Informatik III

CoNe Freiburg

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Arne Vater

Wintersemester 2006/07
12. Vorlesung
01.12.2006

Was heißt abzählbar im Gegensatz zu rekursiv aufzählbar?

- **Definition**
  - Eine Menge M heißt abzählbar, wenn es eine (nicht unbedingt berechenbare) Funktion $f:\mathbf{N} \rightarrow M$ gibt,
    - so dass für jedes $m \in M$ eine natürliche Zahl $i \in \mathbf{N}$ gibt mit $f(i) = m$.
- **Lemma**
  - Jede rekursiv aufzählbare Menge ist abzählbar
  - Jede Teilmenge einer abzählbaren Menge ist abzählbar

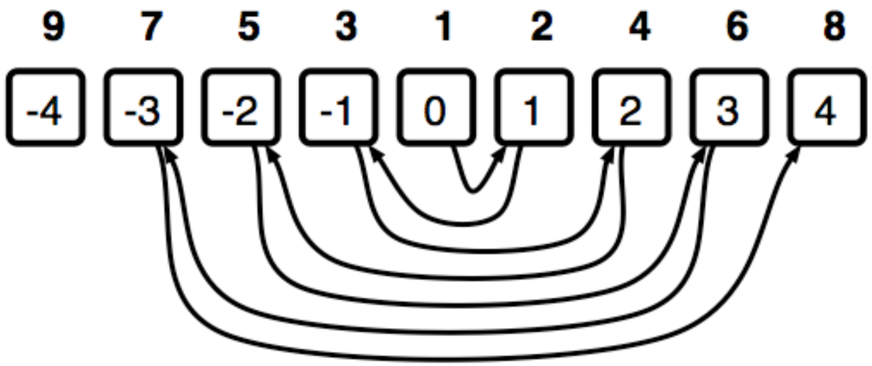
Hilberts Hotel

- **Hilberts Hotel hat unendlich viele Zimmer**
- **Alle Zimmer sind ausgebucht**
  - Es sind also schon unendlich viele Gäste da!

- **Kann der Hotelier dennoch weitere Gäste aufnehmen?**
  - Ein neuer Gast
  - Ein Bus mit unendlich vielen neuen Gästen
  - Unendlich viele Busse mit unendlich vielen neuen Gästen

Die ganzen Zahlen sind abzählbar

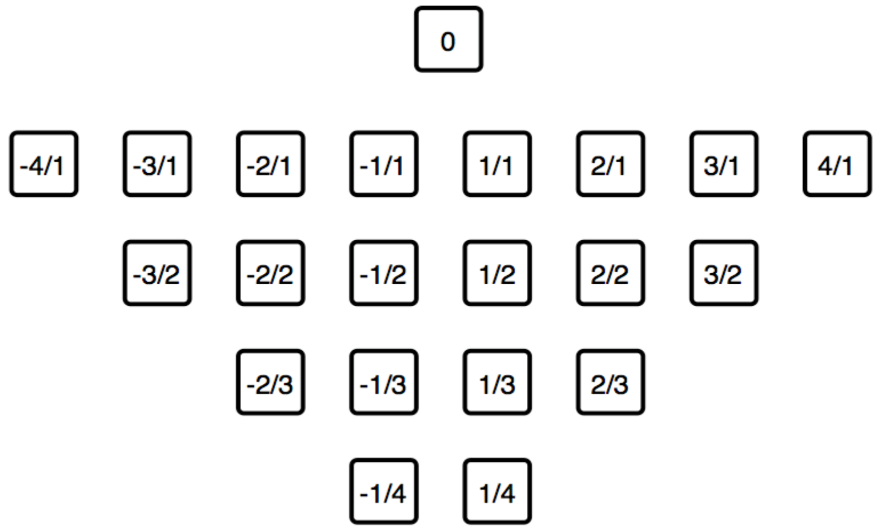
- **Theorem**
  - Die Menge der ganzen Zahlen ist abzählbar
- **Beweis**
  - Konstruiere eine Abzählung aller ganzen Zahlen:
  - 0 erhält die Nummer 1
  - alle positiven Zahlen x erhalten die Nummer 2x
  - alle negativen Zahlen x erhalten die Nummer 2(-x)+1

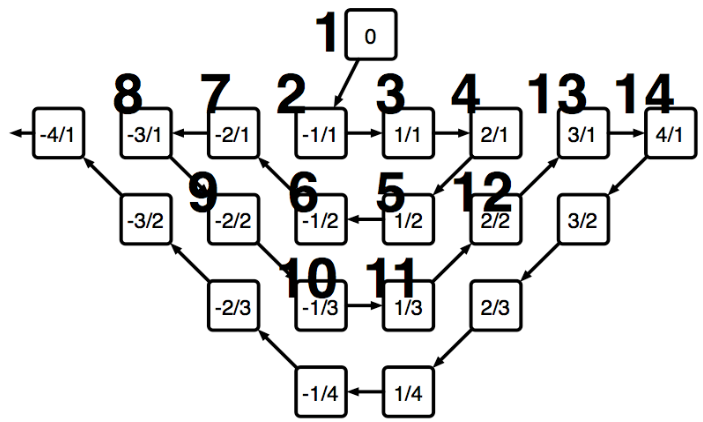


Die rationalen Zahlen sind abzählbar

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**
  - Die Menge der rationalen Zahlen ist abzählbar
- **Beweis**
  - Die rationalen Zahlen sind definiert als Tupel aus einer ganzen Zahl und einer natürlichen Zahl
  - Zähle alle diese Paare geeignet auf
    - Mehrfachaufzählungen sind irrevelant (=egal)

