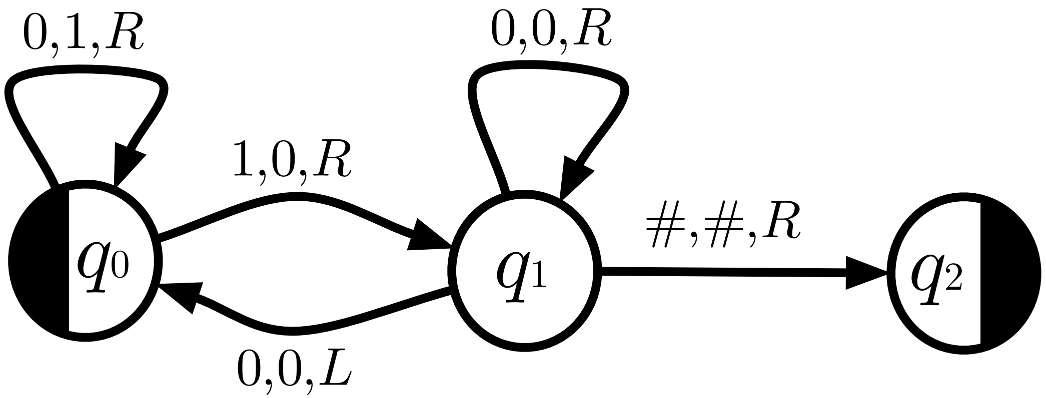
Beispiel: NTM

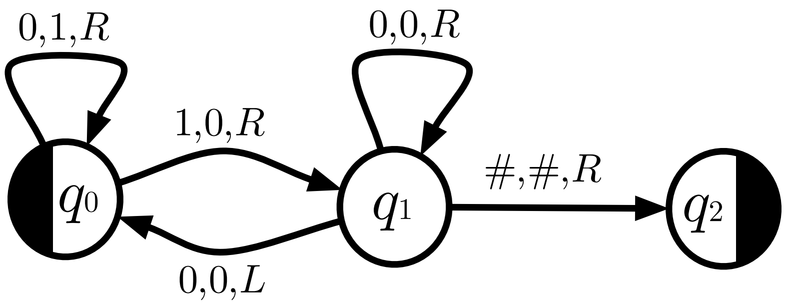
$$M = (\{q_0, q_1, q_2\}, \{0,1\}, \{0,1,\#\}, \delta, q_0, \#, \{q_2\})$$



Zustand	0	1	#
q_0	$\{(1, R, q_0)\}$	$\{(0, R, q_1)\}$	$\emptyset$
q_1	$\{(0, R, q_1), (0, L, q_0)\}$	$\{(1, R, q_1), (1, L, q_0)\}$	$\{(\#, R, q_2)\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$

Aufgabe: Welche Konfigurationen sind bei Eingabe von 01 (bzw. 100) erreichbar?

Beispiel: NTM-Konfigurationen



Eingabe: 01 (bzw. 100)

$$[q_0]01 \vdash_M 1[q_0]1 \vdash_M 10[q_1] \vdash_M 10\#[q_2]$$

$$[q_0]100 \vdash_M 0[q_1]00 \vdash_M 00[q_1]0 \vdash_M 000[q_1] \vdash_M 000\#[q_2]$$

$$[q_0]100 \vdash_M 0[q_1]00 \vdash_M [q_0]000 \vdash_M 1[q_0]00 \vdash_M 11[q_0]0 \vdash_M 111[q_0]$$

$$[q_0]100 \vdash_M 0[q_1]00 \vdash_M 00[q_1]0 \vdash_M 0[q_0]00 \vdash_M 01[q_0]0 \vdash_M 011[q_0]$$

$$[q_0]100 \vdash_M 0[q_1]00 \vdash_M 00[q_1]0 \vdash_M 000[q_1] \vdash_M 00[q_0]0 \vdash_M 001[q_0]$$

Entscheidbare vs. rekursiv aufzählbare Mengen

Akzeptierung/Berechnung

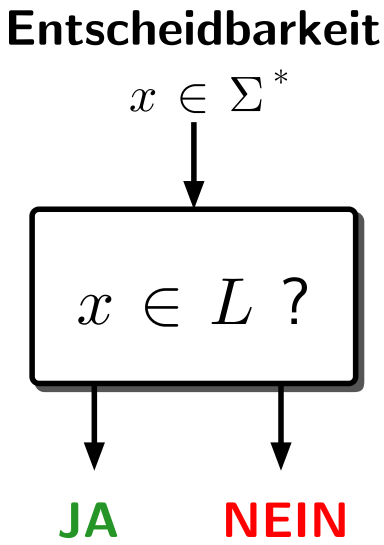
…ein kurzer Rückblick:

- Sprache L wird durch eine TM M akzeptiert gdw. **nur** Wörter aus L eine Erfolgsrechnung auf M haben.
- Funktion f wird von DTM M berechnet, gdw. M für **jede** Eingabe w in einer Endkonfiguration $q_e f(w)$ hält bzw. nicht hält, falls $w \notin \mathrm{Def}(f)$.

