

Randomisierte Algorithmen

3. Probabilistische Komplexitätsklassen

Thomas Worsch

Fakultät für Informatik
Karlsruher Institut für Technologie

Wintersemester 2018/2019

Überblick

Probabilistische Turingmaschinen

Komplexitätsklassen

Beziehungen zwischen Komplexitätsklassen

Überblick

Probabilistische Turingmaschinen

Komplexitätsklassen

Beziehungen zwischen Komplexitätsklassen

3.1 Turingmaschinen (zur Erinnerung)

- ▶ S : Menge der Zustände
- ▶ B : Bandalphabet mit B
- ▶ \square : Blanksymbol
- ▶ $A \subseteq B - \{\square\}$: Eingabealphabet
- ▶ ein Band, ein Kopf
- ▶ δ : Überföhrungsfunktion
 - ▶ deterministisch
$$\delta : S \times B \rightarrow (S \cup \{\text{YES}, \text{NO}\}) \times B \times \{-1, 0, 1\}$$
 - ▶ nichtdeterministisch
$$\delta : S \times B \rightarrow 2^{(S \cup \{\text{YES}, \text{NO}\}) \times B \times \{-1, 0, 1\}}$$
- ▶ für YES und NO keine Regeln nötig
- ▶ akzeptierte Sprache
$$L(T) = \{w \in A^+ \mid \text{für } w \text{ exist. YES-Berechnung}\}$$

3.2 Definition: probabilistische Turingmaschine

- ▶ formal wie nichtdeterministische TM
- ▶ zusätzliche Einschränkung: für alle Paare (s, b) : $1 \leq |\delta(s, b)| \leq 2$.
- ▶ *Bei zwei Möglichkeiten wird jede mit gleicher Wahrscheinlichkeit 1/2 gewählt.*
- ▶ **NB:** 1/2 ist ein „harmloser“ Wert
Unterschiede zu NTM:
 - ▶ lokale Quantifizierung
 - ▶ Interesse für *quantitative* Eigenschaften des globalen Verhaltens

3.4 Definition: PTM in Normalform

- ▶ Für jedes Paar $(s, b) \in S \times B$: $|\delta(s, b)| = 2$.
- ▶ Für jede Eingabe sind alle Berechnungen gleich lang.
- ▶ das ist o. B. d. A. (siehe Übungsaufgaben)

3.5 PTM in Normalform

- ▶ Berechnungsbaum einer PTM in Normalform ist also für jede Eingabe eine vollständiger balancierter binärer Baum.
- ▶ *Zeitkomplexität* $t(n)$ dann naheliegend definiert
- ▶ PTM arbeitet in *Polynomialzeit*, wenn $t(n) \leq p(n)$ für ein Polynom $p(n)$.

Überblick

Probabilistische Turingmaschinen

Komplexitätsklassen

Beziehungen zwischen Komplexitätsklassen

3.6 „Klassische“ Komplexitätsklassen

Klasse C	Kriterium für $L \in C$
P	L kann von DTM in poly. Zeit erkannt werden
NP	L kann von NTM in poly. Zeit erkannt werden
PSPACE	L kann von TM in poly. Platz erkannt werden

- ▶ Für Komplexitätsklasse C sei $\text{co-}C = \{L \mid \bar{L} \in C\}$.
- ▶ Beispiele:
 - ▶ $\mathbf{P} = \text{co-P}$
 - ▶ $\mathbf{P} \subseteq \mathbf{NP} \cap \text{co-NP}$
 - ▶ Beziehung zwischen \mathbf{NP} und co-NP unklar
 - ▶ $\mathbf{NP} \cup \text{co-NP} \subseteq \mathbf{PSPACE}$

3.6 „Klassische“ Komplexitätsklassen (2)

- ▶ Polynomialzeitreduktion von L_1 auf L_2 :
 - ▶ in Polynomialzeit berechenbare Abbildung $f : A^+ \rightarrow A^+$, so dass
 - ▶ für alle $w \in A^+$: $w \in L_1 \iff f(w) \in L_2$.
- ▶ Dann folgt z. B. aus $L_2 \in \mathbf{NP}$ sofort auch $L_1 \in \mathbf{NP}$.
- ▶ Eine Sprache H ist **NP-hart**, wenn jede Sprache aus **NP** auf H polynomialzeitreduziert werden kann.
- ▶ Eine Sprache ist **NP-vollständig**, wenn sie **NP-hart** und aus **NP** ist.