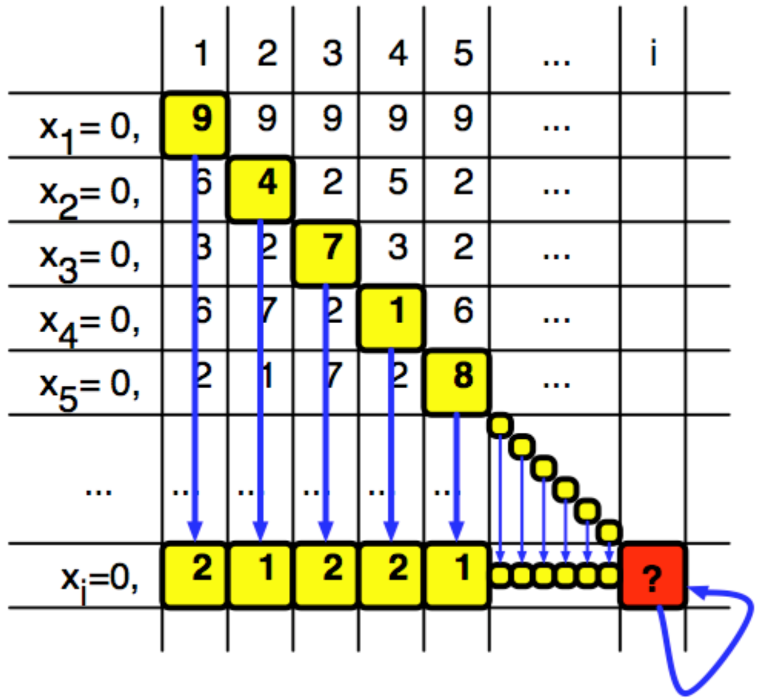




Die reellen Zahlen sind nicht abzählbar

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**
  - Die Menge der reellen Zahlen sind nicht abzählbar
- **Beweis**
  - Betrachte alle reellen Zahlen aus [0,1[in der Dezimaldarstellung
  - Angenommen alle reellen Zahlen sind abzählbar mit $x_1,x_2,x_3,\ldots$
  - Betrachte reelle Zahl z, wobei
    - j-te Ziffer ist 1, falls j-te Ziffer von x_j gerade ist
    - j-te Ziffer ist 2, falls j-te Ziffer von x_j ungerade ist
  - Diese Zahl hat auch einen Index i,
    - d.h. es gib ein i mit x_i=z
    - wenn die Annahme stimmt
  - Ist die i-te Ziffer von x_i jetzt gerade oder ungerade
    - weder noch: Widerspruch
  - Also ist die Annahme falsch



i-te Ziffer von x_i ist 2, falls i-te Ziffer von x_i ungerade
ist 1, falls i-te Ziffer von x_i gerade

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Definition**
  - Eine Menge M heißt abzählbar, wenn es eine (nicht unbedingt berechenbare) Funktion f:**N** → M gibt,
    - so dass für jedes m ∈ M eine natürliche Zahl i ∈ **N** gibt mit f(i) = m.
- **Theorem**
  - Die Menge aller Funktionen die von **N** auf {0,1} abbilden, ist nicht abzählbar.
- **Beweis:**
  - Angenommen es gibt eine Funktion die alle Funktionen $f_1,f_2,\ldots$ abzählt.
  - Betrachte die Funktion $1-f_i(i)$.
  - Diese Funktion ist nicht in $f_1,f_2,\ldots$, da für jedes i gilt $f_i(i) \neq 1-f_i(i)$.
  - Diese Funktion ist also in der Abzählung $f_1,f_2,\ldots$ nicht enthalten (sollte aber).

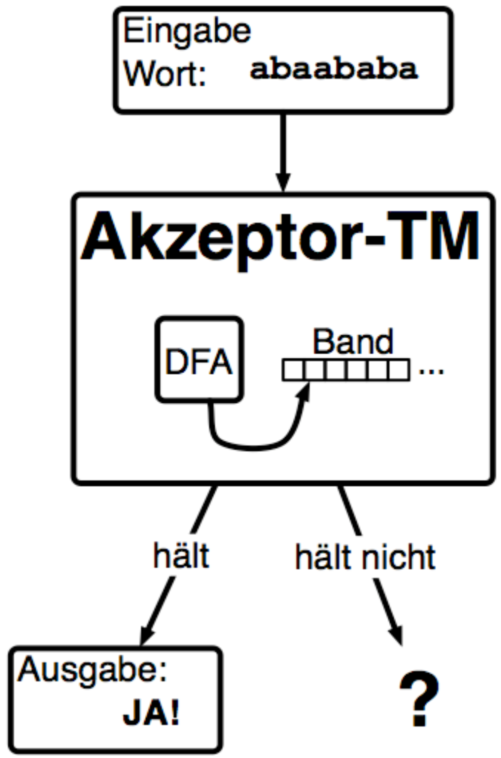
	1	2	3	4	5	...	i	...
$f_1(n)$	**0**	0	0	0	0	...		
$f_2(n)$	1	**1**	1	1	1	...		
$f_3(n)$	0	1	**0**	1	0	...		
$f_4(n)$	1	0	1	**0**	1	...		
$f_5(n)$	0	1	1	0	**1**	...		
...	...	...	...	...	...	...		
$f_i(n)=1-f_n(n)$	**1**	**0**	**1**	**1**	**0**	...	?	

Das TM-Wortproblem

- **Definition**
  - Das Wortproblem der Turing-Maschinen ist definiert als
  - gegeben:
    - eine Turingmaschine M
    - ein Wort w
  - gesucht:
    - akzeptiert M das Wort w?
- **Die alternative Darstellung als Sprache ist:**

$$A_{\mathrm{TM}} = \{\langle \mathrm{M}, \mathrm{w}\rangle \mid M \text{ ist eine TM und } M \text{ akzeptiert } w\}$$

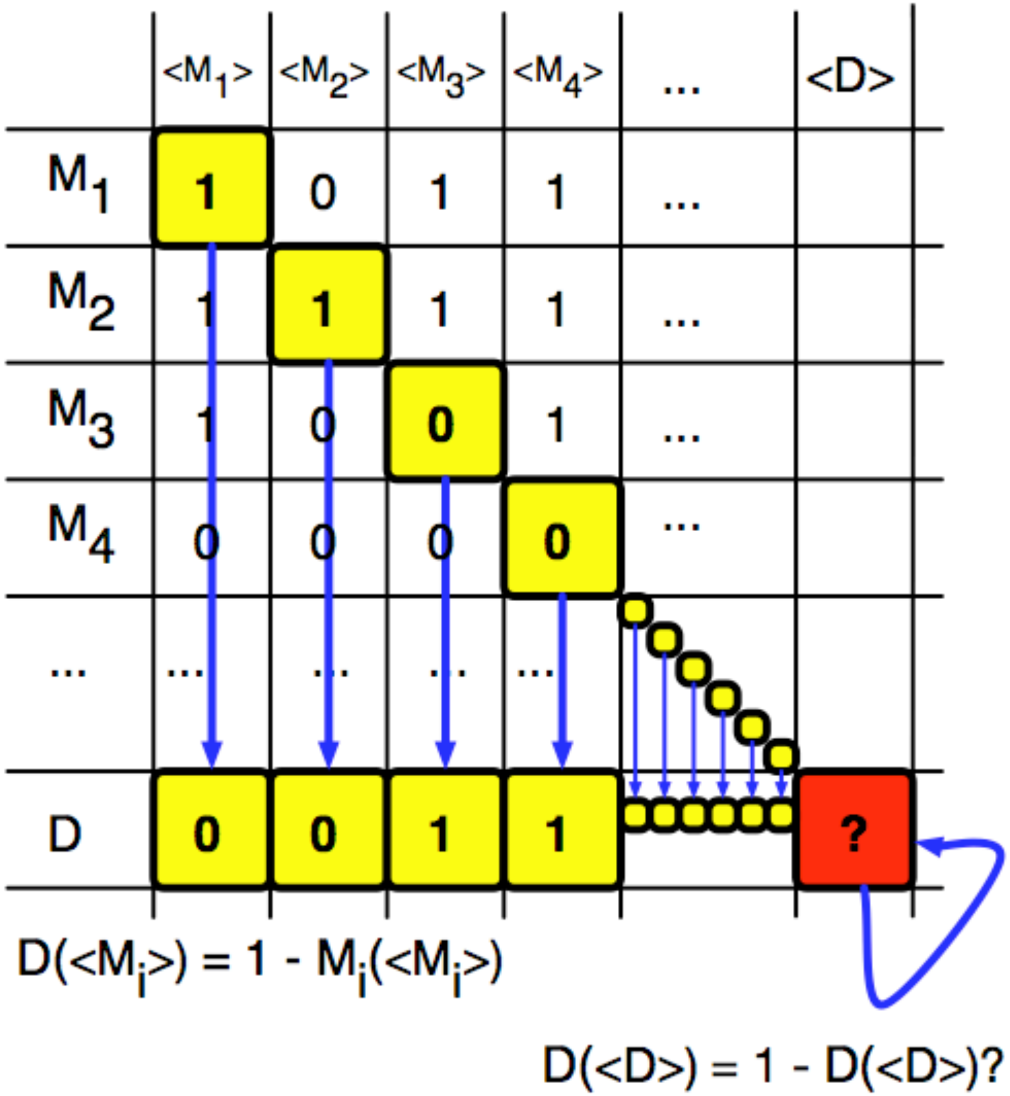
  - hierbei ist <M,w> eine geeignete Kodierung der TM M und des Wortes w



Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

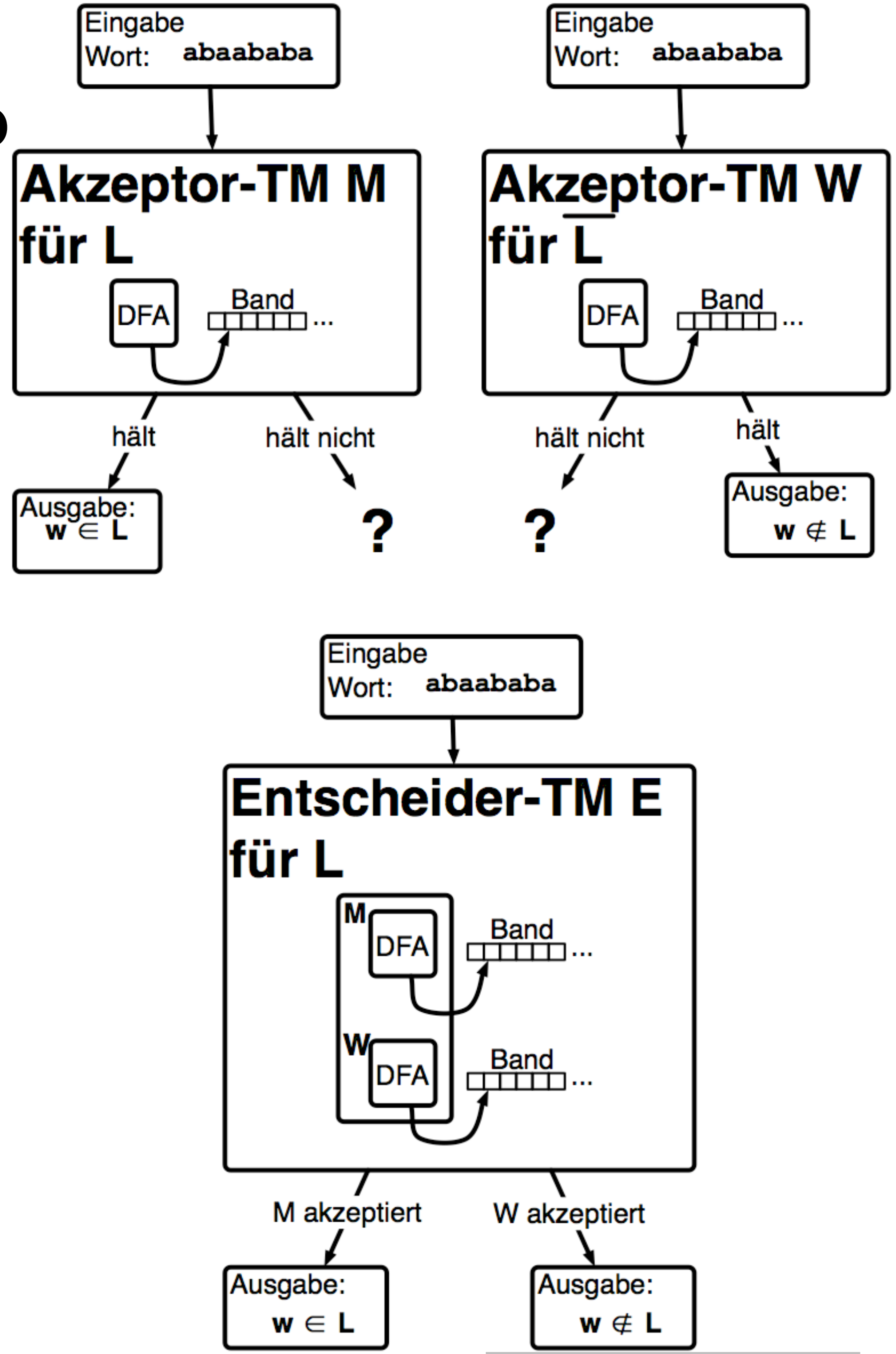
Dial D for Diagonalization!

- Angenommen: Das TM-Wortproblem ist berechenbar
- Dann kann D existieren!
- Das führt zu einem Widerspruch!



Rekursiv = Aufzählbar + Ko Aufzählbar

- **Definition**
  - Eine Sprache ist **rekursiv ko-aufzählbar**, wenn das Komplement der Menge rekursiv aufzählbar ist.
- **Theorem**
  - Eine Sprache L ist rekursiv, genau dann
    - wenn sie rekursiv aufzählbar
    - und rekursiv ko-aufzählbar ist.
- **Beweis (Rückrichtung)**
  - Betrachte Akzeptor-TM M für L
  - und Akzeptor-TM W für $\Sigma^* \setminus L$
  - Konstruiere TM E für Eingabe x
    - Berechne parallel M(x) und W(x)
  - Falls M(x) zuerst akzeptiert:
    - akzeptiere
  - Falls W(x) zuerst akzeptiert:
    - halte und verwerfe
- **Beweis (Hinrichtung):**
  - Jede entscheidbare Sprache ist aufzählbar
  - Das Komplement einer entscheidbaren Sprache ist wiederum entscheidbar:
    - Vertausche akzeptierenden und ablehnenden Zustand



Die Simulator-TM

- **Lemma**
  - Es gibt eine TM, die einen Schritt einer gegebenen TM M und einer Konfiguration berechnen kann,
    - d.h. die Nachfolgekonfiguration ausgeben
- **Beweis**
  - S verfügt über
    - ein Band für Kodierung der 1-Band-TM M
    - ein Band für den aktuellen Zustand
    - ein Band für den Bandinhalt von M mit markierter Kopfposition
    - ein Band für eigene Berechnungen (Zählen)
  - S sucht nach der aktuellen Kopfposition
  - S sucht in der Kodierung von M nach Übergang
  - S schreibt Buchstaben auf das “Band”-Band
  - S bewegt die Kopfposition und markiert entsprechendes Zeichen auf dem Band
  - S schreibt Folgezustand

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Das TM-Wortproblem ist rekursiv aufzählbar

- **Theorem**
  - Die Sprache A_{TM} ist rekursiv aufzählbar.
- **Beweis**
  - Betrachte folgende TM A:
  - A = “Für gegebene TM M und Eingabe x
    - Kodiere Anfangskonfiguration für M und x für Simulator S
    - Solange kein akzeptierende Endkonfiguration
      - S simuliert einen Schritt von M
    - Falls akzeptierende Endkonfiguration erreicht wird, halte und akzeptiere”
- **Beobachtung:**
  - A akzeptiert genau dann, wenn M die Eingabe x akzeptiert

Eine nicht rekursiv aufzählbare Sprache

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem:**
  – Das Komplement der Sprache A_{TM} des TM-Wortproblems ist nicht rekursiv aufzählbar

- **Beweis**
  – Angenommen doch.

