

Berechenbarkeit und Komplexitätstheorie

Wintersemester 2021/2022

Aufgabenblatt 1

Abgabe: 12. November 2021 um 12 Uhr

Aufgabe 1.1 (2 + 3 Punkte)

Gegeben ist eine Turingmaschine $M = (S, \Sigma, \Gamma, \delta, z_0, \square, F)$ mit Zustandsmenge $S = \{z_0, z_1, z_2, z_e\}$, Eingabealphabet $\Sigma = \{0, 1\}$, Bandalphabet $\Gamma = \{0, 1, \square\}$, Blanksymbol $\square$, Startzustand z_0 , Endzustandsmenge $F = \{z_e\}$ und Zustandsüberföhrungsfunktion δ mit:

$\delta(z, w)$	0	1	$\square$
z_0	$(z_0, 0, R)$	$(z_0, 1, R)$	$(z_1, 0, R)$
z_1	$(z_1, 0, L)$	$(z_1, 0, L)$	$(z_2, 1, L)$
z_2	$(z_2, 0, L)$	$(z_2, 1, L)$	$(z_e, \square, R)$

- Geben Sie die Folge der Konfigurationen an, die M bei Eingabe 101 durchläuft.
- Welche Funktion $s : \{0, 1\}^* \rightarrow \{0, 1\}^*$ und welche Funktion $f : \mathbb{N} \rightarrow \mathbb{N}$ berechnet M ?

Lösung

(Die TM sucht den rechten Rand der Eingabe und hängt 01 hinter das Wort. Dann geht sie zurück an den linken Rand des Worts.)

- $z_0 101 \vdash 1 z_0 01 \vdash 10 z_0 1 \vdash 101 z_0 \square \vdash 1010 z_1 \square \vdash 101 z_2 01 \vdash 10 z_2 101 \vdash 1 z_2 0101 \vdash z_2 10101 \vdash z_2 \square 10101 \vdash z_e 10101$
- Die Wortfunktion ist $s : \{0, 1\}^* \rightarrow \{0, 1\}^*$, $w \mapsto w \cdot 01$. Die Funktion auf den natürlichen Zahlen ist $f : \mathbb{N} \rightarrow \mathbb{N}$, $n \mapsto 2 \cdot (2 \cdot n) + 1$, was man zusammenfassen kann zu $n \mapsto 4n + 1$. (Falls man die Eingabe als Zahl zu einer anderen Basis b interpretiert, dann ergibt sich als Funktion $n \mapsto b^2 \cdot n + 1$.)

Aufgabe 1.2 (2 + 3 Punkte)

- Zeigen Sie, dass man zu jeder Turingmaschine eine neue Turingmaschine konstruieren kann, die in jedem Schritt den Schreib-Lese-Kopf entweder eine Position nach links oder eine Position nach rechts bewegt. (D. h. der Kopf darf in keinem Schritt stehen bleiben – die Kopfbewegungen sind aus $\{L, R\}$.)
- Zeigen Sie, dass man zu jeder Turingmaschine eine neue Turingmaschine konstruieren kann, die keine Felder links der *initialen Eingabe* benutzt. D. h. das Arbeitsband ist nur rechtsseitig unendlich; die linke Seite besitzt einen Rand.

(Hinweis: Bei beiden Teilaufgaben reicht jeweils eine Beweisskizze – es sind keine formalen Beweise gefordert. Überlegen Sie sich bei der ersten Teilaufgabe, wie Sie das “Stehenbleiben” mithilfe von zusätzlichen Zuständen umgehen können. Überlegen Sie sich bei der zweiten Teilaufgabe, wie Sie sicherstellen können, dass die TM links vom aktuellen Wort immer ein freies Feld für weitere Symbole hat.)