Definition: rekursiv

Eine Menge $L \subseteq \Sigma^*$ heißt (relativ zu Σ^*) **entscheidbar** oder **rekursiv** gdw. ihre charakteristische Funktion $\chi_L : \Sigma^* \longrightarrow \{0, 1\}$ berechenbar ist.

Die *Klasse aller entscheidbaren Mengen* wird mit $\mathcal{REC}$ (*recursive sets*) bezeichnet.

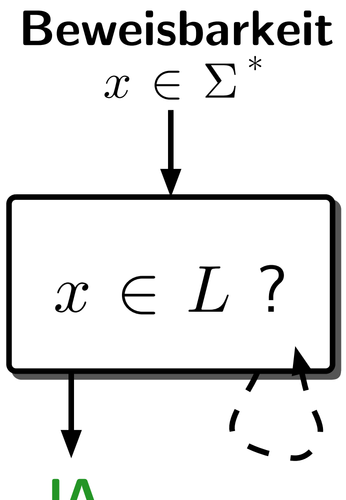


Definition: rekursiv aufzählbar

Eine Menge $L \subseteq \Sigma^*$ heißt **beweisbar** oder **rekursiv aufzählbar** gdw. $L = \emptyset$ ist, oder eine totale Turing-berechenbare Funktion $g : I\!N \to \Sigma^*$ existiert, für die $g(I\!N) = L$ ist.

Die *Klasse aller aufzählbaren Mengen* wird mit $\mathcal{RE}$ (*recursively enumerable sets*) bezeichnet.

…oder anders dargestellt:



Eigenschaften

Theorem: L ist *rekursiv aufzählbar* gdw. eine TM M mit $L = L(M)$ existiert. **Beweisidee:** $L = L(M) \Rightarrow \exists$ TM A, die alle Wörter aus L auf die Ausgabe schreibt:

- Generiere sukzessive Paare aus $(i, j) \in I\!N^2$.
- Simuliere M auf dem i-ten Wort w_i.
- Falls M nach j Schritten hält, schreibe w_i und ein Trennsymbol.

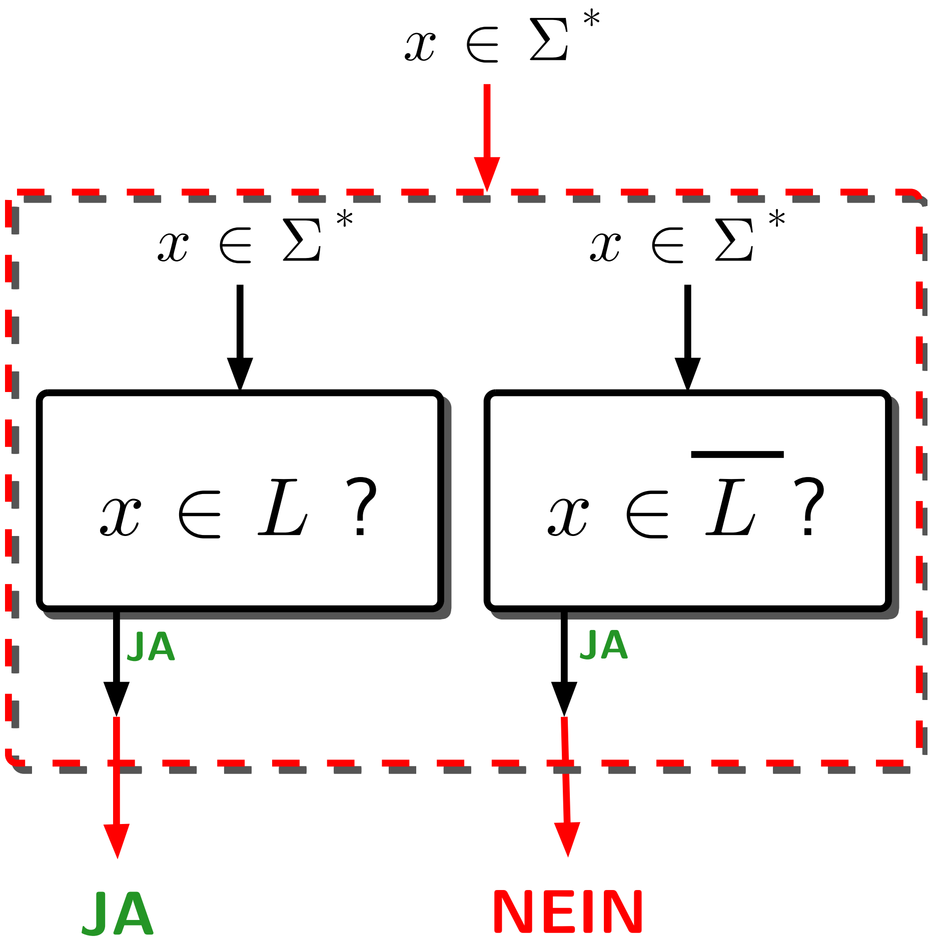
$\exists$ TM A, die alle Wörter aus L schreibt $\Rightarrow L = L(M)$

- Verwende Zähler i
- Kontrolliere, ob w_i (Ausgabe von A) das Eingabewort w ist? Ja: akzeptiere! Nein: inkrementiere i.