  – Dann ist A_{TM} rekursiv aufzählbar und rekursiv ko-aufzählbar

  – dann ist A_{TM} rekursiv (also entscheidbar).

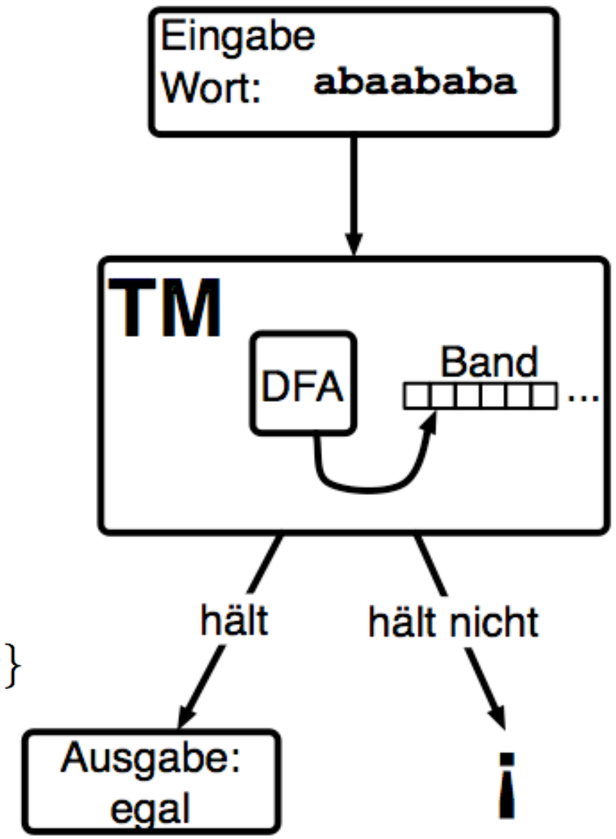
  – **Widerspruch**!

Das Halteproblem

- **Definition**
  - Das Halteproblem der Turing-Maschinen ist definiert als
  - Gegeben:
    - eine Turingmaschine M
    - ein Wort w
  - Gesucht:
    - hält die Turingmaschine auf Eingabe w?
- **Die alternative Darstellung als Sprache ist:**

$$\mathsf{HALT_{TM}} = \{\langle \mathsf{M}, \mathsf{w}\rangle \mid M \text{ ist eine TM und hält auf Eingabe } w\}$$

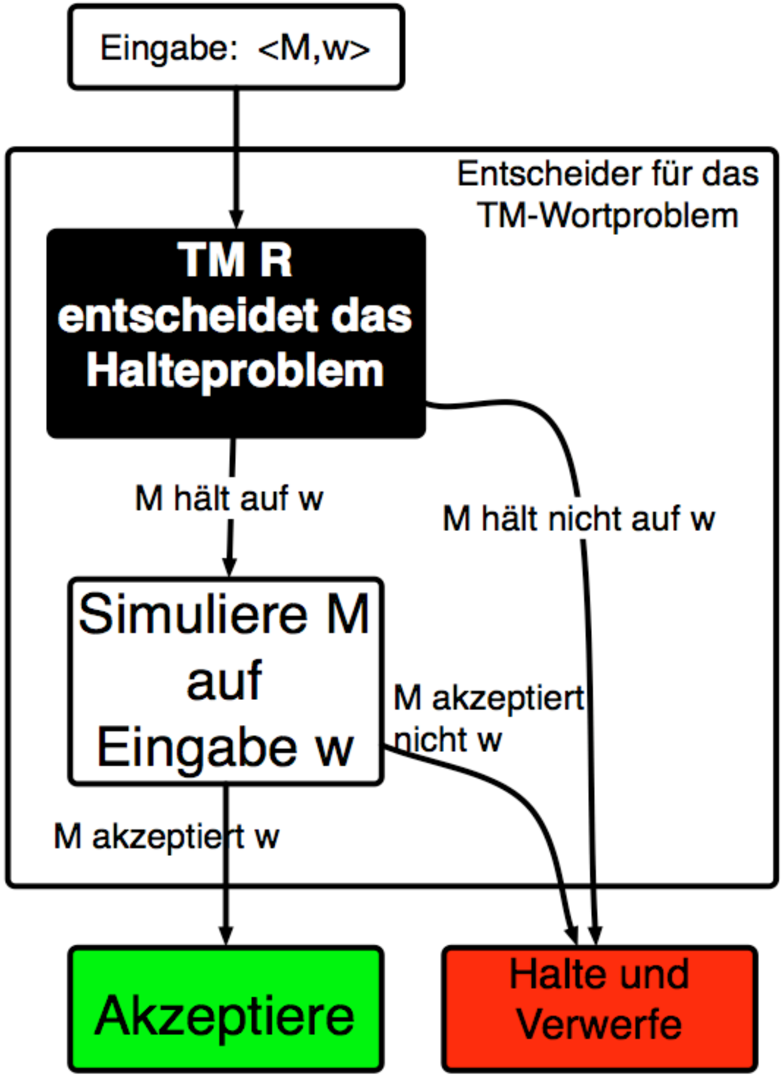
  - hierbei ist <M,w> eine geeignete Kodierung der TM M und des Wortes w



Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Das Halteproblem ist nicht entscheidbar

- **Theorem**
  - $HALT_{TM}$ ist nicht entscheidbar.
- **Beweis**
  - Angenommen, es gibt eine TM R die $HALT_{TM}$ entscheidet.
  - Konstruiere nun TM S wie folgt:
  - S = “Auf Eingabe <M,w>, Kodierung einer TM und Zeichenkette w
    - Lasse R auf <M,w> laufen
    - Falls R verwirft, verwirft S
    - Falls R akzeptiert,
      - simuliere M auf Eingabe w bis M hält
      - Falls M akzeptiert, dann akzeptiert S
      - Falls M verwirft, dann verwirft S”
  - S entscheidet das (unentscheidbare) TM-Wortproblem
  - Widerspruch ⇒ Annahme ist falsch.



Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Kann man entscheiden, ob eine TM überhaupt etwas akzeptiert?