Lösung

- a) Hier muss aus einer beliebigen TM eine neue TM konstruiert werden, die das “Stehenbleiben” der alten TM ersetzt. Die einfachste Variante ist es, dass die neue TM statt stehen zu bleiben einen Schritt nach links und danach wieder nach rechts geht. Sie muss sich dann nur merken, in welchem Zustand sie nach der Rechtsbewegung sein müsste. Etwas formaler: Man ersetzt jeden Übergang $\delta(z_i, w) = (z_j, v, N)$ bei dem sich der Kopf nicht bewegt durch die beiden Übergänge $\delta(z_i, w) = (\hat{z}_j, v, L)$ und $\delta(\hat{z}_j, w) = (z_j, w, R)$.
- b) Die einfache (wenn auch zeitintensive) Methode ist es, jedes Mal, wenn der Kopf den linken Rand erreicht, das Wort auf dem Arbeitsband nach rechts zu schieben. Damit stellt man sicher, dass der Bandinhalt nur nach rechts wächst. Eine andere Idee, welche die Laufzeit nicht verändert, ist das Band der einseitig beschränkten TM in zwei Spuren aufzuteilen. Hierzu erweitert man das Arbeitsalphabet Γ zu Γ^2 ; jedes Symbol im neuen Arbeitsalphabet ist ein Tupel (w, v) (bzw. $\begin{pmatrix} w \\ v \end{pmatrix}$) mit $w, v \in \Gamma$. Die ersten Komponenten entsprechen der oberen und die zweiten Komponenten der unteren Spur. Dann interpretiert man die obere Spur als die rechte und die untere Spur als die linke Hälfte des beidseitig unbeschränkten Bandes der ursprünglichen TM. Wenn die ursprüngliche TM von der linken in die rechte Hälfte wechselt (oder umgekehrt), dann muss die neue TM die entsprechende Spur betrachten. Das kann man mithilfe der Zustände steuern. Zu jedem Zustand z_i führt man einen weiteren Zustand z'_i ein. Die nicht-markierten Zustände werden dazu genutzt, dass die TM die obere Spur betrachtet und die markierten Zustände dazu, dass die TM die untere Spur betrachtet. Je nach Definition der einseitig beschränkten TM muss man eventuell das linkeste Feld explizit als Rand markieren.

Aufgabe 1.3 (4 + 1 Punkte)

Als Erweiterung der in der Vorlesung eingeführten Turingmaschine kann man *Zweiband-Turingmaschinen* (ZTM) definieren. Eine ZTM besitzt zwei Arbeitsbänder mit jeweils eigenen Schreib-Lese-Köpfen. Formal ist eine ZTM auch ein 7-Tupel $(S, \Sigma, \Gamma, \delta, s_0, \square, F)$. Der Unterschied ist, dass die Übergangsfunktion alle Arbeitsbänder betrachtet, also als $\delta: S \times \Gamma \times \Gamma \rightarrow S \times \Gamma \times \Gamma \times \{L, N, R\} \times \{L, N, R\}$ definiert ist.¹

- a) Konstruieren Sie eine Zweiband-Turingmaschine, welche als Eingabe zwei Zahlen $x, y \in \mathbb{N}$ bekommt und ausgibt, ob beide Zahlen gleich sind, ob die erste Zahl kleiner als die zweite Zahl ist, oder ob sie größer als die zweite Zahl ist. Formal soll die ZTM also die folgende Funktion berechnen:

$$f: \mathbb{N}^2 \rightarrow \{-1, 0, 1\}, (x, y) \mapsto \begin{cases} -1, & x < y \\ 0, & x = y \\ 1, & x > y \end{cases}$$

Nehmen Sie ruhig an, dass die beiden Zahlen binär kodiert sind und durch ein Trennzeichen getrennt auf dem ersten Arbeitsband als Eingabe stehen.

- b) Welchen Vorteil bietet eine ZTM gegenüber einer TM?

Lösung

¹Man kann die Definition allgemein auf k Bänder erweitern.

- a) Grob kann man die Arbeitsweise der TM wie folgt beschreiben: Die TM sucht auf dem ersten Band das Trennzeichen bzw. die zweite Zahl. Diese Zahl überträgt sie dann Ziffer für Ziffer auf das zweite Band. Dann geht sie auf dem ersten Band wieder nach links, bis zur letzten Ziffer der ersten Zahl (und löscht dabei die Ziffern der zweiten Zahl und das Trennzeichen auf dem ersten Band.) Jetzt stehen beide Köpfe über den letzten Ziffern beider Zahlen. Nun kann die TM auf beiden Bändern die Köpfe Schritt für Schritt nach links bewegen, bis unter einem der Köpfe ein Blank-Symbol gelesen wird. Wird unter dem anderen Kopf *kein* Blank gelesen, dann ist die zugehörige Zahl länger und somit größer. Falls unter dem anderen Kopf jedoch auch ein Blank gelesen wird, sind beide Zahlen gleich lang und müssen genauer betrachtet werden. Hierzu wandert die Maschine von links nach rechts, Ziffer für Ziffer, über die beiden Zahlen, bis sie eine Stelle findet, in der sich die Ziffern unterscheiden. Die Zahl, bei der die größere Ziffer gelesen wird (bei binärer Kodierung also die 1), ist dann die größere Zahl. Falls die TM am Ende beider Zahlen ankommt und keine unterschiedlichen Ziffern findet, waren beide Zahlen gleich groß. Die TM muss dann entsprechend die Bänder bereinigen und -1 , 0 oder 1 auf das erste Band schreiben.

