Grundlagen der Theoretischen Informatik

Turingmaschinen und rekursiv aufzählbare Sprachen (III)

27.06.2018

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de

Übersicht

1. Motivation
2. Terminologie
3. Endliche Automaten und reguläre Sprachen
4. Kellerautomaten und kontextfreie Sprachen
5. Turingmaschinen und rekursiv aufzählbare Sprachen
6. Berechenbarkeit, (Un-)Entscheidbarkeit
7. Komplexitätsklassen P und NP

TM-Flussdiagramme

Folgende Elemente stehen zur Verfügung:

- L: eine DTM, die nach dem Starten ein Feld nach links geht und danach hält.
- R: eine DTM, die nach dem Starten ein Feld nach rechts geht und danach hält.
- a: TM, die a auf das Band schreibt und danach hält.
- Startschritt: mit einer Pfeilspitze $>$ bezeichnet

TM-Flussdiagramme

- $\mathcal{M}_1 \longrightarrow \mathcal{M}_2$ oder abgekürzt $\mathcal{M}_1\mathcal{M}_2$ (falls $\mathcal{M}_1$, $\mathcal{M}_2$ die Flußdiagramme zweier DTM sind): eine DTM die zuerst wie $\mathcal{M}_1$ arbeitet und dann, falls $\mathcal{M}_1$ hält, wie $\mathcal{M}_2$ weiterarbeitet.

- $\mathcal{M}_1 \xrightarrow{a} \mathcal{M}_2$: $\mathcal{M}_2$ wird nur dann aufgeführt, wenn nach der Beendigung von $\mathcal{M}_1$ der aktuelle Bandbuchstabe a ist.

- Sind $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_n$ Turing-Maschinen, $a_i \in \Sigma$ für $1 \leq i \leq n$, so ist

$$\begin{array}{ccl} & \mathcal{M}_1 & \\ & \uparrow^{a_1} & \\ > & \mathcal{M}_0 & \stackrel{a_2}{\to} \mathcal{M}_2 \\ & \downarrow^{a_n} & \quad \ddots \\ & \mathcal{M}_n & \end{array}$$

die Turing-Maschine, die zuerst wie $\mathcal{M}_0$ arbeitet und dann, falls $\mathcal{M}_0$ mit dem Buchstaben a_i auf dem Arbeitsfeld hält, wie $\mathcal{M}_i$ weiterarbeitet.

TM-Flussdiagramme

$\boldsymbol{\sigma}$ — eine Schreibabkürzung für einen beliebigen Buchstaben aus Σ.

- Die Maschine $\mathcal{M} \xrightarrow{\sigma} \ldots \sigma$ zum Beispiel ist eine Abkürzung für

$$\begin{array}{ll} & \ldots a_1 \\ & \uparrow^{a_1} \\ > & \mathcal{M} \xrightarrow{a_2} \ldots a_2 \\ & \downarrow^{a_n} \qquad \ddots \\ & \ldots a_n \end{array}$$

  falls $\Sigma = \{a_1, \ldots, a_n\}$ ist.

Weitere Schreibabkürzungen sind:

- $\mathcal{M}_1 \xrightarrow{\sigma \neq a} \mathcal{M}_2$: $\mathcal{M}_2$ wird nur dann aufgeführt, wenn $\mathcal{M}_1$ hält und Lesekopf auf einem Buchstaben, der nicht a ist positioniert ist.

- $\boldsymbol{\mathcal{M}}_1 \xrightarrow{a,b} \boldsymbol{\mathcal{M}}_2$ falls nach der Ausführung von $\mathcal{M}_1$ sowohl für den Bandbuchstaben a als auch für b nach $\mathcal{M}_2$ verzweigt werden soll.

Beispiele: Letzte Vorlesung

Beispiel:

Die DTM $\mathcal{M}^+ = (\{s, q_1, q_2, q_3, q_4\}, \{|, \#\}, \delta, s)$ addiert zwei natürliche Zahlen in Unärdarstellung.

Sie rechnet

$$s, \#|^n\#|^m\underline{\#} \ \vdash^*_{\mathcal{M}^+} \ h, \#|^{n+m}\underline{\#}$$

Der Trick: Sie löscht den letzten Strich von $|^m$ und schreibt ihn in den Zwischenraum zwischen $|^n$ und $|^m$.

