

Mathematische Logik

SS 2010

Prof. Dr. Erich Grädel

Mathematische Grundlagen der Informatik
RWTH Aachen

Inhaltsverzeichnis

1	Aussagenlogik	1
1.1	Syntax und Semantik der Aussagenlogik	1
1.2	Aussagenlogik und Boolesche Funktionen	7
1.3	Horn-Formeln	12
1.4	Der Kompaktheitssatz der Aussagenlogik	15
1.5	Aussagenlogische Resolution	21
1.6	Der aussagenlogische Sequenzenkalkül	28
2	Syntax und Semantik der Prädikatenlogik	37
2.1	Strukturen	38
2.2	Ein Zoo von Strukturen	40
2.3	Syntax der Prädikatenlogik	45
2.4	Semantik der Prädikatenlogik	49
2.5	Normalformen	53
2.6	Spieltheoretische Semantik	61



This work is licensed under:

<http://creativecommons.org/licenses/by-nc-nd/3.0/de/>

Dieses Werk ist lizenziert unter:

<http://creativecommons.org/licenses/by-nc-nd/3.0/de/>

© 2010 Mathematische Grundlagen der Informatik, RWTH Aachen.

<http://www.logic.rwth-aachen.de>

1 Aussagenlogik

1.1 Syntax und Semantik der Aussagenlogik

Die Aussagenlogik (AL) untersucht Ausdrücke, die aus atomaren Aussagen (den Aussagenvariablen) allein mit Hilfe der aussagenlogischen Junktoren gebildet werden. Die Aussagenvariablen werden interpretiert durch die Wahrheitswerte 0 (für *falsch*) und 1 (für *wahr*).

Für mathematische Zwecke ist die Aussagenlogik relativ uninteressant, da sie zu ausdruckschwach ist. Viele grundlegende Aspekte stärkerer Logiken lassen sich jedoch im Kontext der Aussagenlogik übersichtlich behandeln und veranschaulichen. Zudem ergeben sich in der Aussagenlogik zahlreiche interessante *algorithmische Probleme* mit fundamentaler Bedeutung für die Informatik, so etwa die Komplexität des Erfüllbarkeitsproblems, die Suche nach effizienten Beweissystemen, sowie die Spezifikation und effiziente Berechnung Boolescher Funktionen.

Syntax

Formeln sind *syntaktische Objekte*, d.h. Wörter in einer formalen Sprache. Die Menge der aussagenlogischen Formeln ist induktiv als Wortmenge über einem Alphabet definiert, welches aus folgenden Symbolen besteht:

- einer festen Menge τ von *Aussagenvariablen*,
- den Booleschen Konstanten 0, 1,
- den *aussagenlogischen Junktoren* \neg , \wedge , \vee und \rightarrow ,
- den Klammersymbolen $(,)$.

Meistens wird eine feste, abzählbar unendliche Menge $\tau = \{X_0, X_1, X_2 \dots\}$ von Aussagenvariablen zugrundegelegt. Für gewisse

Anwendungen der Aussagenlogik ist es jedoch sinnvoll, beliebige (auch überabzählbare) Mengen τ zuzulassen.

Definition 1.1. Die Menge AL der *aussagenlogischen Formeln* ist induktiv definiert durch

- (1) $0, 1 \in \text{AL}$ (die Booleschen Konstanten sind Formeln).
- (2) $\tau \subseteq \text{AL}$ (jede Aussagenvariable ist eine Formel).
- (3) Wenn $\psi, \varphi \in \text{AL}$, dann sind auch die Wörter $\neg\psi$, $(\psi \wedge \varphi)$, $(\psi \vee \varphi)$ und $(\psi \rightarrow \varphi)$ Formeln aus AL.

Boolesche Konstanten und Aussagenvariablen nennen wir auch *atomare Formeln*. Die Formel $\neg\psi$ wird gelesen „nicht ψ “ und ist die *Negation* von ψ . Die Formeln $(\psi \vee \varphi)$, gelesen „ ψ oder φ “, und $(\psi \wedge \varphi)$, gelesen „ ψ und φ “, heißen die *Disjunktion* bzw. *Konjunktion* von ψ und φ . Wir nennen $(\psi \rightarrow \varphi)$ die *Implikation* von ψ nach φ und lesen sie „ ψ Pfeil φ “ oder „ ψ impliziert φ “.

Konventionen zur Notation von Formeln. Zur Verbesserung der Lesbarkeit bedienen wir uns abkürzender oder vereinfachender Schreibweisen. Zum Beispiel werden wir in Formeln oft Klammern weglassen, welche für das Verständnis überflüssig sind. Wir vereinbaren, dass \neg stärker bindet als alle andern Junktoren und dass \wedge und \vee stärker binden als \rightarrow . So steht etwa $\psi \wedge \neg\varphi \rightarrow \vartheta$ für $((\psi \wedge \neg\varphi) \rightarrow \vartheta)$. Außerdem vereinbaren wir implizite Linksklammerung bei iterierten Disjunktionen und Konjunktionen: $\psi \wedge \varphi \wedge \eta$ steht für $((\psi \wedge \varphi) \wedge \eta)$. Für iterierte Konjunktionen und Disjunktionen über Formeln $\varphi_1, \dots, \varphi_n$ verwenden wir die Schreibweisen $\bigwedge_{i=1}^n \varphi_i$ und $\bigvee_{i=1}^n \varphi_i$.

INDUKTION ÜBER DEN FORMELAUFBAU. Jede Formel $\psi \in \text{AL}$ ist ein Wort über dem Alphabet $\Gamma := \tau \cup \{0, 1, \neg, \wedge, \vee, \rightarrow, (,)\}$, aber natürlich ist nicht jedes Wort aus Γ^* eine Formel. Definition 1.1 ist ein Beispiel für eine *induktive* (durch Konstruktionsregeln gegebene) Definition. Sie ist so zu verstehen, dass außer den nach den Regeln (1) – (3) festgelegten Formeln keine weiteren Zeichenketten aussagenlogische Formeln sind. Mit andern Worten: AL ist die kleinste Menge von Wörtern aus

Γ^* , welche 0, 1 sowie alle Aussagenvariablen $X \in \tau$ enthält, und die unter der Regel (3) abgeschlossen ist, die also mit ψ und φ auch die Zeichenketten $\neg\psi$, $(\psi \wedge \varphi)$, $(\psi \vee \varphi)$ und $(\psi \rightarrow \varphi)$ enthält.

Der induktive Aufbau von Formeln erlaubt das Prinzip der strukturellen Induktion für Definitionen und Beweise. Induktionsbeweise über den Formelaufbau folgen folgendem Muster. Um nachzuweisen, dass alle Formeln in AL eine Eigenschaft E besitzen, zeigt man:

- Alle atomaren Formeln haben die Eigenschaft E .
- Haben ψ und $\varphi \in \text{AL}$ die Eigenschaft E , so auch $\neg\psi$ und $(\psi \circ \varphi)$, für $\circ \in \{\wedge, \vee, \rightarrow\}$.

Mit diesem Beweisprinzip kann man leicht die *eindeutige Lesbarkeit von Formeln* einsehen: Kein echtes Anfangsstück einer Formel ist selbst eine Formel und daher kann man jede Formel auf genau eine Weise gemäß den Regeln (1) – (3) von Definition 1.1 in ihre unmittelbaren Bestandteile zerlegen.

Daraus folgt insbesondere, dass induktive Definitionen über den Formelaufbau eindeutig sind: So können wir etwa die *Tiefe* $d(\psi)$ einer Formel $\psi \in \text{AL}$ induktiv wie folgt definieren:

- $d(\psi) := 0$ für atomare ψ ,
- $d(\neg\psi) := d(\psi) + 1$,
- $d((\psi \circ \varphi)) := \max(d(\psi), d(\varphi)) + 1$.

Die Tiefe ist oft ein adäquateres Maß für die Komplexität einer Formel als deren Länge. Eine *Unterformel* einer Formel $\psi \in \text{AL}$ ist ein Teilwort von ψ , welches selbst eine Formel ist. Die Unterformeln von $\psi := (X_1 \vee X_3) \wedge (X_2 \vee (X_3 \rightarrow \neg X_1))$ sind

$$\psi, (X_1 \vee X_3), (X_2 \vee (X_3 \rightarrow \neg X_1)), (X_3 \rightarrow \neg X_1), \neg X_1, X_1, X_2, X_3.$$

Die Tiefe von ψ ist 4.

Übung 1.1. Geben Sie eine *induktive* Definition für die Menge der Unterformeln einer aussagenlogischen Formel an. Zeigen Sie:

- (a) Formeln der Länge n haben höchstens n Unterformeln.
- (b) Formeln der Tiefe n haben höchstens $2^{n+1} - 1$ Unterformeln.

- (c) Es existieren für jedes $n \in \mathbb{N}$ Formeln der Tiefe n mit genau $2^{n+1} - 1$ Unterformeln.

Übung 1.2. Zeigen Sie, dass das Prinzip der eindeutigen Lesbarkeit von Formeln erhalten bleibt, wenn wir die sog. *polnische Notation* verwenden, welche ganz ohne Klammern auskommt. Die Regel (3) in Definition 1.1 wird dabei ersetzt durch

- (3) Wenn ψ und φ aussagenlogische Formeln sind, dann auch die Ausdrücke $\neg\psi$, $\wedge\psi\varphi$, $\vee\psi\varphi$ und $\rightarrow\psi\varphi$.

Man zeige andererseits, dass die eindeutige Lesbarkeit nicht mehr gewährleistet ist wenn in Definition 1.1 die Klammern einfach weggelassen werden, d.h. wenn mit ψ und φ auch die Ausdrücke $\psi \wedge \varphi$, $\psi \vee \varphi$ und $\psi \rightarrow \varphi$ als Formeln zugelassen werden.

Semantik

Für jede Formel $\psi \in AL$ sei $\tau(\psi) \subseteq \tau$ die Menge der in ψ tatsächlich vorkommenden Aussagenvariablen. Für Formelmengen $\Phi \in AL$ ist $\tau(\Phi) = \bigcup_{\varphi \in \Phi} \tau(\varphi)$.

Definition 1.2. Eine (*aussagenlogische*) *Interpretation* ist eine Abbildung $\mathcal{I} : \sigma \rightarrow \{0, 1\}$ für ein $\sigma \subseteq \tau$. Sie ist *passend* für eine Formel $\psi \in AL$, wenn $\tau(\psi) \subseteq \sigma$. Jede zu ψ passende Interpretation \mathcal{I} definiert einen Wahrheitswert $\llbracket \psi \rrbracket^{\mathcal{I}} \in \{0, 1\}$, durch die folgenden Festlegungen:

- $\llbracket 0 \rrbracket^{\mathcal{I}} := 0$, $\llbracket 1 \rrbracket^{\mathcal{I}} := 1$.
- $\llbracket X \rrbracket^{\mathcal{I}} := \mathcal{I}(X)$ für $X \in \sigma$.
- $\llbracket \neg\psi \rrbracket^{\mathcal{I}} := 1 - \llbracket \psi \rrbracket^{\mathcal{I}}$.
- $\llbracket \psi \wedge \varphi \rrbracket^{\mathcal{I}} := \min(\llbracket \psi \rrbracket^{\mathcal{I}}, \llbracket \varphi \rrbracket^{\mathcal{I}})$.
- $\llbracket \psi \vee \varphi \rrbracket^{\mathcal{I}} := \max(\llbracket \psi \rrbracket^{\mathcal{I}}, \llbracket \varphi \rrbracket^{\mathcal{I}})$.
- $\llbracket \psi \rightarrow \varphi \rrbracket^{\mathcal{I}} := \llbracket \neg\psi \vee \varphi \rrbracket^{\mathcal{I}}$.

Ein *Modell* einer Formel $\psi \in AL$ ist eine Interpretation \mathcal{I} mit $\llbracket \psi \rrbracket^{\mathcal{I}} = 1$. Statt $\llbracket \psi \rrbracket^{\mathcal{I}} = 1$ schreibt man auch $\mathcal{I} \models \psi$ und sagt \mathcal{I} *erfüllt* ψ .

Ein *Modell* einer Formelmenge $\Phi \subseteq AL$ ist eine Interpretation \mathcal{I} mit $\mathcal{I} \models \psi$ für alle $\psi \in \Phi$, wofür wir auch $\mathcal{I} \models \Phi$ schreiben.

