

FGI 1

Automaten, Formale Sprachen,
Berechenbarkeit

Christopher Habel & Matthias Jantzen

Kap 23 (Landau-Notation, Komplexitätsklassen)

Matthias Jantzen

M. Jantzen, FGI-1, SoSe 2009: 1

Montag, 6. Juli 2009

1

polynomiale Zeitklassen

Definition 74:

$$\mathcal{P} := \left\{ L \mid \text{Es gibt ein Polynom } p : \mathbb{N} \rightarrow \mathbb{R} \text{ und eine } p(n)\text{-zeitbeschränkte DTM } A \text{ mit } L = L(A) \right\}$$

$$\mathcal{NP} := \left\{ L \mid \text{Es gibt ein Polynom } p : \mathbb{N} \rightarrow \mathbb{R} \text{ und eine } p(n)\text{-zeitbeschränkte NTM } A \text{ mit } L = L(A) \right\}$$

Die Prüfung, ob eine vorgebene Wahrheitsbelegung einer booleschen Formel zu WAHR ausgewertet, ist deterministisch in polynomieller Zeit möglich!

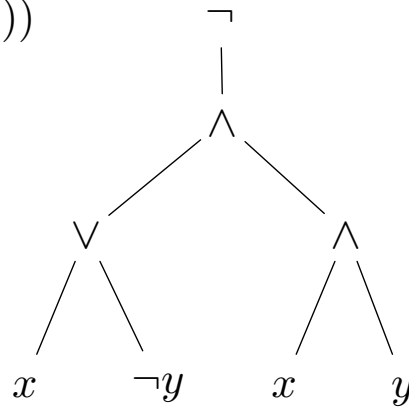
M. Jantzen, FGI-1, SoSe 2009: 2

Montag, 6. Juli 2009

2

Formel in Baumdarstellung bottom-up auswerten

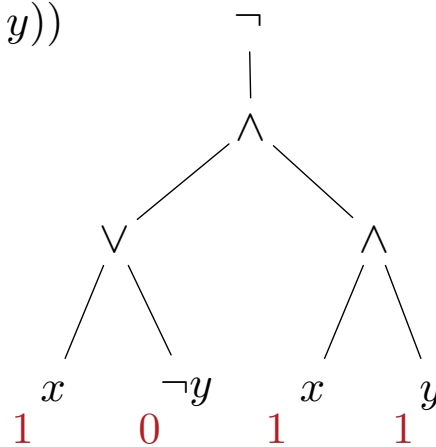
$$\neg((x \vee \neg y) \wedge (x \wedge y))$$