Das Flußdiagramm zur DTM:

$$\begin{tikzcd}
> L \arrow[r, "|"] \arrow[d, "\#"] & L \arrow[loop above, "|"] \arrow[r, "\#"] & {|}R \arrow[loop above, "|"] \arrow[r, "\#"] & L\# \\
\#
\end{tikzcd}$$

Beispiele: Letzte Vorlesung

Die folgenden Turing-Maschinen werden wir als Bestandteile von komplexeren Turing-Maschinen später noch häufig benutzen.

Beispiel: ($R_{\#}$)

$R_{\#}$ bewegt nur den Kopf, ohne zu schreiben:

- Sie geht mindestens einmal nach rechts.
- Dann geht sie solange weiter nach rechts, bis sie ein $\#$ liest.

Die folgende Turing-Maschine $R_{\#}$ läuft zum ersten Blank rechts von der momentanen Position.

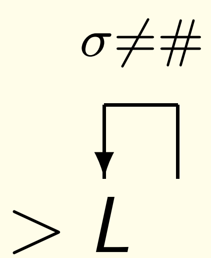
$$\sigma \neq \#$$

$$> R$$

Beispiele: Letzte Vorlesung

Beispiel: ($L_\#$) Analog funktioniert die DTM $L_\#$:

- Sie geht mindestens einmal nach links.
- Dann geht sie solange weiter nach links, bis sie ein $\#$ liest.



TM-Flussdiagramme

Beispiel: ($\mathscr{C}$) Die folgende DTM $\mathscr{C}$ erhält als Eingabe einen String Einsen.

Sie rechnet so:

- Sie bewegt sich nach links auf das Blank vor das Eingabewort,
- geht das Eingabewort von links nach rechts durch,
- merkt sich jeweils ein Zeichen σ von w, markiert die aktuelle Position, indem sie σ mit $\#$ überschreibt,
- und kopiert das Zeichen σ.
- Sie verwendet dabei die Maschinen $L_\#$ und $R_\#$, die wir schon definiert haben.

$$\begin{tikzcd} > L_\# R \arrow[r, "\sigma \neq \#"] \arrow[d, "\#"] & \# R_\# R_\# \sigma L_\# L_\# \sigma \arrow[l, bend right] \\ R_\# \end{tikzcd}$$

TM-Flussdiagramme

Beispiel: Die DTM $\mathscr{S}_R$ bewirkt eine “Verschiebung nach rechts”, das heißt, wenn $\mathscr{S}_R$ das Alphabet Σ besitzt, rechnet sie

$$s, w_1\underline{\#}w_2\#w_3 \vdash^*_{\mathscr{S}_R} h, w_1\#\underline{\#}w_2w_3$$

für alle Wörter $w_1, w_3 \in \Sigma^*$ und $w_2 \in (\Sigma - \{\#\})^*$. (Entgegen der sonstigen Konvention startet sie zwischen zwei Eingabewörtern.)
Sie arbeitet so:

$$\mathscr{S}_R: \quad \begin{tikzcd} > R_\# L \arrow[r, "\sigma\neq\#"] \arrow[d, "\#"] & R\sigma L \arrow[l, bend right] \\ R\# \end{tikzcd}$$

TM-Flussdiagramme

Beispiel: Dazu invers arbeitet die Maschine $\mathscr{S}_L$, die einen “shift nach links” bewirkt. Sie rechnet

$$s, w_1\#w_2\underline{\#}w_3 \vdash^*_{\mathscr{S}_L} h, w_1w_2\underline{\#}\#w_3$$

für alle $w_1, w_3 \in \Sigma^*, w_2 \in (\Sigma - \{\#\})^*$. Sie ist definiert als

$$\begin{tikzcd} \mathscr{S}_L: & > L_\# R \arrow[r, "\sigma\neq\#"] \arrow[d, "\#"] & L\sigma R \arrow[l, bend right] \\ & L\# & \end{tikzcd}$$

Varianten von Turing-Maschinen

Variationen von Turing-Maschinen

Standard-DTM

Die Turing-Maschine, die wir bisher kennen, ...

- ist determiniert
- hat ein einseitig unbeschränktes Band (**Halbband**).

