Kapitel 5 Weitere Komplexitätsklassen

§1 Randomisierte Berechnung

bisher: haben nur deterministische Berechnungen / TMs betrachtet.

jedoch: in vielen Algorithmen in der Praxis werden Zufallszahlen verwendet. (z.B. Quicksort, SAT-Algo. von Schoening, Monte-Carlo-Methoden für Integration)

Def.: Eine probabilistische TM (PTM) M besitzt zwei Zustandsübergangsfkt. $\delta_0, \delta_1 : Q \times \Gamma \to Q \times \Gamma \times \{R, N, L\}$. In jedem Rechenschritt wird eine der beiden Fkt. δ_0, δ_1 mit W.keit $1/2$ ausgewählt. Diese zufällige Wahl ist jeweils unabhängig von den vorhergehenden Wahlen. Die PTM gibt entweder 1 („accept") oder 0 („reject") aus. Bei Eingabe x ist die Ausgabe von M, Notation $M(x) \in \{0,1\}$, eine Zufallsvariable.

Sei $T : \mathbb{N} \to \mathbb{N}$ eine Fkt. Wir sagen M läuft in

Zeit $T(n)$, wenn für jede Eingabe x der Länge n, die PTM nach höchstens $T(n)$ Schritten hält, unabhängig von den zufälligen Entscheidungen.

<u>Notation</u>: Sei $L \subseteq \Sigma^*$ eine Sprache. Führe das Prädikat $L(x) \in \{0,1\}$ für $x \in \Sigma^*$ ein mit

$$L(x) = 1 \iff x \in L.$$

<u>Def.</u>: (<u>Klasse BPP</u> = „probabilistic polynomial time with bounded error")

Sei $L \subseteq \Sigma^*$. Dann ist $L \in BPP$, wenn es eine polynomialzeitbeschränkte PTM M gibt, so dass gilt: $\Pr[M(x) = L(x)] \geq \frac{2}{3}$ für alle $x \in \Sigma^*$.

Mit BPP kann man effiziente probabilistische Algo. beschreiben, die <u>zweiseitige</u> Fehler zulassen.

<u>klar</u>: $P \subseteq BPP$ <u>offen</u>: $BPP \subseteq NP$

<u>Vermutung</u>: $P = BPP$.

Anstatt der W.keit $\frac{2}{3}$ in der Def. von BPP kann jede Zahl zwischen $\frac{1}{2}+\varepsilon$ und $1-\varepsilon$ wählen:

Wiederhole die Berechnung von M eine konstante Anzahl k mal und entscheide mit Mehrheitsentscheidung (dies ist PTM M'):

$$\Pr[M'(x) = L(x)] = \sum_{i=\lceil \frac{k}{2} \rceil}^{k} \binom{k}{i} \left(\frac{2}{3}\right)^i \left(\frac{1}{3}\right)^{k-i}$$

$$= \left(\frac{1}{3}\right)^k \sum_{i=\lceil \frac{k}{2} \rceil}^{k} \binom{k}{i} 2^i$$

$$= \left(\frac{1}{3}\right)^k \left(3^k - \sum_{i=0}^{\lceil \frac{k}{2} \rceil - 1} \binom{k}{i} 2^i\right) \xrightarrow{k \to \infty} 1$$

Viele probabilistische Algorithmen haben nur einseitige Fehler, z.B. falls $x \notin L$, dann ist immer $M(x) = 0$.

<u>Def.</u>: (<u>Klasse RP</u> = „random polynomial")

$L \in RP \iff \exists$ PTM M, die polynomialzeitbeschränkt ist

mit $x \in L \Rightarrow \Pr[M(x) = 1] \geq \frac{2}{3}$

$x \notin L \Rightarrow \Pr[M(x) = 0] = 1.$

bzw. $coRP = \{L : \overline{L} \in RP\}$, wobei $\overline{L} = \Sigma^* \setminus L$.

Alternativ:

$L \in coRP \iff \exists$ PTM M, polynomialzeitbeschränkt

mit $x \in L \Rightarrow Pr[M(x)=1] = 1$

$x \notin L \Rightarrow Pr[M(x)=0] \geq \frac{2}{3}$.

<u>Satz</u> (a) $RP, coRP \subseteq BPP$.

(b) $RP \subseteq NP$, $coRP \subseteq coNP$

<u>Bew.</u>: (a) klar

(b) Verwende die Folge der Wahlen der Zustandsübergangsfkt. δ_0, δ_1 als Zertifikat.

$\boxtimes$

<u>Bsp.</u>: Randomisierter Algo., um zu testen, ob ein bipartiter Graph ein perfektes Matching besitzt (nach Lovász)

Sei $G = (V, E)$ bipartiter Graph mit Bipartition $V = U \dot\cup W$. Sei $|U| = |W|$. G besitzt perfektes Matching, falls Matchingzahl $\mu(G) = |U|$ ist.

Das kann man z.B. mit Hilfe der ungarischen Methode ($\to$ OR) testen.

Geht noch einfacher:

Definiere Matrix $X \in K[\{x_{u,w} : u \in U, w \in W\}]^{U \times W}$

durch $X_{u,w} = \begin{cases} x_{u,w}, & \text{falls } \{u,w\} \in E \\ 0, & \text{sonst} \end{cases}$,

wobei K irgendein Körper ist.

<u>Beh.</u>: G besitzt perfekte Matching $\iff \det X \neq 0$.

<u>Bew.</u>: <u>Leibniz</u>:

$$\det X = \sum_{\substack{\sigma : U \to W \\ \text{bijektiv}}} (-1)^{\operatorname{sgn}(\sigma)} \prod_{u \in U} X_{u,\sigma(u)}$$

Dies ist ein Polynom in den Variablen $x_{u,w}$, das ein Monom für jedes perfekte Matching in G besitzt.

<u>Problem</u>: Müssen testen, ob $\det X$ nicht das Nullpolynom ist. Das kann exponentielle Laufzeit erfordern, wenn man das Polynom $\det X$ hinschreiben möchte.

<u>Ausweg</u>: Setze zufällig Körperelemente $\alpha_{u,w} \in K$ für $x_{u,w}$ ein und überprüfe (z.B. mit Gaußscher Elimination in $O(n^3)$-Zeit), ob

$\det X(\alpha) = 0 \in K$.

Falls $\det X(\alpha) \neq 0$, dann $\det X \neq 0$ und G hat perfektes Matching.

Falls $\det X(\alpha) = 0$, dann ? (einseitiger Fehler)

<u>Satz</u> (Schwartz-Zippel, 1979)

Sei K ein Körper und $p \in K[x_1, \dots, x_m]$, $p \neq 0$, ein Polynom vom Grad d. Sei $S \subseteq K$ endlich. Wähle $\alpha_1, \dots, \alpha_m \in S$ unabhängig, gleichverteilt. Dann ist

$$\Pr[p(\alpha_1, \dots \alpha_m) \neq 0] \geq 1 - \frac{d}{|S|}.$$

Anwendung auf $\det X$: Wähle $K = \mathbb{Q}$ und $S = \{1, 2, \dots, 3n\}$. Dann ist im Fall $\det X \neq 0$

$$\Pr[\det X(\alpha) \neq 0] \geq 1 - \frac{n}{3n} = \frac{2}{3}.$$