Analog: L rekursiv aufzählbar $\Rightarrow \exists$ TM $M : L = L(M)$

Eigenschaften

Theorem: Ist eine Menge L und ihr Komplement $\overline{L}$ rekursiv aufzählbar, so ist L auch rekursiv.



Hierarchie

Folgende Beziehungen wollen wir zeigen:

Klasse der **regulären Mengen**

$\subsetneqq$ Klasse der **kontextfreien Mengen**

$\subsetneqq$ Klasse der **kontextsensitiven Mengen**

$\subsetneqq$ Klasse der **entscheidbaren Mengen**

$\subsetneqq$ Klasse der **aufzählbaren Mengen**

$\subsetneqq$ Klasse der **abzählbaren Mengen**

$\neq$ Klasse der **überabzählbaren Mengen**

Existenz überabzählbarer Mengen ist bekannt.

(z.B. aus Diagonalbeweis).

Abschlusseigenschaften

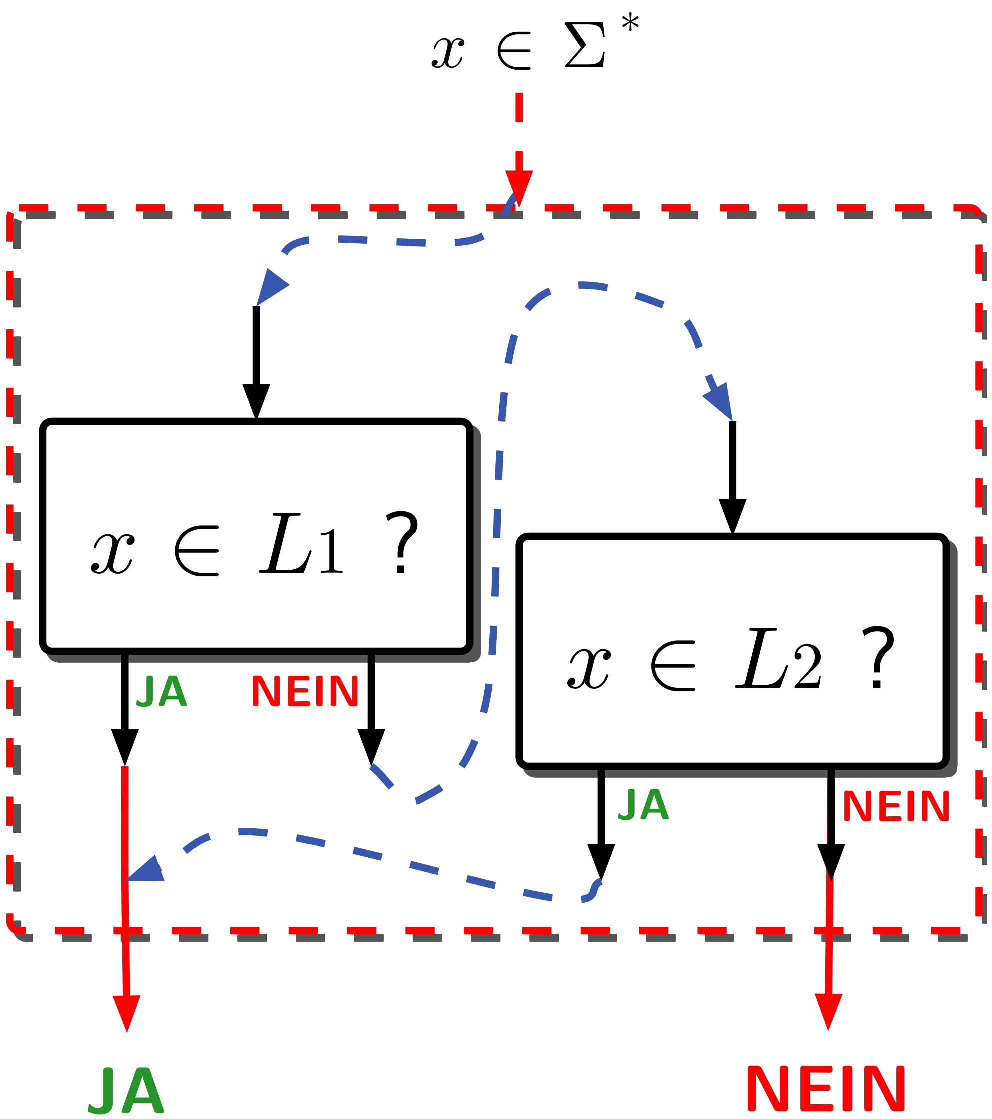
Theorem: Die Klasse der entscheidbaren Mengen bildet eine *boolesche Mengen-Algebra*.

