Logik Grundvorlesung SS 19
Folien Teil 1

Gerhard Brewka

Institut für Informatik
Universität Leipzig
brewka@informatik.uni-leipzig.de

Über diese Vorlesung

- Übungsleitung Ringo Baumann, Frank Loebe
- Di. 09:15, 11:15, 13:15; Mi 11:15 (jeweils A,B)
- Skript/Folien:
  www.informatik.uni-leipzig.de/∼brewka/teaching.html
- Lehrbuch: Uwe Schöning, Logik für Informatiker, 5. Auflage, Spektrum Akademischer Verlag, 2000
- Informatik oft eingeteilt in:
  Technische, Praktische, Angewandte und Theoretische Informatik.
- Theoretische Informatik untersucht grundlegende Konzepte, die für das gesamte Gebiet von Bedeutung sind.
- Die Vorlesung Logik gehört zu Zyklus von 4 Vorlesungen.

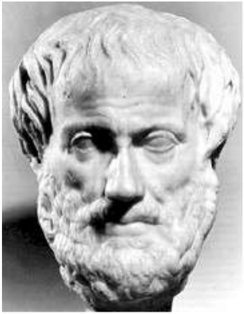
Über diese Vorlesung

- Übungsleitung Ringo Baumann, Frank Loebe
- Di. 09:15, 11:15, 13:15; Mi 11:15 (jeweils A,B)
- Skript/Folien:
  www.informatik.uni-leipzig.de/∼brewka/teaching.html
- Lehrbuch: Uwe Schöning, Logik für Informatiker, 5. Auflage, Spektrum Akademischer Verlag, 2000
- Informatik oft eingeteilt in:
  Technische, Praktische, Angewandte und Theoretische Informatik.
- Theoretische Informatik untersucht grundlegende Konzepte, die für das gesamte Gebiet von Bedeutung sind.
- Die Vorlesung Logik gehört zu Zyklus von 4 Vorlesungen.

Grundvorlesungen Theoretische Informatik

1. Diskrete Strukturen: wichtige Grundbegriffe der theoretischen Informatik; notwendiges Handwerkszeug.
2. Logik: Folgerungsbeziehungen auf Basis präziser Sprachen, in denen exakte Spezifikationen vorgenommen werden können; automatisierbare Schlussverfahren.
3. Automaten und Sprachen: Klassen formaler (Kunst-)Sprachen, ihre Eigenschaften, Beschreibungsmöglichkeiten (Grammatiken) und entsprechende Automatenmodelle.
4. Berechenbarkeit und Komplexität: Präzisierung des Berechenbarkeitsbegriffs; nicht berechenbare, mathematisch präzise beschreibbare Funktionen; Beschreibung der Komplexität von Problemen.

Was ist Logik?



- Lehre von formalen (Folgerungs)Beziehungen zwischen Sätzen
- als eigenständige Wissenschaft begründet von Aristoteles:
    - Lehre vom Begriff: Klassifizierung von Begriffen, Definitionslehre
    - Lehre vom Urteil: Struktur, Klassifikation von Aussagesätzen
    - Lehre vom Schluss: Folgerungsbeziehungen

Syllogismen nach Aristoteles (384-322 v.Chr.)

Katalog von Schlussformen, etwa:

- Alle Rechtecke sind Vierecke.
  Alle Quadrate sind Rechtecke.
  Alle Quadrate sind Vierecke.
- Alle Informatiker sind schlau.
  Einige Studierende sind Informatiker.
  Einige Studierende sind schlau.
- Alle Menschen sind sterblich.
  Sokrates ist ein Mensch.
  Sokrates ist sterblich.

Keine formale, mathematische Definition von Folgerbarkeit

Weiteres zur Geschichte der Logik

- Beiträge von Stoikern: Theorien der Implikation, logische Antinomien
- mittelalterl. Scholastik: Arbeiten zu Sprachphilosophie und log. Semantik
- Leibniz (1646-1716): Vorläufer der modernen Logik; logische Symbolsprache für alle Wissenschaften
- Entwicklung der modernen Logik ab Mitte 19. Jh.: Frege, Boole, Russell

Mathematische Logik

Charakteristika moderner mathematischer Logik:

- formale Kunstsprache
- modelltheoretische Semantik
- korrekte und vollständige Kalküle

Anmerkung: es gibt inzwischen viele Logiken; dazu später mehr

Relevanz der Logik für die Informatik

- Beschreibungsmittel für formale Sachverhalte
- Schaltalgebra
- Semantik von Programmiersprachen
- Programmverifikation
- Wissensrepräsentation
- Logisches Programmieren (PROLOG)
- Automatisches Beweisen
- ...

Logik und Künstliche Intelligenz

- KI in aller Munde: autonome Fahrzeuge; IBMs Watson; Systeme, die die besten Go und Pokerspieler schlagen (AlphaGo, Libratus)
- Oft als Erfolge im Bereich *deep learning* gesehen.
- Logik unnötig? Nein: Nachvollziehbarkeit der Entscheidungen und Empfehlungen von KI-Systemen Voraussetzung für Akzeptanz.
- Logik unabdingbar für Systeme, die ihr Verhalten erklären können.