Nicht alle Aussagenvariablen im Definitionsbereich einer zu ψ passenden Interpretation \mathcal{I} müssen in ψ auch tatsächlich vorkommen. Offensichtlich ist aber für die Definition von $\llbracket \psi \rrbracket^{\mathcal{I}}$ die Interpretation der in ψ gar nicht vorkommenden Aussagenvariablen unerheblich. Dieser Sachverhalt, den man durch eine einfache Induktion über den Formelaufbau nachweisen kann, wird durch das Koinzidenzlemma ausgedrückt.

Lemma 1.3 (Koinzidenzlemma). Sei $\psi \in AL$ eine Formel und seien \mathcal{I} und \mathcal{I}' zwei zu ψ passende Interpretationen, so dass $\mathcal{I}(X) = \mathcal{I}'(X)$ für alle $X \in \tau(\psi)$. Dann ist $\llbracket \psi \rrbracket^{\mathcal{I}} = \llbracket \psi \rrbracket^{\mathcal{I}'}$.

Übung 1.3 (Auswerten aussagenlogischer Formeln). Geben Sie einen (möglichst effizienten) Algorithmus an, welcher zu einer gegebenen Formel $\psi \in AL$ und einer gegebenen Interpretation \mathcal{I} den Wahrheitswert $\llbracket \psi \rrbracket^{\mathcal{I}}$ berechnet. Beurteilen Sie die Laufzeit und den Bedarf an Speicherplatz des Algorithmus.

Übung 1.4. Geben Sie eine Formel ψ an, welche die Aussagenvariablen X_1, X_2, X_3 enthält, so dass für jede Interpretation $\mathcal{I} : \{X_1, X_2, X_3\} \rightarrow \{0, 1\}$ gilt, dass das Ändern jedes Wahrheitswerts $\mathcal{I}(X_i)$ auch den Wahrheitswert $\llbracket \psi \rrbracket^{\mathcal{I}}$ ändert.

Notation. In diesem Kapitel stehen kleine griechische Buchstaben $\psi, \varphi, \vartheta, \dots$ immer für aussagenlogische Formeln und große griechische Buchstaben Φ, Γ für Mengen aussagenlogischer Formeln. Wir verwenden die Schreibweise $\psi(X_1, \dots, X_n)$ um anzudeuten, dass $\tau(\psi)$ eine Teilmenge von $\{X_1, \dots, X_n\}$ ist. Sei $\mathcal{I}(X_1) = w_1, \dots, \mathcal{I}(X_n) = w_n$. Dann schreiben wir auch $\llbracket \psi(w_1, \dots, w_n) \rrbracket$ oder $\llbracket \psi(w) \rrbracket$ für $\llbracket \psi \rrbracket^{\mathcal{I}}$.

Definition 1.4. Zwei Formeln ψ und φ heißen *logisch äquivalent* (kurz: $\psi \equiv \varphi$), wenn für jede zu beiden Formeln passende Interpretation \mathcal{I} gilt, dass $\llbracket \psi \rrbracket^{\mathcal{I}} = \llbracket \varphi \rrbracket^{\mathcal{I}}$.

Hier sind ein paar einfache logische Äquivalenzen. Der Nachweis ergibt sich unmittelbar aus der Definition der Modellbeziehung. Für beliebige Formeln $\psi, \varphi, \vartheta \in AL$ gilt:

- $\neg\neg\psi \equiv \psi$ (Elimination der doppelten Negation)
- $\neg(\psi \wedge \varphi) \equiv \neg\psi \vee \neg\varphi$
 $\neg(\psi \vee \varphi) \equiv \neg\psi \wedge \neg\varphi$ (De Morgan'sche Gesetze)
- $\psi \wedge (\varphi \vee \vartheta) \equiv (\psi \wedge \varphi) \vee (\psi \wedge \vartheta)$
 $\psi \vee (\varphi \wedge \vartheta) \equiv (\psi \vee \varphi) \wedge (\psi \vee \vartheta)$ (Distributivgesetze)
- $\psi \rightarrow \varphi \equiv \neg\varphi \rightarrow \neg\psi$ (Kontraposition)
- $\psi \wedge (\psi \vee \varphi) \equiv \psi \vee (\psi \wedge \varphi) \equiv \psi$ (Absorption)
- $\psi \wedge \psi \equiv \psi$
 $\psi \vee \psi \equiv \psi$ (Idempotenz von \wedge und \vee)
- $\psi \wedge \varphi \equiv \varphi \wedge \psi$
 $\psi \vee \varphi \equiv \varphi \vee \psi$ (Kommutativität von \wedge und \vee)
- $\psi \wedge (\varphi \wedge \vartheta) \equiv (\psi \wedge \varphi) \wedge \vartheta$
 $\psi \vee (\varphi \vee \vartheta) \equiv (\psi \vee \varphi) \vee \vartheta$ (Assoziativität von \wedge und \vee)

Die Assoziativität, Kommutativität und Idempotenz von \wedge und \vee impliziert, dass es bei der Bildung der Konjunktion bzw. Disjunktion über eine endliche Folge $\varphi_1, \dots, \varphi_n$ von Formeln nicht auf die Reihenfolge und Multiplizität der Formeln ankommt. Dies rechtfertigt, dass wir Konjunktionen und Disjunktionen über endliche Formelmengen $\Phi = \{\varphi_1, \dots, \varphi_n\}$ bilden; anstelle von $\bigwedge_{i=1}^n \varphi_i$ verwenden wir auch die Schreibweisen $\bigwedge_{\varphi \in \Phi} \varphi$ oder $\bigwedge \Phi$, und analog $\bigvee_{\varphi \in \Phi} \varphi$ oder $\bigvee \Phi$ für die Disjunktion. (Dabei ist natürlich immer vorauszusetzen, dass Φ endlich ist!) Wenn Φ die leere Menge ist, identifizieren wir $\bigwedge \Phi$ mit 1 und $\bigvee \Phi$ mit 0.

Übung 1.5. Beweisen oder widerlegen Sie folgende Aussagen:

- (a) $\psi \wedge (\varphi \rightarrow \vartheta) \equiv (\psi \wedge \varphi) \rightarrow \vartheta \equiv \varphi \wedge (\psi \rightarrow \vartheta)$
- (b) $\neg\varphi_1 \vee \neg\varphi_2 \vee \dots \vee \neg\varphi_n \vee \psi \equiv \varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \psi$
 $\neg\varphi_1 \vee \neg\varphi_2 \vee \dots \vee \neg\varphi_n \equiv \varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow 0$
- (c) $\psi \rightarrow (\varphi \wedge \vartheta) \equiv (\psi \rightarrow \varphi) \wedge (\psi \rightarrow \vartheta)$
 $\psi \wedge \varphi \rightarrow \vartheta \equiv (\psi \rightarrow \vartheta) \vee (\varphi \rightarrow \vartheta)$
 $(\psi \vee \varphi) \rightarrow \vartheta \equiv (\psi \rightarrow \vartheta) \wedge (\varphi \rightarrow \vartheta)$

Definition 1.5. Hat eine Formel ein Modell, dann heißt sie *erfüllbar*, andernfalls *unerfüllbar*. Eine Formel ψ heißt *allgemeingültig* oder eine

Tautologie, wenn jede zu ψ passende Interpretation ein Modell von ψ ist. Man schreibt $\models \psi$ um anzudeuten, dass ψ eine Tautologie ist.

Lemma 1.6. Eine Formel ψ ist erfüllbar genau dann, wenn $\neg\psi$ keine Tautologie ist.

Es gibt ein offensichtliches Verfahren um festzustellen, ob eine aussagenlogische Formel $\psi(X_1, \dots, X_n)$ erfüllbar (oder allgemeingültig) ist: Man prüft für alle Interpretationen $\mathcal{I} : \{X_1, \dots, X_n\} \rightarrow \{0, 1\}$ mittels des in Übung 1.3 entwickelten Auswertungsalgorithmus nach, ob $\mathcal{I} \models \psi$. Obwohl für jede einzelne Interpretation \mathcal{I} dies sehr schnell nachgeprüft werden kann, ist das Verfahren insgesamt doch extrem ineffizient, da es bei n Aussagenvariablen 2^n mögliche Interpretationen gibt. Für Formeln mit Hunderten von Aussagenvariablen (was in praktischen Anwendungen durchaus realistisch ist) wären also selbst die schnellsten Rechner hoffnungslos überfordert. Natürlich gibt es bessere Verfahren, aber es ist nicht bekannt, ob das exponentielle Wachstum der Berechnungszeit durch raffiniertere Algorithmen vermieden werden kann. Man vermutet, dass dies nicht der Fall ist, dass also das Erfüllbarkeitsproblem der Aussagenlogik (genannt SAT) inhärent exponentiell schwierig ist, da es zu den NP-vollständigen Problemen gehört.

Übung 1.6. Beweisen Sie das *aussagenlogische Interpolationstheorem*: Sei $\psi \rightarrow \varphi$ eine aussagenlogische Tautologie. Dann existiert eine aussagenlogische Formel ϑ mit $\tau(\vartheta) \subseteq \tau(\psi) \cap \tau(\varphi)$, so dass $\psi \rightarrow \vartheta$ und $\vartheta \rightarrow \varphi$ Tautologien sind.

Hinweis: Führen Sie einen Induktionsbeweis über die Anzahl der Aussagenvariablen, die in ψ , aber nicht in φ vorkommen.

1.2 Aussagenlogik und Boolesche Funktionen

Eine (n -stellige) Boolesche Funktion ist eine Funktion $f : \{0, 1\}^n \rightarrow \{0, 1\}$. Sei B^n die Menge aller n -stelligen Booleschen Funktionen und $B = \bigcup_{n \in \mathbb{N}} B^n$. B^0 enthält die konstanten Funktionen 0 und 1. B^1 enthält vier Funktionen $f_{00}, f_{01}, f_{10}, f_{11}$ mit

$$f_{00}(0) = f_{00}(1) = 0, \quad f_{11}(0) = f_{11}(1) = 1,$$

$$f_{01}(w) = w, \quad f_{10}(w) = 1 - w.$$

B^n enthält 2^{2^n} verschiedene Funktionen.

Jede Formel $\psi(X_1, \dots, X_n) \in AL$ definiert eine Boolesche Funktion $h_\psi \in B^n$, durch die Vorschrift $h_\psi(w_1, \dots, w_n) := \llbracket \psi(w_1, \dots, w_n) \rrbracket$. Der folgende Satz zeigt, dass sich umgekehrt jede Boolesche Funktion durch eine aussagenlogische Formel darstellen lässt.

Satz 1.7. Zu jeder Funktion $f \in B^n$ gibt es eine Formel $\psi(X_1, \dots, X_n)$ mit $h_\psi = f$.

Beweis. Die Funktionen in B^0 werden durch die Formeln 0 und 1 dargestellt. Sei nun $n > 0$ und $f \in B^n$. Für jede Aussagenvariable X setzen wir $X^1 := X$ und $X^0 := \neg X$. Weiter definieren wir für jedes $v = v_1, \dots, v_n$ die Formel $\varphi^v := X_1^{v_1} \wedge \dots \wedge X_n^{v_n}$. Man beachte, dass für alle $v, w \in \{0, 1\}^n$ gilt:

$$\llbracket \varphi^v(w) \rrbracket = 1 \text{ gdw. } v = w$$

Die Funktion f wird nun dargestellt durch die Formel

$$\psi(X_1, \dots, X_n) := \bigvee_{\substack{v \in \{0,1\}^n \\ f(v)=1}} \varphi^v.$$

Wir müssen zeigen, dass $f(w) = \llbracket \psi(w) \rrbracket$ für alle $w \in \{0, 1\}^n$.

Sei $f(w) = 1$. Dann ist φ^w ein Disjunktionsglied von ψ , und da $\llbracket \varphi^w(w) \rrbracket = 1$, ist auch $\llbracket \psi(w) \rrbracket = 1$. Wenn aber $f(w) = 0$, dann gilt für jede Teilformel φ^v von ψ , dass $v_i \neq w_i$ für mindestens ein i , und daher $\llbracket \varphi^v(w) \rrbracket = 0$. Also ist $\llbracket \psi(w) \rrbracket = 0$. Q.E.D.

Aus dem Beweis von Satz 1.7 ergeben sich noch weitere wichtige Konsequenzen.

DISJUNKTIVE UND KONJUNKTIVE NORMALFORM. Ein *Literal* ist eine Aussagenvariable X oder deren Negation $\neg X$. Mit \bar{Y} bezeichnen wir das zu Y komplementäre Literal, also $\bar{X} := \neg X$ und $\overline{\neg X} := X$ für jede Aussagenvariable X .

