Berechenbarkeit und Komplexität

Vorlesung #3: Mehrband- und Universelle Turingmaschine

Prof. Berthold Vöcking
präsentiert durch Prof. Joost-Pieter Katoen

24. Oktober 2008

Hörsaaländerung

Momentane Planung

Vorlesung Freitags 12:00 - 13:30 zweiwöchtentlich
Hörsaal: Eph

Ab 07.11.2008

Vorlesung Freitags 12:00 - 13:30 zweiwöchtentlich
Hörsaal: Hörsaal 1 im Hauptgebude

Trick 1: Speicher im Zustandsraum

Für beliebiges festes $k \in \mathbb{N}$, können wir k Zeichen unseres Bandalphabets im Zustand abspeichern, indem wir den Zustandsraum um den Faktor $|\Gamma|^k$ vergößern, d.h. wir setzen

$$Q_{neu} := Q \times \Gamma^k \quad .$$

Trick 2: Mehrspurmaschinen

Bei einer k-spurigen TM handelt es sich um eine TM, bei der das Band in k sogenannte Spuren eingeteilt ist, d.h. in jeder Bandzelle stehen k Zeichen, die der Kopf gleichzeitig einlesen kann. Das können wir erreichen, indem wir das Bandalphabet um k-dimensionale Vektoren erweitern, z.B.

$$\Gamma_{neu} := \Sigma \cup \Gamma^k \quad .$$

Beispiel: Addition mittels 3-spuriger TM

Die Verwendung einer mehrspurigen TM erlaubt es häufig, Algorithmen einfacher zu beschreiben.

Wir verdeutlichen dies am Beispiel der Addition. Aus der Eingabe $bin(i_1)\#bin(i_2)$ für $i_1, i_2 \in \mathbb{N}$ soll $bin(i_1 + i_2)$ berechnet werden.
Wir verwenden eine 3-spurige TM mit den Alphabeten
$\Sigma = \{0, 1, \#\}$ und

$$\Gamma = \left\{ 0, 1, \#, \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \ldots, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, B \right\} .$$

Beispiel: Addition mittels 3-spuriger TM

- *Schritt 1:* Transformation in Spurendarstellung: Schiebe die Eingabe so zusammen, dass die Binärkodierungen von i_1 und i_2 in der ersten und zweiten Spur rechtsbündig übereinander stehen. Aus der Eingabe 0011#0110 wird beispielsweise

$$B^* \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} B^* \ .$$

- *Schritt 2:* Addition nach der Schulmethode indem der Kopf das Band von rechts nach links durchläuft. Überträge werden im Zustand gespeichert. Als Ergebnis auf Spur 3 ergibt sich

$$B^* \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} B^* \ .$$

- *Schritt 3:* Rücktransformation von Spur 3 ins Einspur-Format: Ausgabe 1001.

Techniken zur Programmierung von TMen

Standardtechniken aus der Programmierung können auch auf TMen implementiert werden.

- *Schleifen* haben wir bereits in Beispiel 2.1.3 gesehen.
- *Variablen* können realisiert werden, indem wir pro Variable eine Spur reservieren.
- *Felder (Arrays)* können ebenfalls auf einer Spur abgespeichert werden.
- *Unterprogramme* können implementiert werden indem wir eine Spur des Bandes als Prozedurstack verwenden.

Basierend auf diesen Techniken können wir auch auf bekannte Algorithmen z.B. zum Sortieren von Daten zurückgreifen.

k-Band TM

Eine k-Band-TM ist eine Verallgemeinerung der Turingmaschine und verfügt über k Arbeitsbänder mit jeweils einen unabhängigen Kopf. Die Zustandsübergangfunktion ist entsprechend von der Form

$$\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{L, R, N\}^k \quad .$$

Band 1 fungiert als Ein-/Ausgabeband wie bei der (1-Band) TM. Die Bänder $2, \ldots, k$ sind initial mit B^* beschrieben.

Simulation k-Band TM durch 1-Band TM

Satz:

Eine k-Band TM M, die mit Rechenzeit $t(n)$ und Platz $s(n)$ auskommt, kann von einer (1-Band) TM M' mit Zeitbedarf $O(t^2(n))$ und Platzbedarf $O(s(n))$ simuliert werden.