Die Übergangsfunktion δ ist (etwas) genauer in der Tabelle auf der nachfolgenden Seite angegeben. Startzustand ist z_s , Finalzustände sind $\{z_{-1}, z_0, z_1\}$. Die anderen Parameter (Blanksymbol, Zustandsmenge, Alphabete) ergeben sich aus der Tabelle. Der Inhalt beider Bänder ist als Tupel angegeben; die Tupel, die das Trennsymbol ($\#$) als zweite Komponente beinhalten, stehen nicht in der Tabelle. Da die Eingabe auf dem ersten Band steht und die TM selbst das Trennsymbol nicht auf das zweite Band schreibt, muss keine dieser Kombinationen betrachtet werden. Weiter sind in der Tabelle einige Übergänge mit “ $-$ ” markiert, da diese ebenfalls nicht vorkommen können. An dieser Stelle könnte man auch Übergänge einführen, die die TM in einen Zustand überführen, der anzeigt, dass die Eingabe nicht gültig war. An einigen Stellen steht als Funktionswert nur “ z'_{-1} ” (bzw. “ z'_1 ”) – hier wollte ich mir die Arbeit sparen, zu überlegen, in welche Richtung die Köpfe sinnvollerweise gehen und wie sie die Symbole löschen. (Die Arbeitsweise der TM in diesen beiden Zustände ist in der Tabelle selbst auch nur grob beschrieben.)

Die von mir angegebene TM ist nur eine von vielen. Es gibt mit Sicherheit noch effizientere Möglichkeiten, die Zahlen auf Gleichheit zu überprüfen.

- b) Der Vorteil einer Mehrband-TM ist, dass man auf den verschiedenen Bändern Arbeitsschritte gleichzeitig machen kann. In der vorigen Aufgabe bspw. konnte man so in einem Schritt zwei Symbole vergleichen. Auf einer üblichen TM müsste man dazu das Zeichen lesen, sich das Zeichen merken und dann das andere Zeichen suchen. Damit hat man einen (linearen) Mehraufwand. (Allerdings kann eine Mehrband-TM *nicht* mehr berechnen als eine Einband-TM; man kann jede Mehrband-TM durch eine Einband-TM simulieren.)

	$\square, \square$	$\square, 0$	$\square, 1$	$\#, \square$	$\#, 0$	$\#, 1$	$0, \square$	$0, 0$	$0, 1$	$1, \square$	$1, 0$	$1, 1$
z_s	–	–	–	$(z_c, \#, \square, R, N)$	–	–	$(z_s, 0, \square, R, N)$	–	–	$(z_s, 1, \square, R, N)$	–	–
z_c	$(z_b, \square, \square, L, L)$	–	–	–	–	–	$(z_c, 0, 0, R, R)$	–	–	$(z_c, 1, 1, R, R)$	–	–
z_b	–	–	–	$(z_l, \square, \square, L, N)$	$(z_l, \square, 0, L, N)$	$(z_l, \square, 1, L, N)$	$(z_b, \square, \square, L, N)$	$(z_b, \square, 0, L, N)$	$(z_b, \square, 1, L, N)$	$(z_b, \square, \square, L, N)$	$(z_b, \square, 0, L, N)$	$(z_b, \square, 1, L, N)$
z_l	$(z_g, \square, \square, R, R)$	–	–	–	–	–	–	$(z_l, 0, 0, L, L)$	$(z_l, 0, 1, L, L)$	$(z_l, \square, \square, L, L)$	$(z_l, 1, 0, L, L)$	$(z_l, 1, 1, L, L)$
z_g	$(z_0, 0, \square, N, N)$	–	–	–	–	–	–	$(z_g, \square, \square, R, R)$	$(z_g, 0, 1, L, L)$	$(z_g, \square, \square, L, L)$	$(z_g, 1, 0, L, L)$	$(z_g, \square, \square, R, R)$
z'_{-1}									$(z'_{-1}, \square, \square, R, R)$		$(z'_{-1}, 1, 0, L, L)$	
z'_1									$(z'_1, \square, \square, R, R)$		$(z'_1, 1, 0, L, L)$	

lösche alle Symbole auf beiden Bändern, schreibe -1 auf das erste Band und gehe in Zustand z_{-1}
lösche alle Symbole auf beiden Bändern, schreibe 1 auf das erste Band und gehe in Zustand z_1