Definition 1.8. Eine Formel $\psi \in AL$ ist in *disjunktiver Normalform* (DNF), wenn sie eine Disjunktion von Konjunktionen von Literalen ist, d.h. wenn sie die Form $\bigvee_{i=1}^n \bigwedge_{j=1}^{m_i} Y_{ij}$ hat, wobei die Y_{ij} Literale sind. Der duale Begriff ist die *konjunktive Normalform* (KNF); Formeln in KNF sind Konjunktionen von Disjunktionen von Literalen, also Formeln der Gestalt $\bigwedge_{i=1}^n \bigvee_{j=1}^{m_i} Y_{ij}$.

Die im Beweis von Satz 1.7 konstruierte Formel

$$\psi(X_1, \dots, X_n) := \bigvee_{\substack{v \in \{0,1\}^n \\ f(v)=1}} \varphi^v = \bigvee_{\substack{(v_1, \dots, v_n) \in \{0,1\}^n \\ f(v_1, \dots, v_n)=1}} X_1^{v_1} \wedge \dots \wedge X_n^{v_n}$$

zur Darstellung der Booleschen Funktion f ist in disjunktiver Normalform. Da jede Formel eine Boolesche Funktion definiert folgt unmittelbar, dass es zu jeder Formel $\psi \in AL$ eine äquivalente DNF-Formel gibt.

Die analoge Aussage zur KNF erhalten wir wie folgt. Da zu jeder Formel eine äquivalente Formel in DNF existiert, gilt dies insbesondere auch für $\neg\psi$:

$$\neg\psi \equiv \bigvee_{i=1}^n \bigwedge_{j=1}^{m_i} Y_{ij}.$$

Aus den De Morgan'schen Gesetzen folgt, dass für beliebige Formeln $\vartheta_1, \dots, \vartheta_n$ gilt:

$$\neg \bigvee_{k=1}^m \vartheta_k \equiv \bigwedge_{k=1}^m \neg \vartheta_k, \quad \neg \bigwedge_{k=1}^m \vartheta_k \equiv \bigvee_{k=1}^m \neg \vartheta_k.$$

Also folgt:

$$\psi \equiv \neg \bigvee_{i=1}^n \bigwedge_{j=1}^{m_i} Y_{ij} \equiv \bigwedge_{i=1}^n \neg \bigwedge_{j=1}^{m_i} Y_{ij} \equiv \bigwedge_{i=1}^n \bigvee_{j=1}^{m_i} \bar{Y}_{ij} =: \psi_K.$$

ψ_K ist in KNF und hat die geforderten Eigenschaften. Damit haben wir folgenden Satz bewiesen.

Satz 1.9. Zu jeder Formel $\psi \in \text{AL}$ gibt es äquivalente Formeln ψ_D in DNF und ψ_K in KNF.

Übung 1.7. Führen Sie einen alternativen Beweis für Satz 1.7, indem Sie per Induktion nach n nachweisen, dass es 2^n nicht-äquivalente aussagenlogische Formeln $\psi(X_1, \dots, X_n)$ gibt.

Übung 1.8. Geben Sie einen Algorithmus an, welcher unter Verwendung elementarer Umformungsregeln, z.B. der De Morgan'schen Regeln und der Distributivgesetze, eine gegebene aussagenlogische Formel in äquivalente DNF bzw. KNF-Formeln überführt. Wenden Sie dieses Verfahren auf die Formel $(X_1 \rightarrow X_2) \wedge ((X_1 \wedge X_3) \rightarrow X_2) \wedge (X_2 \rightarrow X_3)$ an. Zeigen Sie, dass in gewissen Fällen die resultierenden DNF- bzw. KNF-Formeln exponentiell länger werden als die gegebene Formel.

Übung 1.9. Zwei Formeln heißen *erfüllbarkeitsäquivalent*, wenn beide erfüllbar oder beide unerfüllbar sind. (Erfüllbarkeitsäquivalente Formeln müssen natürlich nicht unbedingt äquivalent sein.) Eine aussagenlogische Formel ist in 3-KNF, wenn sie folgende Gestalt hat:

$$\bigwedge_{i=1}^n Y_{i1} \vee Y_{i2} \vee Y_{i3} \quad (Y_{ij} \text{ Literale})$$

Zeigen Sie, dass man zu jeder Formel ψ in KNF eine erfüllbarkeitsäquivalente Formel in 3-KNF konstruieren kann, und zwar mit einem Verfahren, dessen Laufzeit durch ein Polynom in der Länge von ψ beschränkt ist.

Hinweis: Man fasse überzählige Literale mit Hilfe neuer Aussagenvariablen zusammen.

Übung 1.10. Zeigen Sie, dass das Erfüllbarkeitsproblem für DNF-Formeln durch einen Algorithmus mit linearer Laufzeit (bezüglich der Länge der Formel) gelöst werden kann.

FUNKTIONAL VOLLSTÄNDIGE MENGEN. Die Konstanten $0, 1$ und die Junktoren $\neg, \wedge, \vee, \rightarrow$ können als Funktionen in B^0, B^1 bzw. B^2 aufgefasst werden. Umgekehrt kann man aus jeder Booleschen Funktion $f \in B^n$ einen aussagenlogischen Junktorkonstruktor definieren: Aus Formeln $\varphi_1, \dots, \varphi_n \in$

AL bildet man eine neue Formel $f(\varphi_1, \dots, \varphi_n)$, deren Semantik auf naheliegende Weise festgelegt ist:

$$\llbracket f(\varphi_1, \dots, \varphi_n) \rrbracket^{\mathcal{J}} := f(\llbracket \varphi_1 \rrbracket^{\mathcal{J}}, \dots, \llbracket \varphi_n \rrbracket^{\mathcal{J}}).$$

Die im Beweis von Satz 1.7 konstruierten Formeln benutzen (für $n > 0$) nur die Junktoren \wedge, \vee, \neg . Also lassen sich aus diesen Funktionen (oder Junktoren) alle anderen Booleschen Funktionen kombinieren.

Definition 1.10. Eine Menge $\Omega \subseteq B$ von Booleschen Funktionen ist *funktional vollständig*, wenn sich daraus jede Boolesche Funktion $f \in B^n$ ($n \geq 1$) im Sinne von Satz 1.7 definieren lässt.

Wir wissen, dass neben $\{\wedge, \vee, \neg\}$ auch bereits $\{\wedge, \neg\}$ und $\{\vee, \neg\}$ funktional vollständig sind, denn es gilt:

$$\psi \wedge \varphi \equiv \neg(\neg\psi \vee \neg\varphi),$$

$$\psi \vee \varphi \equiv \neg(\neg\psi \wedge \neg\varphi).$$

Es gibt aber noch weitere funktional vollständige Mengen:

- (1) $\{\rightarrow, \neg\}$ ist funktional vollständig, da $\{\vee, \neg\}$ funktional vollständig ist und $\psi \vee \varphi \equiv \neg\psi \rightarrow \varphi$.
- (2) $\{\rightarrow, 0\}$ ist funktional vollständig. Dies folgt aus (1) und $\neg\psi \equiv \psi \rightarrow 0$.
- (3) Sei \oplus die Addition modulo 2 (das „exklusive oder“). Die Menge $\{\wedge, \oplus, 1\}$ ist funktional vollständig, da $\neg\psi \equiv 1 \oplus \psi$. Boolesche Funktionen entsprechen also genau den Polynomen über dem Körper \mathbb{F}_2 .
- (4) Sei $(u \mid v) := 0$, wenn $u = v = 1$ und $(u \mid v) := 1$ sonst, also $(\psi \mid \varphi) \equiv \neg(\psi \wedge \varphi)$. Dann ist $\{\mid\}$ funktional vollständig, da $\neg\psi \equiv \psi \mid \psi$ und $\psi \wedge \varphi \equiv \neg(\psi \mid \varphi) \equiv (\psi \mid \varphi) \mid (\psi \mid \varphi)$.
- (5) Hingegen ist $\{\wedge, \vee, \rightarrow\}$ nicht funktional vollständig, da für jede nur mit diesen Junktoren gebildete Formel $\psi(X_1, \dots, X_n)$ gilt, dass $\psi[1, \dots, 1] = 1$. Insbesondere kann mit $\wedge, \vee, \rightarrow$ keine zu $\neg X$ äquivalente Formel gebildet werden.

Für gewisse Zwecke, z.B. für Beweissysteme oder Schaltkreise, ist es durchaus zweckmäßig, Formeln aus anderen funktional vollständigen Mengen als $\{\wedge, \vee, \neg\}$ aufzubauen.

Übung 1.11. Die Funktion $\text{sel} \in B^3$ sei definiert durch $\text{sel}(u, v, w) = v$, wenn $u = 0$ und $\text{sel}(u, v, w) = w$, wenn $u = 1$. Zeigen Sie, dass $\{\text{sel}, 0, 1\}$ funktional vollständig ist.

Übung 1.12. Zeigen Sie, dass die Menge $\{\wedge, \vee, 0, 1\}$ funktional unvollständig ist, dass aber jede Erweiterung durch eine Funktion, welche nicht über $\{\wedge, \vee, 0, 1\}$ definierbar ist, funktional vollständig ist.

Übung 1.13. Eine Boolesche Funktion $f \in B^n$ ist *linear*, wenn sie durch ein lineares Polynom $f(X_1, \dots, X_n) = a_0 + a_1 X_1 + \dots + a_n X_n$ über dem Körper \mathbb{F}_2 beschrieben werden kann. Zeigen Sie, dass die meisten Booleschen Funktion nicht linear sind.

Übung 1.14. Die zu $f \in B^n$ *duale Funktion* $f^\delta \in B^n$ ist definiert durch $f^\delta(x_1, \dots, x_n) := \neg f(\neg x_1, \dots, \neg x_n)$.

- (a) Geben Sie die zu $\vee, \wedge, \rightarrow, \neg$ dualen Funktionen an.
 (b) Eine Funktion f ist *selbstdual*, wenn $f^\delta = f$. Sei T_k^n die n -stellige Boolesche Funktion mit

$$T_k^n(x_1, \dots, x_n) = 1 \text{ gdw. } |\{i : x_i = 1\}| \geq k.$$

Beschreiben Sie die zu T_k^n duale Funktion. Für welche n, k ist T_k^n selbstdual?

- (c) Zeigen Sie, dass die über der Junktorenmenge $\{\neg, T_2^3\}$ definierbaren Funktionen gerade die selbstdualen Funktionen sind.

1.3 Horn-Formeln

Eine in der Praxis sehr wichtige Klasse von Formeln sind Horn-Formeln (benannt nach dem Logiker Alfred Horn). Insbesondere ist das Erfüllbarkeitsproblem für Horn-Formeln durch einen einfachen und effizienten Algorithmus entscheidbar.

Definition 1.11. Eine (*aussagenlogische*) *Horn-Formel* ist eine Formel $\psi = \bigwedge_i \bigvee_j Y_{ij}$ in KNE, wobei jede Disjunktion $\bigvee_j Y_{ij}$ höchstens ein positives Literal enthält.

Horn-Formeln können auch als Konjunktionen von Implikationen geschrieben werden:

- (1) $\neg X_1 \vee \dots \vee \neg X_k \vee X \equiv X_1 \wedge \dots \wedge X_k \rightarrow X$;
 (2) $\neg X_1 \vee \dots \vee \neg X_k \equiv X_1 \wedge \dots \wedge X_k \rightarrow 0$.

Implikationen vom Typ (1) mit $k = 0$ werden in der Form $(1 \rightarrow X)$ geschrieben. Horn-Formeln, die keine solchen Implikationen enthalten, sind trivialerweise erfüllbar, indem man alle Aussagenvariablen mit 0 bewertet. Offensichtlich ist auch jede Horn-Formel erfüllbar, die keine Implikation der Form (2) enthält, z.B. indem man alle Aussagenvariablen mit 1 belegt.

Horn-Formeln können mit dem folgendem Markierungsalgorithmus in polynomialer Zeit auf Erfüllbarkeit getestet werden.