Umformulierung der Definition von \mathbf{P}

$L \in \mathbf{P}$ genau dann, wenn es eine DTM T gibt,

- ▶ die in Polynomialzeit arbeitet und
- ▶ bei der für alle Eingaben $w \in A^+$ gilt:
 - ▶ $w \in L \implies T$ akzeptiert w
 - ▶ $w \notin L \implies T$ akzeptiert w nicht

Umformulierung der Definition von \mathbf{P}

$L \in \mathbf{P}$ genau dann, wenn es eine DTM T gibt,

- ▶ die in Polynomialzeit arbeitet und
- ▶ bei der für alle Eingaben $w \in A^+$ gilt:
 - ▶ $w \in L \implies T$ akzeptiert w
 - ▶ $w \notin L \implies T$ akzeptiert w nicht
- ▶ bei der für alle Eingaben $w \in A^+$ gilt:
 - ▶ $w \in L \implies \mathbf{P}(T \text{ akzeptiert } w) = 1$
 - ▶ $w \notin L \implies \mathbf{P}(T \text{ akzeptiert } w) = 0$

3.7. Probabilistische Komplexitätsklassen

Im folgenden alles für alle $w \in A^+$ quantifiziert:

- ▶ $L \in \mathbf{RP}$, wenn es eine Polynomialzeit-PTM T gibt mit
 - ▶ $w \in L \implies \mathbf{P}(T \text{ akzeptiert } w) \geq 1/2$
 - ▶ $w \notin L \implies \mathbf{P}(T \text{ akzeptiert } w) = 0$
- ▶ $\mathbf{ZPP} = \mathbf{RP} \cap \text{co-}\mathbf{RP}$.
- ▶ $L \in \mathbf{BPP}$, wenn es eine Polynomialzeit-PTM T gibt mit
 - ▶ $w \in L \implies \mathbf{P}(T \text{ akzeptiert } w) > 3/4$
 - ▶ $w \notin L \implies \mathbf{P}(T \text{ akzeptiert } w) < 1/4$
- ▶ $L \in \mathbf{PP}$, wenn es eine Polynomialzeit-PTM T gibt mit
 - ▶ $w \in L \implies \mathbf{P}(T \text{ akzeptiert } w) > 1/2$
 - ▶ $w \notin L \implies \mathbf{P}(T \text{ akzeptiert } w) \leq 1/2$

3.8 Bemerkung

- ▶ **RP**: Möglichkeit eines *einseitigen* Fehlers
- ▶ **BPP, PP**: Möglichkeit eines *zweiseitigen* Fehlers
- ▶ **ZPP**: Möglichkeit, fehlerlose Maschinen zu konstruieren

allgemein:

- ▶ **Monte Carlo-Algorithmen**: Antworten evtl. falsch
- ▶ **Las Vegas-Algorithmen**: Antworten immer richtig
(und endliche erwartete Laufzeit)
- ▶ **Macao-Algorithmen**: Las Vegas-Algorithmen, die stets terminieren

3.10 Komplexitätsklassen für Normalform-PTM

- ▶ $L \in \mathbf{P}$, wenn es eine Polynomialzeit-DTM T gibt mit
 - ▶ $w \in L \implies$ alle Berechnungen liefern YES.
 - ▶ $w \notin L \implies$ alle Berechnungen liefern NO.
- ▶ $L \in \mathbf{RP}$, wenn es eine Polynomialzeit-PTM T gibt mit
 - ▶ $w \in L \implies$ mind. $1/2$ aller Berechnungen liefern YES.
 - ▶ $w \notin L \implies$ alle Berechnungen liefern NO.
- ▶ $L \in \mathbf{BPP}$, wenn es eine Polynomialzeit-PTM T gibt mit
 - ▶ $w \in L \implies$ mind. $3/4$ aller Berechnungen liefern YES.
 - ▶ $w \notin L \implies$ mind. $3/4$ aller Berechnungen liefern NO.
- ▶ $L \in \mathbf{PP}$, wenn es eine Polynomialzeit-PTM T gibt mit
 - ▶ $w \in L \implies$ mehr als $1/2$ aller Berechnungen liefern YES.
 - ▶ $w \notin L \implies$ mind. $1/2$ aller Berechnungen liefern NO.

3.11 Problem

3.11 Problem

- ▶ Für eine PTM ist es nicht ersichtlich, ob sie ein Beweis dafür ist, dass die von ihr erkannte Sprache in **RP** ist.
- ▶ Das ist sogar *unentscheidbar*.
- ▶ Die analoge Aussage gilt auch für **BPP**.