Beweisidee: Z.z. ist der Abschluss gegen

- Komplementbildung,
- Durchschnitt,
- Vereinigung.

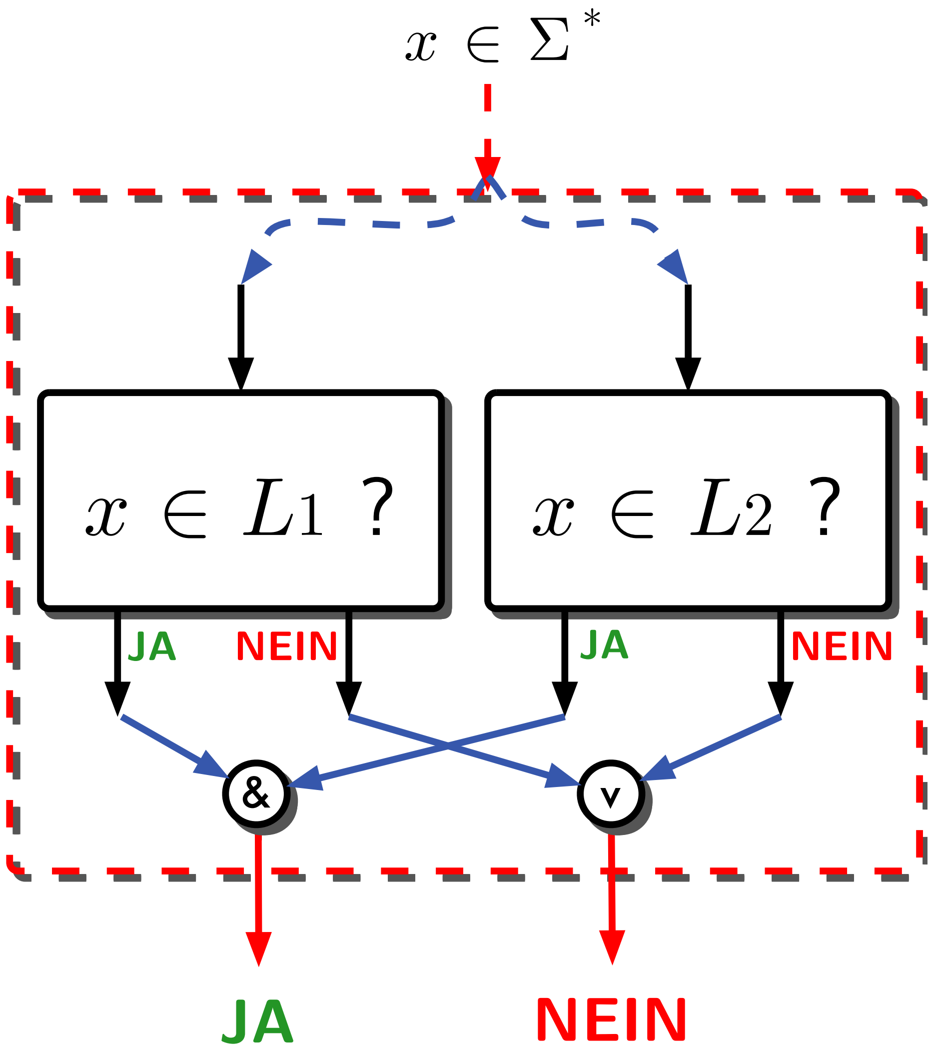
Komplementabschluss:
…einfach JA und NEIN vertauschen.

Vereinigung

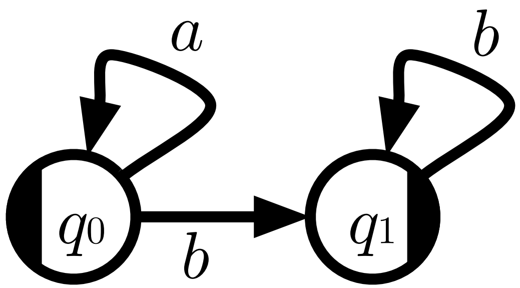


Durchschnitt

. . . folgt nach De Morgan, oder direkt:



Gödelisierung von FA's



. . . wird kodiert durch:

a	x
b	xx
q_0	z
q_1	zz

Kante (q_0, b, q_1): -zxxzz

Insgesamt: -zxz-zxxzz+zzxxzz

. . . ähnlich auch für Turing-Maschinen!

nicht aufzählbare Menge

- Wir zeigen, dass es Mengen gibt, die nicht rekursiv aufzählbar sind:

Theorem: Sei G ein Alphabet zur Kodierung von Turing-Maschinen bzw. Wörtern, sowie w_i das i-te Wort und M_i die i-te DTM in der lexikalischen Aufzählung der Wörter $\langle M_i \rangle \in G^*$.