Algorithmus 1.1 Erfüllbarkeitstest für Horn-Formeln

Input: Eine aussagenlogische Hornformel $\psi = \bigwedge_i C_i$

$N := \emptyset$

$M := \{X \in \tau(\psi) : \psi \text{ enthält } C_i \text{ der Form } (1 \rightarrow X)\}$

while $N \neq M$ **do**

$N := M$

$M := M \cup \{X : \psi \text{ enthält } C_i \text{ der Form } (X_1 \wedge \dots \wedge X_k) \rightarrow X$

mit $\{X_1, \dots, X_k\} \subseteq M\}$

if $[\psi \text{ enthält } C_i \text{ der Form } (X_1 \wedge \dots \wedge X_k) \rightarrow 0$

mit $\{X_1, \dots, X_k\} \subseteq M]$ **then**

output „ ψ unerfüllbar“ **end**

end do

output „ ψ erfüllbar“, **output** M **end**

Die ausgegebene Menge M definiert eine Belegung \mathcal{J}_M mit $\mathcal{J}_M(X) = 1$ genau dann, wenn $X \in M$.

Satz 1.12. Der angegebene Erfüllbarkeitstest für Horn-Formeln ist korrekt. Wenn ψ erfüllbar ist, dann ist \mathcal{I}_M ein Modell von ψ . Für Formeln mit n Aussagenvariablen hält der Erfüllbarkeitstest nach höchstens $n + 1$ Iterationen der while-Schleife.

Beweis. Sei \mathcal{I} ein beliebiges Modell von ψ . Offensichtlich muss $\mathcal{I}(X) = 1$ sein für alle Aussagenvariablen X , welche im Laufe dieser Prozedur markiert werden (d.h. die zu M hinzugefügt werden). Weiter kann es keine Teilformel C_i der Form $X_1 \wedge \dots \wedge X_k \rightarrow 0$ mit $X_1, \dots, X_k \in M$ geben, da sonst $\llbracket \psi \rrbracket^{\mathcal{I}} = 0$. Also stellt der Algorithmus korrekt die Erfüllbarkeit von ψ fest.

Wenn der Algorithmus ausgibt, dass ψ erfüllbar ist, dann ist \mathcal{I}_M tatsächlich ein Modell von ψ , denn die schließlich erzeugte Menge M hat folgende Eigenschaften:

- Für alle Unterformeln $X_1 \wedge \dots \wedge X_k \rightarrow X$ gilt: Wenn $\{X_1, \dots, X_k\} \subseteq M$, dann ist $X \in M$ (sonst würde die while-Schleife noch nicht verlassen).
- Für alle Unterformeln $X_1 \wedge \dots \wedge X_k \rightarrow 0$ gilt: $\{X_1, \dots, X_k\} \not\subseteq M$ (sonst würde der Algorithmus die Unerfüllbarkeit feststellen).

Da in jedem Durchlauf der Schleife eine neue Aussagenvariable in M eingefügt wird, oder festgestellt wird, dass keine neuen mehr hinzugefügt werden müssen und die Schleife verlassen wird, folgt auch die letzte Behauptung. Q.E.D.

Bemerkung. Das durch den Markierungsalgorithmus gefundene Modell \mathcal{I}_M von ψ (falls es existiert) ist das *kleinste Modell* von ψ , d.h. für jedes andere Modell $\mathcal{I} \models \psi$ gilt: Wenn $\mathcal{I}_M(X) = 1$, dann auch $\mathcal{I}(X) = 1$.

Im Gegensatz zu DNF- oder KNF-Formeln ist die Klasse der Horn-Formeln *keine* Normalform.

Satz 1.13. Es gibt aussagenlogische Formeln, die nicht zu einer Horn-Formel äquivalent sind.

Beweis. Horn-Formeln sind entweder unerfüllbar oder haben ein kleinstes Modell. Dies trifft z.B. nicht auf die Formel $X \vee Y$ zu. Q.E.D.

1.4 Der Kompaktheitssatz der Aussagenlogik

In vielen Anwendungen der Aussagenlogik hat man Erfüllbarkeit und Folgerungsbeziehungen für *unendliche* Formelmengen zu untersuchen. Ein grundlegender Satz, der Kompaktheits- oder Endlichkeitsatz, erleichtert diese Aufgabe, indem er sie auf die Untersuchung *endlicher* Teilmengen zurückführt.

Bevor wir ihn formulieren, erläutern wir die *Folgerungsbeziehung* zwischen Formelmengen und Formeln, einer der wichtigsten Begriffe in der Logik überhaupt, nicht nur für die Aussagenlogik sondern insbesondere für ausdrucksstärke Logiken und deren Anwendungen.

Definition 1.14 (Semantische Folgerungsbeziehung). Ein Modell einer Formelmenge $\Phi \subseteq \text{AL}$ ist eine Interpretation \mathcal{I} , so dass $\llbracket \varphi \rrbracket^{\mathcal{I}} = 1$ für alle $\varphi \in \Phi$. Wir sagen, dass ψ aus Φ folgt (kurz: $\Phi \models \psi$), wenn jede zu $\Phi \cup \{\psi\}$ passende Interpretation, welche Modell von Φ ist, auch Modell von ψ ist. Wenn $\Phi = \{\varphi\}$, schreiben wir auch $\varphi \models \psi$ anstelle von $\{\varphi\} \models \psi$.

Wenn $\Phi \models \psi$, dann legt die durch Φ festgelegte (axiomatisierte) Information bereits fest, dass auch ψ gilt, unabhängig von Variationen zwischen verschiedenen Modellen von Φ .

Man beachte, dass dasselbe Symbol \models sowohl für die Modellbeziehung ($\mathcal{I} \models \psi$, bzw. $\mathcal{I} \models \Phi$) als auch für die Folgerungsbeziehung ($\Phi \models \psi$) verwendet wird. Missverständnisse sind ausgeschlossen, da die linke Seite die Bedeutung festlegt.

Übung 1.15 (Beispiele und elementare Eigenschaften der Folgerungsbeziehung). Verifizieren Sie die folgenden Aussagen:

- $\{\psi, \varphi\} \models \psi \wedge \varphi$,
 $\{\psi, \psi \rightarrow \varphi\} \models \varphi$.
- Wenn $\Phi \cup \{\psi\} \models \varphi$ und $\Phi \cup \{\neg\psi\} \models \varphi$, dann gilt bereits $\Phi \models \varphi$.
- $\Phi \cup \{\psi\} \models \varphi$ genau dann, wenn $\Phi \models (\psi \rightarrow \varphi)$.
- ψ ist genau dann eine Tautologie, wenn ψ aus der leeren Menge folgt. (Dies rechtfertigt die Notation $\models \psi$ als abgekürzte Schreibweise für $\emptyset \models \psi$.)

- (e) Es gilt $\Phi \models \varphi$ für jedes $\varphi \in \Phi$.
- (f) Wenn $\Phi \models \psi$, dann gilt auch $\Phi' \models \psi$ für alle Obermengen $\Phi' \supseteq \Phi$.
- (g) ψ und φ sind genau dann äquivalent, wenn $\psi \models \varphi$ und $\varphi \models \psi$.
- (h) $\Phi \models \psi$ gilt genau dann, wenn $\Phi \cup \{\neg\psi\}$ unerfüllbar ist.
- (i) Wenn $\Phi \models \psi$ und $\Phi \models \neg\psi$, dann ist Φ unerfüllbar. Umgekehrt gilt für unerfüllbare Formelmengen Φ , dass $\Phi \models \psi$ für alle $\psi \in \text{AL}$.

Satz 1.15 (Kompaktheits- oder Endlichkeitssatz). Sei $\Phi \subseteq \text{AL}$, $\psi \in \text{AL}$.

- (i) Φ ist erfüllbar genau dann, wenn jede endliche Teilmenge von Φ erfüllbar ist.
- (ii) $\Phi \models \psi$ genau dann, wenn eine endliche Teilmenge $\Phi_0 \subseteq \Phi$ existiert, so dass $\Phi_0 \models \psi$.

Wir lassen hier Formelmengen beliebiger Mächtigkeit zu und verwenden im Beweis das *Lemma von Zorn*, ein fundamentales Beweisprinzip in der Mathematik. Wenn man nur abzählbare Formelmengen Φ (und daher auch nur abzählbare Mengen von Aussagenvariablen) zulässt, dann könnte man den Beweis induktiv und ohne das Lemma von Zorn (aber nicht wirklich einfacher) führen.

Lemma 1.16 (Zorn). Sei $(A, <)$ eine nicht-leere partielle Ordnung, in der jede Kette nach oben beschränkt ist. Dann besitzt $(A, <)$ ein maximales Element.

Im Fall den wir hier betrachten, wird A ein bestimmtes System von Formelmengen (also eine Menge von Mengen) sein, welches durch die Inklusionsbeziehung \subseteq partiell geordnet ist. Eine Kette ist dann also eine Teilmenge B von A , so dass für alle $X, Y \in B$ entweder $X \subseteq Y$ oder $Y \subseteq X$ gilt. Die Voraussetzung, dass eine solche Kette B nach oben beschränkt sei, bedeutet, dass in A eine Menge S_B existiert, so dass $Y \subseteq S_B$ für alle $Y \in B$. Wenn diese Voraussetzung für alle Ketten B nachgewiesen werden kann, dann gibt es nach dem Lemma von Zorn ein maximales Element für ganz A , welches uns dann unmittelbar das gewünschte Modell liefern wird. Nach diesen vorbereitenden Bemerkungen können wir nun den Kompaktheitssatz beweisen.

Beweis (Beweis des Kompaktheitssatzes). Wir zeigen zunächst, dass (ii) aus (i) folgt: Falls $\Phi_0 \models \psi$ für $\Phi_0 \subseteq \Phi$, dann gilt offensichtlich auch

$\Phi \models \psi$. Es gelte umgekehrt $\Phi \models \psi$. Beweis durch Widerspruch: Zu jedem endlichen $\Phi_0 \subseteq \Phi$ gibt es ein $\mathcal{J} : \tau \rightarrow \{0, 1\}$ mit $\mathcal{J} \models \Phi_0$ aber $\llbracket \psi \rrbracket^{\mathcal{J}} = 0$. Dies bedeutet, dass $\Phi_0 \cup \{\neg\psi\}$ für jedes endliche $\Phi_0 \subseteq \Phi$ erfüllbar ist. Also ist jede endliche Teilmenge von $\Phi \cup \{\neg\psi\}$ erfüllbar und damit, nach (i), auch $\Phi \cup \{\neg\psi\}$ selbst. Dies ist aber ein Widerspruch zu $\Phi \models \psi$.

Es bleibt (i) zu zeigen. Es ist klar, dass mit Φ auch jede endliche Teilmenge von Φ erfüllbar ist. Für die Umkehrung nehmen wir an, dass jede endliche Teilmenge $\Phi_0 \subseteq \Phi$ erfüllbar ist und setzen

$$A := \{\Psi : \Psi \supseteq \Phi \text{ und jede endl. Teilmenge von } \Psi \text{ ist erfüllbar}\}.$$

A ist partiell geordnet durch die Inklusionsbeziehung und nicht leer (da $\Phi \in A$).

Wir zeigen zuerst, dass die Voraussetzung des Zornschen Lemmas erfüllt ist. Sei $K \subseteq A$ eine Kette, d.h. es gilt $\Theta \subseteq \Psi$ oder $\Psi \subseteq \Theta$ für alle $\Psi, \Theta \in K$. Offensichtlich ist $\Gamma := \bigcup K$, die Vereinigung aller Mengen aus K , eine obere Schranke für K . Zu zeigen ist, dass Γ selbst in A enthalten ist, d.h. dass jede endliche Teilmenge $\Gamma_0 \subseteq \Gamma$ erfüllbar ist. Jede Formel $\gamma \in \Gamma_0$ ist in einer Menge $\Psi(\gamma) \in K$ enthalten. Da K eine Kette ist, gibt es unter den endlich vielen Mengen $\Psi(\gamma)$ (für $\gamma \in \Gamma_0$) eine maximale, welche ganz Γ_0 enthält. Jede endliche Teilmenge dieser Menge ist erfüllbar, insbesondere also Γ_0 .