3.12 Satz

Für jedes Polynom $q(n) \geq 1$ kann man in der Definition von **RP** anstelle von $1/2$ auch $1 - 2^{-q(n)}$ einsetzen, also Fehlerwahrscheinlichkeit $2^{-q(n)}$ statt $1/2$ fordern, ohne an der Klasse etwas zu verändern.

3.13 Beweis

Gegeben: eine **RP**-PTM R , die L mit Fehlerwahrscheinlichkeit $1/2$ erkennt.

Gesucht: eine **RP**-PTM R' , die L mit Fehlerwahrscheinlichkeit $2^{-q(n)}$ erkennt.

3.13 Beweis

Gegeben: eine **RP**-PTM R , die L mit Fehlerwahrscheinlichkeit $1/2$ erkennt.

Gesucht: eine **RP**-PTM R' , die L mit Fehlerwahrscheinlichkeit $2^{-q(n)}$ erkennt.

Konstruktion:

- ▶ w eine Eingabe der Länge n .
- ▶ R' verwaltet einen Zähler, mit $q(n)$ initialisiert.
- ▶ In einer Schleife wird
 - ▶ Zähler auf Null heruntergezählt
 - ▶ dabei jedes Mal eine Berechnung von R für die Eingabe w simuliert.
- ▶ Wenn R bei einem Versuch YES liefern würde,
 - ▶ dann hält R' mit YES,
 - ▶ sonst hält R' mit Antwort NO.

3.13 Beweis (2)

- ▶ Fehlerwahrscheinlichkeit:
 - ▶ Ist $w \in L(R)$, dann antwortet R' falsch mit Wahrscheinlichkeit $(1/2)^{q(n)} = 2^{-q(n)}$.
 - ▶ Ist $w \notin L(R)$, dann antwortet R und daher auch R' immer richtig mit NO.
- ▶ Laufzeit:
 - ▶ Ist $p(n)$ die Laufzeit von R , dann
 - ▶ ist $p(n)q(n)$ die Laufzeit von R' , also auch polynomiell.

3.14 Satz

Für jedes Polynom $q(n) \geq 2$ kann man in der Definition von **BPP** anstelle der Fehlerwahrscheinlichkeit von $1/4$ auch $2^{-q(n)}$ einsetzen, ohne etwas zu ändern.

3.15 Beweis

Gegeben: eine **BPP**-PTM R , die L mit Fehlerwahrscheinlichkeit $1/4$ erkennt.

Gesucht: eine **BPP**-PTM R' , die L mit Fehlerwahrscheinlichkeit $2^{-q(n)}$ erkennt.

3.15 Beweis

Gegeben: eine **BPP**-PTM R , die L mit Fehlerwahrscheinlichkeit $1/4$ erkennt.

Gesucht: eine **BPP**-PTM R' , die L mit Fehlerwahrscheinlichkeit $2^{-q(n)}$ erkennt.

Konstruktion:

- ▶ w : Eingabe der Länge n .
- ▶ $m = 2q + 1$: ungerade Zahl, später geeignet festgelegt.
- ▶ R' verwaltet Zähler, mit m initialisiert.
- ▶ in einer Schleife:
 - ▶ Zähler auf Null heruntergezählt
 - ▶ immer eine Berechnung von R für Eingabe w simuliert
- ▶ zähle, wie oft R Antwort YES bzw. NO gibt
- ▶ Antwort von R' : die, die von R häufiger gegeben wurde

3.15 Beweis (2)

Wahrscheinlichkeit für falsche Antwort ist

$$\begin{aligned} & \sum_{i=0}^q \binom{m}{i} \left(\frac{3}{4}\right)^i \left(\frac{1}{4}\right)^{m-i} \\ & \leq \sum_{i=0}^q \binom{m}{i} \left(\frac{3}{4}\right)^{m/2} \left(\frac{1}{4}\right)^{m/2} \\ & = \left(\frac{3}{16}\right)^{m/2} \sum_{i=0}^q \binom{m}{i} \\ & \leq \left(\frac{3}{16}\right)^{m/2} \frac{1}{2} 2^m = \frac{1}{2} \left(\frac{3}{4}\right)^{m/2} \end{aligned}$$

3.15 Beweis (3)

Man wählt nun m so, dass gilt:

$$\frac{1}{2} \left(\frac{3}{4} \right)^{m/2} \leq 2^{-q(n)}$$

$$\iff 2 \left(\frac{4}{3} \right)^{m/2} \geq 2^{q(n)}$$

$$\iff \left(\frac{4}{3} \right)^{m/2} \geq 2^{q(n)-1}$$

$$\iff \frac{m}{2} (2 - \log 3) \geq q(n) - 1$$

$$\iff m \geq \frac{2(q(n) - 1)}{2 - \log 3}$$

- ▶ geeignete Anzahl von Wiederholungen linear in $q(n)$
- ▶ mit der Laufzeit von R ist auch die von R' polynomiell

3.16 Satz

Wenn $L \in \mathbf{ZPP}$ ist, dann gibt es eine PTM, die

- ▶ L entscheidet (ohne Fehler) und
- ▶ für die der Erwartungswert der Laufzeit polynomiell ist.