Belegung an Blätter notieren

$$\neg((x \vee \neg y) \wedge (x \wedge y))$$

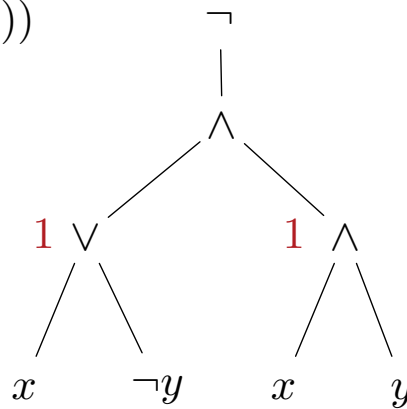
$x := 1$
 $y := 1$



Operator bekommt Auswertungswert

$$\neg((x \vee \neg y) \wedge (x \wedge y))$$

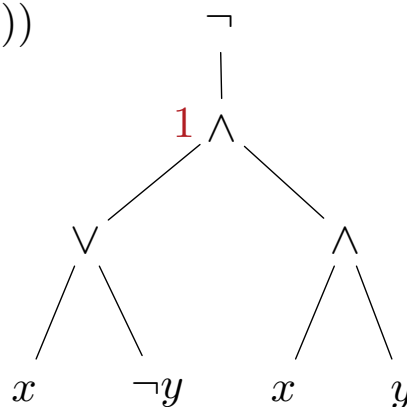
$$x:=1$$
$$y:=1$$



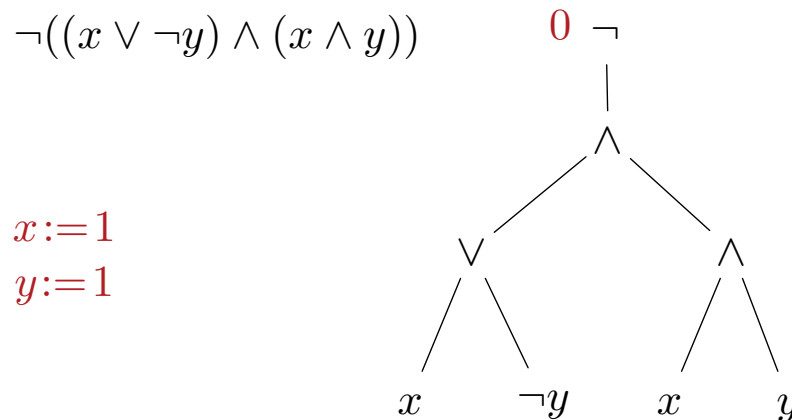
Operator bekommt Auswertungswert

$$\neg((x \vee \neg y) \wedge (x \wedge y))$$

$$x:=1$$
$$y:=1$$



Wurzel bekommt Auswertungswert der Formel



Anzahl aller Knoten ist n . An jedem Knoten ist konstanter Aufwand zur Wertberechnung nötig. Insgesamt ist Berechnung also polynomial in der Länge der Formel.

M. Jantzen, FGI-1, SoSe 2009: 7

Montag, 6. Juli 2009

7

Aus den Definitionen ergibt sich:

Korollar:

Die sich direkt aus den Definitionen ergebenden trivialen Beziehungen zwischen diesen und den zuvor definierten Komplexitätsklassen sind offensichtlich die folgenden:

$$\mathcal{D}Space(f) \subseteq \mathcal{N}Space(f)$$

$$\mathcal{D}Time(f) \subseteq \mathcal{N}Time(f)$$

$$\mathcal{D}Time(f) \subseteq \mathcal{D}Space(f)$$

$$\mathcal{N}Time(f) \subseteq \mathcal{N}Space(f)$$

$$\mathcal{P} \subseteq \mathcal{NP}$$

$$\mathcal{P}Space \subseteq \mathcal{NP}Space$$

M. Jantzen, FGI-1, SoSe 2009: 8

Montag, 6. Juli 2009

8

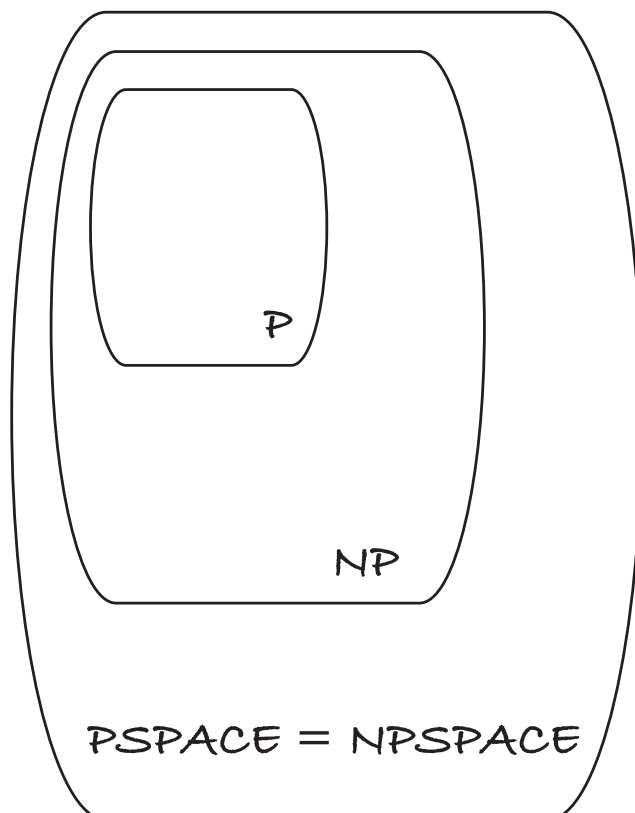
polynomiale Platzklassen

Definition 75:

$$\mathcal{P}Space := \bigcup_{i \geq 1} \mathcal{D}Space (i)$$

$$\mathcal{NP}Space := \bigcup_{i \geq 1} \mathcal{N}Space (i)$$

Zusammenhang



Walter J. Savitch
zeigte 1970:

$$\mathcal{N}Space(s(n)) \in \mathcal{D}Space(s(n)^2)$$

P und die Praxis

- Von Problemen aus der Komplexitätsklasse \mathcal{P} sagt man, dass sie **effizient lösbar** seien.
- Probleme in \mathcal{P} sind anscheinend wirklich in der Praxis noch mit vertretbarem Aufwand lösbar.
- Nehmen wir also an, dass tatsächlich alle Probleme in \mathcal{P} in der Praxis vernünftig gelöst werden können.
 - Was ist dann mit \mathcal{NP} ?
 - Sind Probleme aus \mathcal{NP} nicht mehr praktisch verwendbar/lösbar?
 - Handelt es sich nur um praktisch nicht relevante Probleme?