Nach dem Lemma von Zorn hat demnach A ein maximales Element Φ_{\max} . Wir behaupten, dass für jede Formel ψ entweder $\psi \in \Phi_{\max}$ oder $\neg\psi \in \Phi_{\max}$. Andernfalls betrachten wir die Erweiterungen $\Phi_{\max} \cup \{\psi\}$ und $\Phi_{\max} \cup \{\neg\psi\}$. Aufgrund der Maximalität von Φ_{\max} gehört keine dieser Mengen zu A . Also gibt es endliche Teilmengen $\Psi_0, \Psi_1 \subseteq \Phi_{\max}$, so dass $\Psi_0 \cup \{\psi\}$ und $\Psi_1 \cup \{\neg\psi\}$ unerfüllbar sind. Aber dann ist $\Psi_0 \cup \Psi_1$ eine endliche unerfüllbare Teilmenge von Φ_{\max} , im Widerspruch zu $\Phi_{\max} \in A$. Wir definieren nun eine Interpretation \mathcal{J} durch die Vorschrift

$$\mathcal{J}(X) = 1 \text{ gdw. } X \in \Phi_{\max}.$$

Per Induktion über den Formelaufbau zeigen wir, dass $\mathcal{J} \models \psi$ genau dann, wenn $\psi \in \Phi_{\max}$:

- Für atomare ψ folgt dies unmittelbar aus der Definition.
- Sei $\psi = \neg\varphi$. Dann ist nach Induktionsvoraussetzung und nach der soeben gezeigten Eigenschaft von Φ_{\max}

$$\mathcal{J} \models \psi \text{ gdw. } \mathcal{J} \not\models \varphi \text{ gdw. } \varphi \notin \Phi_{\max} \text{ gdw. } \psi \in \Phi_{\max}.$$

- Sei $\psi = \varphi \wedge \vartheta$. Nach Induktionsvoraussetzung folgt, dass genau dann $\mathcal{J} \models \psi$ gilt, wenn $\varphi, \vartheta \in \Phi_{\max}$. Aber das ist genau dann der Fall, wenn auch $\psi \in \Phi_{\max}$.

Wenn nämlich $\psi \notin \Phi_{\max}$, dann $\neg\psi \in \Phi_{\max}$ was unmöglich ist, da Φ_{\max} dann mit $\{\varphi, \vartheta, \neg(\varphi \wedge \vartheta)\}$ eine unerfüllbare endliche Teilmenge enthalten würde. Wenn aber $\psi \in \Phi_{\max}$, dann müssen auch φ und ϑ in Φ_{\max} liegen, da sonst Φ_{\max} mit $\{\varphi \wedge \vartheta, \neg\varphi\}$ oder $\{\varphi \wedge \vartheta, \neg\vartheta\}$ wieder eine endliche unerfüllbare Teilmenge enthielte.

- Die Argumentation in allen andern Fällen ist analog. (Es wird empfohlen, zur Übung mindestens einen dieser Fälle, z.B. für Formeln $(\varphi \rightarrow \vartheta)$ selbst nachzuvollziehen.)

Also ist \mathcal{J} ein Modell von Φ_{\max} und damit auch von Φ . Q.E.D.

DAS LEMMA VON KÖNIG. Ein Baum mit Wurzel w ist ein zusammenhängender, zyklischer, gerichteter Graph $T = (V, E)$ mit einem ausgezeichneten Knoten $w \in V$, so dass keine Kante in w endet (d.h. $(v, w) \notin E$ für alle $v \in V$) und in jedem andern Knoten genau eine Kante endet. Ein solcher Baum heißt *endlich verzweigt*, wenn von jedem $v \in V$ nur endlich viele Kanten ausgehen. Als Anwendung des Kompaktheitssatzes beweisen wir das folgende Lemma.

Lemma 1.17 (König). Sei T ein endlich verzweigter Baum mit Wurzel w , in dem es beliebig lange endliche Wege gibt. Dann gibt es auch einen unendlichen Weg in T (der bei der Wurzel w beginnt).

Beweis. Für den gegebenen Baum $T = (V, E)$ mit Wurzel w und $n \in \mathbb{N}$ sei

$$S_n = \{v \in V : \text{es gibt einen Weg der Länge } n \text{ von } w \text{ nach } v\}.$$

Alle S_n sind endlich, da der Baum endlich verzweigt ist. Weiter ist $S_0 = \{w\}$ und alle S_n nicht leer, da es beliebig lange Wege in T gibt.

Ein unendlicher, von w ausgehender Weg ist eine Menge $W \subseteq V$, welche folgende Bedingungen erfüllt:

- $|W \cap S_n| = 1$ für alle n ;
- Wenn $v \in W$ und $(u, v) \in E$, dann ist auch $u \in W$.

Zu zeigen ist die Existenz einer solchen Menge W . Dazu ordnen wir jedem $v \in V$ eine Aussagenvariable X_v zu und setzen:

$$\alpha_n := \bigvee_{v \in S_n} X_v,$$

$$\beta_n := \bigwedge_{u, v \in S_n, u \neq v} \neg(X_u \wedge X_v),$$

$$\Phi := \{\alpha_n : n \in \mathbb{N}\} \cup \{\beta_n : n \in \mathbb{N}\} \cup \{(X_v \rightarrow X_u) : (u, v) \in E\}.$$

Jede endliche Teilmenge $\Phi_0 \subseteq \Phi$ ist erfüllbar. Um dies einzusehen, nehmen wir das größte $n \in \mathbb{N}$ mit $\alpha_n \in \Phi_0$ oder $\beta_n \in \Phi_0$. Dann wählen wir ein $z \in S_n$ und den von w nach z führenden Weg $W(w, z)$. Sei

$$\mathcal{J}(X_v) := \begin{cases} 1 & v \in W(w, z) \\ 0 & \text{sonst.} \end{cases}$$

Offensichtlich ist \mathcal{J} Modell von Φ_0 . Mit dem Kompaktheitssatz folgt, dass es ein Modell \mathcal{J} für Φ gibt. Setze $W := \{v \in V : \mathcal{J}(X_v) = 1\}$. Es folgt, dass W einen unendlichen Weg von w aus definiert:

- Da $\alpha_n, \beta_n \in \Phi$, gibt es genau ein v in $W \cap S_n$.
- Sei $v \in W$ und $(u, v) \in E$. Da $\mathcal{J} \models X_v$ und $\mathcal{J} \models X_v \rightarrow X_u$ gilt auch $\mathcal{J} \models X_u$, also $u \in W$. Q.E.D.

Bemerkung. Man beachte, dass das Lemma von König nicht trivial ist. Es gilt z.B. nicht für Bäume mit unendlichen Verzweigungen. Man betrachte etwa den Baum in Abbildung 1.1. In diesem Baum gibt es für jedes n , ausgehend von w , einen Weg der Länge n , aber es gibt keinen unendlichen Weg.

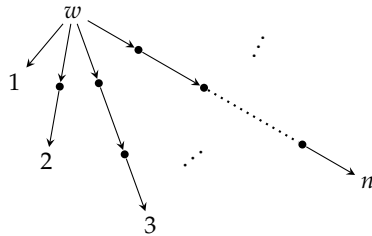


Abbildung 1.1. Ein unendlicher Baum ohne unendlichen Weg

Übung 1.16. Ein *Dominosystem* sei eine endliche Menge von quadratischen Dominosteinen gleicher Größe, deren vier Kanten (oben, unten, links, rechts) gefärbt sind. Eine *Parkettierung* der Ebene (oder eines Teils davon) ist eine vollständige Überdeckung mit Dominosteinen, ohne Lücken und Überlappungen, so dass aneinandergrenzende Kanten dieselbe Farbe tragen. (Rotation der Steine ist nicht erlaubt.) Zeigen Sie mit Hilfe des Lemmas von König, dass für jedes Dominosystem folgendes gilt: Wenn beliebig große endliche Quadrate parkettiert werden können, dann auch die ganze Ebene.

Übung 1.17. Eine Formelmeng $\Phi \subseteq \text{AL}$ ist *endlich axiomatisierbar*, wenn eine endliche Formelmeng $\Phi_0 \subseteq \text{AL}$ existiert, welche die gleichen Modelle hat wie Φ . Sei $\Phi = \{\varphi_n : n \in \mathbb{N}\}$ eine Formelmeng, so dass für alle $n \in \mathbb{N}$ gilt: $\varphi_{n+1} \models \varphi_n$, aber $\varphi_n \not\models \varphi_{n+1}$. Zeigen Sie, dass Φ nicht endlich axiomatisierbar ist.

Übung 1.18. Ein ungerichteter Graph $G = (V, E)$ heißt *k-färbbar*, wenn es eine Funktion $f : V \rightarrow \{1, \dots, k\}$ gibt, so dass $f(p) \neq f(q)$ für alle Kanten $(p, q) \in E$. Zeigen Sie, dass ein ungerichteter Graph G *k-färbbar* ist, wenn jeder endliche Untergraph von G *k-färbbar* ist.

Hinweis: Konstruieren Sie zu jedem endlichen Untergraphen von G eine aussagenlogische Formel, die genau dann erfüllbar ist, wenn der Untergraph *k-färbbar* ist. Führen Sie dazu zu jedem Knoten $g \in V$ und jeder Farbe i mit $1 \leq i \leq k$ eine Aussagenvariable $X_{g,i}$ ein, die besagt, dass der Knoten g die Farbe i hat.

Übung 1.19. Sei $A \subseteq \{0, 1\}^*$ eine unendliche Menge von Wörtern.

Zeigen Sie, dass es eine unendliche Folge w_0, w_1, w_2, \dots gibt, so dass jedes w_i ein Anfangsstück von w_{i+1} und von mindestens einem Wort aus A ist.

Übung 1.20 (Definierbarkeitstheorem). Sei $\Phi \subseteq \text{AL}$ eine Formelmeng, $X \in \tau(\Phi)$ eine Aussagenvariable. X heißt *explizit definierbar* in Φ , wenn eine Formel $\varphi \in \text{AL}$ existiert, welche X nicht enthält, so dass $\Phi \models X \leftrightarrow \varphi$. (In Modellen von Φ ist also der Wahrheitswert von X durch eine Formel, die nicht von X abhängt, explizit festgelegt). Demgegenüber heißt X *implizit definierbar* in Φ , wenn für alle Modelle $\mathcal{I}, \mathcal{I}'$ von Φ gilt: Wenn $\mathcal{I}(Z) = \mathcal{I}'(Z)$ für alle Aussagenvariablen $Z \neq X$, dann auch $\mathcal{I}(X) = \mathcal{I}'(X)$. (In Modellen von Φ ist also der Wahrheitswert von X durch die Wahrheitswerte der andern Variablen implizit festgelegt).

Beweisen Sie das *aussagenlogische Definierbarkeitstheorem*: Wenn X implizit in Φ definierbar ist, dann ist X auch explizit in Φ definierbar.

Hinweis: Die Formelmeng Φ' entstehe dadurch, dass man X in allen Formeln von Φ durch eine neue Aussagenvariable $X' \notin \tau(\Phi)$ ersetzt. Die implizite Definierbarkeit von X in Φ besagt dann, dass $\Phi \cup \Phi' \models X \leftrightarrow X'$. Benutzen Sie den Kompaktheitssatz, um Φ durch eine endliche Formelmeng zu ersetzen, und verwenden Sie das aussagenlogische Interpolationstheorem (Übung 1.6), um eine explizite Definition von X in Φ zu konstruieren.

1.5 Aussagenlogische Resolution

Resolution ist ein syntaktisches Verfahren, um die *Unerfüllbarkeit* von Formeln in KNF nachzuweisen. Es ist dabei nützlich, Formeln in KNF als Mengen von *Klauseln* darzustellen.

Definition 1.18. Eine *Klausel* ist eine endliche Menge von Literalen. Mit \square bezeichnet man die leere Klausel. Einer Formel $\psi = \bigwedge_{i=1}^n \bigvee_{j=1}^{m_i} Y_{ij}$ in KNF wird eine endliche *Klauselmeng* $K(\psi)$ wie folgt zugeordnet: Jeder Disjunktion $\bigvee_{j=1}^{m_i} Y_{ij}$ ordnet man die Klausel $C_i = \{Y_{ij} : j = 1, \dots, m_i\}$ zu und setzt $K(\psi) := \{C_1, \dots, C_n\}$.

Bemerkung. Die Mengennotation ergibt gewisse Vereinfachungen: Ele-

mente einer Menge haben keine Reihenfolge und keine Multiplizität. Daher gilt:

- Formeln, die sich nur durch Reihenfolge der auftretenden Teilformeln unterscheiden, ergeben dieselbe Klauselmenge.
- Mehrfach auftretende Literale in Disjunktionen, bzw. mehrfach auftretende Klauseln verschmelzen zu einem einzigen Element der Klauseln bzw. Klauselmengen.

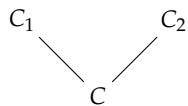
Beispiel. Die Formeln $(X_1 \vee \neg X_2) \wedge X_3$, $(X_1 \vee X_1 \vee \neg X_2) \wedge (X_3 \vee X_3) \wedge X_3$ und $X_3 \wedge (X_1 \vee \neg X_2) \wedge (\neg X_2 \vee X_1)$ haben alle dieselbe Klauselmenge $K = \{\{X_1, \neg X_2\}, \{X_3\}\}$.