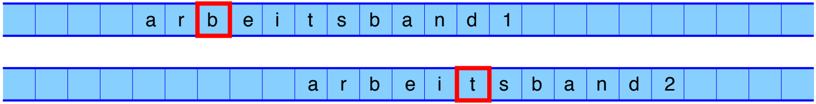
Beweis: Die TM M' verwendet $2k$ Spuren. Nach Simulation des t-ten Schrittes für $0 \leq t \leq t(n)$ gilt

- Die ungeraden Spuren $1, 3, \ldots, 2k-1$ enthalten den Inhalt der Bänder $1, \ldots, k$ von M.
- Auf den geraden Spuren $1, 2, \ldots, 2k$ sind die Kopfposition auf diesen Bändern mit dem Zeichen $\#$ markiert.

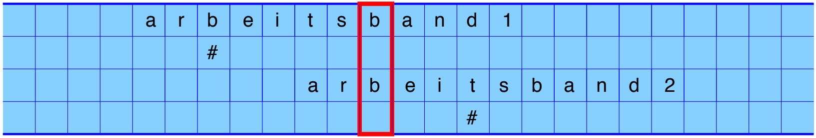
Diese Initialisierung der Spuren ist in Zeit $O(1)$ möglich.

Simulation k-Band TM durch 1-Band-TM – Illustration

simulierte 2–Band–TM M



simulierende 4–spurige TM M' (zu Beginn eines Simulationsschrittes)



Jeder Rechenschritt von M wird durch M' wie folgt simuliert.

- Am Anfang stehe der Kopf von M' auf dem linkesten $\#$ und M' kenne den Zustand von M.
- Der Kopf von M' läuft nach rechts bis zum rechtesten $\#$, wobei die k Zeichen an den mit $\#$ markierten Spurpositionen im Zustand abgespeichert werden.
- Am rechtesten $\#$-Zeichen angekommen kann M' die Übergangsfunktion von M auswerten und kennt den neuen Zustand von M sowie die erforderlichen Übergänge auf den k Bändern.
- Nun läuft der Kopf von M' zurück, verändert dabei die Bandinschriften an den mit $\#$ markierten Stellen und verschiebt, falls erforderlich, auch die $\#$-Markierungen um eine Position nach links oder rechts.

Laufzeitanalyse:

Wieviele Bandpositionen können zwischen den linkesten und dem rechtesten # liegen?

Nach t Schritten können diese Markierungen höchstens $2t$ Positionen auseinanderliegen.

Also ist der Abstand zwischen diesen Zeichen und somit auch die Laufzeit zur Simulation eines Schrittes durch $O(t(n))$ beschränkt.

Insgesamt ergibt das zur Simulation von $t(n)$ Schritten eine Laufzeitschranke von $O(t(n)^2)$. □

Special versus General Purpose Rechner

- Bisher haben wir für jedes Problem eine eigene TM entworfen, einen *special purpose* Rechner.
- Real existierende Maschinen sind jedoch programmierbare *general purpose* Rechner.
- Wir konstruieren jetzt eine programmierbare Variante der TM, die sogenannte *universelle TM*.

Verhaltensbeschreibung der universellen TM

- Das Programm der universellen TM U ist die Kodierung einer beliebigen TM M.
- Diese Kodierung heißt die *Gödelnummer* von M und wird mit $\langle M \rangle$ bezeichnet.
- Als Eingabe erhält U einen String der Form $\langle M \rangle w$ bestehend aus der Gödelnummer $\langle M \rangle$ und einem beliebigen Wort w.
- Die universelle TM simuliert das Verhalten der TM M auf der Eingabe w.
- Bei inkorrekter Eingabe (d.h. die Eingabe beginnt nicht mit einer Gödelnummer) gibt U eine Fehlermeldung aus.

Gödelnummern

Definition

Mit *Gödelnummern* bezeichnet man die eindeutige prefixfreie Kodierung von TM über einem festen Alphabet.

