# Inhaltsverzeichnis

# 1 Berechenbarkeitstheorie

## 1.1 Intuitiver Berechenbarkeitsbegriff und Churchsche These

- Es gibt eine intuitive Vorstellung, welche Funktionen (auf den natürlichen Zahlen) berechenbar sind.
- Auf Basis dieser intuitiven Vorstellung können allerdings keine Beweise geführt werden, daß eine bestimmte Funktion nicht berechenbar ist.
- Deshalb ist eine formale mathematische Definition der Berechenbarkeit notwendig.
- Der Nachweis der Nichtberechenbarkeit einer Funktion besteht dann darin nachzuweisen, daß die betrachtete Funktion nicht der Definition entspricht.
- neues Problem: Begründung, daß die formale Definition genau den intuitiven Berechenbarkeitsbegriff erfaßt.
  Dazu ist kein formaler Beweis führbar, denn der intuitive Berechenbarkeitsbegriff ist nicht formal erfaßt.

**Beispiele für Funktionen, die intuitiv berechenbar sind:**

Eine Funktion $f : \mathbb{N}^k \to \mathbb{N}$ ist berechenbar, falls es einen Algorithmus, also ein mechanisches Rechenverfahren (z.B. MODULA–Programm) gibt, das $f$ berechnet,
also bei Eingabe von $(n_1, \ldots, n_k) \in \mathbb{N}^k$ nach endlich vielen Schritten mit der Ausgabe $f(n_1, \ldots, n_k)$ stoppt.

Eine partielle Funktion $f : \mathbb{N}^k \to \mathbb{N}$ ist berechenbar, falls der Algorithmus bei Eingabe von $(n_1, \ldots, n_k) \notin \mathrm{Def}(f)$ nicht stoppt.

1. Der Algorithmus:

   ```
   INPUT(n);
   REPEAT UNTIL FALSE;
   ```

   "berechnet" die total undefinierte Funktion $\Omega : n \to undef$.

2. Die Funktion

$$f_\pi(n) = \begin{cases} 1 & \text{falls } n \text{ Anfangsabschnitt der} \\ & \text{Dezimalbruchentwicklung von } \pi \text{ ist} \\ 0 & \text{sonst} \end{cases}$$

ist berechenbar, denn es gibt ein beliebig genaues Näherungsverfahren für die Berechnung von $\pi$ .

3. Ob die Funktion

$$g(n) = \begin{cases} 1 & \text{falls } n \text{ irgendwo in der Dezimalbruchentwicklung} \\ & \text{von } \pi \text{ vorkommt} \\ 0 & \text{sonst} \end{cases}$$

berechenbar ist, ist bisher unklar.

4. Die Funktion

$$h(n) = \begin{cases} 1 & \text{falls in der Dezimalbruchentwicklung von } \pi \\ & \text{irgendwo mindestens } n\text{–mal hintereinander} \\ & \text{eine 0 vorkommt} \\ 0 & \text{sonst} \end{cases}$$

ist berechenbar!

1.Fall: In $\pi$ kommen beliebig lange 0–Folgen vor.

$\Rightarrow h(n) = 1$ für alle $n$

2.Fall: In $\pi$ kommen 0–Folgen der Länge $n_0$,
aber nicht der Länge $\geq n_0 + 1$ vor.

$$\Rightarrow h(n) = \begin{cases} 1 & \text{falls } n \leq n_0 \\ 0 & \text{sonst} \end{cases}$$

Es tritt entweder Fall 1 oder Fall 2 ein.

**Achtung:** Das Verfahren ist nicht konstruktiv;
trotzdem existiert ein Algorithmus.

Beispiel 2 zeigt, daß $f_\pi$ berechenbar ist.
(Da ein Näherungsverfahren existiert.)
Auch $f_e$ ist berechenbar.

**Behauptung.**
Es gibt (viele) $r \in \mathbb{R}$, für die $f_r$ nicht berechenbar ist.

Beweis.
Es gibt überabzählbar viele $r \in \mathbb{R}$, aber nur abzählbar viele Rechenverfahren:
Ein Rechenverfahren muß sich durch einen endlichen Text beschreiben lassen und es gibt höchstens abzählbar viele endliche Texte. ■

**Bemerkung**
Berechenbarkeit von $f_r$ hat nichts mit rational/irrational zu tun!
Aber es gilt: für alle rationalen Zahlen $\mathbb{Q}$ ist $f_q$ berechenbar.

## 1.2 Churchsche These

Es gibt verschiedene Vorschläge, den intuitiven Berechenbarkeitsbegriff formal zu erfassen:

- Turing–Maschinen (Turing 1936)
- WHILE–Programme
- GOTO–Programme
- $\mu$–rekursive Funktionen

Interessanter Weise kann bewiesen werden, daß alle diese Definitionen äquivalent sind. Es gibt bisher keinen Vorschlag, der nicht äquivalent wäre.

