# Theoretische Informatik I

## Einheit 4.3

## Eigenschaften von $\mathcal{L}_0/\mathcal{L}_1$-Sprachen

1. Abschlußeigenschaften
2. Prüfen von Eigenschaften
3. Grenzen der Sprachklassen

# Sprachklassen

- **Semi-entscheidbare Sprache**
  – Sprache, die von einer Turingmaschine akzeptiert wird
  – Auch aufzählbare Sprache genannt
- **Entscheidbare Sprache**
  – Sprache, die von einer Turingmaschine akzeptiert wird, die bei jeder Eingabe terminiert
  – Auch rekursive Sprache genannt
- **Kontextsensitive Sprache**
  – Sprache, die von einem linear beschränkten Automaten akzeptiert wird

# Sprachklassen

- **Semi-entscheidbare Sprache**
  - Sprache, die von einer Turingmaschine akzeptiert wird
  - Auch aufzählbare Sprache genannt
- **Entscheidbare Sprache**
  - Sprache, die von einer Turingmaschine akzeptiert wird, die bei jeder Eingabe terminiert
  - Auch rekursive Sprache genannt
- **Kontextsensitive Sprache**
  - Sprache, die von einem linear beschränkten Automaten akzeptiert wird
- **Entscheidbare Sprachen sind auch semi-entscheidbar**
  - Offensichtlich, da engere Bedingung
- **Kontextsensitive Sprachen sind entscheidbar**
  - Ein LBA hat bei Eingabe $w$ maximal $(|\Gamma| + |Q|)^{|w|+1}$ Konfigurationen
  - Wenn der $LBA$ nach $(|\Gamma| + |Q|)^{|w|+1}$ Schritten nicht akzeptiert, dann gehört $w$ nicht zur Sprache und die Berechnung kann anhalten

# Abschlusseigenschaften summarisch

- **Alle drei Sprachklassen sind abgeschlossen unter**
  - Vereinigung $L_1 \cup L_2$
  - Durchschnitt $L_1 \cap L_2$
  - Spiegelung $L^R$
  - Verkettung $L_1 \circ L_2$
  - Hüllenbildung $L^*$
  - Homomorphismen $h(L)$
  - Inverse Homomorphismen $h^{-1}(L)$
  - Urbild berechenbarer Funktionen $f^{-1}(L)$
- **Typ-1 und entscheidbare Sprachen zusätzlich**
  - Komplement $\overline{L}$
  - Differenz $L_1$ - $L_2$
  - Aufzählbare Sprachen: Bild berechenbarer Funktionen $f(L)$

# Nachweis der Abschlusseigenschaften

## Beweisführung mit Grammatiken

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(G_i)$, wobei $G_i = (V_i, T_i, P_i, S_i)$ disjunkt
  - Wähle $G = (V_1 \cup V_2 \cup \{S\}, T_1 \cup T_2, P_1 \cup P_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\}, S)$
  - Die Eigenschaften der $G_i$ bleiben erhalten und es gilt $L(G) = L_1 \cup L_2$

# Nachweis der Abschlusseigenschaften

## Beweisführung mit Grammatiken

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(G_i)$, wobei $G_i = (V_i, T_i, P_i, S_i)$ disjunkt
  - Wähle $G = (V_1 \cup V_2 \cup \{S\}, T_1 \cup T_2, P_1 \cup P_2 \cup \{S \to S_1, S \to S_2\}, S)$
  - Die Eigenschaften der $G_i$ bleiben erhalten und es gilt $L(G) = L_1 \cup L_2$
- **Spiegelung $L^R$**
  - Bilde Spiegelgrammatik zu $G = (V, T, P, S)$ mit $L = L(G)$
    - · Setze $\boldsymbol{G_R} = (V, T, P_R, S)$ mit $\boldsymbol{P_R} = \{l^R \to \alpha^R \mid l \to \alpha \in P\}$
  - Die Eigenschaften von $G$ bleiben erhalten und es gilt $L(G_R) = L^R$