$L \in \mathbf{RP}$ und $L \in \mathbf{co-RP}$

	RP-TM R für L	<i>Antwort von R</i>	
		« $w \in L$ »	« $w \notin L$ »
<i>real</i>	$w \in L$	$\geq 1/2$	$\leq 1/2$
	$w \notin L$	0	1

	RP-TM \bar{R} für \bar{L}	<i>Antwort von \bar{R}</i>	
		« $w \in \bar{L}$ »	« $w \notin \bar{L}$ »
<i>real</i>	$w \in \bar{L}$	$\geq 1/2$	$\leq 1/2$
	$w \notin \bar{L}$	0	1

$L \in \mathbf{RP}$ und $L \in \mathbf{co-RP}$

	RP-TM R für L	Antwort von R	
		$\langle\langle w \in L \rangle\rangle$	$\langle\langle w \notin L \rangle\rangle$
<i>real</i>	$w \in L$	$\geq 1/2$	$\leq 1/2$
	$w \notin L$	0	1

	RP-TM \bar{R} für \bar{L}	Antwort von \bar{R}	
		$\langle\langle w \in \bar{L} \rangle\rangle$	$\langle\langle w \notin \bar{L} \rangle\rangle$
<i>real</i>	$w \in \bar{L}$	$\geq 1/2$	$\leq 1/2$
	$w \notin \bar{L}$	0	1

	RP-TM \bar{R} für \bar{L}	Antwort von \bar{R}	
		$\langle\langle w \notin L \rangle\rangle$	$\langle\langle w \in L \rangle\rangle$
<i>real</i>	$w \notin L$	$\geq 1/2$	$\leq 1/2$
	$w \in L$	0	1

$L \in \mathbf{RP}$ und $L \in \mathbf{co-RP}$

	RP-TM R für L	<i>Antwort von R</i>	
		« $w \in L$ »	« $w \notin L$ »
<i>real</i>	$w \in L$	$\geq 1/2$	$\leq 1/2$
	$w \notin L$	0	1

	RP-TM \bar{R} für \bar{L}	<i>Antwort von \bar{R}</i>	
		« $w \in \bar{L}$ »	« $w \notin \bar{L}$ »
<i>real</i>	$w \in \bar{L}$	$\geq 1/2$	$\leq 1/2$
	$w \notin \bar{L}$	0	1

	RP-TM \bar{R} für \bar{L}	<i>Antwort von \bar{R}</i>	
		« $w \notin L$ »	« $w \in L$ »
<i>real</i>	$w \notin L$	$\geq 1/2$	$\leq 1/2$
	$w \in L$	0	1

	RP-TM \bar{R} für \bar{L}	<i>Antwort von \bar{R}</i>	
		« $w \in L$ »	« $w \notin L$ »
<i>real</i>	$w \in L$	1	0
	$w \notin L$	$\leq 1/2$	$\geq 1/2$

3.17 Beweis

Es sei R eine **RP**-PTM und \bar{R} eine **RP**-PTM für \bar{L} .

⟨Eingabe: Wort w ⟩

⟨Ausgabe: YES falls $w \in L$, NO falls $w \notin L$ ⟩

repeat

$r \leftarrow R(w)$

$r' \leftarrow \mathbf{not} \bar{R}(w) \quad \langle \bar{R}(w) = YES \implies w \in \bar{L}, w \notin L \rangle$

until $r = r'$

return r

Antwort *immer* korrekt:

- ▶ falls $w \notin L$: Antwort von R garantiert richtig
- ▶ falls $w \in L, w \notin \bar{L}$: Antwort von \bar{R} garantiert richtig

3.17 Beweis (2)

Laufzeit

- ▶ Sei $p(n)$ polynomielle Schranke für die Laufzeiten von R bzw. \bar{R} .
- ▶ Erwartungswert für die Laufzeit ist

$$\begin{aligned} &\leq \frac{1}{2}2p(n) + \frac{1}{4}4p(n) + \frac{1}{8}6p(n) + \dots \\ &= 2p(n) \sum_{i=1}^{\infty} 2^{-i}i \end{aligned}$$

- ▶ und weil $\sum_{i=0}^k 2^{-i}i = 2 - (k+2)2^{-k} \leq 2$ ist, ist
- ▶ $E[\text{Laufzeit}] \leq 4p(n)$.