Umgekehrt entspricht einer Klausel C die Formel $\bigvee_{Y \in C} Y$. Einer endlichen Klauselmenge K entspricht die Formel $\bigwedge_{C \in K} \bigvee_{Y \in C} Y$.

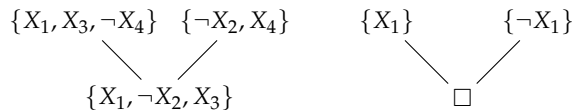
Wir können also Klauseln und Klauselmengen wie Formeln und Formelmengen behandeln und benutzen Begriffe wie Erfüllbarkeit und Äquivalenz entsprechend. Insbesondere ist eine Klauselmenge erfüllbar, wenn es eine Interpretation \mathfrak{J} gibt, so dass jede Klausel $C \in K$ ein Literal Y enthält mit $\llbracket Y \rrbracket^{\mathfrak{J}} = 1$. Beachte:

- Die leere Klauselmenge ist erfüllbar.
- Wenn $\square \in K$, dann ist K unerfüllbar.

Definition 1.19. Seien C, C_1, C_2 Klauseln. C ist *Resolvente* von C_1 und C_2 genau dann, wenn es ein Literal Y gibt mit $Y \in C_1, \bar{Y} \in C_2$ und $C = (C_1 \setminus \{Y\}) \cup (C_2 \setminus \{\bar{Y}\})$. Dies wird folgendermaßen notiert:



Beispiel.



Lemma 1.20 (Resolutionslemma). Sei K eine Klauselmenge, $C_1, C_2 \in K$ und C Resolvente von C_1 und C_2 . Dann sind K und $K \cup \{C\}$ äquivalent.

Beweis. Wenn $\llbracket K \cup \{C\} \rrbracket^{\mathfrak{J}} = 1$, dann offensichtlich erst recht $\llbracket K \rrbracket^{\mathfrak{J}} = 1$. Sei umgekehrt $\llbracket K \rrbracket^{\mathfrak{J}} = 1$ und $C = (C_1 \setminus \{Y\}) \cup (C_2 \setminus \{\bar{Y}\})$.

- Wenn $\llbracket Y \rrbracket^{\mathfrak{J}} = 1$, dann ist $\llbracket C_2 \setminus \{\bar{Y}\} \rrbracket^{\mathfrak{J}} = 1$, da sonst $\llbracket C_2 \rrbracket^{\mathfrak{J}} = 0$. Also ist $\llbracket C \rrbracket^{\mathfrak{J}} = 1$.
- Wenn $\llbracket Y \rrbracket^{\mathfrak{J}} = 0$, dann ist $\llbracket C_1 \setminus \{Y\} \rrbracket^{\mathfrak{J}} = 1$ und also wiederum $\llbracket C \rrbracket^{\mathfrak{J}} = 1$.

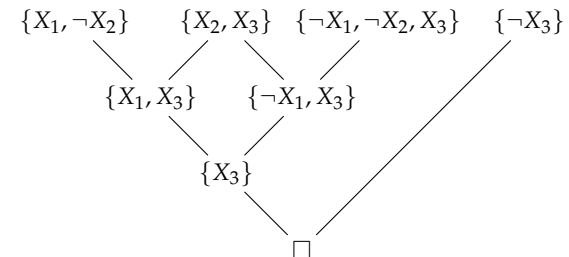
Also ist $\llbracket K \cup \{C\} \rrbracket^{\mathfrak{J}} = 1$.

Q.E.D.

Definition 1.21. Für jede Klauselmenge K sei

- $\text{Res}(K) := K \cup \{C : C \text{ Resolvente zweier Klauseln aus } K\}$.
- $\text{Res}^0(K) := K$, $\text{Res}^{n+1}(K) := \text{Res}(\text{Res}^n(K))$ für $n \in \mathbb{N}$.
- $\text{Res}^*(K) := \bigcup_{n \in \mathbb{N}} \text{Res}^n(K)$.

Beispiel. Sei $\psi = (X_1 \vee \neg X_2) \wedge \neg X_3 \wedge (\neg X_1 \vee \neg X_2 \vee X_3) \wedge (X_2 \vee X_3)$. Dann ist $K(\psi) = \{\{X_1, \neg X_2\}, \{\neg X_3\}, \{\neg X_1, \neg X_2, X_3\}, \{X_2, X_3\}\}$. Die leere Klausel ist wie folgt aus $K(\psi)$ ableitbar:



KORREKTHEIT UND VOLLSTÄNDIGKEIT. Ein Beweiskalkül ist *korrekt*, wenn keine falschen Aussagen darin ableitbar sind, und *vollständig*, wenn alle wahren Aussagen ableitbar sind. Der Resolutionskalkül ist ein Verfahren, um die *Unerfüllbarkeit* einer Klauselmenge K nachzuweisen, indem durch wiederholte Anwendung des Operators Res die leere Klausel abgeleitet wird. Die Korrektheit und Vollständigkeit des Resolutionskalküls wird durch den Resolutionssatz ausgedrückt.

Satz 1.22 (Resolutionssatz). Eine Klauselmenge K ist genau dann unerfüllbar, wenn $\square \in \text{Res}^*(K)$.

Beweis. (Korrektheit) Aus dem Resolutionslemma folgt $K \equiv \text{Res}(K)$ und damit per Induktion $K \equiv \text{Res}^*(K)$. Wenn also $\square \in \text{Res}^*(K)$, dann ist $\text{Res}^*(K)$ und damit auch K unerfüllbar.

(Vollständigkeit) Sei K unerfüllbar. Nach dem Kompaktheitsatz gibt es eine endliche unerfüllbare Teilmenge $K_0 \subseteq K$. Dann gibt es ein $n \in \mathbb{N}$, so dass K_0 höchstens die Aussagenvariablen X_0, \dots, X_{n-1} enthält. Wir zeigen per Induktion nach n , dass $\square \in \text{Res}^*(K_0) \subseteq \text{Res}^*(K)$.

Sei $n = 0$. Es gibt nur zwei Klauselmengen ohne Aussagenvariablen, nämlich \emptyset und $\{\square\}$. Da die leere Klauselmenge erfüllbar ist, muss $K_0 = \{\square\}$ sein. Für den Induktionsschluss nehmen wir an, dass alle Aussagenvariablen von K_0 in $\{X_0, \dots, X_n\}$ enthalten seien. Wir konstruieren zwei Klauselmengen K_0^+ und K_0^- , in denen X_n nicht vorkommt:

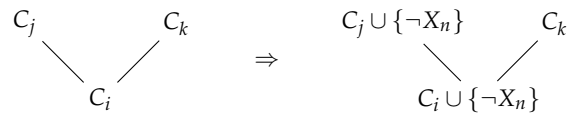
$$K_0^+ := \{C \setminus \{\neg X_n\} : X_n \notin C, C \in K_0\},$$

$$K_0^- := \{C \setminus \{X_n\} : \neg X_n \notin C, C \in K_0\}$$

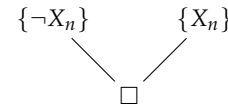
(d.h. wir streichen aus K_0 alle Klauseln, in denen X_n bzw. $\neg X_n$ vorkommt und streichen $\neg X_n$ bzw. X_n aus allen verbleibenden Klauseln).

K_0^+ und K_0^- sind unerfüllbar. Andernfalls gäbe es etwa eine Interpretation $\mathcal{J} : \{X_0, \dots, X_{n-1}\} \rightarrow \{0, 1\}$, so dass $\llbracket K_0^+ \rrbracket^{\mathcal{J}} = 1$. Erweitere \mathcal{J} durch $\mathcal{J}(X_n) = 1$. Es gilt dann $\llbracket K_0 \rrbracket^{\mathcal{J}} = 1$ im Widerspruch zur Unerfüllbarkeit von K_0 .

Aus der Induktionsvoraussetzung folgt, dass $\square \in \text{Res}^*(K_0^+)$ und $\square \in \text{Res}^*(K_0^-)$. Also gibt es Klauseln C_1, C_2, \dots, C_m , so dass $C_m = \square$, und für $i = 1, \dots, m$ gilt $C_i \in K_0^+$ oder C_i ist Resolvente von C_j, C_k für $j, k < i$. Einige der Klauseln C_i können aus Klauseln in K_0 durch Streichen von $\neg X_n$ entstanden sein. Wenn nicht, dann sind C_1, \dots, C_m auch in $\text{Res}^*(K_0)$, also $\square \in \text{Res}^*(K_0)$. Wenn ja, erhalten wir durch Wiedereinfügen von $\neg X_n$ eine Folge von Klauseln C'_1, \dots, C'_m , welche beweist, dass $\{\neg X_n\} \in \text{Res}^*(K_0)$.



Analog folgt aus $\square \in \text{Res}^*(K_0^-)$, dass entweder $\square \in \text{Res}^*(K_0)$ oder $\{X_n\} \in \text{Res}^*(K_0)$. Mit



folgt, dass $\square \in \text{Res}^*(K_0)$.

Q.E.D.

Wenn K nur die Aussagenvariablen X_0, \dots, X_{n-1} enthält, dann gilt dies auch für $\text{Res}^*(K)$, denn eine Resolvente zweier Klauseln C, C' enthält nur Literale, die bereits in C oder C' enthalten sind. Insbesondere folgt, dass die Kette

$$K = \text{Res}^0(K) \subseteq \text{Res}^1(K) \subseteq \dots \subseteq \text{Res}^m(K) \subseteq \dots$$

nach höchstens 2^{2^n} Schritten abbricht, d.h. $\text{Res}^*(K) = \text{Res}^{2^{2^n}}(K)$, denn es gibt nur 2^{2^n} verschiedene Klauseln mit Literalen $X_0, \dots, X_{n-1}, \neg X_0, \dots, \neg X_{n-1}$.

Für endliche Klauselmengen K erhält man also folgenden Algorithmus um zu entscheiden, ob K erfüllbar ist:

Algorithmus 1.2 Erfüllbarkeitstest mit Resolution

Input K (endliche Klauselmenge)

$R := \emptyset, S := K$

while $R \neq S$ **do**

$R := S$

$S := \text{Res}(R)$

if $\square \in S$ **then**

output „ K unerfüllbar“

end do

output „ K erfüllbar“ **end**

Dieser Algorithmus hat (im *worst case*) exponentielle Komplexität. Es ist auch nicht zu erwarten, dass es einen effizienten (in polyno-

mialer Zeit laufenden) Algorithmus für dieses Problem gibt, denn das Erfüllbarkeitsproblem für KNF-Formeln ist NP-vollständig.

Die Erfüllbarkeit einer Formel ist durch eine Existenzaussage ausgedrückt (es gibt ein Modell). Die Unerfüllbarkeit (oder die Allgemeingültigkeit) einer Formel ist eine Aussage über alle möglichen Interpretationen, ihrer Natur nach also eine universelle Aussage. Der Resolutionskalkül (wie jeder korrekte und vollständige Beweiskalkül) erlaubt nun, solche universellen Aussagen durch äquivalente Existenzaussagen auszudrücken: ψ ist unerfüllbar, wenn eine Deduktion der leeren Klausel existiert.

Man beachte aber folgende Asymmetrie: Das Aufschreiben eines Modells für ψ (also eines „Zeugen“ für die Erfüllbarkeit) ist mit viel weniger Aufwand verbunden als (im worst case) das Aufschreiben eines Resolutionsbeweises (also eines „Zeugen“ für die Unerfüllbarkeit). Dies hängt mit einem der wichtigsten Probleme der Komplexitätstheorie zusammen, dem Problem ob $NP = coNP$.

Für unendliche Klauselmengen kann es durchaus passieren, dass $Res(K) \setminus K$ unendlich ist oder dass die Kette

$$K = Res^0(K) \subset Res^1(K) \subset \dots \subset Res^n(K) \subset \dots$$

nicht stationär wird (auch wenn K erfüllbar ist).

Beispiel. Sei $K = \{\{X_0\}\} \cup \{\{\neg X_n, X_{n+1}\} : n \in \mathbb{N}\}$. Dann ist $X_{n+1} \in Res^{n+1}(K) \setminus Res^n(K)$ für jedes $n \in \mathbb{N}$.

