VL-05: Unentscheidbarkeit II

(Berechenbarkeit und Komplexität, WS 2018)

Gerhard Woeginger

WS 2018, RWTH

Organisatorisches

- Nächste Vorlesung:
  Freitag, November 16, 16:30–18:00 Uhr, Audimax

- Webseite:
  http://algo.rwth-aachen.de/Lehre/WS1819/BuK.php

Wiederholung

Wdh.: Abzählbarkeit

Definition: Abzählbare Menge

Eine Menge M heisst abzählbar,
wenn M leer ist oder
wenn es eine surjektive Funktion $c : \mathbb{N} \to M$ gibt.

Abzählbar: endliche Mengen; $\mathbb{N}, \mathbb{Z}, \mathbb{Q}$; $\{0,1\}^*$;
Menge der Gödelnummern; Menge der TMen; Menge der Algorithmen

Satz

Die Menge $\mathcal{P}(\mathbb{N})$ ist überabzählbar (= nicht abzählbar).

Überabzählbar: $\mathbb{R}$; $\mathcal{P}(\mathbb{N})$; $\mathcal{P}(\{0,1\}^*)$; Menge der Berechnungsprobleme

Triviale Schlussfolgerung

Es existieren nicht-berechenbare Probleme.

Wdh.: Unentscheidbarkeit der Diagonalsprache

Definition (Diagonalsprache)

$$D = \{ w \in \{0,1\}^* \mid w = w_i \text{ und } M_i \text{ akzeptiert } w \text{ nicht} \}$$

Satz

Die Diagonalsprache D ist nicht entscheidbar.

Beweis: Durch Diagonalisierung

Wdh.: Unentscheidbarkeit der Diagonalsprache

$$A_{i,j} = \begin{cases} 1 & \text{falls } M_i \text{ das Wort } w_j \text{ akzeptiert} \\ 0 & \text{sonst} \end{cases}$$

	w_0	w_1	w_2	w_3	w_4	
M_0	**0**	1	1	0	1	$\cdots$
M_1	1	**0**	1	0	1	$\cdots$
M_2	0	0	**1**	0	1	$\cdots$
M_3	0	1	1	**1**	0	$\cdots$
M_4	0	1	0	0	**0**	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		

Die Diagonalsprache lässt sich von der Diagonale der Matrix ablesen. Es gilt

$$D = \{w_i \mid A_{i,i} = 0\}$$

Wdh.: Bisher betrachtete unentscheidbare Probleme

Die folgenden Probleme sind nicht entscheidbar:

Die Diagonalsprache:

$$D = \{w \in \{0,1\}^* \mid w = w_i \text{ und } M_i \text{ akzeptiert } w \text{ nicht}\}$$

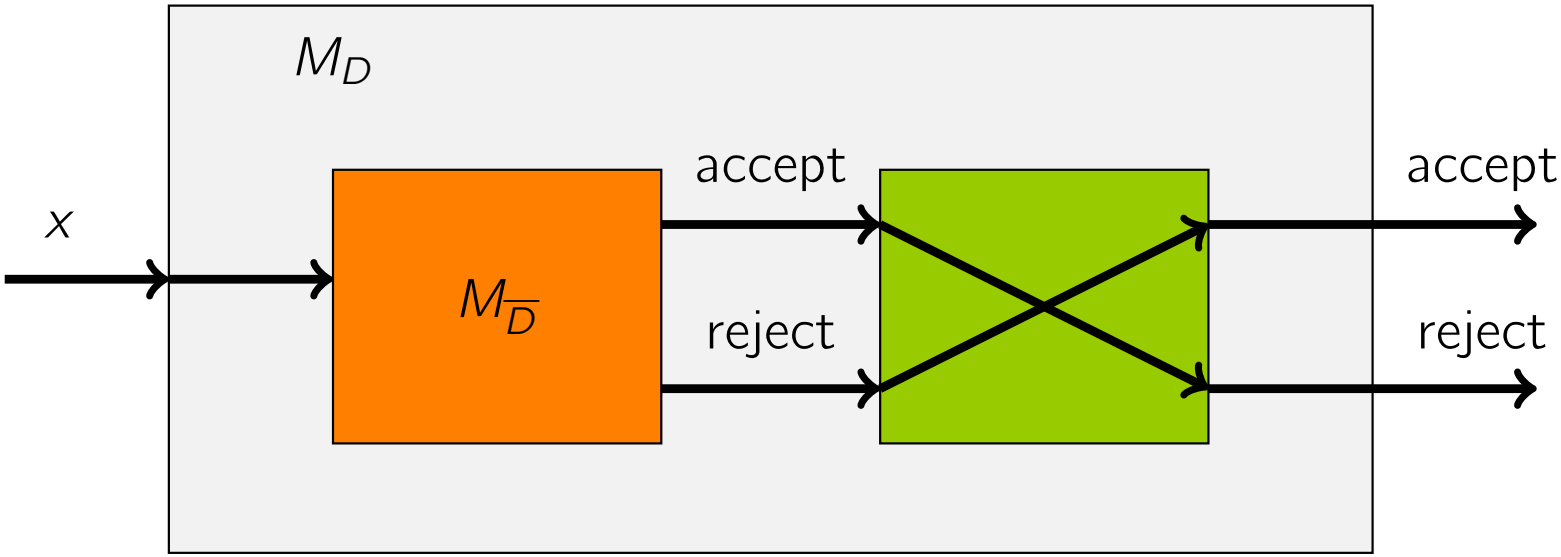
Das Diagonalsprachenkomplement:

$$\overline{D} = \{w \in \{0,1\}^* \mid w = w_i \text{ und } M_i \text{ akzeptiert } w\}$$

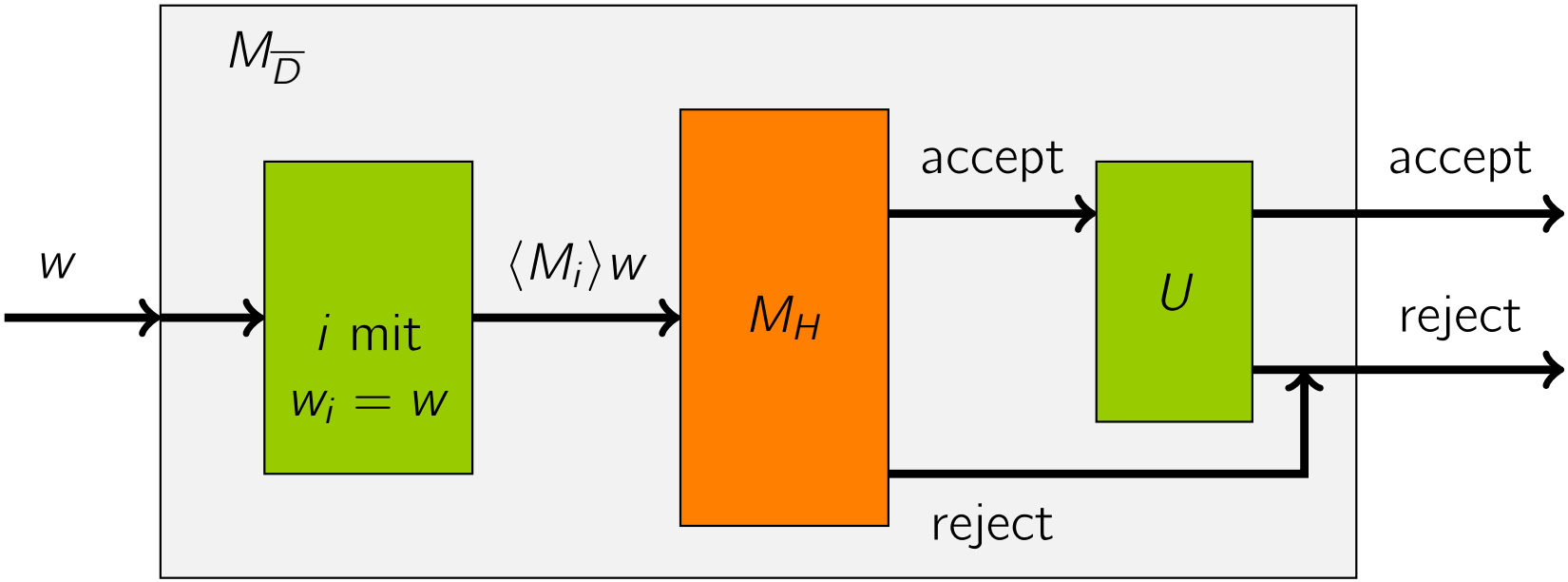
Das Halteproblem:

$$H = \{\langle M \rangle w \mid M \text{ hält auf } w\}$$

Wdh.: Komplement der Diagonalsprache



Wdh.: Halteproblem



Vorlesung VL-05
Unentscheidbarkeit II

- Das Epsilon-Halteproblem

- Der Satz von Rice
- Anwendungen von Rice

Das Epsilon-Halteproblem

Das Epsilon-Halteproblem

Definition (Epsilon-Halteproblem)

$$H_\epsilon = \{\langle M \rangle \mid M \text{ hält auf der Eingabe } \epsilon \}$$

Satz

Das Epsilon-Halteproblem H_ϵ ist nicht entscheidbar.

Beweisidee: Wir benutzen die Unterprogrammtechnik. Aus einer TM M_ϵ, die H_ϵ entscheidet, konstruieren wir eine neue TM M_H, die das (wie wir bereits wissen: nicht-entscheidbare!!) Halteproblem H entscheidet.