- **Das Leerheitsproblem:**
  - Gegeben
    - eine TM M
  - Gesucht:
    - Akzeptiert M kein einziges Wort?
- **Darstellung als Sprache:**

$$\mathsf{E}_{\mathsf{TM}} = \{\langle \mathsf{M} \rangle \mid M \text{ ist eine TM und } L(M) = \emptyset\}$$

- **Theorem**
  - E_{TM} ist nicht entscheidbar

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**
  - E_{TM} ist nicht entscheidbar

$$E_{TM} = \{\langle M \rangle \mid M \text{ ist eine TM und } L(M) = \emptyset\}$$

- **Beweis**
  - Für gegebene TM M und ein gegebenes Wort w kann man eine TM $A_{M,w}$ bauen mit folgender Funktionalität
  - $A_{M,w}$ = “Auf Eingabe x:
    - Falls $x \neq w$, verwerfe
    - Falls x=w, simuliere M auf Eingabe w
      - akzeptiere falls M das Wort w akzeptiert”

- **Angenommen TM R entscheidet die Sprache E_{TM}**
- Betrachte nun TM S:
- S = “Auf Eingabe <M,w> (= Kodierung einer TM und Zeichenkette w)
  - Berechne die Beschreibung der TM $A_{M,w}$
  - Simuliere R auf Eingabe $<A_{M,w}>$,
  - Falls R akzeptiert, verwirft S
  - Falls R verwirft, akzeptiert S”
- S akzeptiert w gdw. R die Eingabe $<A_{M,w}>$ verwirft
- R verwirft gdw. wenn $A_{M,w}$ mindestens ein Wort akzeptiert
- $A_{M,w}$ akzeptiert gdw. wenn M(w) akzeptiert.
- Also: S entscheidet das (unentscheidbare) TM-Wortproblem
- **Widerspruch $\Rightarrow$ Annahme ist falsch**

Entscheidbare und unentscheidbare Sprachen

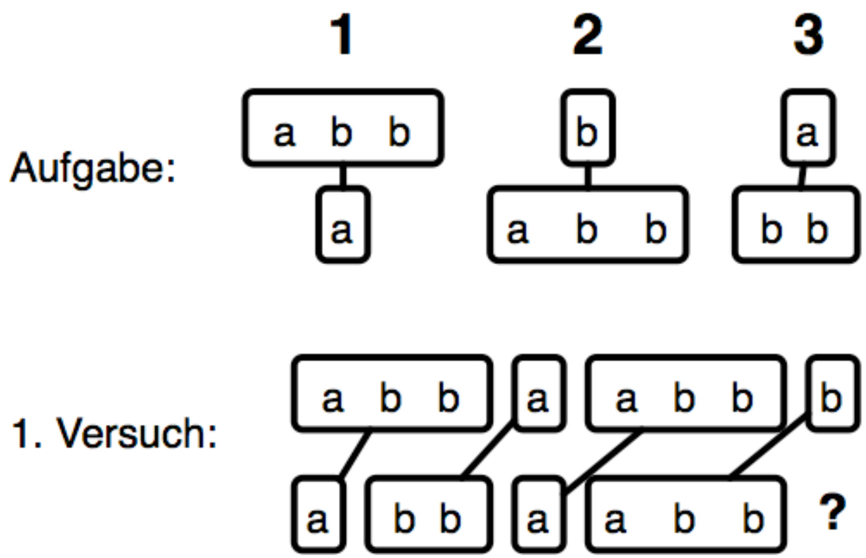
- **Reguläre Sprachen**
  - Wortproblem ist entscheidbar
  - Leerheitsproblem ist entscheidbar
  - Äquivalenzproblem ist entscheidbar
- **Kontextfreie Sprachen**
  - Wortproblem ist entscheidbar
  - Leerheitsproblem ist entscheidbar
  - Äquivalenzproblem ist **nicht** entscheidbar
- **Das Halteproblem**
  - Diagonalisierung
  - Wortproblem der TM ist **nicht** entscheidbar
  - Eine nicht rekursiv aufzählbare Sprache
  - Halteproblem ist **nicht** entscheidbar
  - Leerheitsproblem der TM ist **nicht** entscheidbar

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Ein einfaches unentscheidbares Problem

- **Das Postsche Korrespondenzproblem**
  - Gegeben
    - die Worte $x_1, x_2, .., x_n$ und
    - die Worte $y_1, y_2, .., y_n$
  - Gesucht
    - Gibt es eine Folge von Indizes $i_1, i_2, ..., i_m$ mit $m \geq 1$
    - so dass:

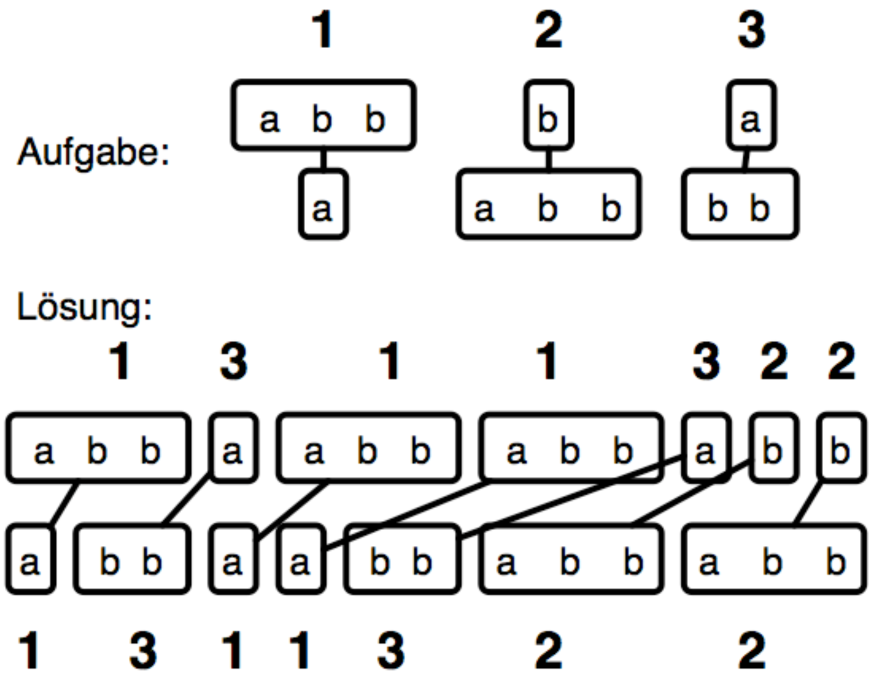
$$x_{i_1} x_{i_2} \ldots x_{i_m} = y_{i_1} y_{i_2} \ldots y_{i_m}$$



- **Theorem**
  - Das Postsche Korrespondenzproblem ist nicht entscheidbar.