Gliederung der Vorlesung

1. Einführung
2. Aussagenlogik
   - Syntax und Semantik; Normalformen
   - Hornformeln und Resolution
   - Davis-Putnam-Verfahren und Tableaus
3. Prädikatenlogik
   - Syntax und Semantik; Normalformen
   - Entscheidbarkeit
   - Herbrand-Theorie und Resolution
4. Logikprogrammierung
   - Definite Programme und minimale Modelle
   - Default Negation und stabile Modelle
   - Answer-Set-Programmierung

Aussagenlogik: Grundannahmen/Basisdefinitionen

- AL untersucht Verknüpfungen wie "und", "oder", "nicht", "wenn ... dann" zwischen atomaren und komplexen Sätzen.
- Sätze selbst entweder wahr oder falsch (tertium non datur).
- Ihre Bedeutung abgesehen davon irrelevant!
- Atomare Sätze durch beliebige Symbole repräsentiert (P, Q, A_1, A_2, ...).

AL: Basisdefinitionen

Definition (Syntax der Aussagenlogik)

Sei A eine Menge von Symbolen (aussagenlogische Variablen oder atomare Formeln). Die Menge der aussagenlogischen Formeln über A ist die kleinste Menge, für die gilt:

1. Jedes Element aus A ist eine (atomare) Formel.
2. Wenn F und G Formeln sind, so sind $(F \wedge G)$ und $(F \vee G)$ Formeln.
3. Wenn F eine Formel ist, so ist $\neg F$ eine Formel.

- $(F \wedge G)$ Konjunktion, $(F \vee G)$ Disjunktion von F und G, $\neg F$ Negation von F.
- Formel F, die als Teil einer Formel G vorkommt: Teilformel von G.
- Abkürzungen: $(F_1 \rightarrow F_2)$ statt $(\neg F_1 \vee F_2)$; $(F_1 \leftrightarrow F_2)$ statt $((F_1 \wedge F_2) \vee (\neg F_1 \wedge \neg F_2))$

AL: Semantik

Definition (Belegung/Interpretation)

Eine (Wahrheits-) Belegung (Interpretation) ist eine Funktion $I : A \rightarrow \{0, 1\}$, wobei A die Menge der atomaren Formeln ist.

Definition (Wahrheitswert komplexer Formeln)

Sei I eine Belegung. Wir erweitern I zu einer Funktion $I^+ : E \rightarrow \{0, 1\}$, wobei E die Menge aller Formeln über A ist:

1. Für jede atomare Formel $A_i \in A$ ist $I^+(A_i) = I(A_i)$,
2. $I^+((F \wedge G)) = 1$ falls $I^+(F) = 1$ und $I^+(G) = 1$, 0 sonst,
3. $I^+((F \vee G)) = 1$ falls $I^+(F) = 1$ oder $I^+(G) = 1$, 0 sonst,
4. $I^+(\neg F) = 1$ falls $I^+(F) = 0$, 0 sonst.

Da I^+ Erweiterung von I ist, schreiben wir vereinfachend I statt I^+.

Definition (Modell, Gültigkeit, Erfüllbarkeit)

Sei F eine Formel. Eine Belegung, die F zu 1 auswertet, heißt Modell von F. Falls I Modell von F, schreiben wir: $I \models F$, falls nicht: $I \not\models F$.

Eine Formel F heißt erfüllbar, wenn F mindestens ein Modell besitzt. Unerfüllbare Formeln heißen auch widersprüchlich (Kontradiktionen).

F heißt allgemeingültig (Tautologie), falls jede Belegung Modell für F ist. Notation: $\models F$ für F ist Tautologie, $\not\models F$ für F ist nicht Tautologie.

Eine Belegung ist Modell einer Menge von Formeln M, wenn sie jedes Element von M zu 1 auswertet. M ist erfüllbar, wenn M mindestens ein Modell besitzt.

$Mod(F)$ [$Mod(M)$] bezeichnet die Menge der Modelle von F [von M].

Satz: Die Formel F ist Tautologie, genau dann wenn $\neg F$ unerfüllbar ist.

Folgerbarkeit

Definition (Logische Folgerung)

Seien $F_1, ..., F_k$ und G Formeln.

G folgt logisch aus $\{F_1, ..., F_k\}$ gdw. jede zu $F_1, ..., F_k$ und G passende Belegung, die Modell von $\{F_1, ..., F_k\}$ ist, auch Modell von G ist.

(I passend zu F: jede aussagenlogische Variable in F im Definitionsbereich von I.)

Notation für G folgt logisch aus $\{F_1, ..., F_k\}$: $\{F_1, ..., F_k\} \models G$.

Falls $k = 1$: $F_1 \models G$.

Man sagt auch: G ist Folgerung aus der Prämissenmenge $\{F_1, ..., F_k\}$.

Die Menge aller Folgerungen einer Formelmenge H nennt man $Th(H)$.

Alternative Charakterisierungen

Satz: Folgende Aussagen sind äquivalent:

1. G folgt logisch aus $\{F_1, ..., F_k\}$.
2. $((...(F_1 \wedge F_2) \wedge ... \wedge F_k) \rightarrow G)$ ist Tautologie.
3. $((...(F_1 \wedge F_2) \wedge ... \wedge F_k) \wedge \neg G)$ ist unerfüllbar.

