π^0_2 Aussage

Turingmaschinen mit unendlicher Laufzeit

		$\bigtriangledown$						
1	1	0	$\sqcup$	1	$\sqcup$	0	1	1 ...

NEU:

- $q_L \in Q$, ein *Limeszustand*

- Eingabe $\in {}^{\mathbb{N}}\Sigma$, eine *unendliche* Folge

- nachdem die Maschine λ viele Schritte gerechnet hat, gerät sie in den Limeszustand und der Kopf wird auf die erste Zelle gesetzt

- Zelleninhalt nach λ vielen Schritten: $lim_{\alpha \to \lambda}(sup(\text{Zelleninhalt zwischen } \alpha \text{ und } \lambda))$

Satz (Hamkins, Lewis 2000). $\{\Phi \mid \Phi$ ist eine wahre arithmetische Aussage $\}$ ist entscheidbar.

Satz (Hamkins, Lewis 2000). $\{x \in {}^{\mathbb{N}}\{0,1\} \mid x$ kodiert eine Wohlordnung $\}$ ist entscheidbar.

Satz (Hamkins, Lewis 2000). Jede analytische Menge ist entscheidbar.

Satz (Hamkins, Lewis 2000). Die hyperarithmetischen Mengen sind genau diejenigen, die in beschränkter Zeit $< \omega_1^{\mathsf{CK}}$ entschieden werden können.

$lh(w) = \omega$ für $w \in {}^{\mathbb{N}}\{0,1\} = \mathbb{R}$.

Definition. $\mathcal{L} \subset \mathbb{R}$ ist in **p**olynomieller Zeit entscheidbar ($\mathcal{L} \in P$) gdw. es eine Turingmaschine T und ein $m \in \mathbb{N}$ gibt, so daß

- $\mathcal{L}$ die Menge aller Eingaben ist, die T akzeptiert, und
- T bei allen Eingaben w nach $lh(w)^m = \omega^m$ vielen Schritten hält

Definition. $\mathcal{L} \subset \mathbb{R}$ ist **n**ichtdeterministisch in **p**olynomieller Zeit entscheidbar gdw. es eine Turingmaschine T und ein $m \in \mathbb{N}$ gibt, so daß

- $\mathcal{L}$ die Menge aller w ist mit $\exists z\ T$ akzeptiert $w \oplus z$, und
- T bei allen Eingaben $w \oplus z$ nach $lh(w)^m = \omega^m$ vielen Schritten hält

Satz. $NP \setminus P \neq \emptyset$ für Turingmaschinen mit unendlicher Laufzeit.

`Beweis:` Sei $A \subset \mathbb{R}$ analytisch, aber nicht Borel. Können A so wählen, daß es ein rekursives R gibt mit

$$x \in A \Leftrightarrow \exists y \forall n \in \mathbb{N}\ (x \upharpoonright n, y \upharpoonright n) \in R.$$

Es gibt eine Maschine, die $\forall n \in \mathbb{N}\ (x \upharpoonright n, y \upharpoonright n) \in R$ in ω vielen Schritten entscheidet. Insbesondere $A \in NP$.

Jede Menge in P ist Borel. Also $A \notin P$.

$\square$

Satz. Es gibt ein Orakel $\mathcal{L} \subset \mathbb{R}$, so daß $P^{\mathcal{L}} = NP^{\mathcal{L}}$ für Turingmaschinen mit unendlicher Laufzeit.