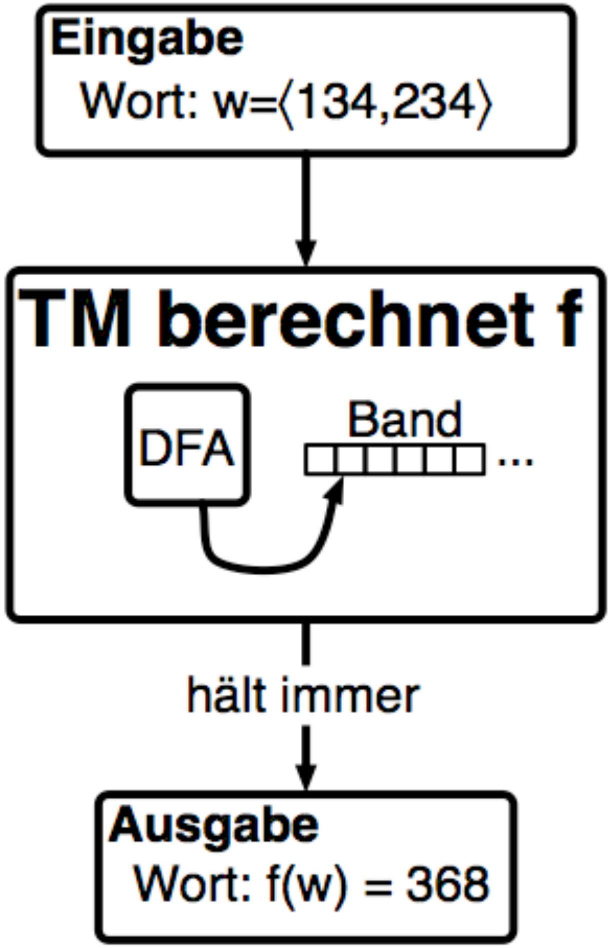
Eine Lösung und eine Anregung

- **Der Beweis der Nichtentscheidbarkeit des Postschen Korrespondenz-Problem kann in Sipsers Buch "Introduction to the Theory of Computation" nachgelesen werden**
  - sehr empfehlenswert



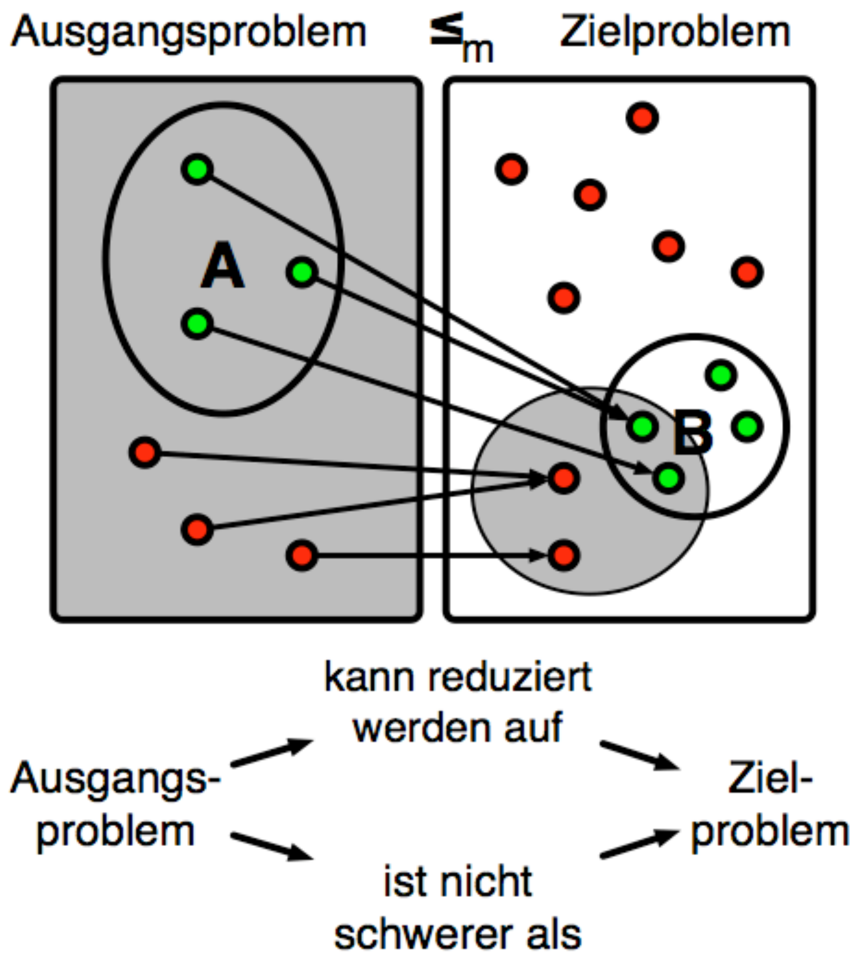
Berechenbare Funktionen

- **Definition**
  - Eine Funktion f: $\Sigma^* \rightarrow \Sigma^*$ ist **berechenbar,** falls eine Turing-Maschine für jede Eingabe w mit dem Ergebnis f(w) auf dem Band hält.
- **Beispiele für berechenbare Funktionen:**
  - Addition, Division, Multiplikation, Vergleich, Sortieren, Division, ...
  - Automatische Generierung von Kodierungen für bestimmte Turingmaschinen
  - Modifizierung der Kodierung einer TM:
    - Kartesisches Produkt
    - Invertierung von Zuständen
    - Initialisierung der Eingabe
    - Verknüpfung mehrerer kodierter TMs



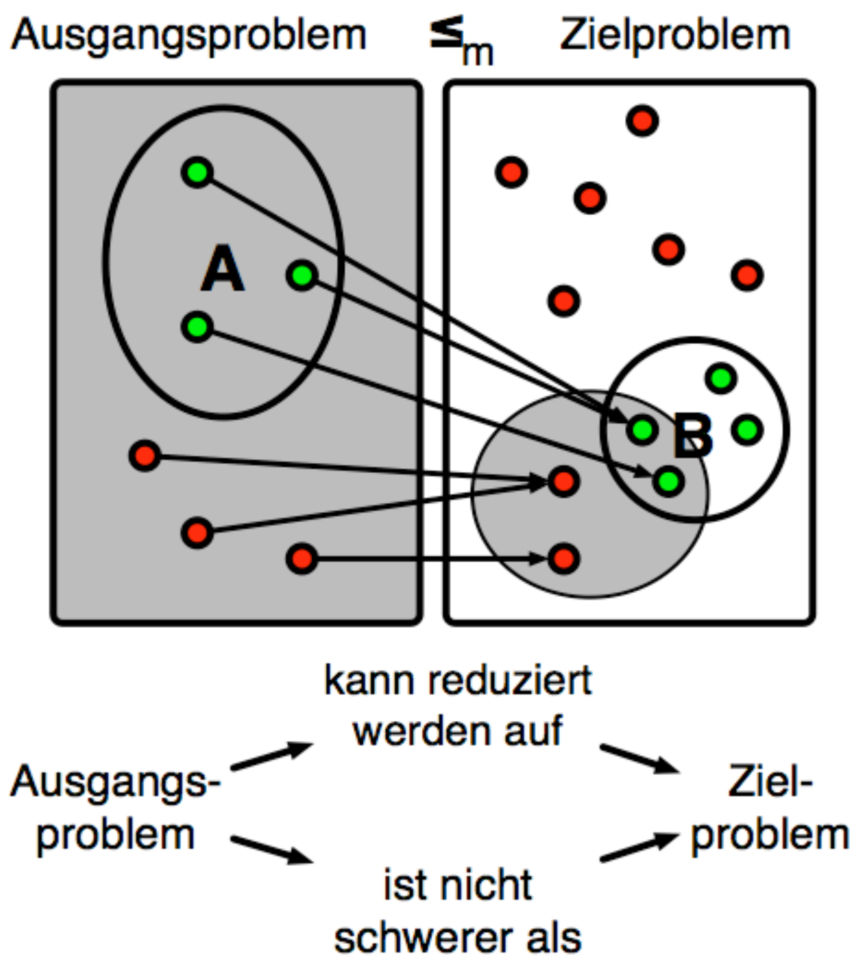
Abbildungsreduktion

- **Definition (Abbildungsreduktion, Mapping Reduction, Many-one)**
  - Eine Sprache A kann durch Abbildung auf eine Sprache B reduziert werden: $\mathbf{A \leq_m B}$,
    - falls es eine berechenbare Funktion f: $\Sigma^* \rightarrow \Sigma^*$ gibt,
    - so dass für alle w: $w \in A \;\Leftrightarrow\; f(w) \in B$
  - Die Funktion f heißt die **Reduktion** von A auf B.