Ab jetzt nennen wir sie auch:

Standard-Turing-Maschine (Standard-DTM oder kurz DTM)

Variationen von Turing-Maschinen

Turing-Maschinen, die nie hängen

Gegeben:

Eine Turing-Maschine $\mathcal{M}$, mit Eingabge $\#w\underline{\#}$

Daraus konstruieren wir eine DTM $\mathcal{M}'$, die

- dasselbe berechnet wie $\mathcal{M}$
- **nie hängt**.

Variationen von Turing-Maschinen

Konstruktion der TM, die nie hängt

Das Bandende ist am Anfang ein Zeichen links vom Eingabewort

DTM $\mathcal{M}'$ rechnet so:

- Sie verschiebt die Eingabe ein Zeichen nach rechts.
- Dann druckt sie ganz links ein **Sonderzeichen** α,
  das das **Bandende** anzeigt.
- Ab dann rechnet sie wie $\mathcal{M}$.
- Aber:
  Wenn sie α erreicht, bleibt sie dort stehen und druckt immer wieder α.

Variationen von Turing-Maschinen

Eigenschaften der nicht-hängenden DTM

- $\mathcal{M}'$ hält für Eingabe w <u>gdw</u> $\mathcal{M}$ hält für Eingabe w.
- $\mathcal{M}'$ hängt nie.
  Wenn $\mathcal{M}$ hängt, rechnet $\mathcal{M}'$ unendlich lang.

O.B.d.A. sollen alle Turing-Maschinen, die wir von jetzt an betrachten, nie hängen.

DTM mit zweiseitig unbeschränktem Band

DTM mit zweiseitig unbeschränktem Band (zw-DTM)

- Die Definition der Maschine bleibt gleich.
- Die Definition der **Konfiguration** ändert sich.
- Sie hat immer noch die Form $q, w\underline{a}u$, aber:
  - w umfasst analog zu u alle Zeichen bis zum letzten nicht-Blank links vom Schreib-/Lesekopf.
  - $w = \epsilon$ bzw. $u = \epsilon$ bedeutet, dass links bzw. rechts vom Schreib-/Lesekopf nur noch Blanks stehen.

DTM mit zweiseitig unbeschränktem Band

Definition (DTM mit zweiseitig unbeschränktem Band, zw-DTM)
Eine **Turing-Maschine mit zweiseitig unbeschränktem Band (zw-DTM)** ist eine DTM, für die die Begriffe der Konfiguration und der Nachfolgekonfiguration wie folgt definiert sind:

Eine **Konfiguration** C einer zw-DTM $\mathcal{M} = (K, \Sigma, \delta, s)$ ist von der Form

$$C = q, w\underline{a}u$$

Dabei ist

- $q \in K \cup \{h\}$ der aktuelle Zustand,
- $w \in (\Sigma - \{\#\})\Sigma^* \cup \{\epsilon\}$ der Bandinhalt links des Kopfes,
- $a \in \Sigma$ das Zeichen unter dem Kopf, und
- $u \in \Sigma^*(\Sigma - \{\#\}) \cup \{\epsilon\}$ der Bandinhalt rechts des Kopfes.

DTM mit zweiseitig unbeschränktem Band

Definition (Forts.)

$C_2 = q_2, w_2\underline{a_2}u_2$ heißt **Nachfolgekonfiguration** von $C_1 = q_1, w_1\underline{a_1}u_1$,

in Zeichen $C_1 \ \vdash_{\mathcal{M}} \ C_2$, falls es einen Übergang $\delta(q_1, a_1) = (q_2, b)$ gibt, mit:

Fall 1: $b \in \Sigma$**.** Dann $w_1 = w_2, u_1 = u_2$ und $a_2 = b$.

Fall 2: $b = L$**.** Für u_2: Wenn $a_1{=}\#$ und $u_1{=}\epsilon$ ist, dann $u_2{=}\epsilon$, sonst $u_2{=}a_1u_1$.
Für a_2 und w_2: Wenn $w_1{=}\epsilon$ ist, dann $w_2{=}\epsilon$ und $a_2{=}\#$; sonst $w_1{=}w_2a_2$.