Dann ist die Menge $L_d := \{w_i \mid w_i \notin L(M_i)\}$ nicht aufzählbar.

(Selbstanwendbarkeitsproblem)

Beweis: $L_d \notin \mathcal{RE}$

Charakteristische Funktion für $L(M_i)$ und L_d:

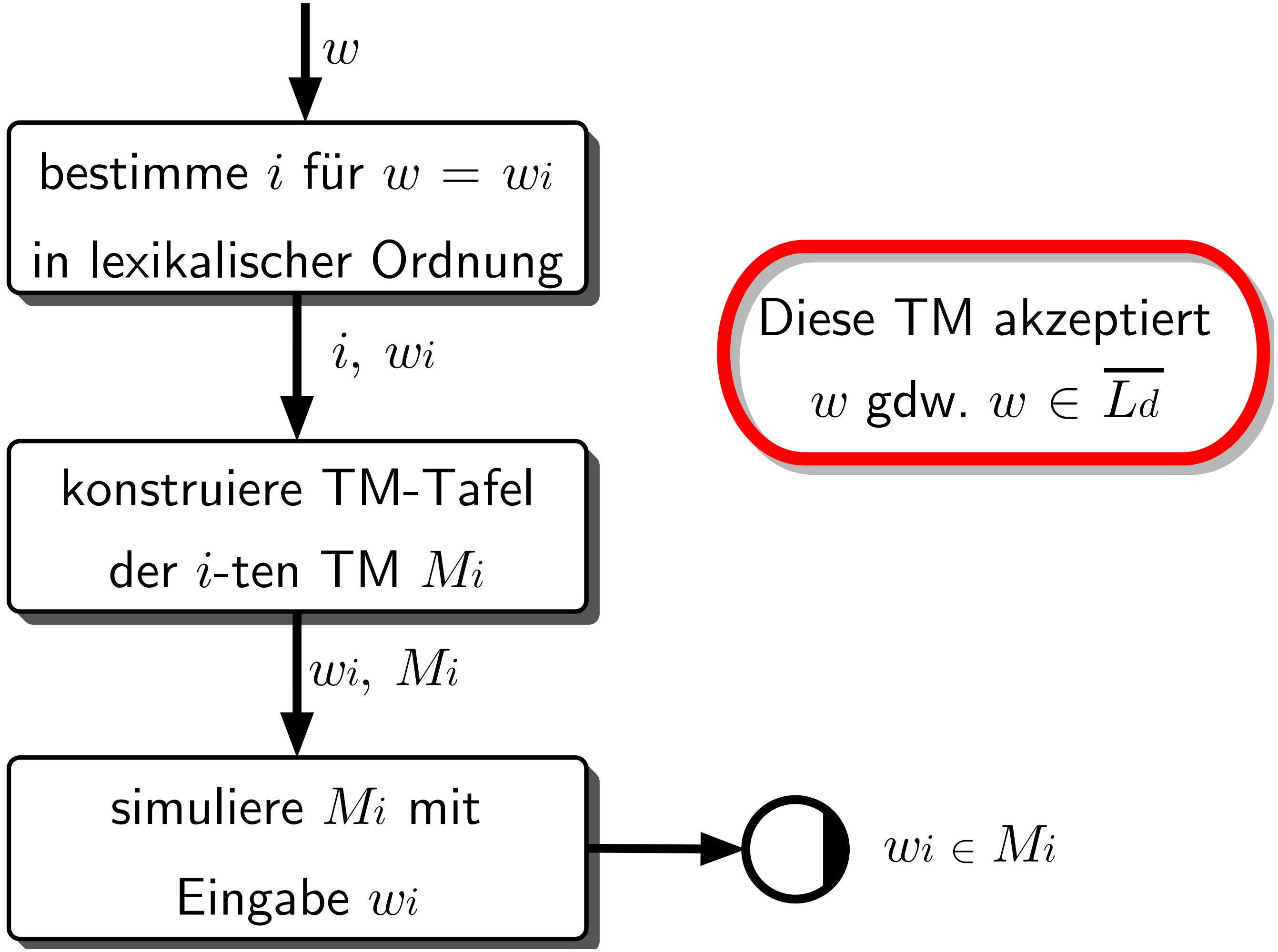
	w_0	w_1	w_2	$\cdots$	w_n	$\cdots$
M_0	0	1	1	$\cdots$	0	$\cdots$
M_1	1	1	0	$\cdots$	1	$\cdots$
M_2	1	0	0	$\cdots$	1	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
M_n	1	1	0	$\cdots$	1	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$
L_d	1	0	1	$\cdots$	0	$\cdots$

Also: $\forall i \in I\!N : w_i \in L(M_i) \iff w_i \notin L_d$.
Somit $\forall i \in I\!N : L(M_i) \neq L_d$

und L_d *nicht aufzählbar*!