# Nachweis der Abschlusseigenschaften

## Beweisführung mit Grammatiken

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(G_i)$, wobei $G_i = (V_i, T_i, P_i, S_i)$ disjunkt
  - Wähle $G = (V_1 \cup V_2 \cup \{S\}, T_1 \cup T_2, P_1 \cup P_2 \cup \{S \to S_1, S \to S_2\}, S)$
  - Die Eigenschaften der $G_i$ bleiben erhalten und es gilt $L(G) = L_1 \cup L_2$
- **Spiegelung $L^R$**
  - Bilde Spiegelgrammatik zu $G = (V, T, P, S)$ mit $L = L(G)$
    - · Setze $\boldsymbol{G_R} = (V, T, P_R, S)$ mit $\boldsymbol{P_R} = \{l^R \to \alpha^R \mid l \to \alpha \in P\}$
  - Die Eigenschaften von $G$ bleiben erhalten und es gilt $L(G_R) = L^R$
- **Analoge Beweise für $L_1 \circ L_2$, $L^*$, $h(L)$**
  - Verzweige aus Startsymbol oder modifiziere rechte Seite der Regeln

# Nachweis der Abschlusseigenschaften

## Beweisführung mit Grammatiken

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(G_i)$, wobei $G_i = (V_i, T_i, P_i, S_i)$ disjunkt
  - Wähle $G = (V_1 \cup V_2 \cup \{S\}, T_1 \cup T_2, P_1 \cup P_2 \cup \{S \to S_1, S \to S_2\}, S)$
  - Die Eigenschaften der $G_i$ bleiben erhalten und es gilt $L(G) = L_1 \cup L_2$
- **Spiegelung $L^R$**
  - Bilde Spiegelgrammatik zu $G = (V, T, P, S)$ mit $L = L(G)$
    - · Setze $\boldsymbol{G_R} = (V, T, P_R, S)$ mit $\boldsymbol{P_R} = \{l^R \to \alpha^R \mid l \to \alpha \in P\}$
  - Die Eigenschaften von $G$ bleiben erhalten und es gilt $L(G_R) = L^R$
- **Analoge Beweise für $L_1 \circ L_2$, $L^*$, $h(L)$**
  - Verzweige aus Startsymbol oder modifiziere rechte Seite der Regeln
- **Grammatiken helfen wenig bei Entscheidbarkeit**
  - Beweisführung mit Turingmaschinen ist sinnvoller

# Nachweis der Abschlusseigenschaften

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(M_i)$, wobei $M_i = (Q_i, \Sigma_i, \Gamma_i, \delta_i, q_{0,i}, B, F_i)$ disjunkt
  - Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern
  - $M$ akzeptiert genau dann, wenn $M_1$ oder $M_2$ akzeptieren
  - Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M) = L_1 \cup L_2$

# Nachweis der Abschlusseigenschaften

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(M_i)$, wobei $M_i = (Q_i, \Sigma_i, \Gamma_i, \delta_i, q_{0,i}, B, F_i)$ disjunkt
  - Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern
  - $M$ akzeptiert genau dann, wenn $M_1$ oder $M_2$ akzeptieren
  - Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M) = L_1 \cup L_2$
- **Durchschnitt $L_1 \cap L_2$**
  - Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern
  - $M$ akzeptiert genau dann, wenn $M_1$ und $M_2$ akzeptieren
  - Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M) = L_1 \cap L_2$