Beweis (1)

Die neue TM M_H mit Unterprogramm M_ϵ arbeitet wie folgt:

(1) Falls die Eingabe nicht mit einer korrekten Gödelnummer beginnt, verwirft M_H die Eingabe.

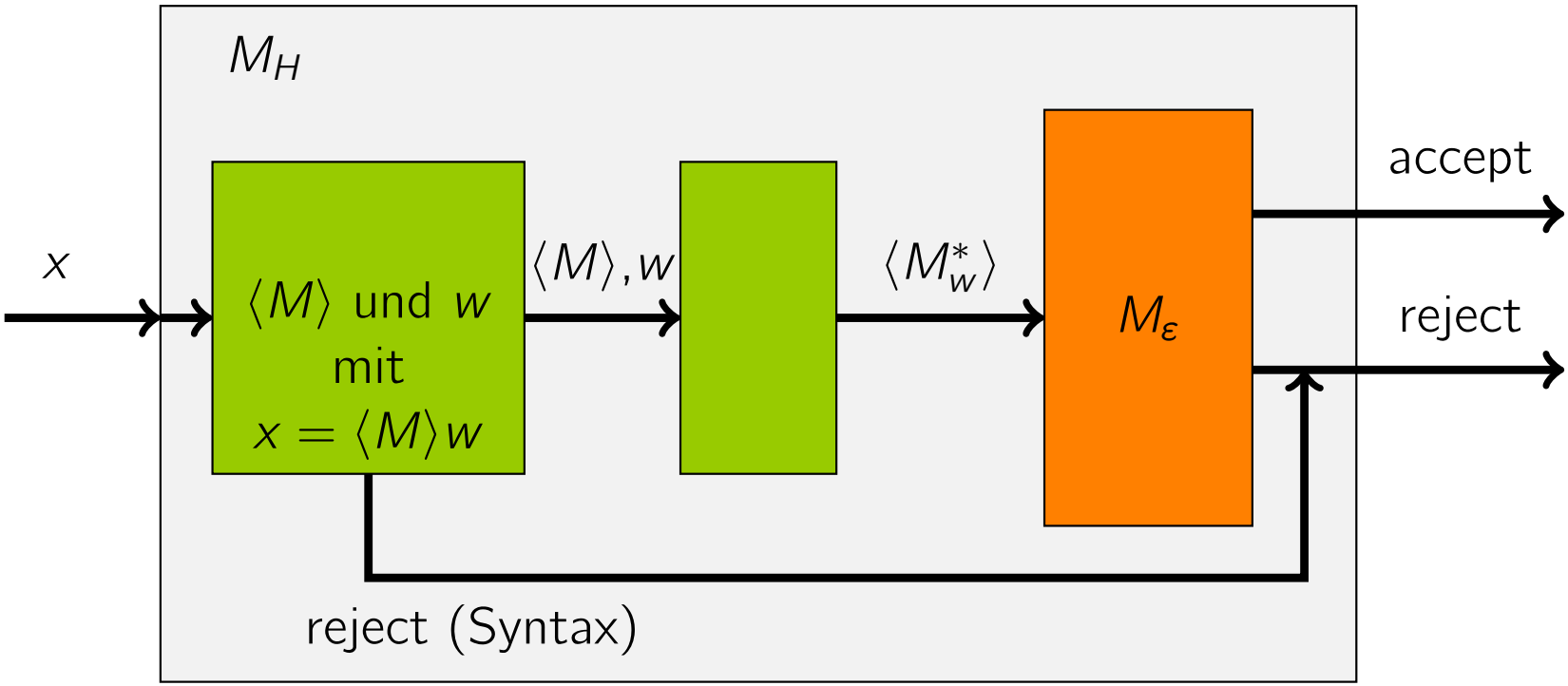
(2) Sonst, also auf Eingaben der Form $\langle M \rangle w$, berechnet M_H die Gödelnummer einer TM M_w^* mit den folgenden Eigenschaften:

- Falls M_w^* die Eingabe ϵ erhält, so schreibt sie zunächst das Wort w aufs Band und simuliert dann die TM M mit der Eingabe w.
- Bei Eingaben ungleich ϵ kann sich M_w^* beliebig verhalten.

(3) M_H startet nun M_ϵ mit der Eingabe $\langle M_w^* \rangle$ und akzeptiert (verwirft) genau dann, wenn M_ϵ akzeptiert (verwirft).

Beweis (2)

Illustration: Aus M_ϵ konstruieren wir M_H.



- Existenz von M_H steht im Widerspruch zur Unentscheidbarkeit von H.
- Daher gibt es M_ϵ nicht.
- Daher ist das Epsilon-Halteproblem H_ϵ nicht entscheidbar.

