Prof. T. Nipkow
K. Kappelmann, J. Rädle, L. Stevens **ABGABE 06.07.2020, 23:59 ($\Sigma$ 5,5 P.)** Technische Universität München
Lehrstuhl für Logik und Verifikation

# Einführung in die Theoretische Informatik

Sommersemester 2020 – Übungsblatt 10

**AUFGABE 10.1.** (*Die Korrespondenz hält sich in Grenzen*) 1,5 Punkte

Für Instanzen $(x_1, y_1), \dots, (x_m, y_m)$ des Post'schen Korrespondenzproblems bezeichnen wir eine Lösung $i_1, \dots, i_n$ als $k$-begrenzt, wenn für alle $j \leq n$ gilt:

$$\Big| \left|x_{i_1} \dots x_{i_j}\right| - \left|y_{i_1} \dots y_{i_j}\right| \Big| \leq k.$$

Zeigen Sie, dass für fixes $k$ entscheidbar ist, ob eine PCP-Instanz eine $k$-begrenzte Lösung hat. **Hinweis:** Konstruieren Sie einen DFA über dem Alphabet $\Sigma = \{1, \dots, m\}$, der ein Wort $i_1 \dots i_n$ genau dann akzeptiert, wenn es eine $k$-begrenzte Lösung ist. Einen Korrektheitsbeweis für die Konstruktion ist nicht notwendig.

**AUFGABE 10.2.** 2 Punkte

Sei $A \subseteq \Sigma^*$ eine Sprache in P. Zeigen Sie, dass die Sprache $\{w_1 \dots w_l \mid w_1, \dots, w_l \in A \land l \leq k\}$ für jede Konstante $k \in \mathbb{N}$ ebenfalls in P liegt.

**AUFGABE 10.3.** ($\Pr[$*Tutor*in akzeptiert Beweis*$] > 0$) 1 + 1 Punkte

Bis hierhin haben wir nur deterministische oder nichtdeterministische Algorithmen betrachtet. Ein bisher von uns unbetrachtetes Feld ist das der randomisierten Algorithmen. Dies soll sich nun ändern:
Eine *probabilistische* Turingmaschine ist ein 8-Tupel $M = (Q, \Sigma, \Gamma, \delta_1, \delta_2, q_0, \square, F)$, wobei

(a) $\delta_1 : Q \times \Gamma \to Q \times \Gamma \times \{L, N, R\}$ die erste Übergangsfunktion und
(b) $\delta_2 : Q \times \Gamma \to Q \times \Gamma \times \{L, N, R\}$ die zweite Übergangsfunktion

der Maschine ist. Die restlichen Komponenten sind analog zu dem klassischen, bereits bekannten Turingmaschinen-Modell.
Bei jedem Schritt der Maschine wird nun zufällig, mit jeweils Wahrscheinlichkeit $1/2$, entweder ein Schritt der ersten oder der zweiten Übergangsfunktion durchgeführt. Das Auswählen der Transitionsfunktion spiegelt somit das Prinzip eines Münzwurfes wieder.
Das zufällige Auswählen der Übergangsfunktion kann zu Uneinstimmigkeiten führen: Es kann passieren, dass in einem Lauf die Maschine eine Wort akzeptiert, in einem weiteren Lauf das Wort jedoch abgelehnt wird. Eine probabilistische Turingmaschine akzeptiert somit Sprachen mit einer gewissen Wahrscheinlichkeit.
Sei $\varepsilon_+, \varepsilon_- \in [0, 1] \subset \mathbb{R}$ und $L$ eine Sprache. Wir sagen, dass die probabilistische Turingmaschine $M$ die Sprache $L$ mit Fehler $(\varepsilon_+, \varepsilon_-)$ akzeptiert falls

(a) $\forall w \in L.\ \Pr[M \text{ akzeptiert } w] \geq 1 - \varepsilon_+$
(b) $\forall w \notin L.\ \Pr[M \text{ lehnt } w \text{ ab }] \geq 1 - \varepsilon_-$

Wir definieren nun die Klasse $\mathsf{RP}_{(\varepsilon_+, \varepsilon_-)}$ als die Menge aller Sprachen, die von einer probabilistischen Turingmaschine mit Fehler $(\varepsilon_+, \varepsilon_-)$ in polynomieller Zeit akzeptiert werden. Zusätzlich setzen wir $\mathsf{RP} := \mathsf{RP}_{(1/2, 0)}$.

(a) Zeigen Sie: $\mathsf{P} \subseteq \mathsf{RP} \subseteq \mathsf{NP}$.

(b) Zeigen Sie: $\mathsf{RP} = \mathsf{RP}_{(\varepsilon_+, 0)}$ für alle $\varepsilon_+ \in (0, 1) \subset \mathbb{R}$.

**Bemerkung:** Halten Sie sich in beiden Teilaufgaben strikt an die in der Vorlesung bzw. hier präsentierten Maschinenmodelle und Definitionen der Zeitklassen. Argumentationen beispielsweise anhand von Javaprogrammen werden nicht gewertet.

> Nearly every example of faulty reasoning that has been published is accompanied by the phrase "of course" or its equivalent.
>
> — Donald Knuth