Fall 3: $b = R$**.** Für w_2: Wenn $a_1{=}\#$ und $w_1{=}\epsilon$ ist, dann $w_2{=}\epsilon$, sonst $w_2{=}w_1a_1$.
Für a_2 und u_2: Wenn $u_1{=}\epsilon$ ist, dann $u_2{=}\epsilon$ und $a_2{=}\#$; ansonsten $u_1{=}a_2u_2$.

DTM mit zweiseitig unbeschränktem Band

Theorem (Simulation von zw-DTM durch DTM)
Zu jeder zw-DTM $\mathcal{M}$, die eine Funktion f berechnet oder eine Sprache L akzeptiert, existiert eine Standard-DTM $\mathcal{M}'$, die ebenfalls f berechnet bzw. L akzeptiert.

DTM mit zweiseitig unbeschränktem Band

Theorem (Simulation von zw-DTM durch DTM)
Zu jeder zw-DTM $\mathcal{M}$, die eine Funktion f berechnet oder eine Sprache L akzeptiert, existiert eine Standard-DTM $\mathcal{M}'$, die ebenfalls f berechnet bzw. L akzeptiert.

Beweis Sei $w = a_1 \dots a_n$ die Eingabe für $\mathcal{M} = (K, \Sigma, \delta, s)$.

Dann sieht das beidseitig unendliche Band zu Beginn der Rechnung so aus:

$$\dots \#\#\# a_1 \dots a_n \underline{\#}\# \dots$$

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) **Idee:**

- $\mathcal{M}$ hat quasi zwei unendlich lange Halbbänder.
- Ziel ist, den Inhalt beider Halbbänder von $\mathcal{M}$ auf einem unterzubringen.
- Dazu: Den Teil des Bandes, der zwei Zeichen links vom Input w beginnt, umklappen:

$$\begin{array}{ll} \text{Spur } 1 & \# \, \# \ldots \# \, \# \\ \text{Spur } 2 & \# \, a_1 \ldots a_n \, \underline{\#} \end{array} \;\ldots$$

- Die DTM $\mathcal{M}'$ hat zwei **Spuren**, d.h. zwei Reihen von Zeichen, die auf demselben Band untergebracht (kodiert) sind.
- Das Bandalphabet von $\mathcal{M}'$ ist $\Sigma' \supseteq \Sigma \times \Sigma$.

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) Sei $\mathcal{M}' = (\ K', \Sigma', \delta', s\ )$. $\mathcal{M}'$ rechnet so:

- $\mathcal{M}'$ legt zunächst eine zweite Spur an,
- simuliert dann die Arbeit von $\mathcal{M}$, und
- transformiert dann das Ergebnis wieder auf nur eine Spur herunter.

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) **Erste Phase der Rechnung:**

$\mathcal{M}'$ rechnet

$$s, \#a_1 \dots a_n\underline{\#} \ \vdash^*_{\mathcal{M}'} \ q, \$ \ \begin{array}{c} \# \, \# \, \dots \, \# \, \# \\ \# \, a_1 \dots a_n \, \underline{\#} \end{array} \# \dots$$

Die zweite Spur wird nur so weit wie nötig angelegt.

Mit dem Symbol \$ markiert $\mathcal{M}'$ das Ende des Halbbands, damit sie nicht hängenbleibt.

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) **Zweite Phase der Rechnung:**

$\mathcal{M}'$ simuliert $\mathcal{M}$.

Dabei muß sie sich immer merken, auf welcher der beiden Spuren sie gerade arbeitet.

Deshalb definieren wir $K' \supseteq K \times \{1, 2\}$.

(q, i) bedeutet, dass die simulierte Maschine $\mathcal{M}$ im Zustand q ist und $\mathcal{M}'$ auf Spur i arbeitet.