EINHEITSRESOLUTION FÜR HORN-FORMELN. Die einer Horn-Formel ψ zugeordnete Klauselmenge $K(\psi)$ enthält nur Klauseln der Form $\{\neg X_1, \dots, \neg X_k\}$ (nur negative Literale) oder $\{\neg X_1, \dots, \neg X_k, X\}$ (ein positives Literal). Solche Klauseln heißen *Horn-Klauseln*. Für $k = 0$ ergibt sich, dass die leere Klausel \square und die Klauseln $\{X\}$, welche aus einer einzigen Aussagenvariablen bestehen, auch Horn-Klauseln sind. Wir präsentieren nun eine eingeschränkte Variante des Resolutionskalküls, welche vollständig für Horn-Formeln ist.

Definition 1.23. Seien C, C_1, C_2 Klauseln. C ist *Einheitsresolvente* von C_1

und C_2 , wenn C Resolvente von C_1 und C_2 ist und entweder $|C_1| = 1$ oder $|C_2| = 1$.

Bei der Einheitsresolution besteht also mindestens eine der Ausgangsklauseln nur aus einem einzigen Literal.

Satz 1.24 (Vollständigkeit der Einheitsresolution für Horn-Formeln). Eine aussagenlogische Horn-Formel ψ ist genau dann unerfüllbar, wenn \square durch Einheitsresolution aus $K(\psi)$ ableitbar ist.

Beweis. Es ist klar, dass ψ unerfüllbar ist, wenn \square aus $K(\psi)$ durch Einheitsresolution (also insbesondere durch Resolution) ableitbar ist.

Für die Umkehrung betrachten wir den Erfüllbarkeitstest für Horn-Formeln. Setze:

$$\begin{aligned} M^0 &:= \{X : K(\psi) \text{ enthält die Klausel } \{X\}\}, \\ M^{i+1} &:= M^i \cup \{X : \text{es gibt } X_1, \dots, X_k \in M^i, \text{ so dass } K(\psi) \\ &\quad \text{die Klausel } \{\neg X_1, \dots, \neg X_k, X\} \text{ enthält}\}, \\ M^* &:= \bigcup_{i \in \mathbb{N}} M^i. \end{aligned}$$

Die Korrektheit des Erfüllbarkeitstests (Satz 1.12) ergibt: ψ ist unerfüllbar genau dann, wenn $X_1, \dots, X_k \in M^*$ existieren, so dass $\{\neg X_1, \dots, \neg X_k\} \in K(\psi)$. Wir zeigen: Wenn $X \in M^*$, dann ist $\{X\}$ per Einheitsresolution aus $K(\psi)$ ableitbar.

Für $X \in M^0$ ist dies klar. Wenn $X \in M^{i+1}$, dann ist entweder $X \in M^i$ (dann greift die Induktionsvoraussetzung) oder es gibt $X_1, \dots, X_k \in M^i$, so dass $\{\neg X_1, \dots, \neg X_k, X\} \in K(\psi)$. Nach Induktionsvoraussetzung lassen sich die Klauseln $\{X_1\}, \dots, \{X_k\}$ aus $K(\psi)$ per Einheitsresolution ableiten. Unter Zuhilfenahme der Klausel $\{\neg X_1, \dots, \neg X_k, X\}$ lässt sich dann auch $\{X\}$ per Einheitsresolution aus $K(\psi)$ ableiten.

Wenn ψ unerfüllbar ist, dann gibt es also $\{\neg X_1, \dots, \neg X_k\} \in K(\psi)$, so dass die Einerklauseln $\{X_1\}, \dots, \{X_k\}$ per Einheitsresolution aus $K(\psi)$ ableitbar sind. Damit folgt nun sofort, dass \square per Einheitsresolution aus $K(\psi)$ abgeleitet werden kann. Q.E.D.

1.6 Der aussagenlogische Sequenzenkalkül

Wir beschreiben durch *Axiome* und *Schlussregeln* einen im wesentlichen auf Gentzen zurückgehenden Beweiskalkül SK, den *Sequenzenkalkül*. Dieser Kalkül operiert auf Paaren von endlichen Formelmengen, welche wir *Sequenzen* nennen. Im Folgenden bezeichnen Γ, Δ endliche Mengen aussagenlogischer Formeln. Wir schreiben Γ, Δ für $\Gamma \cup \Delta$ und Γ, ψ für $\Gamma \cup \{\psi\}$. Die Ausdrücke $\wedge \Gamma$ bzw. $\vee \Gamma$ stehen für die Konjunktion bzw. Disjunktion über alle Formeln in Γ .

Definition 1.25. Eine *Sequenz* ist ein Ausdruck der Form $\Gamma \Rightarrow \Delta$ für endliche Formelmengen $\Gamma, \Delta \subseteq \text{AL}$. Wir nennen Γ das *Antezedens* und Δ das *Sukzedens* der Sequenz $\Gamma \Rightarrow \Delta$.

Die Sequenz $\Gamma \Rightarrow \Delta$ ist *gültig*, wenn jedes Modell von Γ auch ein Modell mindestens einer Formel aus Δ ist, d.h. wenn $\wedge \Gamma \models \vee \Delta$. Wenn also $\Gamma \Rightarrow \Delta$ *nicht* gültig ist, dann existiert eine Interpretation \mathcal{J} in der alle Formeln aus Γ wahr und alle Formeln aus Δ falsch sind. In diesem Fall sagen wir, dass \mathcal{J} die Sequenz $\Gamma \Rightarrow \Delta$ *falsifiziert*.

Beispiel.

- Jede Sequenz $\Gamma \Rightarrow \Delta$ mit $\Gamma \cap \Delta \neq \emptyset$ ist gültig. Solche Sequenzen sind die *Axiome* des Sequenzenkalküls.
- Seien Γ, Δ Mengen von Aussagenvariablen. Die Sequenz $\Gamma \Rightarrow \Delta$ ist genau dann falsifizierbar, wenn Γ und Δ disjunkt sind.
- Eine Sequenz der Form $\Gamma \Rightarrow \emptyset$ ist genau dann gültig, wenn Γ unerfüllbar ist.
- Eine Sequenz $\emptyset \Rightarrow \Delta$ ist genau dann gültig, wenn $\vee \Delta$ gültig ist.

Die genaue Formulierung eines Beweiskalküls hängt von den verwendeten Junktoren ab. Wir behandeln hier den aussagenlogischen Sequenzenkalkül für Formeln, welche aus den Junktoren \neg, \wedge, \vee und \rightarrow aufgebaut sind.

Definition 1.26. Die *Axiome* von SK sind alle Sequenzen der Form $\Gamma, \psi \Rightarrow \Delta, \psi$. Die *Schlussregeln* von SK sind:

$$(\neg \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \psi}{\Gamma, \neg \psi \Rightarrow \Delta} \quad (\Rightarrow \neg) \frac{\Gamma, \psi \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \neg \psi}$$

$$(\vee \Rightarrow) \frac{\Gamma, \psi \Rightarrow \Delta \quad \Gamma, \vartheta \Rightarrow \Delta}{\Gamma, \psi \vee \vartheta \Rightarrow \Delta} \quad (\Rightarrow \vee) \frac{\Gamma \Rightarrow \Delta, \psi, \vartheta}{\Gamma \Rightarrow \Delta, \psi \vee \vartheta}$$

$$(\wedge \Rightarrow) \frac{\Gamma, \psi, \vartheta \Rightarrow \Delta}{\Gamma, \psi \wedge \vartheta \Rightarrow \Delta} \quad (\Rightarrow \wedge) \frac{\Gamma \Rightarrow \Delta, \psi \quad \Gamma \Rightarrow \Delta, \vartheta}{\Gamma \Rightarrow \Delta, \psi \wedge \vartheta}$$

$$(\rightarrow \Rightarrow) \frac{\Gamma \Rightarrow \Delta, \psi \quad \Gamma, \vartheta \Rightarrow \Delta}{\Gamma, \psi \rightarrow \vartheta \Rightarrow \Delta} \quad (\Rightarrow \rightarrow) \frac{\Gamma, \psi \Rightarrow \Delta, \vartheta}{\Gamma \Rightarrow \Delta, \psi \rightarrow \vartheta}$$

Hierbei können jeweils für Γ, Δ, Σ beliebige endliche Formelmengen und für ψ, φ, ϑ beliebige Formeln eingesetzt werden. Jede Regel besteht aus einer oder zwei Sequenzen in der oberen Zeile, genannt *Prämissen* und einer Sequenz in der unteren Zeile, genannt *Konklusion*.

Definition 1.27. Die Menge der *ableitbaren Sequenzen* von SK ist die induktiv durch die Axiome und Schlussregeln definierte Sequenzenmenge, d.h. die kleinste Menge, welche alle Axiome umfasst und mit jeder Instanz der oberen Zeile einer Schlussregel auch die entsprechende Instanz der unteren Zeile enthält.

Ein *Beweis in SK* ist ein Baum, dessen Knoten auf folgende Weise mit Sequenzen beschriftet sind:

- Jedes Blatt ist mit einem Axiom beschriftet.
- Jeder innere Knoten des Baumes ist mit der unteren Zeile einer Schlussregel von SK beschriftet; die Kinder dieses Knotens müssen dann gerade mit den in der oberen Zeile dieser Regel auftretenden Sequenz beschriftet sein. Also hat jeder innere Knoten ein oder zwei Kinder.

Es folgt, dass eine Sequenz genau dann in SK ableitbar ist, wenn sie als Beschriftung eines Knotens in einem Beweis von SK auftritt.

Beispiel. Die Sequenz $\psi, (\varphi \vee \vartheta) \Rightarrow (\psi \wedge \varphi), (\psi \wedge \vartheta)$ kann wie folgt in SK bewiesen werden:

$$\frac{\frac{\psi, \varphi \Rightarrow \psi, (\psi \wedge \vartheta) \quad \psi, \varphi \Rightarrow \varphi, (\psi \wedge \vartheta)}{\psi, \varphi \Rightarrow (\psi \wedge \varphi), (\psi \wedge \vartheta)} \quad \frac{\psi, \vartheta \Rightarrow (\psi \wedge \varphi), \psi \quad \psi, \vartheta \Rightarrow (\psi \wedge \varphi), \vartheta}{\psi, \vartheta \Rightarrow (\psi \wedge \varphi), (\psi \wedge \vartheta)}}{\psi, (\varphi \vee \vartheta) \Rightarrow (\psi \wedge \varphi), (\psi \wedge \vartheta)}$$

Wie bei jedem Beweiskalkül sind auch beim Sequenzenkalkül zwei grundlegende Eigenschaften zu überprüfen:

- *Korrektheit:* Es können *nur* gültige Objekte abgeleitet werden.

- *Vollständigkeit*: Es können *alle* gültigen Objekte abgeleitet werden.

Die Korrektheit des Sequenzenkalküls ist leicht nachzuweisen.

Lemma 1.28. Für jede Regel des Sequenzenkalküls und jede aussagenlogische Interpretation \mathcal{I} (deren Definitionsbereich alle vorkommenden Aussagenvariablen umfasst) gilt: \mathcal{I} falsifiziert die Konklusion der Regel genau dann wenn \mathcal{I} eine Prämisse der Regel falsifiziert. Es folgt, dass die Konklusion genau dann gültig ist, wenn die Prämissen gültig sind.

Übung 1.21. Beweisen Sie dieses Lemma.

Eine unmittelbare Konsequenz ist der Korrektheitssatz für SK.

Satz 1.29 (Korrektheit des Sequenzenkalküls). Jede in SK ableitbare Sequenz $\Gamma \Rightarrow \Delta$ ist gültig.

Aus dem Sequenzenkalkül gewinnen wir unmittelbar auch einen formalen Ableitungsbegriff für *Formeln* (statt Sequenzen).

Definition 1.30. Sei $\Phi \subseteq \text{AL}$ eine Formelmenge. Eine aussagenlogische Formel ψ ist *ableitbar* aus der Hypothesenmenge Φ (kurz: $\Phi \vdash \psi$), wenn eine endliche Teilmenge Γ von Φ existiert, so dass die Sequenz $\Gamma \Rightarrow \psi$ im Sequenzenkalkül ableitbar ist. Insbesondere ist ψ aus der leeren Hypothesenmenge ableitbar (kurz: $\vdash \psi$) wenn die Sequenz $\emptyset \Rightarrow \psi$ in SK abgeleitet werden kann.