- O.B.d.A. gehen wir von Binären Kodierungen über dem Alphabet $\{0, 1\}$ aus.
- *Prefixfrei* bedeutet, dass keine Gödelnummer Prefix (Anfangsteilwort) einer anderen Gödelnummer sein darf.
- Prefixfreiheit können wir beispielsweise erreichen indem
  - alle Gödelnummern auf 111 enden und ansonsten der Teilstring 111 nicht in der Kodierung vorkommt, oder alternativ
  - alle Gödelnummern mit 111 beginnen und auf 111 enden und ansonsten der Teilstring 111 nicht in der Kodierung vorkommt.

Mögliche Realisierung von Gödelnummern

Wir zeigen, wie eine geeignete Kodierung von TM aussehen könnte.

O.B.d.A. beschnken wir uns auf TM der folgenden Form:

- Sei $Q = \{q_1, \ldots, q_t\}$.
- Der Anfangszustand sei q_1 und der Endzustand q_2.
- O.B.d.A. sei $\Gamma = \{0, 1, B\}$. Wir nummerieren das Alphabet durch indem wir $X_1 = 0$, $X_2 = 1$ und $X_3 = B$ setzen.
- Auch die möglichen Kopfbewegungen nummerieren wir indem wir $D_1 = L$, $D_2 = N$ und $D_3 = R$ setzen.

Zur Beschreibung von TM dieser Form müssen wir nur die Übergangsfunktion als Binärstring kodieren.

Mögliche Realisierung von Gödelnummern

Kodierung der Übergangsfunktion:

- Der Übergang $\delta(q_i, X_j) = (q_k, X_\ell, D_m)$ wird kodiert durch den Binärstring
$$0^i 1 0^j 1 0^k 1 0^\ell 1 0^m \quad .$$

- Die Kodierung des $j$ten Übergangs bezeichnen wir mit $code(j)$.
- Die Gödelnummer einer TM M mit s vielen Übergängen ist dann
$$\langle M \rangle \;=\; 111\, code(1)\, 11\, code(2)\, 11 \ldots 11\, code(s)\, 111 \quad .$$

Mögliche Implementierung der universellen TM

Als Eingabe erhält die universelle TM U ein Wort der Form $\langle M \rangle w$ für beliebiges $w \in \{0,1\}^*$.

Wir implementieren U zunächst in Form einer 3-Band TM:

- Band 1 von U simuliert das Band der TM M.
- Band 2 von U enthält die Gödelnummer von M.
- Auf Band 3 speichert U den jeweils aktuellen Zustand von M.

Mögliche Implementierung der universellen TM

Initialisierung:

- U überprüft, ob die Eingabe eine korrekte Gödelnummer enthält. Falls nein, Fehlerausgabe.
- U kopiert die Gödelnummer auf Band 2 und schreibt die Binärkodierung des Anfangszustands auf Band 3.
- U bereitet Band 1 so vor, dass es nur das Wort w enthält. Der Kopf steht unter dem ersten Zeichen von w.

Laufzeit? – Die Laufzeit ist $O(1)$, wobei wir die Kodierungslänge von M als Konstante ansehen.

Mögliche Implementierung der universellen TM

Simulierung eines Schritts von M:

U sucht zu dem Zeichen an der Kopfposition aus Band 1 und dem Zustand auf Band 3 die Kodierung des entsprechenden Übergangs von M auf Band 2.

Wie in der Übergangsfunktion beschrieben

- aktualisiert U die Inschrift auf Band 1,
- bewegt U den Kopf auf Band 1, und
- verndert U den auf Band 3 abgespeicherten Zustand von M.

Laufzeit eines Schrittes: $O(1)$.

Das bedeutet U simuliert M mit konstantem Zeitverlust!

Mögliche Implementierung der universellen TM

Können wir dieses Ergebnis auch mit einer (1-Band) TM erreichen?

Natürlich können wir die beschriebene 3-Band TM auf der 1-Band TM mit mehreren Spuren simulieren.

Aber bei Verwendung dieser Simulation handeln wir uns einen quadratischen Zeitverlust ein

Wir erhalten eine **universelle 1-Band TM mit konstantem Zeitverlust**, wenn wir die Gödelnummer auf Spur 2 und den Zustand auf Spur 3 mit dem Kopf der TM M mitführen.