Beweis (3): Korrektheit

Falls Eingabe nicht von der Form $x = \langle M \rangle w$ ist, verwirft M_H die Eingabe.
Wir nehmen nun an, dass Eingabe der Form $x = \langle M \rangle w$ vorliegt.

Für die Korrektheit ist somit noch zu zeigen:

- $\langle M \rangle w \in H \Rightarrow M_H$ akzeptiert $\langle M \rangle w$
- $\langle M \rangle w \notin H \Rightarrow M_H$ verwirft $\langle M \rangle w$

$$
\begin{aligned}
\langle M \rangle w \in H &\Rightarrow M \text{ hält auf Eingabe } w \\
&\Rightarrow M^*_w \text{ hält auf der Eingabe } \epsilon \\
&\Rightarrow M_\epsilon \text{ akzeptiert } \langle M^*_w \rangle \\
&\Rightarrow M_H \text{ akzeptiert } \langle M \rangle w
\end{aligned}
$$

$$
\begin{aligned}
\langle M \rangle w \notin H &\Rightarrow M \text{ hält nicht auf Eingabe } w \\
&\Rightarrow M^*_w \text{ hält nicht auf der Eingabe } \epsilon \\
&\Rightarrow M_\epsilon \text{ verwirft } \langle M^*_w \rangle \\
&\Rightarrow M_H \text{ verwirft } \langle M \rangle w
\end{aligned}
$$

Entscheidbar vs unentscheidbar

Unentscheidbar versus entscheidbar (1)

Wir haben gesehen, dass die folgenden Probleme unentscheidbar sind:

- Gegeben $\langle M \rangle$ und w, gilt $w \in L(M)$?
- Gegeben $\langle M \rangle$, gilt $\varepsilon \in L(M)$?

Analoge Argumente zeigen, dass folgende Probleme unentscheidbar sind:

Übung

- Gegeben $\langle M \rangle$, gilt $\langle M \rangle \in L(M)$?
- Gegeben $\langle M \rangle$, ist $L(M)$ leer?
- Gegeben $\langle M \rangle$, gilt $L(M) = \Sigma^*$?
- Gegeben $\langle M \rangle$, ist $L(M)$ endlich?
- Gegeben $\langle M \rangle$, ist $L(M)$ unendlich?
- Gegeben $\langle M \rangle$, ist $L(M)$ regulär?
- Gegeben $\langle M \rangle$, ist $L(M)$ kontext-frei?

Unentscheidbar versus entscheidbar (2)

Andrerseits ist jedes der folgenden Probleme entscheidbar:

- Gegeben $\langle M \rangle$, gilt $L(M) \subseteq \{0,1\}^*$?
- Gegeben $\langle M \rangle$, wird $L(M)$ von einer TM akzeptiert?
- Gegeben $\langle M \rangle$, gilt $2222 \in L(M)$?
- Gegeben $\langle M \rangle$, gilt $2222 \notin L(M)$?
- Gegeben $\langle M \rangle$, hat M eine gerade Anzahl von Zuständen?
- Gegeben $\langle M \rangle$, besitzt M einen Endzustand?
- Gegeben $\langle M \rangle$, ist $|\langle M \rangle|$ eine Primzahl?

Partielle Funktionen

TM-berechenbare Funktionen sind partielle Funktionen: Im Allgemeinen halten TMen nicht auf jeder Eingabe und berechnen partielle Funktionen. Das können wir wie folgt formalisieren:

- Die von einer TM M berechnete Funktion ist von der Form

  $$f_M \colon \{0,1\}^* \to \{0,1\}^* \cup \{\bot\}$$

  Das Zeichen $\bot$ steht dabei für undefiniert und bedeutet, dass die Maschine nicht hält.

- Im Fall von Entscheidungsproblemen ist die Funktion von der Form

  $$f_M \colon \{0,1\}^* \to \{0,1,\bot\}$$

Dabei steht 0 für Verwerfen, 1 für Akzeptieren und $\bot$ für Nicht-Halten.

Der Satz von Rice

Henry Gordon Rice (1920–2003)

Wikipedia: Henry Gordon Rice was an American logician and mathematician, best known as the author of Rice's theorem, which he proved in his doctoral dissertation of 1951 at Syracuse University.

He was also a Professor of Mathematics at the University of New Hampshire. After 1960 he was employed by Computer Sciences Corporation in El Segundo.

Mathematics Genealogy Project

- Henry G. Rice
- Ph.D.: Syracuse University 1951
- Dissertation:
  Classes of Recursively Enumerable Sets and Their Decision Problems
- Advisor: Paul Charles Rosenbloom
- No students known

Satz von Rice

Satz

Es sei $\mathcal{R}$ die Menge der von TMen berechenbaren partiellen Funktionen.
Es sei $\mathcal{S}$ eine Teilmenge von $\mathcal{R}$ mit $\emptyset \subsetneq \mathcal{S} \subsetneq \mathcal{R}$.