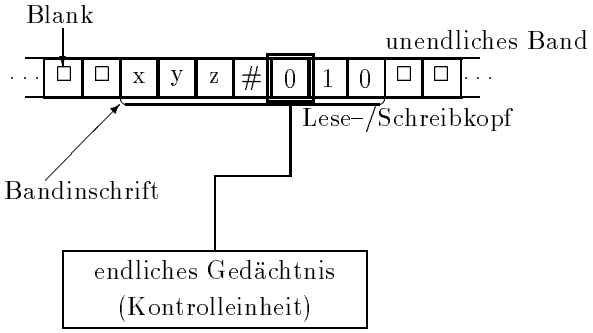
$\Rightarrow$ **Churchsche These:**
*Die durch den formalen Begriff der Turing–Berechenbarkeit (äquivalent: WHILE–Berechenbarkeit, GOTO–Berechenbarkeit, $\mu$ –Rekursivität) erfaßte Klasse von Funktionen stimmt genau mit der Klasse der intuitiv berechenbaren Funktionen überein.*

Die Churchsche These ist nicht beweisbar, da der intuitive Begriff der Berechenbarkeit nicht formal faßbar ist.
Sie ist allgemein akzeptiert.

Man kann auf Basis der Churchschen These zeigen, daß bestimmte Funktionen nicht berechenbar sind, indem man lediglich zeigt, daß keine TM zu ihrer Berechnung existieren kann.

## 1.3 Turing–Berechenbarkeit

Turings Vorschlag war, den Berechenbarkeitsbegriff mit Hilfe einer sehr einfachen Rechenmaschine (heute Turing–Maschine genannt) zu erfassen.




**Definition.**
Eine (Nichtdeterministische) Turingmaschine (kurz TM ) ist ein 7–Tupel

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \square, E)$$

mit

$Q$ endliche Menge von "Zuständen"
$\Sigma$ endliche Menge von "Eingabesymbolen" (Eingabealphabet)
$\Gamma \supset \Sigma$ endliche Menge von "Bandsymbolen" (Arbeitsalphabet)

$$\left.\begin{array}{l} \delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{\mathrm{L,R,N}\} \text{ "deterministische"} \\ \delta : Q \times \Gamma \rightarrow 2^{Q \times \Gamma \times \{\mathrm{L,R,N}\}} \text{ "nichtdeterministische"} \end{array}\right\} \text{Überführungsfunktion}$$

$q_0 \in Q$ "Startzustand"
$\square \in \Gamma - \Sigma$ "Blank" (leeres Bandsymbol)
$E \subseteq Q$ endliche Menge von "Endzuständen"

Informal bedeutet

$$\delta(q, a) = (q', b, m) \text{ bzw. } \delta(q, a) \ni (q', b, m)$$

folgendes:

Befindet sich $M$ in Zustand $q$ und liest Bandsymbol $a$ (unter Lese-/Schreibkopf), so geht $M$ im nächsten Schritt in Zustand $q'$ über, überschreibt $a$ durch $b$ und

führt Kopfbewegung $m \in \{L(inks), R(echts), N(ichts)\}$ aus.

**Definition.**
Eine Konfiguration einer TM $M$ ist ein Wort $k \in \Gamma^* Q \Gamma^*$

Konfigarationen sind "Momentaufnahmen" von $M$:
$k = \alpha q \beta$ bedeutet:

> $\alpha\beta$ ist der nichtleere (bzw. besuchte) Teil der Bandinschrift.
> $q$ ist der Zustand der TM .
> Das erste Zeichen von $\beta$ ist das Zentrum des Lese-/Schreibkopfes.

Startkonfiguration bei Eingabe $w \in \Sigma^*$ : $q_0 w$

$$\square \ldots \square w \square \ldots \square$$
$$\uparrow$$

$w$ steht auf dem Band
Der Lese-/Schreibkopf steht auf dem ersten Symbol von $w$
$M$ ist in Zustand $q_0$

Formale Beschreibung der Arbeit einer TM :

**Definition.**
Beschreibe die Relation $\vdash$ auf der Menge der Konfigurationen von M:

$$a_1 \ldots a_m q b_1 \ldots b_n \vdash \begin{cases} a_1 \ldots a_m q' c b_2 \ldots b_n & \text{falls } (q', c, N) \in \delta(q, b_1)\ m \geq 0, n \geq 1 \\ a_1 \ldots a_m c q' b_2 \ldots b_n & \text{falls } (q', c, R) \in \delta(q, b_1)\ m \geq 0, n \geq 2 \\ a_1 \ldots a_{m-1} q' a_m c b_2 \ldots b_n & \text{falls } (q', c, L) \in \delta(q, b_1)\ m \geq 1, n \geq 1 \\ a_1 \ldots a_m \square q' c & \text{falls } (q', c, R) \in \delta(q, b_1) \text{ und } n = 1 \\ q' \square c b_2 \ldots b_n & \text{falls } (q', c, L) \in \delta(q, b_1) \text{ und } m = 0 \end{cases}$$