Das Komplement von L_d

…ist aufzählbar:



Definition: UTM

- Sei M eine DTM mit Anfangskonfiguration k_0.
- Sei $\langle M \rangle$ die Kodierung über dem Alphabet $G := \{0, 1\}$ von M und $\langle k_0 \rangle$ jene von k_0.

Eine DTM $U := (Z, \Sigma, \Gamma, \delta, q_0, Z_{\text{end}})$ heißt **universelle Turing-Maschine** (UTM), wenn für die Anfangskonfiguration $q_0\langle M \rangle\langle k_0 \rangle$ gilt:

- $k_i \underset{M}{\vdash} k_j \quad \Rightarrow \quad \langle A \rangle\langle k_i \rangle \overset{*}{\underset{U}{\vdash}} \langle A \rangle\langle k_j \rangle$, wobei hier nur die relevante Bandinschrift von U gemeint ist, ohne die Stellungen ihres LSK und der Zustände.

Existenz einer UTM

Definition der **universellen Turing-Maschine** macht noch keine Aussage darüber, ob so etwas auch tatsächlich existiert!

$\boxed{\textbf{Lemma:}\text{ Es gibt universelle Turing-Maschinen.}}$

- Lesen der Kodierung einer TM und eines Eingabewortes,
- Simulation der Einzelschritte der kodierten TM.
- hier **kein Beweis** (durch Angabe einer UTM)

Das Halteproblem

Das Halteproblem lautet:

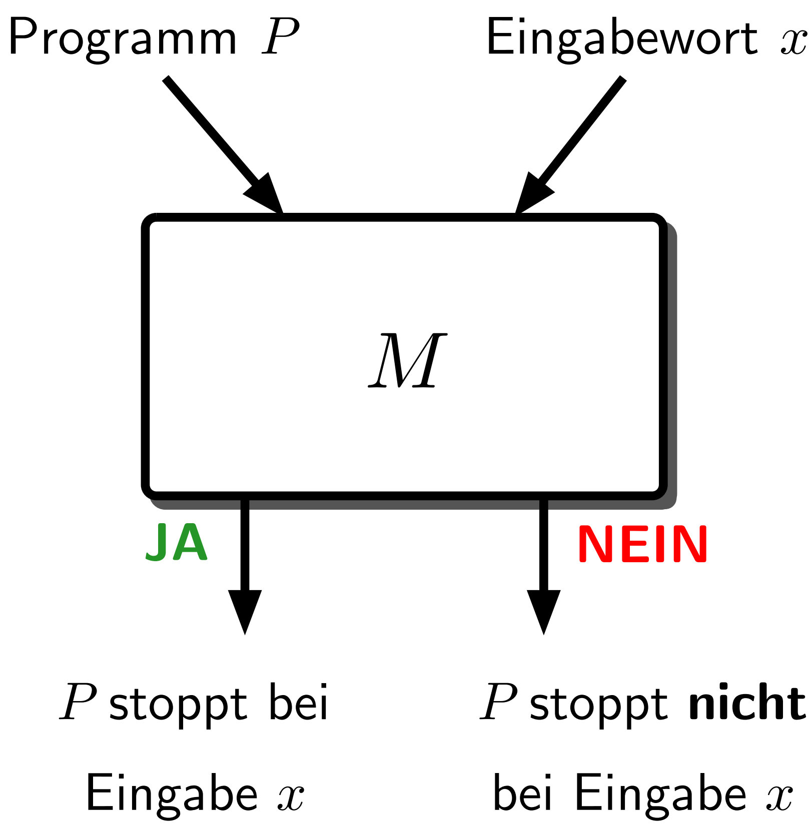
Gegeben:	Ein Computerprogramm P und ein Eingabe x für P.
Gesucht:	Kommt P für x jemals in einen Stop- bzw. Endzustand?
Antwort:	**?**

- Darstellung als Menge:
  $H = \{\langle M\rangle\langle w\rangle \mid \text{TM } M \text{ hält auf } w\}$
- Wichtig zur Erkennung von Endlosschleifen!
- **Aber:** H ist unentscheidbar.