Dann ist die Sprache

$$L(\mathcal{S}) = \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\}$$

nicht entscheidbar.

Mit anderen Worten: Alle nicht-trivialen Aussagen über die von einer TM berechnete Funktion sind unentscheidbar.

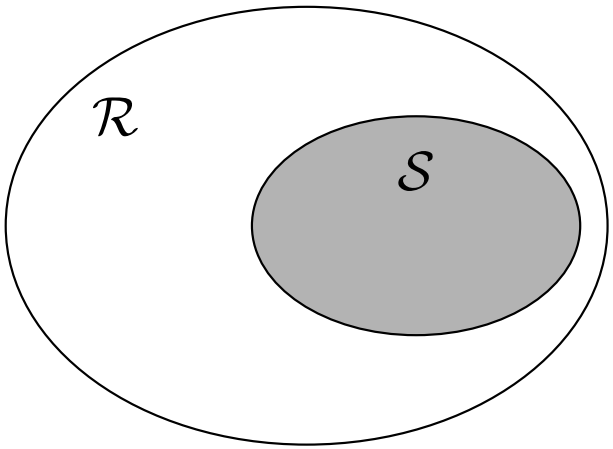
Anwendungsbeispiele (1)

Beispiel 1

- Es sei $\mathcal{S} = \{f_M \mid f_M(\epsilon) \neq \perp\}$.
- Dann ist

$$\begin{aligned} L(\mathcal{S}) &= \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\} \\ &= \{\langle M \rangle \mid M \text{ hält auf Eingabe } \epsilon\} \\ &= H_\epsilon \end{aligned}$$

Gemäss dem Satz von Rice ist das Epsilin-Halteproblem H_ϵ also nicht entscheidbar. (Aber das wussten wir ja schon.)



Anwendungsbeispiele (2)

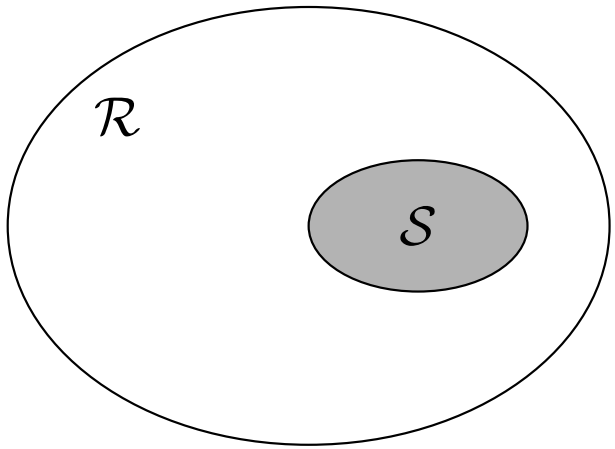
Beispiel 2

- Es sei $\mathcal{S} = \{f_M \mid \forall w \in \{0,1\}^* : f_M(w) \neq \perp\}$.
- Dann ist

$$\begin{aligned} L(\mathcal{S}) &= \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\} \\ &= \{\langle M \rangle \mid M \text{ hält auf jeder Eingabe}\} \end{aligned}$$

- Diese Sprache ist auch als das *totale Halteproblem* H_{tot} bekannt.

Gemäss dem Satz von Rice ist die Sprache H_{tot} nicht entscheidbar.



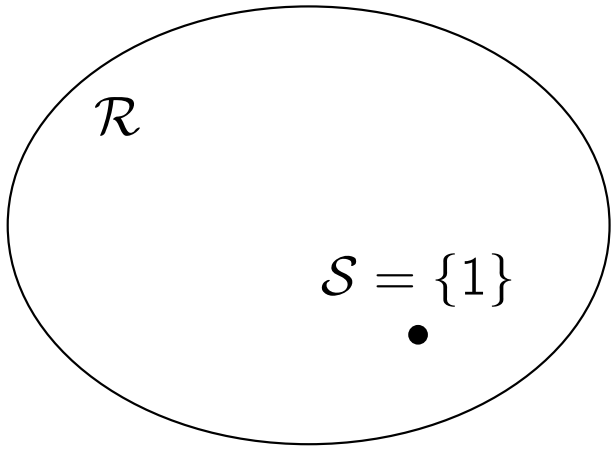
Anwendungsbeispiele (3)

Beispiel 3

- Es sei $\mathcal{S} = \{f_M \mid \forall w \in \{0,1\}^* : f_M(w) = 1\}$.
- Dann ist

$$\begin{aligned} L(\mathcal{S}) &= \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\} \\ &= \{\langle M \rangle \mid M \text{ hält auf jeder Eingabe mit Ausgabe 1}\} \end{aligned}$$

Gemäss dem Satz von Rice ist die Sprache $L(\mathcal{S})$ nicht entscheidbar.