Beweis:

G folgt logisch aus $\{F_1, ..., F_k\}$ gdw.
alle Modelle von $\{F_1, ..., F_k\}$ sind Modelle von G gdw.
jede Interpr. wertet ein F_i ($i \in \{1, ..., k\}$) zu 0 aus oder G zu 1 gdw.
jede Interpr. ist Modell von $((...(F_1 \wedge F_2) \wedge ... \wedge F_k) \rightarrow G)$ gdw.
$((...(F_1 \wedge F_2) \wedge ... \wedge F_k) \rightarrow G)$ ist Tautologie gdw.
keine Interpr. wertet F_i ($i \in \{1, ..., k\}$) zu 1 aus und G zu 0 gdw.
$((...(F_1 \wedge F_2) \wedge ... \wedge F_k) \wedge \neg G)$ ist unerfüllbar.

Monotonie des Folgerungsbegriffs

Satz: Seien H und H' Mengen von Formeln, so dass $H \subseteq H'$. Sei F eine Formel. Falls $H \models F$, dann gilt auch $H' \models F$.
(in anderen Worten: $H \subseteq H'$ impliziert $Th(H) \subseteq Th(H')$.)

Beweis:
Da $H \subseteq H'$, ist jedes Modell von H' auch Modell von H:
jedes Modell von H' wertet ja auch alle Formeln in H zu 1 aus.
Da alle Modelle von H F zu 1 auswerten, tun das auch alle Modelle von H'.

Intuitiv: zusätzliche Information kann nur zu weniger Modellen führen und damit zu mehr Folgerungen.

Manchmal sind auch nichtmonotone Folgerungsrelationen interessant!

Folgerung aus widersprüchlichen Prämissen

Satz: Sei H eine unerfüllbare Menge von Formeln, F eine beliebige Formel. Es gilt $H \models F$.

Beweis:
Da H unerfüllbar ist, gibt es kein Modell von H.
Deshalb werten trivialer Weise alle Modelle von H F zu 1 aus.

Intuitiv: aus widersprüchlicher Information kann man alles folgern.

Nicht immer das, was man möchte!

Äquivalenz

Definition (Äquivalenz)

Zwei Formeln F und G heißen (logisch) äquivalent (Notation: $F \equiv G$), falls für alle Belegungen I gilt: $I(F) = I(G)$.

Satz: Seien F und G Formeln. $F \equiv G$ gdw. $F \leftrightarrow G$ ist Tautologie.

Satz: (Ersetzbarkeit) Seien F und G äquivalente Formeln. Sei H eine Formel, die F als Teilformel besitzt. H' entstehe aus H durch Ersetzen eines Vorkommens von F in H durch G. Dann gilt: $H \equiv H'$.

Beweis: Induktion über den Formelaufbau von H.

Einige gültige Äquivalenzen

$$\begin{array}{rcll}
(F \wedge F) & \equiv & F & \\
(F \vee F) & \equiv & F & \text{Idempotenz} \\[1em]
(F \wedge G) & \equiv & (G \wedge F) & \\
(F \vee G) & \equiv & (G \vee F) & \text{Kommutativität} \\[1em]
((F \wedge G) \wedge H) & \equiv & (F \wedge (G \wedge H)) & \\
((F \vee G) \vee H) & \equiv & (F \vee (G \vee H)) & \text{Assoziativität} \\[1em]
(F \wedge (F \vee G)) & \equiv & F & \\
(F \vee (F \wedge G)) & \equiv & F & \text{Absorption} \\[1em]
(F \wedge (G \vee H)) & \equiv & ((F \wedge G) \vee (F \wedge H)) & \\
(F \vee (G \wedge H)) & \equiv & ((F \vee G) \wedge (F \vee H)) & \text{Distributivität}
\end{array}$$

Weitere Äquivalenzen

$$\neg\neg F \equiv F \qquad \text{doppelte Negation}$$

$$\neg(F \wedge G) \equiv (\neg F \vee \neg G)$$
$$\neg(F \vee G) \equiv (\neg F \wedge \neg G) \qquad \text{de Morgansche Regeln}$$

$$(F \vee G) \equiv F\text{, falls } F \text{ Tautologie}$$
$$(F \wedge G) \equiv G\text{, falls } F \text{ Tautologie} \qquad \text{Tautologieregeln}$$

$$(F \vee G) \equiv G\text{, falls } F \text{ unerfüllbar}$$
$$(F \wedge G) \equiv F\text{, falls } F \text{ unerfüllbar} \qquad \text{Unerfüllbarkeitsregeln}$$

Beweise: jeweils mit Wahrheitstafeln.

Normalformen

Definition (Normalformen)

Ein Literal ist eine atomare Formel (positives Literal) oder die Negation einer atomaren Formel (negatives Literal).

Eine Formel F ist in konjunktiver Normalform (KNF), falls sie eine Konjunktion von Disjunktionen von Literalen ist.

