Vorlesungsmitschrift

Algorithmen und Berechenbarkeit

Vorlesung 17

Letztes Update: 2018/01/20 - 10:45 Uhr

Die Komplexitätsklasse $\mathcal{NP}$

Ein Problem $\mathcal{X}$ ist in der Komplexitätsklasse $\mathcal{P}$, wenn es einen Polynomzeitalgorithmus für $\mathcal{X}$ gibt (*alternativ:* Ein Problem $\mathcal{X}$ ist in der Komplexitätsklasse $\mathcal{P}$, wenn es eine TM $\mathcal{M}$ gibt, die $\mathcal{X}$ in einer polynomiellen Anzahl an Schritten löst).

Definition Akzeptanzverhalten einer NTM: Eine NTM $\mathcal{M}$ akzeptiert eine Eingabe $x \in \Sigma^*$ falls es mindestens eine Sequenz von gültigen Rechenschritten (gemäß Übergangsrelation) gibt, die in einer akzeptierenden Konfiguration endet.

Definition Laufzeit einer NTM: Sei $\mathcal{M}$ eine NTM. Die Laufzeit $T_{\mathcal{M}}(x)$ von $\mathcal{M}$ auf einer Eingabe $x \in L(\mathcal{M})$ ist definiert als

$$T_{\mathcal{M}}(x) := \text{Länge des kürzesten akzeptierenden Rechenwegs von } \mathcal{M} \text{ auf } x$$

Außerdem gilt: Für ein $x \notin L(\mathcal{M})$ ist $T_{\mathcal{M}}(x) = 0$. Die Worst-Case-Laufzeit $t_{\mathcal{M}}(n)$ für $\mathcal{M}$ auf Eingaben der Länge n ist

$$t_{\mathcal{M}}(n) := max\{T_{\mathcal{M}}(x) \mid x \in \Sigma^n\}$$

Definition Komplexitätsklasse $\mathcal{NP}$: $\mathcal{NP}$ ist die Klasse der Entscheidungsprobleme, die durch eine NTM $\mathcal{M}$ erkannt wird, deren Worst-Case-Laufzeit $t_{\mathcal{M}}(n)$ polynomiell **in** n beschränkt ist. $\mathcal{NP}$ bedeutet *Nichtdeterministisch Polynomiell*.

Beispiel für ein $\mathcal{NP}$-Problem: CLIQUE

Das *CLIQUE*-Problem liegt nicht in $\mathcal{P}$ und ist definiert wie folgt: Gegeben sei ein ungerichteter Graph $G(V, E)$ und ein $k \in \{1, \ldots, |V|\}$. Nun möchte man wissen, ob G eine CLIQUE der Größe k hat. Dieses Problem kann naiv in $\mathcal{O}(n^k)$ entschieden werden, was jedoch nicht polynomiell ist (Eine CLIQUE ist dabei eine Teilmenge von Knoten von G, die vollständig untereinander verbunden sind).

Satz: CLIQUE $\in \mathcal{NP}$

Beweis: Es wird eine NTM $\mathcal{M}$ beschrieben mit $L(\mathcal{M})$ =CLIQUE, die polynomielle Laufzeit hat und wie folgt vorgeht:

1. Falls die Eingabe nicht der Form (G, K) entspricht, wird verworfen.
2. Sei nun $G = (V, E)$, $N =$ Anzahl der Knoten ohne Beschränkung der Allgmeinheit und $V = \{1, \dots, N\}$.

   $\mathcal{M}$ schreibt hinter die Eingabe den String $\#^N$, der Kopf bewegt sich über das erste #.
3. $\mathcal{M}$ läuft von links nach rechts über $\#^N$ und ersetzt nichtdeterministisch jedes # durch 0 oder 1. Der resultierende String sei $y = (y_1, y_2, \dots, y_N) \in \{0, 1\}^N$
4. Sei $C = \{i \in V \mid q_i = 1\}$. $\mathcal{M}$ akzeptiert, falls $C =$ K-CLIQUE.

Task $1, 2$ und 4 sind deterministisch, Task 3 ist nichtdeterministisch. Alle Tasks benötigen eine polynomielle Anzahl an Schritten.

Nun muss gezeigt werden, dass $L(\mathcal{M}) =$ CLIQUE.

$a)$ Angenommen, G enthält CLIQUE:

- $\Rightarrow$ Dann existiert mindestens ein y, dass zur Akzeptanz in Task 4 führt
- $\Rightarrow$ Dieses y wird in Task 3 nichtdeterministisch gefunden
- $\Rightarrow$ $\mathcal{M}$ akzeptiert die Eingabe

$b)$ Angenommen, G enthält CLIQUE nicht:

- $\Rightarrow$ Egal was das Ergebnis aus Task 3 ist, Task 4 führt nie zur Akzeptanz

$\Rightarrow$ CLIQUE $\in \mathcal{NP}$. $\square$

(Im Skript Kapitel 3.2.2 lesen)

Beispiel für ein $\mathcal{NP}$-Hart-Problem: Rucksack/Knapsack

Rucksack/Knapsack (Optimierungsvariante)

Gegeben sind N Gegenstände $U = \{u_1, u_2, \dots, u_N\}$ mit den jeweiligen Werten $p_1, p_2, \dots, p_N$, Gewichten $w_1, w_2, \dots, w_N$ und einer Gewichtsschranke B. Nun soll entschieden werden, ob es eine Teilmenge $k \subseteq U$ gibt mit

$$\sum_{u_i \in U} w_i \leq B \textbf{ und } \sum_{u_i \in K} P_i \text{ maximal}$$

Rucksack/Knapsack (Entscheidungsvariante)

Gegeben sind wieder $U, B \in \mathbb{N}, P \in \mathbb{N}$. Man stellt sich nun die Frage: Existiert ein $K \subseteq U$, für das gilt:

$$\sum_{u_i \in K} w_i \leq B \textbf{ und } \sum_{k_i \in K} p_i \geq P$$

Es kann gezeigt werden, dass das Lösen der einen Variante automatisch auch die andere Variante löst.