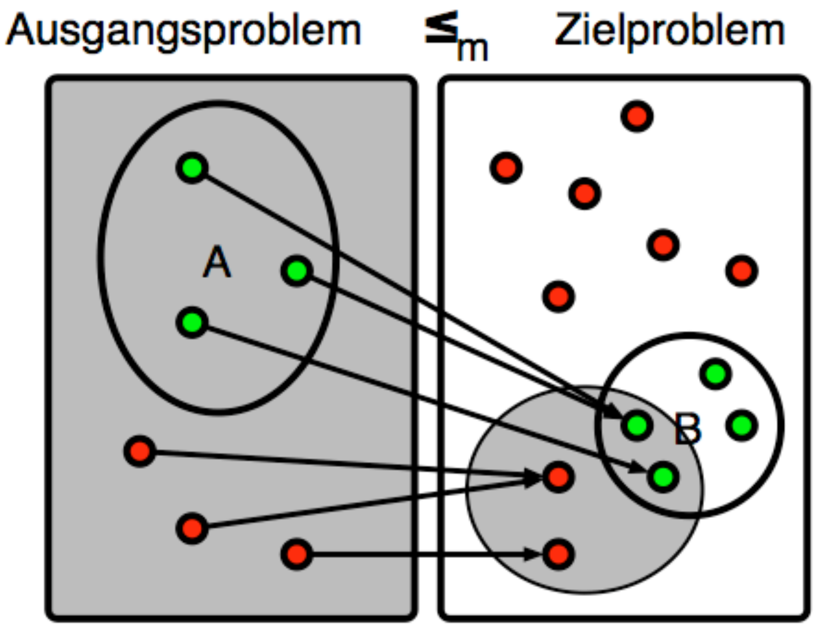
Der Nutzen der Reduktionen

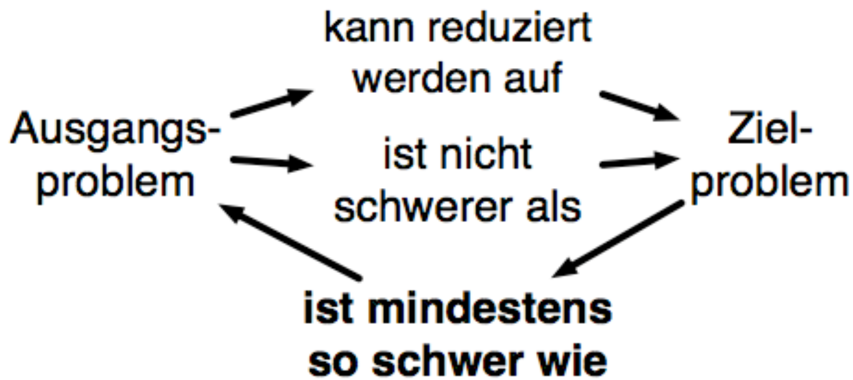
- **Theorem**
  - Falls A $\leq_m$ B und B ist entscheidbar, dann ist A entscheidbar.
- **Beweis**
  - Sei M, eine Turing-Maschine, die B entscheidet.
  - Betrachte die Entscheider-TM:
  - N = “Auf Eingabe w:
    - Berechne f(w)
    - Führe die Berechnung von M auf Eingabe f(w) durch
    - N gibt aus, was M ausgibt”
  - Falls f(w) ∈ B,
    - dann akzeptiert M
    - dann ist auch w ∈ A
  - Falls f(w) ∉ B,
    - dann akzeptiert M nicht
    - dann ist auch w ∉ A



Der Nutzen der Reduktionen

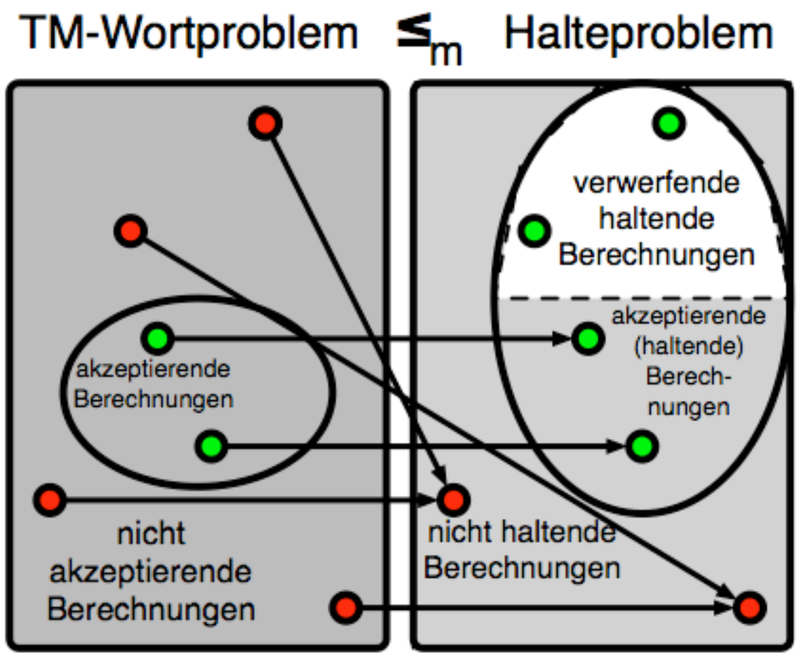
- **Korollar**
  - Falls A $\leq_m$ B und A ist nicht entscheidbar, dann ist B auch nicht entscheidbar.
- **Folgt aus:**
  - $X \wedge Y \Rightarrow Z$ und $X \wedge \neg Y \Rightarrow \neg Z$ sind äquivalent
- **Dieses Korollar ist unser Hauptwerkzeug für den Beweis der Nichtberechenbarkeit**

