Einführung in Berechenbarkeit, Komplexität und Formale Sprachen: Blatt 7

Bernhard Dietrich (6256800)
Lars Fernhomberg (6256030)
Sebastian Kniesburges (6257120)

Übungsgruppe 4
Montag 16:00-18:00
Gunnar Schomaker

Aufgabe 25 (5 Punkte)
Sei folgende Sprache L gegeben. Zeigen Sie mittels Reduktion, dass L unentscheidbar ist.
$L := \left\{ \langle M_1 \rangle \langle M_2 \rangle \middle| M_1 \text{ akzeptiert das Komplement der von } M_2 \text{ akzeptierten Sprache} \right\}$
Dieses Problem lässt sich auf das Äquivalenzproblem
$\ddot{A} = \left\{ \langle M_1 \rangle \langle M_2 \rangle \middle| M_1 \text{ und } M_2 \text{ akzeptieren dieselbe Sprache} \right\}$ reduzieren, so dass gilt $L \leq \ddot{A}$.
Es gibt somit eine Funktion $f : \Sigma^* \to \Sigma^*$, für die gilt: $x \in L \Rightarrow f(x) \in \ddot{A} \wedge x \notin L \Rightarrow f(x) \notin \ddot{A}$.
Somit muss die Funktion M_1 vom Band löschen und das Komplement $\overline{M_1}$ auf das Band schreiben. Somit gilt $f\left(\langle M_1 \rangle \langle M_2 \rangle\right) = \overline{\langle M_1 \rangle} \langle M_2 \rangle$.

Aufgabe 26 (5 Punkte)
Zeigen Sie: Sei M eine $t(n)$ zeit- und $s(n)$ platzbeschränkte NTM. Dann kann M durch eine $O(t(n))$ zeit- und $O(s(n))$ platzbeschränkte NTM simuliert werden, in der jede Konfiguration höchstens zwei Nachfolgekonfigurationen hat.

M besitzt die Übergangsfunktion δ, welche für die Zustände $q_0, ..., q_n$ definiert ist. Um nun eine Turingmaschine zu konstruieren, welche nur maximal zwei Nachfolgekonfigurationen pro Konfiguration hat, müssen die Übergänge, bei denen mehr als zwei Nachfolgekonfigurationen für eine Konfiguration vorhanden sind entsprechend aufgeteilt werden. Dieses geschieht am besten durch eine Baumstruktur, bei der dann die Blätter den eigentlichen, ursprünglichen Konfigurationsübergängen entsprechen und die Wurzel sowie die inneren Knoten nur als Hilfsmittel zu verstehen sind und keinerlei Bewegungen auf dem Band ausführen sowie das aktuelle Zeichen auf dem Band nicht durch einen anderen Wert überschreiben.
Wenn wir nun für einen Übergang n (n sei Zweierpotenz bzw. nächst höhere Zweierpotenz des eigentlichen Wertes) mögliche Folgekonfigurationen haben, so benötigen wir einen binären Baum mit $2^n - 1$ Hilfsknoten bzw. in unserem konkreten Fall $2^n - 1$ Zwischenkonfigurationen, die dann den Entscheidungsbaum repräsentieren.
Da durch die Zwischenkonfigurationen keine Schleifen oder sonstige zeit- und platzaufwendige Operationen eingeführt werden, verändert sich die Laufzeitklasse nicht, so dass die neue NTM auch $O(t(n))$ zeit- und $O(s(n))$-platzbeschränkt ist.

Aufgabe 27 (5 Punkte)
Beschreiben Sie informal die Arbeitsweise einer NTM, die das Travelling Salesperson Problem (TSP) entscheidet.
Das Travelling Salesperson Problem überprüft, ob sich in einem gegebenen Graphen G mit einer gegebenen Gewichtsfunktion c ein oder mehrere Hamiltonkreise befinden, deren Gewicht kleiner oder gleich einem gegebenen Wert k ist.
Eine NTM, die dieses Problem entscheidet rät somit zuerst eine Menge von Kanten, wobei anschließend überprüft werden muss, ob die geratenen Kanten einen Hamiltonkreis bilden und die Summe der Kantengewichte $\leq k$ ist. Ein entsprechender Algorithmus für die Verifikation müsste in einem ersten Schritt prüfen, ob die Summe der Kantengewichte $\leq k$ ist und dann überprüfen, ob mit den geratenen Kanten ein Hamiltonkreis gebildet werden kann, so dass alle Knoten genau einmal vorkommen. Dieses ist am einfachsten zu überprüfen, indem die Kantenmenge abgelaufen wird und dabei geguckt wird, ob ein bereits besuchter Knoten abermals besucht wird, bzw. ob es am Ende Knoten gibt, die noch nicht besucht wurden.

Aufgabe 28 (5 Punkte)

a) Warum sollten Sie zunächst in Jubel ausbrechen?
Wir sollten zunächst in Jubel ausbrechen, da unser Hauptkonkurrent sich für 10 Millionen Euro eine Rechneranlage erworben hat, welche das Problem nicht sehr viel schneller berechnen kann, als der bisher vorhandene Rechner:
Das Problem ist von der Komplexität $O\left(|x|^2\right)$, wie sich unschwer anhand der nachgewiesenen Laufzeit A feststellen lässt.
Wenn sich nun der Konkurrent einen Rechner gekauft hat, der doppelt so schnell arbeitet wie der bereits vorhandene, so bedeutet dies, dass diese in der gleichen Zeit m nun ein Problem von $\sqrt{2} \cdot |x|$ bearbeiten können. Da $\sqrt{2} \approx 1,41$ folgt somit, dass der der Konkurrent in der gleichen Zeit nun $1,41$ mal so große Probleme in der gleichen Zeit schaffen kann. Da es sich aber bei $|x| = 22$ um ein $\frac{22^2}{11^2} = 4$ mal so zeitaufwendiges Problem handelt, ist der Zeitgewinn des Konkurrenten bei der Bearbeitung des Problems nicht sehr groß, allerdings haben wir weiterhin zehn Millionen Euro in der Kasse, der Konkurrent aber nicht.

b) Man kann davon ausgehen, dass sich die Geschwindigkeit von Rechnern jedes Jahr verdoppelt, bzw. der Preis für den selben Rechner sich pro Jahr halbiert. Wann sollten Sie sich einen neuen Rechner kaufen?
Da es sich um ein 4 mal so zeitaufwendiges Problem wie bisher handelt, sollte ein Rechnerneukauf erst anstehen, wenn man das Problem in der gleichen Zeit wie bisher das kleine Problem bearbeiten kann.
Da das Problem viermal so zeitaufwendig ist, muss also gelten:
$\sqrt{x} = 4 \Rightarrow x = 16$. Wir brauchen also 16 mal schnellere Rechner als bisher. Da sich die Leistung der Rechner jedes Jahr verdoppelt, wäre eine Neuanschaffung also in 4 Jahren sinnvoll, da $2^4 = 16$.

c) Warum sollte man sich in der Zwischenzeit bis zum Rechnerkauf Gedanken über Komplexität machen? Ihre Aufgabe ist es nun, so vorzugehen, dass Sie ihren Konkurrenten ein für alle mal loswerden und dabei möglichst viel Geld in der eignen Firma behalten. Wie macht man das? (5 Zusatzpunkte)