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) Für die Simulation von $\mathcal{M}$ durch $\mathcal{M}'$ soll nun gelten:

$\mathcal{M}$ erreicht von $s, \# \vdots \# w \underline{\#}$ aus eine Konfiguration $q, u_1 b \vdots a u_2$

<u>gdw</u>

$\mathcal{M}'$ rechnet $p, \$\begin{smallmatrix}\# & \cdots & \# \\ \# & w & \underline{\#}\end{smallmatrix}\# \vdash^*_{\mathcal{M}'} p', \$\begin{smallmatrix} b & u_1^R & \# \\ a & u_2 & \# \end{smallmatrix} \ldots \begin{smallmatrix}\# \\ \#\end{smallmatrix}\#$

($\vdots$ steht in beiden Konf. zwischen denselben zwei Bandpositionen (an denen das Band “umklappt”))

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) $\mathcal{M}'$ simuliert $\mathcal{M}$ wie folgt:

- Wenn $\mathcal{M}'$ das Zeichen \$ erreicht, wechselt sie die Spur.
- Wenn die simulierte Maschine $\mathcal{M}$ nach rechts (links) geht, geht $\mathcal{M}'$ nach rechts (links) auf Spur 2 und nach links (rechts) auf Spur 1.
- Wenn $\mathcal{M}'$ ein $\#$ erreicht (d.h. sie erreicht den Bandteil, wo noch nicht zwei Spuren angelegt sind), macht sie daraus $\genfrac{}{}{0pt}{}{\#}{\#}$.

Gilt etwa $\delta_{\mathcal{M}}(q, a) = (q', L)$, so muß in $\mathcal{M}$ gelten:

- $\delta_{\mathcal{M}'}\big((q, 2), \genfrac{}{}{0pt}{}{x}{a}\big) = \big((q', 2), L\big)$ für alle möglichen x,
- $\delta_{\mathcal{M}'}\big((q, 1), \genfrac{}{}{0pt}{}{a}{x}\big) = \big((q', 1), R\big)$ (auf der oberen Spur ist der Inhalt des "linken Halbbandes" revers notiert, deshalb muß hier die Laufrichtung entgegengesetzt sein).

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.)

Außerdem gilt immer:

- $\delta_{\mathcal{M}'}\big((q,1),\$\big) = \big((q,2),R\big)$
- $\delta_{\mathcal{M}'}\big((q,2),\$\big) = \big((q,1),R\big)$
  Spurwechsel beim Überschreiten von \$
- $\delta_{\mathcal{M}'}\big((q,i),\#\big) = \big(q,i), \genfrac{}{}{0pt}{}{\#}{\#}\big)$
  Erzeugen eines neuen Doppelspurstücks
- etc.

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.)

Wenn dann $\mathcal{M}$ mit $h, u\underline{\#}$ hält, dann erreicht $\mathcal{M}'$ eine Konfiguration, die eine der folgenden Formen hat:

$$(i) \qquad (h, 1), \$ \genfrac{}{}{0pt}{}{\#}{\#} \ldots \genfrac{}{}{0pt}{}{\overline{\#}}{\#} \genfrac{}{}{0pt}{}{u^R}{\# \cdots \#} \genfrac{}{}{0pt}{}{\#}{\#} \ldots \genfrac{}{}{0pt}{}{\#}{\#} \text{ oder}$$

$$(ii) \qquad (h, 2), \$ \genfrac{}{}{0pt}{}{\#}{\#} \ldots \genfrac{}{}{0pt}{}{\#}{\#} \genfrac{}{}{0pt}{}{\# \cdots \#}{u} \genfrac{}{}{0pt}{}{\#}{\underline{\#}} \ldots \genfrac{}{}{0pt}{}{\#}{\#} \text{ oder}$$

$$(iii) \qquad (h, 2), \$ \genfrac{}{}{0pt}{}{u_1^R}{u_2\underline{\#}} \genfrac{}{}{0pt}{}{\#}{\#} \ldots \genfrac{}{}{0pt}{}{\#}{\#} \text{ mit } u_1 u_2 = u.$$

Bei Konfigurations-Form (iii) kann entweder das u_1^R über das u_2 "hinausragen" oder umgekehrt.

DTM mit zweiseitig unbeschränktem Band

Beweis (Forts.) **Dritte Phase der Rechnung:**

Die Simulation von $\mathcal{M}$ ist abgeschlossen.

$\mathcal{M}'$ muß nun den Bandinhalt von zwei Spuren auf nur eine heruntertransformieren, um danach die Konfiguration $h, \#u\underline{\#}$ zu erreichen.