# Nachweis der Abschlusseigenschaften

- **Vereinigung $L_1 \cup L_2$**
  - Sei $L_i = L(M_i)$, wobei $M_i = (Q_i, \Sigma_i, \Gamma_i, \delta_i, q_{0,i}, B, F_i)$ disjunkt
  - Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern
  - $M$ akzeptiert genau dann, wenn $M_1$ oder $M_2$ akzeptieren
  - Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M)=L_1 \cup L_2$
- **Durchschnitt $L_1 \cap L_2$**
  - Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern
  - $M$ akzeptiert genau dann, wenn $M_1$ und $M_2$ akzeptieren
  - Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M)=L_1 \cap L_2$
- **Spiegelung $L_1^R$**
  - Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort umgedreht auf ein Hilfsband und simuliert $M_1$ auf diesem Band
  - $M$ akzeptiert genau dann, wenn $M_1$ akzeptiert
  - Die Eigenschaften von $M_1$ bleiben erhalten und es gilt $L(M)=L_1^R$

# Nachweis der Abschlusseigenschaften II

- **Verkettung $L_1 \circ L_2$**
  - Bei Eingabe eines Wortes $w$ wählt $M$ nichtdeterministisch eine Zerlegung das Wort $w = w_1 \circ w_2$, kopiert die $w_i$ auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ entsprechend
  - $M$ akzeptiert genau dann, wenn $M_1$ und $M_2$ akzeptieren
  - Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M)=L_1 \circ L_2$

# Nachweis der Abschlusseigenschaften II

- **Verkettung $L_1 \circ L_2$**
  – Bei Eingabe eines Wortes $w$ wählt $M$ nichtdeterministisch eine Zerlegung das Wort $w = w_1 \circ w_2$, kopiert die $w_i$ auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ entsprechend
  – $M$ akzeptiert genau dann, wenn $M_1$ und $M_2$ akzeptieren
  – Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M) = L_1 \circ L_2$
- **Hülle $L_1^*$**
  – Bei Eingabe eines Wortes $w$ wählt $M$ nichtdeterministisch eine Zerlegung des Wortes $w = w_1 \circ .. \circ w_n$, kopiert die $w_i$ der Reihe nach auf ein Hilfsband und simuliert $M_1$ entsprechend
  – $M$ akzeptiert genau dann, wenn $M_1$ alle $w_i$ akzeptiert
  – Die Eigenschaften von $M_1$ bleiben erhalten und es gilt $L(M) = L_1^*$

# Nachweis der Abschlusseigenschaften II

- **Verkettung $L_1 \circ L_2$**
  – Bei Eingabe eines Wortes $w$ wählt $M$ nichtdeterministisch eine Zerlegung das Wort $w = w_1 \circ w_2$, kopiert die $w_i$ auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ entsprechend
  – $M$ akzeptiert genau dann, wenn $M_1$ und $M_2$ akzeptieren
  – Die Eigenschaften der $M_i$ bleiben erhalten und es gilt $L(M) = L_1 \circ L_2$
- **Hülle $L_1^*$**
  – Bei Eingabe eines Wortes $w$ wählt $M$ nichtdeterministisch eine Zerlegung des Wortes $w = w_1 \circ .. \circ w_n$, kopiert die $w_i$ der Reihe nach auf ein Hilfsband und simuliert $M_1$ entsprechend
  – $M$ akzeptiert genau dann, wenn $M_1$ alle $w_i$ akzeptiert
  – Die Eigenschaften von $M_1$ bleiben erhalten und es gilt $L(M) = L_1^*$
- **Homomorphismen $h(L_1)$**
  – Bei Eingabe eines Wortes $w$ wählt $M$ nichtdeterministisch eine Zerlegung das Wort $w = w_1 \circ .. \circ w_n$ mit $w_i = h(a_i)$, kopiert $v = a_1..a_n$ auf ein Hilfsband und simuliert $M_1$ entsprechend
  – $M$ akzeptiert genau dann, wenn $M_1$ das Wort $v$ akzeptiert
  – Die Eigenschaften von $M_1$ bleiben erhalten und es gilt $L(M) = h(L_1)$