Eine Formel F ist in disjunktiver Normalform (DNF), falls sie eine Disjunktion von Konjunktionen von Literalen ist.

Satz: Für jede Formel F gibt es eine äquivalente Formel in KNF und eine äquivalente Formel in DNF.

Beweis durch Induktion über den Formelaufbau von F.

Algorithmus zur Umformung in KNF

Gegeben: Formel F (Abkürzungen $\rightarrow$ und $\leftrightarrow$ seien bereits ersetzt).

1. Ersetze in F jedes Vorkommen einer Teilformel der Bauart
$$\begin{array}{rcl} \neg\neg G & \text{durch} & G \\ \neg(G \vee H) & \text{durch} & (\neg G \wedge \neg H) \\ \neg(G \wedge H) & \text{durch} & (\neg G \vee \neg H) \end{array}$$
bis keine solche Teilformel mehr vorkommt.

2. Ersetze in F jedes Vorkommen einer Teilformel der Bauart
$$\begin{array}{rcl} (F \vee (G \wedge H)) & \text{durch} & ((F \vee G) \wedge (F \vee H)) \\ ((F \wedge G) \vee H)) & \text{durch} & ((F \vee H) \wedge (G \vee H)) \end{array}$$
bis keine solche Teilformel mehr vorkommt (ausdistribuieren).

Umformung in DNF: im letzten Schritt $\vee$ und $\wedge$ vertauschen.

Hornformeln

Definition (Hornformeln)

Eine Formel F ist eine Hornformel, falls F in KNF ist und jede Disjunktion in F höchstens ein positives Literal enthält.

Bsp.: $(A \vee \neg B) \wedge (\neg C \vee \neg A \vee D) \wedge (\neg A \vee \neg B) \wedge D \wedge \neg E$

Äquivalente Darstellung durch folgende Implikationen:

$$(B \rightarrow A) \wedge (C \wedge A \rightarrow D) \wedge (A \wedge B \rightarrow 0) \wedge D \wedge (E \rightarrow 0)$$

Negative Literale werden (nichtnegiert) Vorbedingungen der Implikation, das positive (wenn existent) Nachbedingung (Kopf).

Falls kein positives Literal: 0 als Kopf (steht für Widerspruch).

Falls kein negatives Literal: es entsteht wie im Fall von D ein Fakt.

Beobachtung: die Implikationen enthalten keine Negation!

Erfüllbarkeitstest für Hornformeln

Eingabe: Hornformel F (in Implikationsform)

1. Markiere jedes Vorkommen von A in F, falls A Fakt ist;
2. **while** es gibt in F Teilformel $(A_1 \wedge \ldots \wedge A_n \rightarrow B)$ mit $A_1, \ldots, A_n$ markiert, B unmarkiert **do**

   if $B = 0$ **then** gib aus "unerfüllbar" und stoppe;
   else markiere jedes Vorkommen von B;

3. gib aus "erfüllbar" und stoppe.

Falls F erfüllbar, so ergibt sich ein Modell I aus den markierten Atomen: $I(A_i) = 1$ gdw. A_i markiert.

Dieses Modell ist das kleinste Modell, wobei $I \leq I'$ gdw. für alle Atome A_i: $I(A_i) \leq I'(A_i)$. Dabei gilt $0 < 1$.

Nicht-Hornformeln besitzen nicht immer ein kleinstes Modell.

Komplexität und Korrektheit

Satz: Der Markierungsalgorithmus ist korrekt und stoppt nach maximal n Markierungsschritten (n = Anzahl der Atome in F).

Beweisskizze:
Komplexität klar: jedes Atom kann nur einmal markiert werden.

Korrektheit: Schritte 1 und 2 markieren nur Atome, die in allen Modellen von F den Wert 1 haben müssen. Falls 0 markiert werden müsste (then-Fall), gibt es kein Modell und die Ausgabe ist korrekt.

Ansonsten ist nach Verlassen von 2. die Belegung I mit $I(A_i) = 1$ gdw. A_i markiert ein Modell von F, denn für jede Implikation G in F gilt: entweder sind die Vorbedingung und die Nachbedingung von G wahr in I oder die Vorbedingung ist falsch. In beiden Fällen ist G wahr.

Korollar: Falls Hornformel keine Implikation der Form $A_1 \wedge \ldots \wedge A_n \rightarrow 0$ enthält, so ist sie erfüllbar. Ebenso wenn kein Fakt vorkommt.

Endlichkeitssatz (Kompaktheit)

Satz: Eine Menge M von Formeln ist erfüllbar gdw. jede der endlichen Teilmengen von M erfüllbar ist.

Beweis: "$\Rightarrow$": offensichtlich: jedes Modell von M ist auch Modell jeder Teilmenge von M.

"$\Leftarrow$": 3 Fälle

1. M endlich:
   offensichtlich, da M endliche Teilmenge von sich selbst.
2. M unendlich, aber endlich viele Atome $\{A_1, \ldots, A_n\}$ in M:
   offensichtlich, da es höchstens 2^{2^n} paarweise nicht äquivalente Formeln in M geben kann; jedes Modell dieser endlichen Menge von Formeln ist auch Modell von M.
3. M unendlich und unendlich viele Atome $\{A_1, A_2, A_3, \ldots, \}$ in M:
   schwierig.