- $\mathcal{M}'$ macht zunächst alle $\substack{\#\\\#}$ rechts vom beschriebenen Bandteil zu $\#$. Für Fall (i) und (ii) löscht sie die $\substack{\#\\\#}$ links von u^R bzw. u.
- Für Fall (iii) schiebt $\mathcal{M}'$ dann die untere Spur nach links, bis sie eine Konfiguration $q, \substack{u_1^R\\\#\ldots\#}\, u_2\underline{\#}$ erreicht.
- Für Fall (i) und (iii) muß $\mathcal{M}'$ jetzt u_1^R bzw. u^R auf nur eine Spur transformieren und zugleich invertieren, sie muß also für den allgemeineren Fall (iii) $q, \$\substack{u_1^R\\\#\ldots\#}\, u_2\underline{\#} \vdash^*_{\mathcal{M}'} q', \$u_1u_2\underline{\#}$ rechnen.
- Danach muß $\mathcal{M}'$ nur noch das \$ links löschen und nach rechts neben u laufen.

DTM mit zweiseitig unbeschränktem Band

Damit hat die Standard-DTM $\mathcal{M}'$ die Arbeitsweise der zw-DTM $\mathcal{M}$ vollständig simuliert.

Also kann man mit zw-Turing-Maschinen nicht mehr berechnen als mit Standard DTMs. $\square$

Variationen von Turing-Maschinen

- Standard-DTM
- Turing-Maschinen, die nie hängen
- DTM mit zweiseitig unbeschränktem Band (zw-DTM)
- DTM mit k Halbbändern

DTM mit k Halbbändern

Definition (DTM mit k Halbbändern, k-DTM)
Eine **Turing-Maschine** $\mathcal{M} = (\ K, \Sigma_1, \ldots, \Sigma_k, \delta, s\ )$ **mit k Halbbändern** (mit je einem Kopf) ist eine Turing-Maschine mit einer Übergangsfunktion

$$\delta : K \times \Sigma_1 \times \ldots \times \Sigma_k \rightarrow$$
$$(K \cup \{h\}) \times (\Sigma_1 \cup \{L, R\}) \times \ldots \times (\Sigma_k \cup \{L, R\})$$

Eine **Konfiguration** einer k-Turing-Maschine hat die Form

$$C = q, w_1 \underline{a_1} u_1, \ldots, w_k \underline{a_k} u_k.$$

DTM mit k Halbbändern

DTM mit k Halbbändern

- Die Köpfe einer k-DTM können sich **unabhängig** bewegen (sonst hätten wir nur eine DTM mit k Spuren).
- Die Definition der Nachfolgekonfiguration verläuft analog zu der Definition bei Standard-DTM.
- Für eine k-DTM, die eine Funktion $f : \Sigma_0^m \to \Sigma_0^n$ berechnet, legen wir fest, dass sowohl die m Eingabewerte als auch – nach der Rechnung – die n Ergebniswerte auf dem ersten Band stehen sollen.
- Es übertragen sich alle Begriffe wie *berechenbar, entscheidbar* etc. kanonisch auf k-DTM.

DTM mit k Halbbändern

Theorem [Simulation von k-DTM durch DTM]
Zu jeder k-DTM $\mathcal{M}$, die eine Funktion f berechnet (resp. eine Sprache L akzeptiert), existiert eine DTM $\mathcal{M}'$, die ebenfalls f berechnet (resp. L akzeptiert).

DTM mit k Halbbändern

Beweis (Skizze)

Wir arbeiten mit einer Turing-Maschine mit mehreren Spuren.

Um eine k-**DTM** zu simulieren, verwenden wir $2k$ **Spuren**, also Bandzeichen, die aus $2k$ übereinander angeordneten Buchstaben bestehen.

- In den Spuren mit ungerader Nummer stehen die Inhalte der k Bänder von $\mathcal{M}$.
- Die Spuren mit gerader Nummer verwenden wir, um die Positionen der Köpfe von $\mathcal{M}$ zu simulieren:
  Die $2i$-te Spur enthält an genau einer Stelle ein $\wedge$, nämlich da, wo $\mathcal{M}$ gerade seinen i-ten Kopf positioniert hätte, und ansonsten nur Blanks.

$\mathcal{M}'$ kodiert zunächst die Eingabe von $\mathcal{M}$. Dann simuliert $\mathcal{M}'$ die Maschine $\mathcal{M}$. Am Ende der Rechnung wird noch die Ausgabe dekodiert.