Beweis des Satzes von Rice

Beweis des Satzes von Rice (0)

Hier ist noch einmal der Wortlaut des Satzes:

Satz

Es sei $\mathcal{R}$ die Menge der von TMen berechenbaren partiellen Funktionen.
Es sei $\mathcal{S}$ eine Teilmenge von $\mathcal{R}$ mit $\emptyset \subsetneq \mathcal{S} \subsetneq \mathcal{R}$.

Dann ist die Sprache

$$L(\mathcal{S}) = \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\}$$

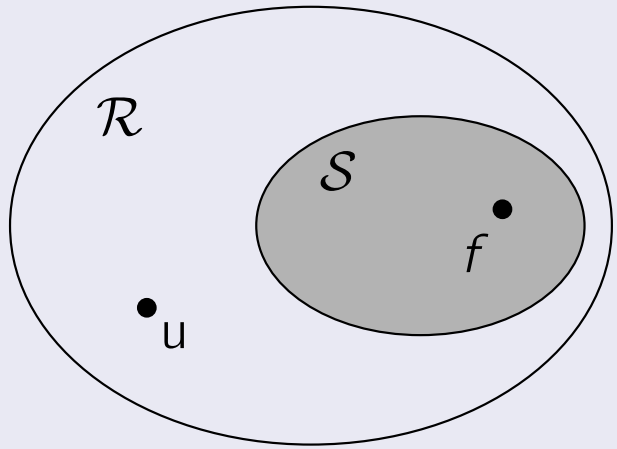
nicht entscheidbar.

Beweis des Satzes von Rice (1)

Wir benutzen die Unterprogrammtechnik:
Aus einer TM $M_{L(\mathcal{S})}$, die $L(\mathcal{S})$ entscheidet, konstruieren wir eine TM M_{H_ϵ}, die das Epsilon-Halteproblem H_ϵ entscheidet.

Einige Vereinbarungen:

- Es sei u die überall undefinierte Funktion $u(w) \equiv \perp$.
- O.B.d.A. $u \notin \mathcal{S}$.
- Es sei f eine Funktion aus $\mathcal{S}$.
- Es sei N eine TM, die f berechnet.



Anmerkung: Falls $u \in \mathcal{S}$ gilt, so betrachten wir einfach das Komplement $\mathcal{R} \setminus \mathcal{S}$ statt $\mathcal{S}$, und zeigen die Unentscheidbarkeit von $L(\mathcal{R} \setminus \mathcal{S})$. Hieraus ergibt sich auch unmittelbar die Unentscheidbarkeit von $L(\mathcal{S})$.

Beweis des Satzes von Rice (2)

Die neue TM M_{H_ϵ} mit dem Unterprogramm $M_{L(\mathcal{S})}$ arbeitet wie folgt:

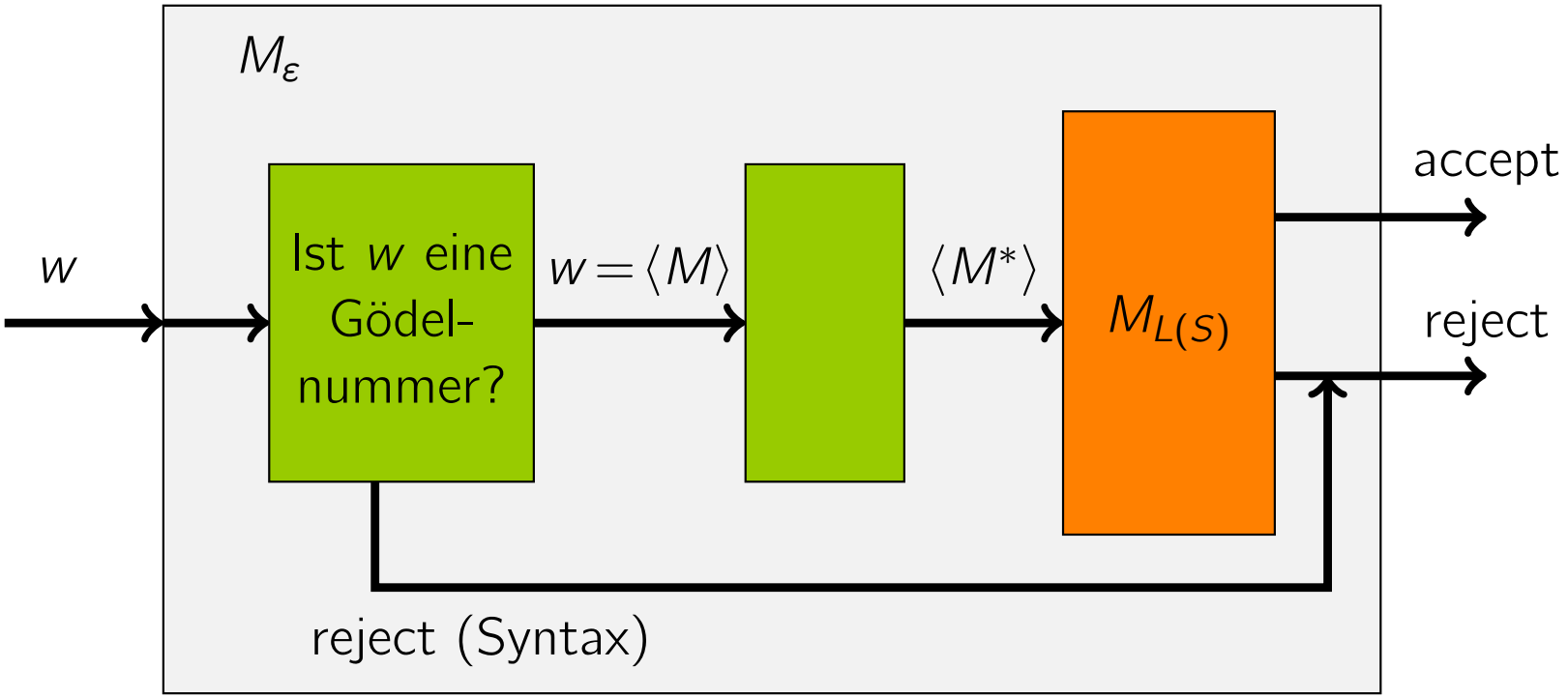
(1) Falls die Eingabe nicht aus einer korrekten Gödelnummer besteht, so verwirft M_{H_ϵ} die Eingabe.

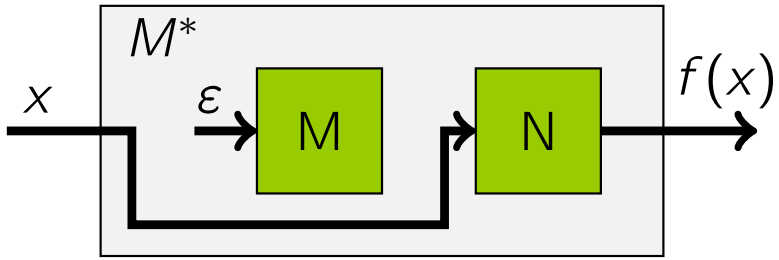
(2) Andernfalls berechnet M_{H_ϵ} aus der Eingabe $\langle M \rangle$ die Gödelnummer der TM M^*:

Verhalten von M^* auf Eingabe x

- Zuerst simuliert M^* das Verhalten von TM M bei Eingabe ϵ auf einer für diesen Zweck reservierten Spur.
- Danach simuliert M^* das Verhalten von TM N bei Eingabe x. M^* hält, sobald N hält, und übernimmt die Ausgabe.

(3) Schlussendlich starten wir $M_{L(\mathcal{S})}$ mit der Eingabe $\langle M^* \rangle$. Wir akzeptieren (verwerfen) genau dann, wenn $M_{L(\mathcal{S})}$ akzeptiert (verwirft).


M_ε
w
Ist w eine Gödel-nummer?
$w = \langle M \rangle$
$\langle M^* \rangle$
$M_{L(S)}$
accept
reject
reject (Syntax)


M^*
x
ε
M
N
$f(x)$

Beweis des Satzes von Rice (3)

Korrektheit:

- Bei Eingabe von w, wobei w keine Gödelnummer ist, verwirft M_{H_ϵ}.
- Bei Eingabe von $w = \langle M \rangle$ gilt:

$$
\begin{aligned}
w \in H_\epsilon \quad &\Rightarrow \quad M \text{ hält auf } \epsilon \\
&\Rightarrow \quad M^* \text{ berechnet } f \\
&\overset{f \in \mathcal{S}}{\Rightarrow} \quad \langle M^* \rangle \in L(\mathcal{S}) \\
&\Rightarrow \quad M_{L(\mathcal{S})} \text{ akzeptiert } \langle M^* \rangle \\
&\Rightarrow \quad M_{H_\epsilon} \text{ akzeptiert } w
\end{aligned}
$$

$$
\begin{aligned}
w \notin H_\epsilon \quad &\Rightarrow \quad M \text{ hält nicht auf } \epsilon \\
&\Rightarrow \quad M^* \text{ berechnet } u \\
&\overset{u \notin \mathcal{S}}{\Rightarrow} \quad \langle M^* \rangle \notin L(\mathcal{S}) \\
&\Rightarrow \quad M_{L(\mathcal{S})} \text{ verwirft } \langle M^* \rangle \\
&\Rightarrow \quad M_{H_\epsilon} \text{ verwirft } w
\end{aligned}
$$

Satz von Rice für Java Programme

Konsequenzen für Java:

Es gibt keine algorithmische Methode (von Hand oder automatisiert; heute oder morgen oder in ferner Zukunft) um festzustellen, ob ein gegebenes Java Programm einer nicht-trivialen Spezifikation entspricht.