Machbarkeit bei Polynomzeit?

- Jedes Problem $L \in \mathcal{NP}$, ist mit exponentiellem Zeitaufwand (in $\mathcal{DTime}(2^{p(n)})$ für ein Polynom p) deterministisch lösbar.
 - Bessere Transformationen sind i.a. nicht bekannt.
- Viele der *praktisch wichtigen* Fragestellungen haben Lösungen mit Algorithmen, die **nur dann in Polynomzeit arbeiten, wenn sie nicht-deterministisch sind**, jedenfalls kennt man zur Zeit noch nicht Besseres!
- Eine **Implementierung** erfordert ein **deterministisches Verfahren**.
- Unterscheiden sich \mathcal{P} und \mathcal{NP} überhaupt?!

Beispiele von Problemen in P und NP:

- Ähnliche Probleme liegen trotzdem oft in (vielleicht wirklich) unterschiedlichen Komplexitätsklassen!

Gegeben: Matrix $A \in \mathbb{Z}^{m \times n}$ und Vektor $b \in \mathbb{Z}^m$	
Frage: Gibt es $x \in \mathbb{Z}^n$ mit $Ax = b$	Frage: Gibt es $x \in \mathbb{N}^n$ mit $Ax = b$
\mathcal{P}	\mathcal{NP}

Gegeben: Ungerichteter, kantenbewerteter Graph $G = (V, E)$, eine Gewichtsfunktion $g : E \rightarrow \mathbb{N}$, zwei Knoten $s, t \in V$ und eine Schranke $k \in \mathbb{N}$	
Frage: Gibt es einen einfachen Pfad p von s nach t mit $g(p) \leq k$? („Kürzester Weg zwischen zwei Knoten“)	Frage: Gibt es einen einfachen Pfad p von s nach t mit $g(p) \geq k$? („Längster Weg zwischen zwei Knoten“)
\mathcal{P}	\mathcal{NP}

M. Jantzen, FGI-1, SoSe 2009: T3

Montag, 6. Juli 2009

13

weitere Beispiele von Problemen in \mathcal{P} und \mathcal{NP} :

- $L_w := \{\langle G, s, t, k \rangle \mid G \text{ besitzt einen Pfad } p \text{ von } s \text{ nach } t \text{ mit } g(p) \geq k\}$ ist die dem Problem „Längster Weg zwischen zwei Knoten“ zugeordnete Sprache.

- **Theorem:** Das Problem zu L_w ist nichtdeterministisch in Polynomzeit zu lösen, d.h. $L_w \in \mathcal{NP}$.
Idee: Raten eines geeigneten Pfades.

- $K_w := \{\langle G, s, t, k \rangle \mid G \text{ besitzt einen Pfad } p \text{ von } s \text{ nach } t \text{ mit } g(p) \leq k\}$ ist die Sprache, die zu dem Problem „Kürzester Weg zwischen zwei Knoten“ gehört.

- **Theorem:** Das Problem zu K_w ist in \mathcal{P} .
Idee: Systematische Suche der minimalen Pfadlängen.

M. Jantzen, FGI-1, SoSe 2009: 14

Montag, 6. Juli 2009

14

weitere Probleme in P und NP

Gegeben: Ungerichteter Graph $G = (V, E)$

Frage: Gibt es einen geschlossenen Kreis, in dem jede Kante genau einmal auftritt? („Euler-Kreis“)

Frage: Gibt es einen geschlossenen Kreis, in dem jeder Knoten genau einmal auftritt? („Hamilton-Kreis“)

\mathcal{P}

\mathcal{NP}

- (Leider) sind häufig Probleme aus \mathcal{NP} die (für die Praxis) interessantesten!
- Manchmal kann man das aber auch ausnutzen. Z.B. für die Kryptographie.