Beweis Kompaktheit

"$\Leftarrow$": schwieriger Fall: M unendlich und unendlich viele Atome $\{A_1, A_2, A_3, \ldots, \}$ in M

Für $n \geq 1$ sei M_n die Menge der Formeln in M, die nur die Atome $A_1, ..., A_n$ enthalten.

Es gibt $k \leq 2^{2^n}$ paarweise nicht äquivalente Formeln $F_1 \ldots F_k$ in M_n.

Also: für jedes F in M_n gibt es $i \leq k$ mit $F \equiv F_i$.

Jedes Modell für $\{F_1, ..., F_k\}$ ist Modell für M_n. So ein Modell existiert nach Voraussetzung.

Wir nennen es I_n. Es ist auch Modell für $M_1, M_2, ..., M_{n-1}$ (denn diese Mengen sind Teilmengen von M_n).

Beweis Kompaktheit: Konstruktion des Modells I für M

Notation: $(A_n, 1) \in I$ statt $I(A_n) = 1$. Ebenso für 0

Stufe 0: $I := \emptyset$;
$J := \{1, 2, 3, ...\}$.

Stufe $n > 0$: if es gibt unendlich viele Indizes $i \in J$ mit $I_i(A_n) = 1$ then
$I := I \cup \{(A_n, 1)\}$;
$J := J \setminus \{i \mid I_i(A_n) \neq 1\}$
else $I := I \cup \{(A_n, 0)\}$;
$J := J \setminus \{i \mid I_i(A_n) \neq 0\}$.

In Stufe n wird I um $(A_n, 0)$ oder $(A_n, 1)$ erweitert, nie um beides. $I : \{A_1, A_2, ...\} \rightarrow \{0, 1\}$ daher wohldefiniert. Sei F beliebige Formel in M. Der höchste Index eines Atoms in F sei k, d.h. $F \in M_k, M_{k+1}, ...$ und jede der Belegungen $I_k, I_{k+1}, ...$ ist Modell von F. Bei der Konstruktion von I wird J nie endlich. Damit verbleiben unendlich viele Indizes $i > k$ in J. Für all diese i gilt: $I_i(A_1) = I(A_1), ..., I_i(A_k) = I(A_k)$, deshalb ist I Modell von F.

Resolution

Syntaktische Umformungsregel, die aus zwei Formeln neue erzeugt.
Verwendet, um Unerfüllbarkeit einer Formel(menge) zu testen.
Setzt voraus, dass Formel(menge) in KNF vorliegt.

Darstellung der KNF in Mengennotation:

$$F = \{\{L_{1,1}, \ldots, L_{1,n_1}\}, \ldots, \{L_{k,1}, \ldots, L_{k,n_k}\}\}$$

statt

$$F = (L_{1,1} \vee \ldots \vee L_{1,n_1}) \wedge \ldots \wedge (L_{k,1} \vee \ldots \vee L_{k,n_k})$$

Jedes Element von F heißt Klausel.

Definition (Resolvente)

Seien K_1, K_2 Klauseln. R heißt Resolvente von K_1 und K_2, falls es ein Literal L gibt mit $L \in K_1$ und $-L \in K_2$ und $R = (K_1 \setminus \{L\}) \cup (K_2 \setminus \{-L\})$. Hierbei ist $-L = \neg A_i$ falls $L = A_i$, $-L = A_i$ falls $L = \neg A_i$.

Beispiel: $\{A, \neg C\}$ ist Resolvente aus $\{A, B\}$ und $\{\neg B, \neg C\}$.

Resolution

Spezialfall: die leere Klausel $\{\}$ entsteht durch Resolvieren von widersprüchlichen einelementigen Klauseln, etwa $\{A\}$,$\{\neg A\}$.

Sie entspricht falsch und wird auch als $\Box$ oder $\perp$ repräsentiert.

Resolutionslemma: Sei F eine Formel in KNF (in Klauselform), R Resolvente zweier Klauseln aus F. F und $F \cup \{R\}$ sind äquivalent.

Beweis: Sei I eine Belegung. Zu zeigen: $I \models F \cup \{R\}$ gdw. $I \models F$.

Falls $I \models F \cup \{R\}$, dann gilt offensichtlich auch $I \models F$.

Es gelte umgekehrt $I \models F$. Dann gilt $I \models K$ für jede Klausel $K \in F$.
Ferner sei $R = (K_1 \setminus \{L\}) \cup (K_2 \setminus \{-L\})$.

Fall 1: $I \models L$. Dann gilt $I \models K_2 \setminus \{-L\}$ und deshalb $I \models R$.
Fall 2: $I \not\models L$. Dann gilt $I \models K1 \setminus \{L\}$ und deshalb $I \models R$.

Resolutionshülle Res^*

Definition

Sei F eine Klauselmenge. Wir definieren:

$$\begin{array}{rcl} Res(F) & = & F \cup \{R \mid R \text{ Resolvente zweier Klauseln in } F\} \\ Res^0(F) & = & F \\ Res^{n+1}(F) & = & Res(Res^n(F)) \\ Res^*(F) & = & \bigcup_{n \geq 0} Res^n(F) \end{array}$$

Bemerkung: es kann 4^n verschiedene Klauseln aus n Atomen geben (jedes A_i kann nicht, positiv, negativ, positiv und negativ vorkommen).