Analoge Konsequenzen gelten für alle anderen höheren Programmiersprachen wie C, C++, Pascal, Algol, COBOL, Python, FORTRAN, LISP, Prolog, Haskell, Scala, Idris, etc.

Weitere Anwendungsbeispiele

Anwendungsbeispiele (4)

Beispiel 4

- Es sei
  $L_{17} = \{\langle M \rangle \mid M \text{ berechnet bei Eingabe der Zahl 17 die Zahl 42}\}$.
- Es ist $L_{17} = L(\mathcal{S})$ für $\mathcal{S} = \{f_M \mid f_M(\text{bin}(17)) = \text{bin}(42)\}$.
- Da $\emptyset \subsetneq \mathcal{S} \subsetneq \mathcal{R}$ gilt, ist diese Sprache L_{17} gemäss dem Satz von Rice nicht entscheidbar.

Anwendungsbeispiele (5)

Beispiel 5

- Es sei $H_{32} = \{\langle M \rangle \mid$ auf jeder Eingabe hält M nach höchstens 32 Schritten $\}$.
- Über diese Sprache sagt der Satz von Rice nichts aus!

Ist H_{32} entscheidbar?

Anwendungsbeispiele (6)

Beispiel 6

- Es sei $L_{44} = \{\langle M \rangle \mid$ Es existiert ein Wort w, sodass die TM M bei Abbarbeitung von w mindestens einmal im Zustand q_{44} ist $\}$.
- Über diese Sprache sagt der Satz von Rice nichts aus!

Ist L_{44} entscheidbar?

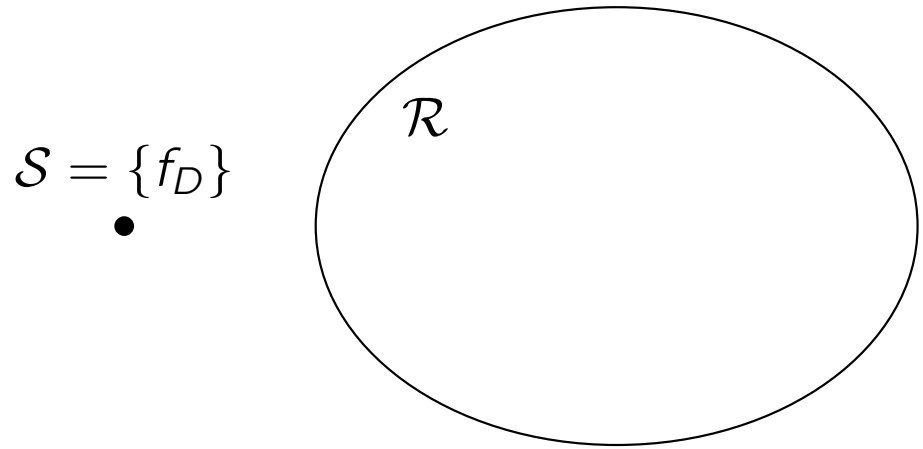
Anwendungsbeispiele (7)

Beispiel 7

- Es sei $L_D = \{\langle M \rangle \mid M \text{ entscheidet die Diagonalsprache}\}$.
- Dann ist $L_D = L(\mathcal{S})$ für $\mathcal{S} = \{f_D\}$ wobei

$$f_D(w) = \begin{cases} 1 & \text{wenn } w \in D \\ 0 & \text{sonst.} \end{cases}$$

- Über diese Sprache sagt der Satz von Rice nichts aus!
- Aber: Diese Sprache ist entscheidbar, denn $L_D = \{\}$.



Noch einmal: Das Collatz Problem

Die Collatz'sche Iterationsgleichung lautet:

$$x \leftarrow \begin{cases} x/2 & \text{wenn } x \text{ gerade} \\ 3x+1 & \text{wenn } x \text{ ungerade} \end{cases}$$

Collatz Problem

Erreicht die Collatz'sche Iteration von jedem natürlichen Startwert x aus irgendwann einmal die Zahl $x = 1$?

- Das Collatz-Problem ist eine konkrete Instanz des totalen Halteproblems.
- Wir wissen nicht, ob diese Instanz eine Ja- oder eine Nein-Instanz ist.
- Der Satz von Rice ist für konkrete Probleminstanzen nutzlos.