Der Sequenzenkalkül erlaubt die *systematische Suche und Analyse* von Beweisen. Dies ist ein wichtiger Vorteil gegenüber vielen andern Beweiskalkülen (z.B. dem Hilbertkalkül). Wir werden einen Algorithmus angeben, welcher zu jeder gegebenen Sequenz $\Gamma \Rightarrow \Delta$ entweder einen Beweis konstruiert, oder aber eine Interpretation findet, welche jede Formel aus Γ , aber keine aus Δ erfüllt und damit den Nachweis liefert, dass $\Gamma \Rightarrow \Delta$ nicht ableitbar ist. Wir erläutern diesen Algorithmus zunächst an zwei Beispielen.

Beispiel.

- Betrachte die Formel $\psi := (X \rightarrow Y) \rightarrow (\neg Y \rightarrow \neg X)$. Wir suchen also einen Beweis in SK für die Sequenz $\emptyset \Rightarrow \psi$. Wir beobachten zunächst, dass ψ die Form $(\varphi \rightarrow \vartheta)$ hat. Die einzige Regel, die

zu einer Sequenz der Form $\emptyset \Rightarrow (\varphi \rightarrow \vartheta)$ führen kann, ist die Regel $(\Rightarrow \rightarrow)$. Diese Regel kann aber nur angewandt werden, wenn vorher die Sequenz $\varphi \Rightarrow \vartheta$, d.h. die Sequenz $(X \rightarrow Y) \Rightarrow (\neg Y \rightarrow \neg X)$ abgeleitet wurde. Wir beginnen also die Konstruktion des Ableitungsbaums so:

$$\frac{(X \rightarrow Y) \Rightarrow (\neg Y \rightarrow \neg X)}{\emptyset \Rightarrow (X \rightarrow Y) \rightarrow (\neg Y \rightarrow \neg X)}$$

Um nun $(X \rightarrow Y) \Rightarrow (\neg Y \rightarrow \neg X)$ abzuleiten, können wir entweder mit der Regel $(\rightarrow \Rightarrow)$ auf dem Antezedens oder mit der Regel $(\Rightarrow \rightarrow)$ auf dem Sukzedens arbeiten. Die erste Möglichkeit führt zu einer Verzweigung des Ableitungsbaums:

$$\frac{\frac{\emptyset \Rightarrow X, (\neg Y \rightarrow \neg X) \quad Y \Rightarrow (\neg Y \rightarrow \neg X)}{(X \rightarrow Y) \Rightarrow (\neg Y \rightarrow \neg X)}}{\emptyset \Rightarrow (X \rightarrow Y) \rightarrow (\neg Y \rightarrow \neg X)}$$

Die beiden Blätter werden nun mit den Regeln $(\Rightarrow \rightarrow)$ und dann $(\neg \Rightarrow)$ und $(\Rightarrow \neg)$ weiter bearbeitet. Dies führt schließlich zu folgendem Ableitungsbaum:

$$\frac{\frac{\frac{X, \neg Y \Rightarrow X}{\neg Y \Rightarrow X, \neg X} \quad \frac{Y \Rightarrow \neg X, Y}{Y, \neg Y \Rightarrow \neg X}}{\emptyset \Rightarrow X, (\neg Y \rightarrow \neg X) \quad Y \Rightarrow (\neg Y \rightarrow \neg X)}}{(X \rightarrow Y) \Rightarrow (\neg Y \rightarrow \neg X)}{\emptyset \Rightarrow (X \rightarrow Y) \rightarrow (\neg Y \rightarrow \neg X)}$$

Die Blätter dieses Baumes sind Axiome, und wir haben damit einen Beweis für die gegebene Sequenz gefunden.

Wenn wir nach dem ersten Ableitungsschritt die zweite Möglichkeit gewählt hätten und mit der Regel $(\Rightarrow \rightarrow)$ auf dem Sukzedens weitergearbeitet hätten, dann wären wir schließlich zum Beweis

$$\frac{\frac{\frac{X \Rightarrow X, Y \quad X, Y \Rightarrow Y}{(X \rightarrow Y), X \Rightarrow Y}}{X \rightarrow Y, \neg Y \Rightarrow \neg X}}{(X \rightarrow Y) \Rightarrow (\neg Y \rightarrow \neg X)}{\emptyset \Rightarrow (X \rightarrow Y) \rightarrow (\neg Y \rightarrow \neg X)}$$

gekommen. Wir sehen also, dass es verschiedene Beweise derselben Sequenz gibt.

- Als zweites Beispiel betrachten wir die Sequenz $(X \vee Y) \Rightarrow (X \wedge Y)$. Die Konstruktion des Ableitungsbaums führt mit der Regel $(\Rightarrow \wedge)$ zunächst auf den Baum

$$\frac{X \vee Y \Rightarrow X \quad X \vee Y \Rightarrow Y}{X \vee Y \Rightarrow X \wedge Y}.$$

Mit der Regel $(\vee \Rightarrow)$ erhalten wir dann den Ableitungsbaum

$$\frac{\frac{X \Rightarrow X \quad Y \Rightarrow X}{X \vee Y \Rightarrow X} \quad \frac{X \Rightarrow Y \quad Y \Rightarrow Y}{X \vee Y \Rightarrow Y}}{X \vee Y \Rightarrow X \wedge Y}.$$

Die Blätter bestehen nur aus Aussagenvariablen, aber nur die äußeren beiden sind Axiome. Die beiden Blätter $Y \Rightarrow X$ und $X \Rightarrow Y$ werden durch die Interpretationen falsifiziert, welche eine der Aussagenvariablen X, Y mit wahr, die andere aber mit falsch belegen. Diese Interpretationen falsifizieren auch die Ausgangssequenz $(X \vee Y) \Rightarrow (X \wedge Y)$. Der Versuch, einen Beweis für diese Sequenz zu konstruieren führt also zu einer Interpretation, welche die Sequenz falsifiziert und damit (wegen der Korrektheit des Sequenzenkalküls) nachweist, dass kein Beweis existiert.

Die systematische Beweissuche beruht darauf, dass zu jeder Sequenz $\Gamma \Rightarrow \Delta$ und jeder darin vorkommenden nicht-atomaren Formel ψ genau eine Regel mit der Konklusion $\Gamma \Rightarrow \Delta$ existiert, in deren Prämissen ψ nicht vorkommt. Der Algorithmus baut nun wie in den beiden Beispielen ausgehend von der zu beweisenden Sequenz einen Ableitungsbaum auf, indem er rückwärts von der Konklusion und einer daraus ausgewählten Formel die entsprechende Regel bestimmt und den Baum um die Prämissen dieser Regel erweitert, bis entweder eine rein atomare, falsifizierbare Sequenz gefunden wird oder alle Blätter mit Axiomen beschriftet sind.

Definition 1.31. Ein *Ableitungsbaum* T für eine Sequenz S ist ein Baum, dessen Wurzel mit S beschriftet ist, so dass jeder innere Knoten von T mit der unteren Zeile einer Schlussregel und die Kinder dieses Knotens

mit den in der oberen Zeile derselben Regel auftretenden Sequenzen beschriftet sind.

Ein mit einem Axiom beschriftetes Blatt eines Ableitungsbaums nennen wir *positiv*. Ein Blatt ist *negativ*, wenn es mit einer Sequenz $\Gamma \Rightarrow \Delta$ beschriftet ist, wobei Γ und Δ disjunkte Mengen von Aussagenvariablen sind. Ein Ableitungsbaum ist *vollständig*, wenn alle seine Blätter positiv oder negativ sind.

Ein Beweis ist demnach ein Ableitungsbaum, dessen Blätter alle positiv sind (und welcher daher insbesondere vollständig ist). Ein Ableitungsbaum, der ein negatives Blatt enthält, nennen wir eine *Widerlegung*.

Wir können nun einen Algorithmus angeben, welcher zu jeder aussagenlogischen Sequenz einen Beweis oder eine Widerlegung findet.

Satz 1.32. Algorithmus 1.3 terminiert auf jeder gegebenen Sequenz $\Gamma \Rightarrow \Delta$ in endlich vielen Schritten. Er findet genau dann einen Beweis wenn $\Gamma \Rightarrow \Delta$ gültig ist; andernfalls findet er eine falsifizierende Interpretation für $\Gamma \Rightarrow \Delta$.

Beweis. Die Komplexität einer Sequenz sei die Anzahl der in ihr vorkommenden Junktoren. Für jede Regel von SK gilt, dass die Komplexität der Konklusion echt größer ist als die Komplexität der Prämissen. Deshalb kann die Tiefe des konstruierten Ableitungsbaum nicht größer sein als die Komplexität der Ausgangssequenz; der Algorithmus muss also terminieren.

Wenn der Algorithmus auf $\Gamma \Rightarrow \Delta$ einen Ableitungsbaum T findet, dessen Blätter alle mit (+) markiert sind (deren Beschriftungen also Axiome sind), dann ist T offensichtlich ein Beweis für $\Gamma \Rightarrow \Delta$. Aufgrund der Korrektheit des Sequenzenkalküls ist $\Gamma \Rightarrow \Delta$ dann gültig.

Andernfalls enthält der konstruierte Ableitungsbaum ein negatives Blatt mit einer Beschriftung $\Gamma' \Rightarrow \Delta'$, so dass Γ' und Δ' disjunkte Mengen von Aussagenvariablen sind. Indem man die Aussagenvariablen in Γ' mit wahr, diejenigen in Δ' mit falsch und alle übrigen beliebig belegt, gewinnt man eine Interpretation, welche $\Gamma' \Rightarrow \Delta'$ falsifiziert. Aus Lemma 1.28 folgt, dass diese Interpretation auch die Ausgangssequenz $\Gamma \Rightarrow \Delta$ falsifiziert. Q.E.D.

Algorithmus 1.3. Beweissuche im aussagenlogischen Sequenzenkalkül

Input: Eine aussagenlogische Sequenz $\Gamma \Rightarrow \Delta$.

Ein Ableitungsbaum für $\Gamma \Rightarrow \Delta$ wird induktiv wie folgt aufgebaut. Zu Beginn sei T der Baum, der nur aus der Wurzel besteht, beschriftet mit $\Gamma \Rightarrow \Delta$. Solange T noch unmarkierte Blätter enthält, werden folgende Operationen ausgeführt:

Wähle ein unmarkiertes Blatt ℓ ; sei $\Gamma' \Rightarrow \Delta'$ die Beschriftung von ℓ .

Wenn ℓ negativ ist, dann wird die Interpretation konstruiert welche alle Aussagenvariablen in Γ' mit wahr und alle andern mit falsch bewertet. Diese wird als falsifizierende Interpretation für $\Gamma \Rightarrow \Delta$ ausgegeben. Die Prozedur ist damit beendet.

Wenn ℓ positiv ist wird ℓ mit (+) markiert.

Andernfalls wird eine nicht-atomare Formel ψ aus $\Gamma' \Rightarrow \Delta'$ ausgewählt und die (eindeutig festgelegte) Regel bestimmt, deren Konklusion $\Gamma' \Rightarrow \Delta'$ ist und deren Prämissen ψ nicht mehr enthalten. Dann wird T um ein oder zwei Nachfolgeknoten von ℓ erweitert, welche mit den Prämissen dieser Regel beschriftet werden.

Wenn alle Blätter mit (+) markiert sind, wird T als Beweis für $\Gamma \Rightarrow \Delta$ ausgegeben und die Prozedur beendet.

Der Sequenzenkalkül liefert also sogar ein *Entscheidungsverfahren* für die gültigen aussagenlogischen Sequenzen und damit auch für die aussagenlogischen Tautologien. Insbesondere folgt aus Satz 1.32, dass der aussagenlogische Sequenzenkalkül vollständig ist.

Folgerung 1.33 (Vollständigkeit des Sequenzenkalküls). Jede gültige aussagenlogische Sequenz ist im Sequenzenkalkül ableitbar.

Übung 1.22. Geben Sie Schlussregeln ($\oplus \Rightarrow$) und ($\Rightarrow \oplus$) für den Junktor \oplus („exklusives oder“) an. Konstruieren Sie im entsprechend

erweiterten Sequenzenkalkül einen Beweis für die Sequenz $(\psi \oplus \varphi) \oplus \vartheta \Rightarrow \psi \oplus (\varphi \oplus \vartheta)$.

Übung 1.23. Modifizieren Sie den Suchalgorithmus für den Sequenzenkalkül zu einem Entscheidungsverfahren für die Erfüllbarkeit aussagenlogischer Formeln, also zu einem Algorithmus, welcher zu jeder gegebenen aussagenlogischen Formel ψ entscheidet, ob ψ erfüllbar ist oder nicht.