Übersicht

- Determinierte Turing-Maschinen (DTMs)
- Varianten von Turing-Maschinen
- Indeterminierte Turing-Maschinen (NTMs)
- Universelle determinierte Turing-Maschinen
- Entscheidbar/Aufzählbar
- Determinierte Turing-Maschinen entsprechen Typ 0
- Unentscheidbarkeit

Indeterminierte Turing-Maschinen (NTMs)

Definition (Indeterminierte Turing-Maschine, NTM)

Eine **indeterminierte Turing-Maschine** $\mathcal{M}$ ist ein Tupel

$$M = (K, \Sigma, \Delta, s)$$

Dabei sind K, Σ, s definiert wie bei determinierten Turing-Maschinen.

Übergangs**relation**:

$$\Delta \subseteq (K \times \Sigma) \times \big((K \cup \{h\}) \times (\Sigma \cup \{L, R\})\big)$$

Indeterminierte Turing-Maschinen (NTMs)

Mehrere Nachfolgekonfigurationen

Konfigurationen sind definiert wie bei DTMs.

Nun kann eine Konfiguration aber mehrere mögliche Nachfolgekonfigurationen haben.

Definition (NTM: Halten, Hängen, Akzeptieren) Sei $\mathcal{M} = (K, \Sigma, \Delta, s_0)$ eine indeterminierte Turing-Maschine.

- $\mathcal{M}$ **hält** bei Input w, falls es **unter den möglichen Rechnungen**, die $\mathcal{M}$ wählen kann, **eine gibt**, so dass $\mathcal{M}$ eine Haltekonfiguration erreicht.
- $\mathcal{M}$ **hängt** in einer Konfiguration, wenn es keine (durch Δ definierte) Nachfolgekonfiguration gibt.
- $\mathcal{M}$ **akzeptiert** ein Wort w, falls sie **von $s, \#w\underline{\#}$ aus einen Haltezustand erreichen kann**, und $\mathcal{M}$ akzeptiert eine Sprache L, wenn sie genau alle Wörter $w \in L$ akzeptiert.

Indeterminierte Turing-Maschinen (NTMs)

Bemerkung

Wenn es nicht nur darauf ankommt, ob die Maschine hält, sondern auch mit welchem Bandinhalt:

Welche der vielen Haltekonfigurationen sollte dann gelten?

Um dieses Problem zu umgehen, übertragen wir die Begriffe des *Entscheidens* und *Aufzählens* **nicht** auf NTMs. Im Allgemeinen verwendet man NTMs auch nicht dazu, Funktionen zu berechnen.

Indeterminierte Turing-Maschinen (NTMs)

Wie rechnet eine indeterminierte Turing-Maschine?

- Die Regeln einer determinierten DTM kann man sich als Programm (aus sehr einfachen Schritten) vorstellen.
- **Bei NTM ist das anders!**
- **Eine NTM ist nicht einfach eine Maschine, die immer richtig rät!**

Dieselbe Diskussion hatten wir bei indeterminierten endlichen Automaten.

Indeterminierte Turing-Maschinen (NTMs)

Vorstellung von einer NTM

- Korrekte Vorstellung:
  - Übergänge von Konfiguration zu Nachfolgekonfiguration entspr. Regelmenge
  - **plus Suchverfahren!**

  oder: Eine NTM beschreitet alle möglichen Rechenwege parallel.

- **Eine NTM akzeptiert ein Wort, wenn es mindestens einen Berechnungsweg gibt, der in einer Haltekonfiguration endet.**

- Sprechweise **"Die NTM rät"**: Wir verwenden diese Sprechweise, sie ist aber mit Vorsicht zu genießen!

Indeterminierte Turing-Maschinen (NTMs)

Beispiel.

Sei

$$L = \{|^n \mid n \text{ ist nicht prim und } n \geq 2\}$$

Eine NTM kann diese Sprache wie folgt azeptieren:

1. Eine Zahl „raten“ und (nach rechts) aufs Band schreiben.
2. Noch eine Zahl „raten“ und daneben schreiben.
3. Die beiden Zahlen miteinander multiplizieren.
4. Das Ergebnis mit der Eingabe vergleichen.
5. Genau dann, wenn beide gleich sind, anhalten.