Beweis: Halteproblem $\notin \mathcal{REC}$

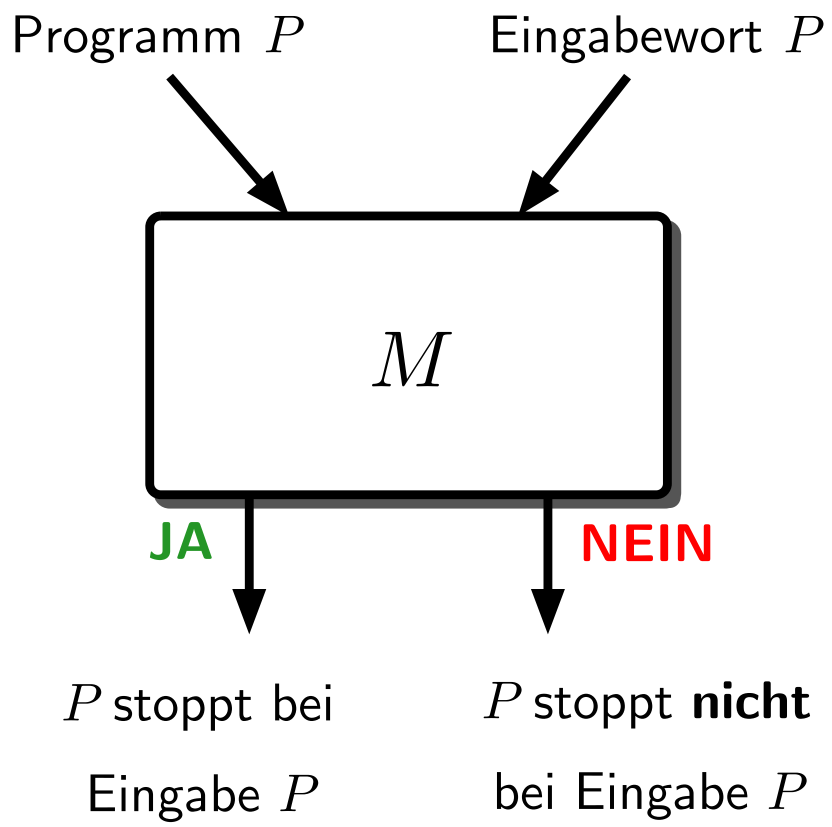
Erster Beweis: Konstruktion mit Widerspruch

Angenommen M löst das Halteproblem:



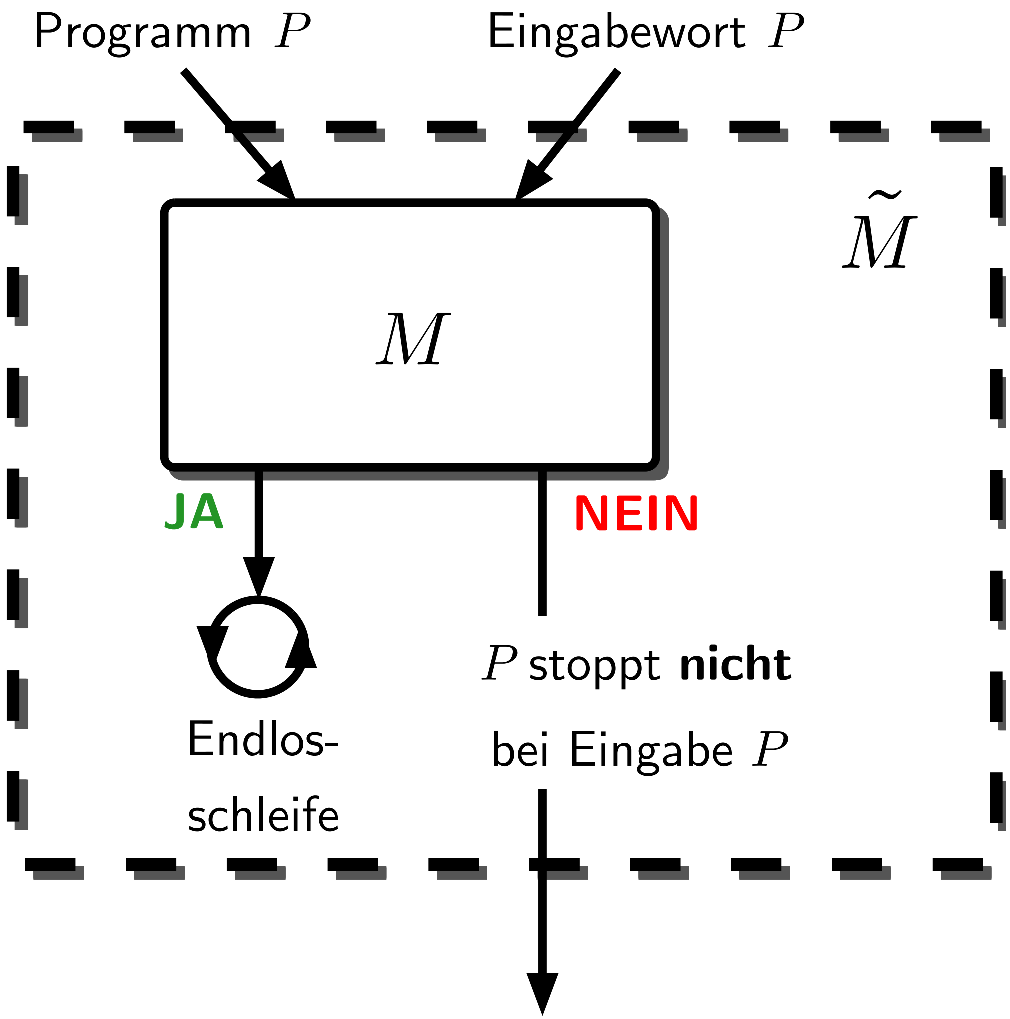
Halteproblem (2)