Daraus folgt für endliche Klauselmengen:
es gibt ein k, so dass $Res^k(F) = Res^{k+1}(F) = \ldots = Res^*(F)$.

Res^* nicht immer ganz zu berechnen. Herleitung von $\square$ genügt:

Herleitung

Definition (Herleitung)

Eine Herleitung (Deduktion, Beweis) einer Klausel C aus einer Klauselmenge F ist eine Folge $K_1, ..., K_m$ von Klauseln, so dass:

1. $K_m = C$,
2. für $1 \leq i \leq m$: K_i aus F oder K_i Resolvente von K_j, K_k mit $j, k < i$.

1. Lineare Anordnung erzwingt, dass Resolventen nur aus bereits hergeleiteten Klauseln oder aus Klauseln in F erzeugt werden: keine unerwünschten Zyklen.

2. Darstellung in Resolutionsgraphen oft übersichtlicher: Knoten Klauseln, Kanten von resolvierten Klauseln zur Resolvente. Darstellungen ineinander überführbar.

Satz: $C \in Res^*(F)$ gdw. es eine Herleitung von C aus F gibt.

Ein nützliches Lemma

Lemma: Sei F Klauselmenge, A Atom in F. $F_{A=0}$ entstehe aus F durch

1. Streichen der Klauseln, in denen $\neg A$ vorkommt,
2. Streichen jedes Vorkommens von A in einer Klausel.

(Belegung von A mit 0 fixieren.) Analog entstehe $F_{A=1}$ aus F durch

1. Streichen der Klauseln, in denen A vorkommt,
2. Streichen jedes Vorkommens von $\neg A$ in einer Klausel.

(Belegung von A mit 1 fixieren.)

Es gilt: F ist unerfüllbar gdw. $F_{A=0}$ und $F_{A=1}$ unerfüllbar sind.

Beweisidee: $F_{A=0}$ erfüllbar gdw. es Modell M von F gibt mit $M(A) = 0$. Analog für $F_{A=1}$.

Resolutionssatz der Aussagenlogik

Satz: Eine Klauselmenge F ist unerfüllbar gdw. $\square \in Res^*(F)$.

Beweis: "<=" folgt aus Resolutionslemma: da $\square \in Res^k(F)$ für ein k und F äquivalent $Res^k(F)$, ist F unerfüllbar.

"=>" Sei F unerfüllbar. Falls F unendlich, muss es wegen Kompaktheit endliche Teilmenge von F geben, die unerfüllbar ist, d.h. es genügt, endliche Mengen F zu betrachten. Wir zeigen $\square \in Res^*(F)$ durch Induktion über die Anzahl n der in F vorkommenden Atome:

Induktionsanfang: $n = 0$, dann ist $F = \{\square\}$, also $\square \in Res^*(F)$

Induktionsschritt: Sei F Klauselmenge mit Atomen $A_1, ..., A_{n+1}$. Wir definieren aus F zwei Klauselmengen F_0 und F_1 wie folgt: F_0 (F_1) entsteht aus F durch Streichen jedes Vorkommens von A_{n+1} ($\neg A_{n+1}$) sowie durch Streichen der Klauseln, in denen $\neg A_{n+1}$ (A_{n+1}) vorkommt, d.h. wir fixieren die Belegung von A_{n+1} mit 0 (1).

Resolutionssatz: Fortsetzung Beweis

F_0 und F_1 sind unerfüllbar. Wäre etwa F_0 erfüllbar, so könnte man aus einem Modell M für F_0 ein Modell M' für F konstruieren, indem man $A_1, ..., A_n$ wie in M auswertet und A_{n+1} den Wert 0 gibt. Analog für F_1.

Nach Induktionsvoraussetzung muss es Resolutionsbeweis $K_1, ..., K_m$ aus F_0 für $\square$ geben. Benutzt man statt $K_1, ..., K_m$ die ursprünglichen Klauseln, erhält man einen Resolutionsbeweis für $\square$ oder für A_{n+1}. Im ersten Fall ist bewiesen, dass $\square$ aus F ableitbar ist.

Ebenso gibt es einen Resolutionsbeweis für $\square$ aus F_1, der durch Verwendung der ursprünglichen Klauseln zu Beweis für $\square$ oder für $\neg A_{n+1}$ wird. Wieder ist im ersten Fall die Behauptung bewiesen.

Ansonsten lässt sich aus A_{n+1} und $\neg A_{n+1}$ in einem weiteren Resolutionsschritt $\square$ ableiten, d.h. $\square \in Res^*(F)$.

Resolutionsalgorithmus

Aus Resolutionssatz lässt sich folgender Algorithmus ableiten:

Eingabe: Formel F in KNF (repräsentiert als Klauselmenge)
repeat $G := F$;
$F := Res(F)$
until $\square \in F$ or $F = G$;
if $\square \in F$ **then** "unerfüllbar" **else** "erfüllbar".