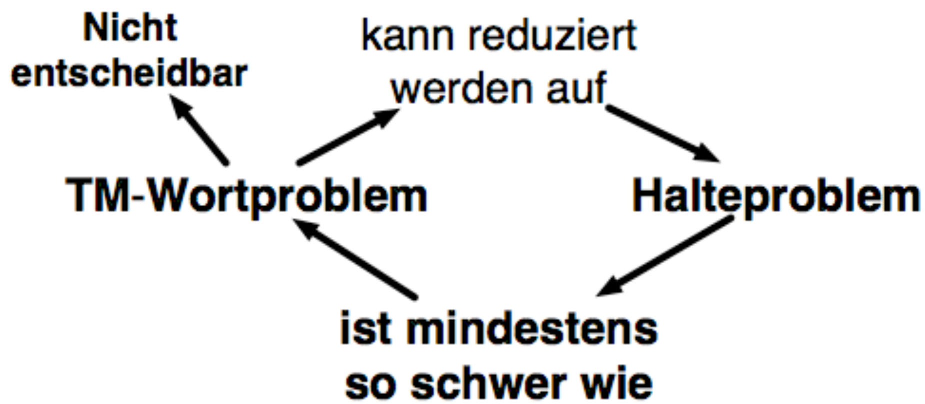




Ein alternativer Beweis für die Nichtberechenbarkeit des Halteproblems

- **Theorem:**
  - $A_{TM} \leq_m HALT_{TM}$
- **Beweis**
  - Betrachte Reduktionsfunktion F:
  - F = “Auf Eingabe <M, w>:
    - Konstruiere TM M’:
      - M’ = “Auf Eingabe x:
        - Führe M auf Eingabe x aus
        - Falls M akzeptiert, akzeptiert M’
        - Falls M verwirft, hält M’ nicht”
    - F gibt <M’,w> aus”





Ein alternativer Beweis für das Leerheitsproblem der TMs

Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

- **Theorem**

  $E_{TM} = \{\langle M\rangle \mid M \text{ ist eine TM und } L(M) = \emptyset\}$
  - E_{TM} ist nicht entscheidbar
- **Beweis: $A_{TM} \leq_m \overline{E_{TM}}$**
  - Betrachte Reduktionsfunktion F:
  - F = “Auf Eingabe <M, w>:
    - Konstruiere TM M’:
      - M’ = “Für jede Eingabe x:
        - ∗ Führe M auf Eingabe w aus
        - ∗ Falls M akzeptiert, akzeptiert M’
        - ∗ Falls M verwirft, verwirft M’ ”
    - F gibt M’ aus”
- **Zu zeigen:**
  - F ist eine berechenbare Funktion
    - M’ kann effektiv aus M und w konstruiert werden
  - <M, w> $\in A_{TM} \Leftrightarrow$ F(<M, w>) $\in \overline{E_{TM}}$
- **Falls <M, w> $\in A_{TM}$**
  - dann ist L(M’) = Σ^*
  - dann ist F(<M, w>) $\notin E_{TM}$
- **Falls <M, w> $\notin A_{TM}$**
  - dann ist L(M’) = $\emptyset$
  - dann ist F(<M, w>) $\in E_{TM}$
- **Nun ist $A_{TM} \leq_m \overline{E_{TM}}$**
  - da A_{TM} nicht rekursiv ist
  - ist auch $\overline{E_{TM}}$ nicht rekursiv
  - damit ist auch E_{TM} nicht rekursiv

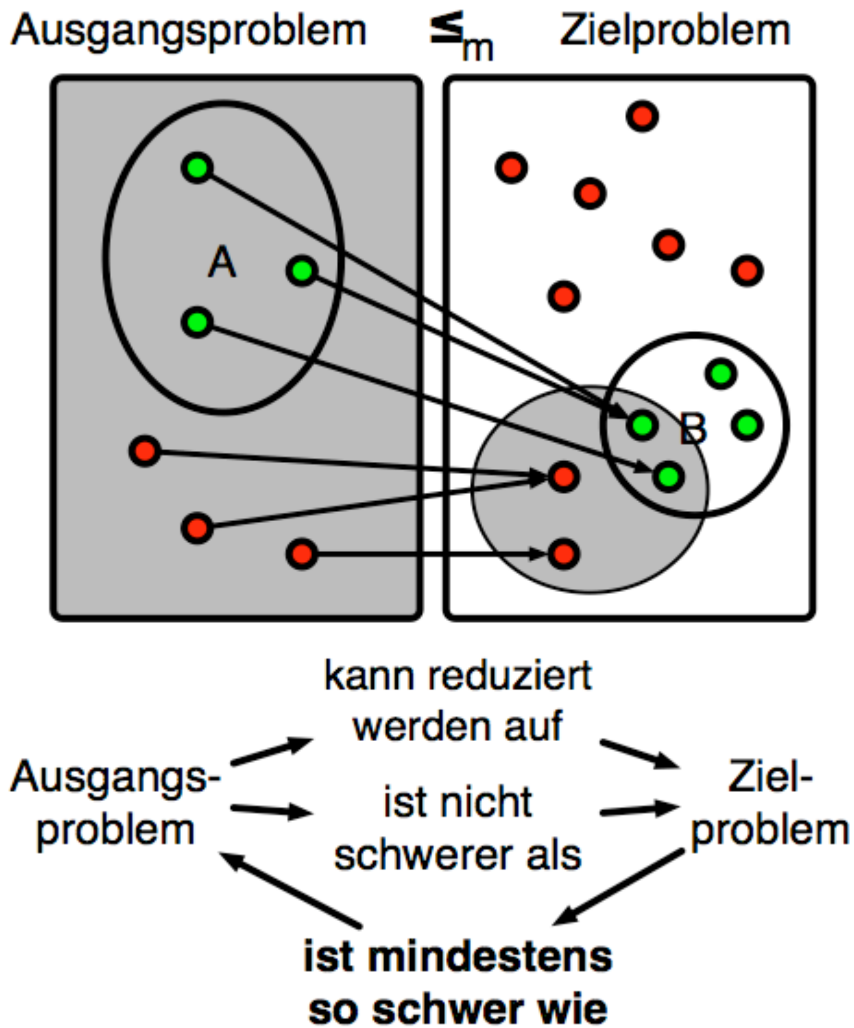
Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Reduktionen und Rekursive Aufzählbarkeit

- **Theorem**
  - Falls A $\leq_m$ B und B ist rekursiv aufzählbar, dann ist A rekursiv aufzählbar.
- **Beweis**
  - Sei M, eine Turing-Maschine, die B akzeptiert.
  - Betrachte die Akzeptor-TM N:
  - N = "Auf Eingabe w:
    - Berechne f(w)
    - Führe die Berechnung von M auf Eingabe f(w) durch
    - N gibt aus, was M ausgibt"
  - Falls f(w) $\in$ B,
    - dann akzeptiert M
    - dann ist auch w $\in$ A
  - Falls f(w) $\notin$ B,
    - dann akzeptiert M nicht
    - dann ist auch w $\notin$ A

Nicht-Rekursive Aufzählbarkeit und Reduktionen

- **Theorem**
  - Falls $A \leq_m B$ und B ist rekursiv aufzählbar, dann ist A rekursiv aufzählbar.
- **Korollar**
  - Falls $A \leq_m B$ und A ist nicht rekursiv aufzählbar, dann ist B nicht rekursiv aufzählbar.



Ende der 12. Vorlesung

CoNe Freiburg

Albert-Ludwigs-Universität Freiburg
Rechnernetze und Telematik
Prof. Dr. Christian Schindelhauer

Informatik III
Arne Vater
01.12.2006