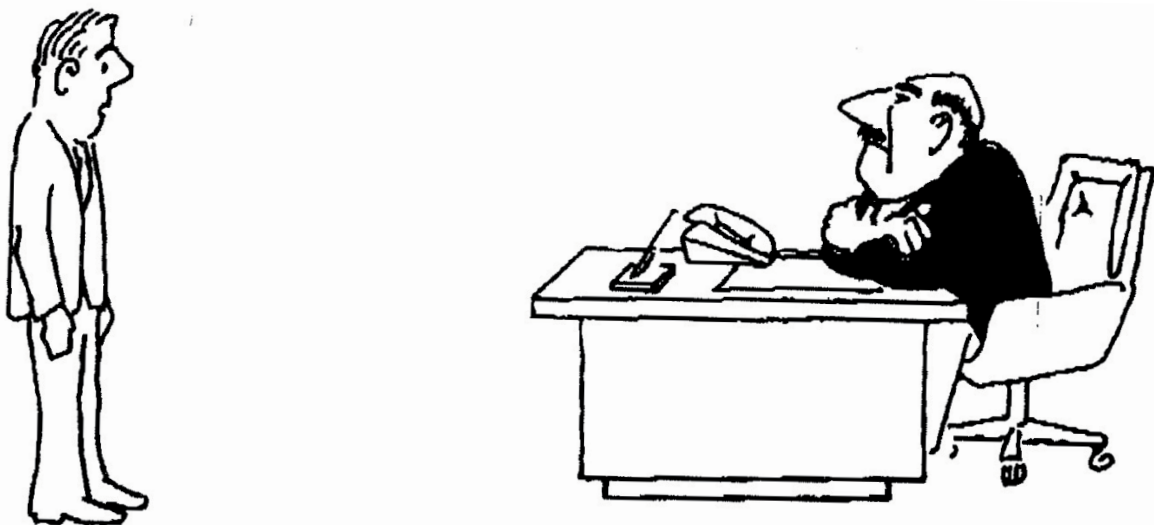
Warum ist Eulerkreis in \mathcal{P} ?

M. Jantzen, FGI-1, SoSe 2009: 15

Montag, 6. Juli 2009

15

Eine wirklich dumme Antwort



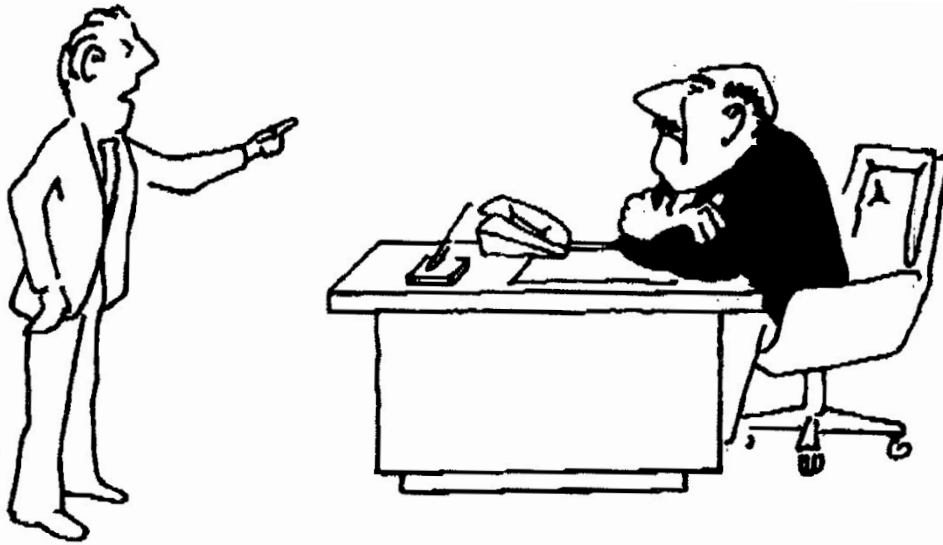
“I can't find an efficient algorithm, I guess I'm just too dumb.”

M. Jantzen, FGI-1, SoSe 2009: 16

Montag, 6. Juli 2009

16

Obwohl korrekt, für Praktiker inakzeptabel!



“I can’t find an efficient algorithm, because no such algorithm is possible!”

M. Jantzen, FGI-1, SoSe 2009: 17

Montag, 6. Juli 2009

17

Das sichert Ihre Position



“I can’t find an efficient algorithm, but neither can all these famous people.”

M. Jantzen, FGI-1, SoSe 2009: 18

Montag, 6. Juli 2009

18