Es handelt sich um ein Entscheidungsverfahren:

- korrekt: Ausgabe immer richtig.
- terminiert immer.

Beispiel

Ist $\mathit{Haustier} \wedge \mathit{JagtKatzen}$ ableitbar aus:

1)	$4\mathit{Beiner} \wedge \mathit{Miaut} \rightarrow \mathit{Katze}$	$\{\neg 4\mathit{Beiner}, \neg \mathit{Miaut}, \mathit{Katze}\}$
2)	$4\mathit{Beiner} \wedge \mathit{Bellt} \rightarrow \mathit{Hund}$	$\{\neg 4\mathit{Beiner}, \neg \mathit{Bellt}, \mathit{Hund}\}$
3)	$\mathit{Katze} \rightarrow \mathit{Haustier}$	$\{\neg \mathit{Katze}, \mathit{Haustier}\}$
4)	$\mathit{Hund} \rightarrow \mathit{Haustier}$	$\{\neg \mathit{Hund}, \mathit{Haustier}\}$
5)	$\mathit{Katze} \rightarrow \mathit{JagtMäuse}$	$\{\neg \mathit{Katze}, \mathit{JagtMäuse}\}$
6)	$\mathit{Hund} \rightarrow \mathit{JagtKatzen}$	$\{\neg \mathit{Hund}, \mathit{JagtKatzen}\}$
7)	$4\mathit{Beiner}$	$\{4\mathit{Beiner}\}$
8)	Bellt	$\{\mathit{Bellt}\}$

Äquivalent:
Prämissenmenge plus $\{\neg \mathit{Haustier}, \neg \mathit{JagtKatzen}\}$ unerfüllbar?

Beispiel, Fortsetzung

1)	$\{\neg 4Beiner, \neg Miaut, Katze\}$	10)	$\{\neg Bellt, Hund\}$	2,7
2)	$\{\neg 4Beiner, \neg Bellt, Hund\}$	11)	$\{Hund\}$	10,8
3)	$\{\neg Katze, Haustier\}$	12)	$\{Haustier\}$	11,4
4)	$\{\neg Hund, Haustier\}$	13)	$\{\neg JagtKatzen\}$	12,9
5)	$\{\neg Katze, JagtMäuse\}$	14)	$\{JagtKatzen\}$	11,6
6)	$\{\neg Hund, JagtKatzen\}$	15)	$\Box$	14,13
7)	$\{4Beiner\}$			
8)	$\{Bellt\}$			
9)	$\{\neg Haustier, \neg JagtKatzen\}$			

Prämissen plus negiertes Ziel unerfüllbar $\Rightarrow$ Formel folgerbar

Beschränkungen des Resolutionsverfahrens

Unit-Resolution:
mindestens eine resolvierte Klausel einelementig (Einer-Klausel).

Input-Resolution:
mindestens eine resolvierte Klausel Eingabe-Klausel (aus F).

Satz: Unit- und Input-Resolution sind vollständig für Hornklauseln:
falls F Hornklauselmenge, so ist $\square$ ableitbar aus F gdw. F unerfüllbar.

Gegenbeispiel im allgemeinen Fall (Unit):

$$\{\neg A, \neg B\}, \{\neg A, B\}, \{A, \neg B\}, \{A, B\}.$$

unerfüllbar, aber keine Unit-Resolution möglich.

Lineare Resolution: jede Resolution (bis auf die erste) verwendet zuletzt erzeugte Resolvente (mit beliebiger anderer Klausel).

Satz: Lineare Resolution ist vollständig.

Das Davis-Putnam-Verfahren

Erfüllbarkeitstest für Formel F in Klauselform

Definition (reduzierte Klauselmenge)

Sei F eine Menge von Klauseln, L ein Literal. Die um L reduzierte Klauselmenge F_L entsteht aus F durch

1. Streichen aller Klauseln, die L enthalten,
2. Streichen des Komplements von L aus verbleibenden Klauseln.

Beispiel:	$F = \{\{A, B\}, \{C, \neg A\}, \{\neg B\}\}$
Reduzieren um A:	$F_A = \{\{C\}, \{\neg B\}\}$
Reduzieren um $\neg B$:	$F_{\neg B} = \{\{A\}, \{C, \neg A\}\}$

Jedes Modell M von F, das L zu wahr auswertet, ist Modell von F_L.
Zu jedem Modell M' von F_L gibt es Modell M von F, das Atome in F_L so auswertet wie M': $M(L) = 1$, M wie M' sonst.
also: F_L erfüllbar gdw. F Modell hat, das L zu wahr auswertet.

Das Davis-Putnam-Verfahren

Beobachtungen:

1. $S = \{\}$: F erfüllbar.
2. $\Box$ in F: F ist unerfüllbar.
3. Einerklausel $\{L\}$ in F: F erfüllbar gdw. F_L erfüllbar.
4. L Literal in F: F erfüllbar gdw. F_L erfüllbar oder F_{-L} erfüllbar.