Eingabewort x ersetzt durch das Programm P:



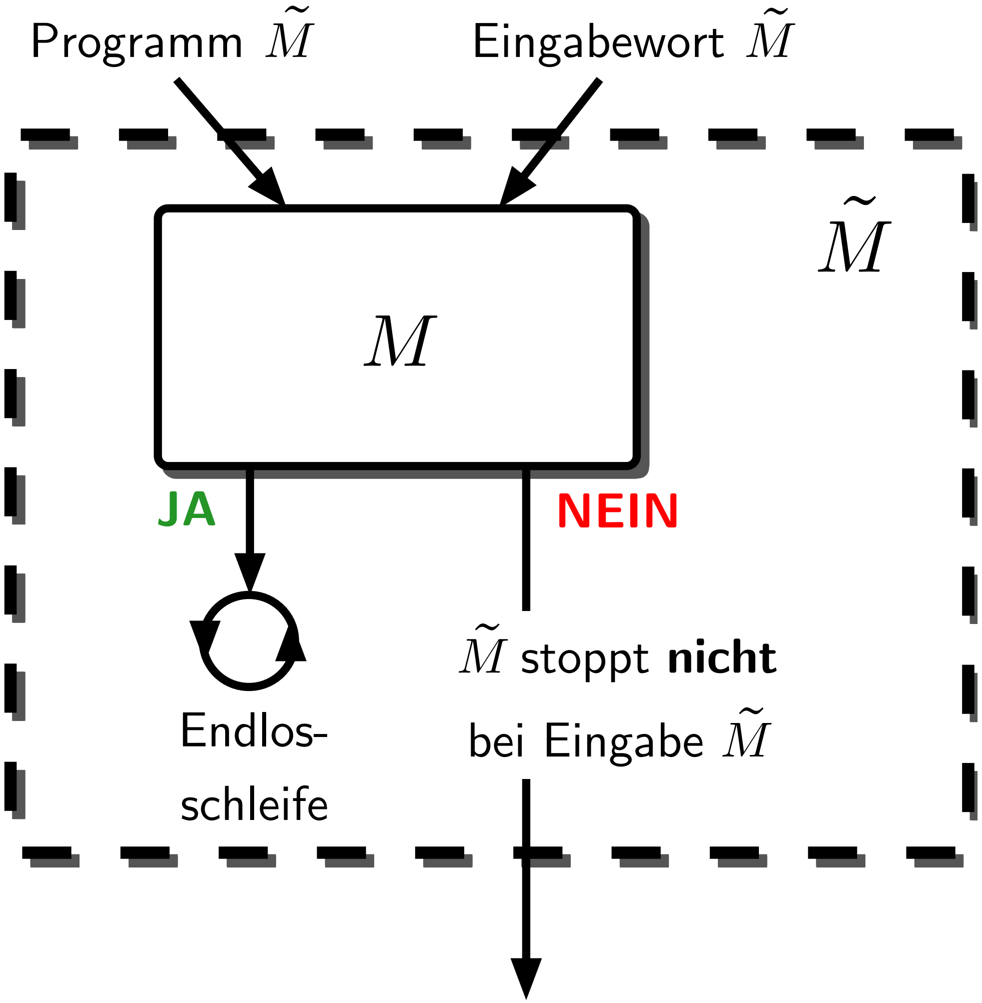
Halteproblem (3)

Neue Maschine $\tilde{M}$ mit Endlosschleife bei JA:



Halteproblem (4)

Als Programm P verwenden wir jetzt $\tilde{M}$:



Halteproblem und L_d

- Kodieren wir L_d und H mit dem Alphabet $\{0, 1\}$.
- Die Unentscheidbarkeit des Halteproblems H kann dann durch Reduktion von $L_d^{\{0,1\}}$ auf H gezeigt werden (bei geeigneter Kodierung):

$$w \in L_d^{\{0,1\}} \iff \langle w \rangle \langle w \rangle \notin H$$

ALSO: Wenn H entscheidbar, dann auch $L_d^{\{0,1\}}$.

Halteproblem ist aufzählbar

$H = \{\langle M\rangle\langle w\rangle \mid$ TM M hält auf $w\}$ ist rekursiv aufzählbar.

Die Begründung dafür ist Übungsaufgabe 3.2!

Ausblick

- Weitere wichtige Unentscheidbarkeitsresultate
- Weitere Berechenbarkeitsmodelle
- Realistische Maschinenmodelle