# Nachweis der Abschlusseigenschaften III

- **Inverse Homomorphismen $h^{-1}(L_1)$**
  – Bei Eingabe eines Wortes $w = a_1..a_n$ bestimmt $M$ das Wort $v = h(a_1)..h(a_n)$, kopiert es auf ein Hilfsband und simuliert $M_1$
  – $M$ akzeptiert genau dann, wenn $M_1$ das Wort $v$ akzeptiert
  – Es gilt $L(M)$=$h^{-1}(L_1)$
  – Beweis gilt in dieser Form nur für (semi-)entscheidbare Sprachen
  Für LBA's ist Simulation eines Bandes $k$-facher Länge erforderlich

# Nachweis der Abschlusseigenschaften III

- **Inverse Homomorphismen $h^{-1}(L_1)$**
  - Bei Eingabe eines Wortes $w = a_1..a_n$ bestimmt $M$ das Wort $v = h(a_1)..h(a_n)$, kopiert es auf ein Hilfsband und simuliert $M_1$
  - $M$ akzeptiert genau dann, wenn $M_1$ das Wort $v$ akzeptiert
  - Es gilt $L(M)=h^{-1}(L_1)$
  - Beweis gilt in dieser Form nur für (semi-)entscheidbare Sprachen
    Für LBA's ist Simulation eines Bandes $k$-facher Länge erforderlich
- **Komplement $\overline{L_1}$**
  - Bei Eingabe eines Wortes $w$ simuliert $M$ die Berechnung von $M_1$ und akzeptiert genau dann, wenn $M_1$ nicht akzeptiert
  - Es gilt $L(M)=\overline{L_1}$
  - Die Eigenschaften von $M_1$ bleiben nur erhalten, wenn $M_1$ terminiert
    Bei semi-entscheidbaren Sprachen terminiert die Berechnung für $w \in \overline{L_1}$ nicht

# Nachweis der Abschlusseigenschaften III

- **Inverse Homomorphismen $h^{-1}(L_1)$**
  - Bei Eingabe eines Wortes $w = a_1..a_n$ bestimmt $M$ das Wort $v = h(a_1)..h(a_n)$, kopiert es auf ein Hilfsband und simuliert $M_1$
  - $M$ akzeptiert genau dann, wenn $M_1$ das Wort $v$ akzeptiert
  - Es gilt $L(M)$=$h^{-1}(L_1)$
  - Beweis gilt in dieser Form nur für (semi-)entscheidbare Sprachen
    Für LBA's ist Simulation eines Bandes $k$-facher Länge erforderlich
- **Komplement $\overline{L_1}$**
  - Bei Eingabe eines Wortes $w$ simuliert $M$ die Berechnung von $M_1$ und akzeptiert genau dann, wenn $M_1$ nicht akzeptiert
  - Es gilt $L(M)$=$\overline{L_1}$
  - Die Eigenschaften von $M_1$ bleiben nur erhalten, wenn $M_1$ terminiert
    Bei semi-entscheidbaren Sprachen terminiert die Berechnung für $w \in \overline{L_1}$ nicht
- **Differenz $L_1$-$L_2$**
  - Mathematische Begründung: $L_1$-$L_2$ = $L_1 \cap \overline{L_2}$
  - Abgeschlossenheit unter Differenz gilt für aufzählbare Sprachen nicht!

# Aufzählbarkeit vs. Entscheidbarkeit

- **$L$ entscheidbar $\Leftrightarrow$ $L$ und $\overline{L}$ semi-entscheidbar**

  "$\Rightarrow$": Entscheidbare Sprachen sind abgeschlossen unter Komplement

  "$\Leftarrow$": Sei $L{=}L(M_1)$ und $\overline{L}{=}L(M_2)$.

  – Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern

  – $M$ akzeptiert genau dann, wenn $M_1$ akzeptiert und terminiert ohne zu akzeptieren, wenn $M_2$ akzeptiert

  – Da eine der beiden Maschinen das Wort $w$ akzeptieren muß, terminiert $M$ und es gilt $L(M){=}L$

# Aufzählbarkeit vs. Entscheidbarkeit

- **$L$ entscheidbar $\Leftrightarrow$ $L$ und $\overline{L}$ semi-entscheidbar**

  "$\Rightarrow$": Entscheidbare Sprachen sind abgeschlossen unter Komplement

  "$\Leftarrow$": Sei $L{=}L(M_1)$ und $\overline{L}{=}L(M_2)$.

  – Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern

  – $M$ akzeptiert genau dann, wenn $M_1$ akzeptiert und terminiert ohne zu akzeptieren, wenn $M_2$ akzeptiert

  – Da eine der beiden Maschinen das Wort $w$ akzeptieren muß, terminiert $M$ und es gilt $L(M){=}L$

- **Jede endliche Sprache $L$ ist entscheidbar**

  – Jede endliche Sprache ist als Liste von Wörtern $[w_1; ..; w_n]$ darstellbar

  – Bei Eingabe eines Wortes $w$ vergleicht $M$ das Wort mit dieser Liste

# Aufzählbarkeit vs. Entscheidbarkeit

- **$L$ entscheidbar $\Leftrightarrow$ $L$ und $\overline{L}$ semi-entscheidbar**

  "$\Rightarrow$": Entscheidbare Sprachen sind abgeschlossen unter Komplement

  "$\Leftarrow$": Sei $L{=}L(M_1)$ und $\overline{L}{=}L(M_2)$.

  – Bei Eingabe eines Wortes $w$ kopiert $M$ das Wort auf zwei Hilfsbänder und simuliert $M_1$ und $M_2$ auf den beiden Bändern

  – $M$ akzeptiert genau dann, wenn $M_1$ akzeptiert und terminiert ohne zu akzeptieren, wenn $M_2$ akzeptiert

  – Da eine der beiden Maschinen das Wort $w$ akzeptieren muß, terminiert $M$ und es gilt $L(M){=}L$

- **Jede endliche Sprache $L$ ist entscheidbar**

  – Jede endliche Sprache ist als Liste von Wörtern $[w_1; ..; w_n]$ darstellbar

  – Bei Eingabe eines Wortes $w$ vergleicht $M$ das Wort mit dieser Liste

- **$L$ semi-entscheidbar $\Leftrightarrow$ es gibt ein entscheidbares $L' \subseteq \Sigma^* \times \Sigma^*$ mit $L = \{w \mid \exists v.\, (w, v) \in L'\}$** (Projektionssatz)

  – Aufwendiger Beweis, benötigt schrittweise Simulation von Maschinen

# Prüfen von Eigenschaften summarisch

- **"$x \in L$" kann automatisch geprüft werden für**
  - Kontextsensitive und entscheidbare Sprachen

    (Folgt unmittelbar aus der Definition von Entscheidbarkeit)
  - Aber **nicht für aufzählbare Sprachen**

    (Folgt aus Existenz einer aufzählbaren, aber unentscheidbaren Sprache)

# Prüfen von Eigenschaften summarisch

- **“$x \in L$” kann automatisch geprüft werden für**
  – Kontextsensitive und entscheidbare Sprachen
  (Folgt unmittelbar aus der Definition von Entscheidbarkeit)
  – Aber **nicht für aufzählbare Sprachen**
  (Folgt aus Existenz einer aufzählbaren, aber unentscheidbaren Sprache)

- **Für keine Sprachklasse kann getestet werden ob**
  – eine Sprache $L$ der Klasse leer ist
  – zwei Sprachen $L_1$ und $L_2$ der Klasse gleich sind
  – zwei Sprachen $L_1$ und $L_2$ der Klasse ineinander enthalten sind
  – der Durchschnitt zweier Sprachen der Klasse leer ist

Beweise benötigen Beispiele für Sprachen, die nicht zur Klasse gehören

# Grenzen der Sprachklassen

- **Entscheidbare, nicht kontextsensitive Sprache**
  - Menge aller äquivalenten regulären Ausdrücke (gelesen als Text), wenn diese eine Iteration $E^k = \underbrace{E \circ E \ldots \circ E}_{k-mal}$ enthalten dürfen
  - Äquivalenztest benötigt exponentiell großen Speicherplatz

# Grenzen der Sprachklassen

- **Entscheidbare, nicht kontextsensitive Sprache**
  - Menge aller äquivalenten regulären Ausdrücke (gelesen als Text), wenn diese eine Iteration $E^k = \underbrace{E \circ E \ldots \circ E}_{k-mal}$ enthalten dürfen
  - Äquivalenztest benötigt exponentiell großen Speicherplatz
- **Aufzählbare, nicht entscheidbare Sprache**
  - Selbstanwendbarkeitsproblem: Menge aller Programme von Turingmaschinen, die bei Eingabe des eigenen Programms als Text terminieren

# Grenzen der Sprachklassen

- **Entscheidbare, nicht kontextsensitive Sprache**
  - Menge aller äquivalenten regulären Ausdrücke (gelesen als Text), wenn diese eine Iteration $E^k = \underbrace{E \circ E \ldots \circ E}_{k-mal}$ enthalten dürfen
  - Äquivalenztest benötigt exponentiell großen Speicherplatz
- **Aufzählbare, nicht entscheidbare Sprache**
  - Selbstanwendbarkeitsproblem: Menge aller Programme von Turingmaschinen, die bei Eingabe des eigenen Programms als Text terminieren
- **Nicht aufzählbare Sprache**
  - Totale Berechenbarkeit: Menge aller Programme von Turingmaschinen, die bei jeder Eingabe terminieren

# Grenzen der Sprachklassen

- **Entscheidbare, nicht kontextsensitive Sprache**
  - Menge aller äquivalenten regulären Ausdrücke (gelesen als Text), wenn diese eine Iteration $E^k = \underbrace{E\circ E\ldots\circ E}_{k-mal}$ enthalten dürfen
  - Äquivalenztest benötigt exponentiell großen Speicherplatz
- **Aufzählbare, nicht entscheidbare Sprache**
  - Selbstanwendbarkeitsproblem: Menge aller Programme von Turingmaschinen, die bei Eingabe des eigenen Programms als Text terminieren
- **Nicht aufzählbare Sprache**
  - Totale Berechenbarkeit: Menge aller Programme von Turingmaschinen, die bei jeder Eingabe terminieren

**Mehr dazu in Theoretischer Informatik II**

# Zusammenfassung: Typ-0 und Typ-1 Sprachen

- **Turingmaschine als allgemeinstes Maschinenmodell**
  - Deterministischer endlicher Automat mit unendlichem Speicherband
  - Gleiche Ausdruckskraft wie reale Computer (aber einfacher strukturiert)
  - Nichtdeterministische Variante mit exponentiellem Aufwand simulierbar
  - Äquivalent zu Typ-0 Grammatiken
  - Bei linearer Bandbeschränkung äquivalent zu Typ-1 Grammatiken
  - Entscheidbare Sprachen stehen zwischen $\mathcal{L}_0$ und $\mathcal{L}_1$
- **Wichtige Eigenschaften der Sprachklassen**
  - Abgeschlossen unter $\cup$, $\cap$, ${}^R$, $\circ$, ${}^*$, $h$, $h^{-1}$
  - $\mathcal{L}_1$ und entscheidbare Sprachen zusätzlich unter $\overline{\phantom{x}}$, -
  - Viele Eigenschaften können nicht automatisch getestet werden
    - Fast alle nichtrivialen Eigenschaften sind für keine Klasse entscheidbar
    - Für $\mathcal{L}_0$ ist selbst das Wortproblem nicht mehr entscheidbar