Daraus ergibt sich der folgende rekursive Erfüllbarkeitstest:

boolean Satisfiable(S)
begin
  if $S = \{\}$ **then** return true;
  if $\Box \in S$ **then** return false;
  if Einer-Klausel in S **then** select $\{L\} \in S$; return Satisfiable(S_L)
  else select Literal L; return (Satisfiable(S_L) or Satisfiable(S_{-L}));
end;

Tableau für F: Baum, Knoten mit Formeln markiert, Wurzel mit F.

Pfad von Wurzel zu Blatt: Konjunktion der Formeln an Knoten.

Gesamter Baum: Disjunktion dieser Konjunktionen

Grundidee: F unerfüllbar, falls für jeden Pfad von Wurzel zu Blatt die entsprechende Konjunktion unerfüllbar.

Erzeugungsregeln für Tableaus:

$$\frac{\neg\neg H}{H} \qquad \frac{G_1 \wedge G_2}{\begin{array}{c} G_1 \\ G_2 \end{array}} \qquad \frac{\neg(G_1 \wedge G_2)}{\neg G_1 \mid \neg G_2} \qquad \frac{G_1 \vee G_2}{G_1 \mid G_2} \qquad \frac{\neg(G_1 \vee G_2)}{\begin{array}{c} \neg G_1 \\ \neg G_2 \end{array}}$$

lies: wenn in Pfad $\neg\neg H$ vorkommt, erweitere ihn um H,
wenn in Pfad $G_1 \wedge G_2$ vorkommt, erweitere ihn um G_1 und um G_2,
wenn in Pfad $\neg(G_1 \wedge G_2)$ vorkommt, verzweige und erweitere um linken Nachfolger $\neg G_1$ und rechten Nachfolger $\neg G_2$, etc.

Tableau-Verfahren

Definition (Tableau für beliebige Formel F (KNF nicht nötig))

Die Menge der Tableaus für Formel F ist induktiv wie folgt definiert:

1. der Baum, der aus einem mit F markierten Knoten besteht, ist ein Tableau für F,
2. wenn T ein Tableau für F ist und T' durch Anwendung einer Erzeugungsregel aus T entsteht, so ist T' ein Tableau für F.

Definition (abgeschlossenes Tableau)

Ein Ast eines Tableaus ist ein Pfad von der Wurzel zu einem Blatt.
Ein Ast ist abgeschlossen, wenn in ihm eine Formel und ihre Negation vorkommt.
Ein Tableau ist abgeschlossen, wenn jeder Ast abgeschlossen ist.

Satz: F ist unerfüllbar gdw. es ein abgeschlossenes Tableau für F gibt.

Beispiele

$$
\begin{array}{c}
\neg(\neg(A \wedge B) \vee A) \\
| \\
\neg\neg(A \wedge B) \\
| \\
\neg A \\
| \\
A \wedge B \\
| \\
A \\
| \\
B \\
\times
\end{array}
\qquad\qquad
\begin{array}{ccc}
 & \neg(\neg(A \vee B) \vee A) & \\
 & | & \\
 & \neg\neg(A \vee B) & \\
 & | & \\
 & \neg A & \\
 & | & \\
 & A \vee B & \\
/ & & \backslash \\
A & & B \\
\times & & \checkmark
\end{array}
$$

Modell: $A \mapsto 0$, $B \mapsto 1$

Sie finden zu Hause einen Zettel:

"Es ist noch Pizza und Bier da. Mama kommt Freitag den 13."

aber: Freitag ist der 12. Gehen Sie Bier kaufen?

Idee: betrachte maximale konsistente Teilmengen der Prämissen.

Definition (maximal konsistente Teilmenge)

Sei H Formelmenge. $H' \subseteq H$ heißt maximal konsistente Teilmenge von H gdw. H' konsistent und kein H'' mit $H' \subset H'' \subseteq H$ ist konsistent.

Beobachtungen:

Jedes H besitzt mindestens 1 maximal konsistente Teilmenge.

Es kann mehrere solche Teilmengen geben: $H = \{A, \neg A\}$

Falls H konsistent, so ist H selbst die einzige solche Teilmenge.

Folgerungsoperator für inkonsistente Information

Definition (*mc*-Folgerungsoperator)

Sei H Formelmenge, $mc(H)$ die Menge der maximal konsistenten Teilmengen von H. Es gelte $H \models_{mc} G$ gdw. $G \in \bigcap_{S \in mc(H)} Th(S)$.

$$H = \{P, B, F, 13, F \rightarrow \neg 13\}$$

$$mc(H) = \{\{P, B, F, 13\}, \{P, B, F, F \rightarrow \neg 13\}, \{P, B, 13, F \rightarrow \neg 13\}\}$$

Jetzt gilt: $\quad H \models_{mc} P \quad H \models_{mc} B \quad H \models_{mc} F \vee 13$

aber keine unerwünschten Folgerungen: $H \not\models_{mc} \neg B \wedge \neg P$

Beobachtungen:

Die Menge $\{F \mid H \models_{mc} F\}$ ist erfüllbar.

$\models_{mc}$ ist nichtmonoton: $\{A\} \models_{mc} A$, aber $\{A, \neg A\} \not\models_{mc} A$.

Falls H konsistent, so ist $\models = \models_{mc}$.