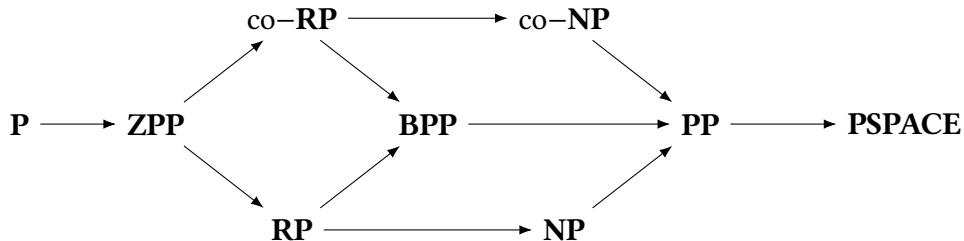
Überblick

Probabilistische Turingmaschinen

Komplexitätsklassen

Beziehungen zwischen Komplexitätsklassen

3.19 Beziehungen zwischen Komplexitätsklassen



3.20 – 3.23 Sätze

$$\mathbf{P} \subseteq \mathbf{ZPP} \subseteq \mathbf{RP} \quad \text{und} \quad \mathbf{ZPP} \subseteq \text{co-}\mathbf{RP}$$

$$\mathbf{RP} \subseteq \mathbf{NP} \quad \text{und} \quad \text{co-}\mathbf{RP} \subseteq \text{co-}\mathbf{NP}$$

$$\mathbf{RP} \subseteq \mathbf{BPP} \quad \text{und} \quad \text{co-}\mathbf{RP} \subseteq \mathbf{BPP}$$

$$\mathbf{BPP} \subseteq \mathbf{PP}$$

3.24 Satz

NP \subseteq **PP** und **co-NP** \subseteq **PP**

3.25 Beweis

- ▶ Gegeben: Polynomialzeit-NTM N in Normalform.
- ▶ konstruiere PTM N' wie folgt:

3.25 Beweis

- ▶ Gegeben: Polynomialzeit-NTM N in Normalform.
- ▶ konstruiere PTM N' wie folgt:
 - ▶ am Ende jeder Berechnung weiterer Schritt:
 - ▶ die eine Antwort: die ursprüngliche Antwort von N
 - ▶ die andere Antwort: auf jeden Fall YES
- ▶ Ist $w \in L(N)$, dann gibt es bei N mindestens eine akzeptierende Berechnung.
- ▶ Folglich ist bei N' mehr als die Hälfte aller Berechnungen akzeptierend.
- ▶ Ist $w \notin L(N)$, dann ist nur genau die Hälfte aller Berechnungen akzeptierend.
- ▶ Mit N arbeitet auch N' in Polynomialzeit.

3.26 Satz

$$\mathbf{PP} \subseteq \mathbf{PSPACE}$$

3.27 Beweis

P sei **PP**-PTM in Normalform mit Zeitkomplexität $p(n)$.

\langle Eingabe: w \rangle

$a \leftarrow 0$ \langle Zähler für akzeptierende Berechnungen \rangle

$k \leftarrow p(|w|)$ \langle Anzahl Schritte von P \rangle

\langle für alle Bitfolgen der Länge k : \rangle

for $(b_k b_{k-1} \cdots b_1) \leftarrow (000 \cdots 0)$ **to** $(111 \cdots 1)$ **do**

$r \leftarrow$ Ergebnis der Simulation von $P(w)$ mit
 Zufallsentscheidungen gemäß den b_i

if $r = \text{YES}$ **then** $a \leftarrow a + 1$ **fi**

od

if $a > 2^{k-1}$ **then**

return YES

else

return NO

fi

3.27 Beweis (2)

- ▶ Platzbedarf: dominiert von den Bits b_i und dem für die Simulationen von $P(w)$.
- ▶ Anzahl k der Bits ist polynomiell in $|w|$.
- ▶ Jede Simulation dauert polynomiell lange, hat also auch nur polynomiellen Platzbedarf.

Zusammenfassung

- ▶ Fehler: keiner, einseitig, zweiseitig
- ▶ Las Vegas (und Macao) und Monte Carlo
- ▶ Reduktion der Fehlerwahrscheinlichkeit auf Kosten der Anzahl der Zufallsbits