$\vdash^*$ bezeichne den transitiven Abschluß von $\vdash$

**Beispiel.**
Eine TM , die die Eingabe $w \in \Sigma^*$ als Binärzahl interpretiert und 1 hinzuaddiert:

$$M = (\{q_0, q_1, q_2, q_e\}, \{0, 1\}, \{0, 1, \square\}, \delta, q_0, \square, \{q_e\})$$

mit

$$\begin{aligned} \delta(q_0, 0) &= (q_0, 0, R) \\ \delta(q_0, 1) &= (q_0, 1, R) \\ \delta(q_0, \square) &= (q_1, \square, L) \end{aligned}$$

$$\delta(q_1, 0) = (q_2, 1, L)$$

$$\begin{aligned}\delta(q_1, 1) &= (q_1, 0, L)\\ \delta(q_1, \Box) &= (q_e, 1, N)\end{aligned}$$

$$\begin{aligned}\delta(q_2, 0) &= (q_2, 0, L)\\ \delta(q_2, 1) &= (q_2, 1, L)\\ \delta(q_2, \Box) &= (q_e, \Box, R)\end{aligned}$$

**Beispiel.**

$w$=101

$$q_0 \vdash 1q_001 \vdash 10q_01 \vdash 101q_0\Box \vdash 10q_1\Box \vdash 1q_100\Box \vdash q_2110\Box \vdash q_2\Box110\Box \vdash \Box q_e110$$

**Definition.**
Eine Funktion $f : I\!N^* \rightarrow I\!N$ heißt *Turing–berechenbar*, falls es eine (deterministische) TM $M$ gibt, so daß für alle $n_1, \ldots, n_k, m \in I\!N$ gilt:

$$f(n_1, \ldots, n_k) = m$$

$\Longleftrightarrow$

$$q_0 bin(n_1)\#bin(n_2)\# \ldots \#bin(n_k) \vdash^* q_e bin(m)$$

wobei $q_e \in E$.

($bin(n)$ bezeichnet die Binärdarstellung von $n \in I\!N$ ohne führende Nullen)

**Definition.**
Eine Funktion $f : \Sigma^* \rightarrow \Sigma$ heißt *Turing–berechenbar*, falls es eine (deterministische) TM $M$ gibt, so daß für alle $x, y \in \Sigma^*$ gilt:

$$f(x) = y$$

$\Longleftrightarrow$

$$q_0 x \vdash^* q_e y$$

wobei $q_e \in E$.

Damit ist ausgedrückt, daß im Falle $f(x) = undef.$ $M$ in eine unendliche Schleife geht.

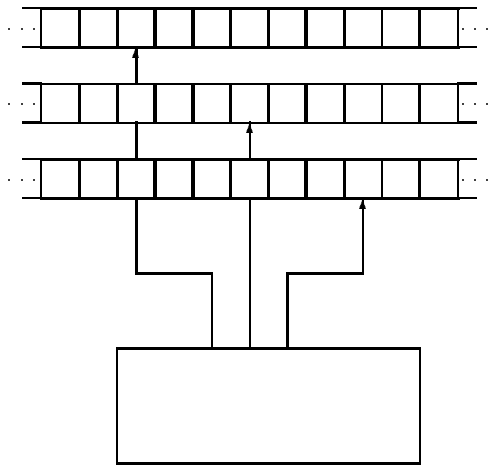
**Beispiel.**

1. $f : n \to n + 1$ ist Turing–berechenbar.
   Das Beispiel überführt $bin(n)$ in $bin(n + 1)$
2. die überall nichtdefinierte Funktion ist Turing–berechenbar;
   z.B.: durch eine TM mit

   $$\delta(q_0, a) = (q_0, a, R) \text{ für alle } a \in \Gamma$$

3. Typ 0- (semientscheidbare) Sprachen sind berechenbar.

**Mehrband–TM:**



Die Lese-/Schreibköpfe können sich unabhängig voneinander bewegen.

$\delta : Q \times \Gamma^k \to Q \times \Gamma^k \times \{L, R, N\}^k$

**Satz.**
*Zu jeder Mehrband-TM $M$ gibt es eine 1-Band TM $M'$ die dieselbe Funktion berechnet wie $M$.*

Beweis.
Sei $k$ die Anzahl der Bänder, $\Gamma$ Arbeitsalphabet von $M$.

Idee: