Die Turingmaschine

- DTM = Deterministische Turingmaschine
- NTM = Nichtdeterministische Turingmaschine
- TM = DTM oder NTM

Intuitiv gilt:

- DTM = (DFA + dynamischer Speicher)
- NTM = (NFA + dynamischer Speicher)
- Der dynamische Speicher ist ein (in Zellen unterteiltes) zweiseitig unendliches Band versehen mit einem Lese–Schreibkopf. Es enthält anfangs die Eingabe, dient aber auch als Arbeitsspeicher.
- Die Einschränkung, den Kopf auf dem Band nur von links nach rechts bewegen zu dürfen, wird fallen gelassen.
- Die Einschränkung, den Arbeitsspeicher kellerartig organisieren zu müssen, wird ebenfalls fallen gelassen.

DTM (formale Definition)

Eine DTM M besteht aus den folgenden Komponenten:

- Z, die Zustandsmenge (eine endliche Menge)
- Σ, das Eingabealphabet (ebenfalls endlich)
- $\Gamma \supset \Sigma$, das Arbeitsalphabet (ebenfalls endlich)
- $\delta : Z \times \Gamma \rightarrow\ Z \times \Gamma \times \{L, R, N\}$, die partiell definierte Überführungsfunktion
- $z_0 \in Z$, der Startzustand
- $\Box \in \Gamma \setminus \Sigma$, das Blank (auch Leerzeichen genannt)
- $E \subseteq Z$ die Menge der Endzustände:
  $\delta(z_e, A)$ ist undefiniert für alle $z_e \in E$ und alle $A \in \Gamma$.

Arbeitsweise der DTM

- Anfangs befindet sich M im Startzustand z_0, ihr Band enthält das Eingabewort $w \in \Sigma^*$ (umrahmt von Blanks) und der Kopf steht auf dem ersten Zeichen von w (bzw. auf einem Blank, falls $w = \varepsilon$).
- Falls sich M im Zustand $z \in Z$ befindet, der Kopf das Zeichen $A \in \Gamma$ liest und
$$\delta(z, A) = (z', A', d) \in Z \times \Gamma \times \{L, R, N\} \quad ,$$
dann geht M in den Zustand z' über und ersetzt A durch A'. Für $d = L$ bzw. $d = R$ erfolgt zusätzlich eine Kopfbewegung auf die linke bzw. rechte Nachbarzelle des Bandes.
- M stoppt **gdw** M in einem Zustand z ist, ein Symbol A liest und $\delta(z, A)$ undefiniert ist.
- Die Eingabe wird akzeptiert **gdw** M im Laufe der Rechnung in einen Endzustand gerät (und dann automatisch stoppt).

Konfigurationen einer TM

Die Konfiguration einer TM besteht aus

- dem aktuellen Zustand $z \in Z$,
- der Position des Kopfes auf dem Band.
- dem Bandinhalt $\gamma \in \Gamma^*$ (im Bereich der Eingabe sowie der im Laufe der Rechnung bereits besuchten Zellen)

Notation: $\alpha z \beta$, wobei $\gamma = \alpha\beta$ der aktuelle Bandinhalt und der Kopf auf dem ersten Zeichen von β positioniert ist

Anfangskonfiguration bei Eingabe w: $z_0 w$ (hier: $\alpha = \varepsilon, \beta = w$)

Akzeptierende Endkonfiguration: $\alpha z \beta$ für jedes $z \in E, \alpha, \beta \in \Gamma^*$

Stoppkonfiguration: $\alpha z A \beta'$ für jedes $z \in Z, \alpha, \beta' \in \Gamma^*$, $A \in \Gamma$ mit $\delta(z, A)$ ist undefiniert.

Beobachtung: Da δ auf Endzuständen undefiniert ist, ist jede akzeptierende Endkonfiguration auch eine Stoppkonfiguration.

Folgekonfigurationen

Eine „Rechnung“ einer TM lässt sich als Folge von Konfigurationen beschreiben.

Definition:

1. $\alpha' z' \beta'$ heißt unmittelbare Folgekonfiguration von $\alpha z \beta$ **gdw** $\alpha' z' \beta'$ aus $\alpha z \beta$ durch einen „Rechenschritt“ (einmalige Verwendung der Überführungs-funktion) resultieren kann.
   Notation: $\alpha z \beta \vdash \alpha' z' \beta'$.

2. $\alpha' z' \beta'$ heißt Folgekonfiguration von $\alpha z \beta$ **gdw** $\alpha' z' \beta'$ aus $\alpha z \beta$ durch eine (evtl. leere) Folge von Rechenschritten resultieren kann.
   Notation: $\alpha z \beta \vdash^* \alpha' z' \beta'$.
   Formal ist „$\vdash^*$” die reflexive–transitive Hülle von „$\vdash$“.

Im Falle einer DTM ist die unmittelbare Folgekonfiguration stets eindeutig bestimmt und es gibt nur eine mögliche Rechnung auf der Eingabe.

Beispiel

bin(n) bezeichne die Binärdarstellung einer Zahl $n \geq 0$.

Aufgabe: Implementiere einen Binärzähler, der, gestartet auf bin(n),

- bin($n + 1$) berechnet,
- den Kopf auf dem ersten Zeichen von bin($n + 1$) positioniert
- und sich dann in einen Endzustand begibt und stoppt.

Idee: Verwende vier Zustände für folgende Phasen der Berechnung:

- z_0: Suche das Bit am weitesten rechts.
- z_1: Inkrementiere den Zähler (unter Beachtung des Übertrages).
- z_2: Suche das Bit am weitesten links.
- z_e: Stoppe.

Beispiel (fortgesetzt)

Komponenten der „Binärzähler“–DTM:

- Zustandsmenge $\{z_0, z_1, z_2, z_e\}$
- Eingabealphabet $\{0, 1\}$
- Arbeitsalphabet $\{0, 1, \Box\}$
- Überführungsfunktion δ (weiter unten spezifiziert)
- Startzustand z_0
- Blank $\Box$
- Menge $\{z_e\}$ der Endzustände

Beispiel (fortgesetzt)

„Turing–Tafel“ von M (tabellarische Angabe von δ):

δ	0	1	$\Box$
z_0	$(z_0, 0, R)$	$(z_0, 1, R)$	$(z_1, \Box, L)$
z_1	$(z_2, 1, L)$	$(z_1, 0, L)$	$(z_e, 1, N)$
z_2	$(z_2, 0, L)$	$(z_2, 1, L)$	$(z_e, \Box, R)$

`--- Macht das Sinn ?? Erläutere !! ---`

Sprache einer TM

Die folgende Definition der von der TM M erkannten Sprache $T(M)$ entspricht unserer Vereinbarung über das Akzeptieren mit Endzustand:

$$T(M) := \{w \in \Sigma^* |\ \exists z \in E, \alpha, \beta \in \Gamma^* : z_0 w \vdash^* \alpha z \beta\}$$

In Worten: Wort w gehört zur Sprache $T(M)$ **gdw** M durch Verarbeitung der Eingabe w aus der Anfangskonfiguration in eine akzeptierende Endkonfiguration gelangen kann.