Christoph Lüth, Serge Autexier

Einführung in die Formale Logik

Sommersemester 2023

Vorlesungsbegleitende Unterlagen

Stand vom 14. September 2023.

Inhaltsverzeichnis

Kapitel 1

Aussagenlogik

Vorlesung vom 11.04.2023: Aussagenlogik I

1.1 Logische Systeme

Definition 1.1 *Ein* logisches System *besteht aus:*

*(1) einer formalen Sprache (*Syntax*) mit einem Alphabet von Zeichen, und Regeln, die daraus erlaubte Worte (*Formeln*) bilden;*

(2) einer Semantik*, welche Formeln eine Bedeutung zuordnet;*

(3) Schlussregeln, welche ein oder mehrere Formeln in eine andere Formel umformen.

Am Anfang werden alle unsere Formeln entweder wahr (1) oder falsch (0) sein; später werden wir das etwas aufweichen.

Beispiele für logische Systeme: Programmiersprachen?

1.2 Aussagen

Aussagenlogik beruht auf atomaren Aussagen, die wir mit Konnektiven zu komplexeren Formeln kombinieren.

Definition 1.2 *Eine (logische)* Aussage *ist etwas, was entweder wahr oder falsch sein kann. Eine logische Aussage ist* atomar*, wenn sie nicht in weitere Aussagen zerlegt werden kann.*

Welches der folgenden sind Aussagen? Welche sind atomar?

(1) Säugetiere sind Warmblüter.

*Atomare Aussage (*1*).*

(2) Die Sonne umkreist die Erde.
Atomare Aussage (0)

(3) Jede Sekunde verbrennt die Sonne 4 Millionen Tonnen ihrer Masse zu Energie.
Atomare Aussage (1, anscheinend)

(4) Fiete ist größer als Kalle.
Atomare Aussage

(5) Kalle ist kleiner als Fiete.
Atomare Aussage, (fast) äquivalent zu der davor.

(6) Ist Fiete schlauer als Kalle?
Keine Aussage— keine Wahrheitswert.

(7) Kalle hat doch gar kein Auto.
Atomare Aussage.

(8) Kalle hat doch gar kein Auto?
Aussage wird zur Frage — keine Aussage.

(9)
$$\vec{v}(t) = \frac{\delta \vec{s}(t)}{\delta t}$$
Atomare Aussage, formal hingeschrieben.

(10) Geschwindigkeit ist die Ableitung des Ortsvektors nach der Zeit.
Atomare Aussage. Definition.

(11) Bitte nicht bewegen.
Keine Aussage

(12) Kalle kommt mit und Fiete kommt mit.
Komplexe Aussage.

(13) Kalle und Fiete kommen mit.
Komplexe Aussage.

(14) Fiete kommt mit, wenn Kalle auch kommt.
Komplexe Aussage.

(15) Kommen Kalle oder Fiete mit?
Keine Aussage

(16) Sie kommen nicht mit.
Wer sind sie? Aussage mit einer Variablen.

(17) Kalle und Fiete heiraten.
Aussage. Mehrdeutig — wenn sei einander *heiraten atomar, sonst komplex.*

(18) Der König ist tot, lang lebe der König!
Der erste Teil ist eine atomare Aussage, der zweite Teil ist keine Aussage.

(19) Solange die Sonne scheint, regnet es nicht.
Komplexe Aussage.

(20) Wenn der Kuchen spricht schweigen die Krümel.
Aussage, kann aber auch eine Aufforderung sein.

Folgendes sind also keine Aussagen:

- Fragen,
- Aufforderungen,
- Interjektionen (spontante Ausrufe).

Komplexe Aussagen werden aus atomaren Aussagen durch *Konnektive* verbunden. In den Aussagen haben wir verschiedene Konnektive:

- "und" (Konjunktion),
- "oder" (Disjunktion),
- "wenn. . . dann" (Implikation),
- "wenn", "solange" (zeitliche Abfolge).

Umgangssprachliche Konnektive sind mehrdeutig. Hier noch ein paar Beispiele:

(1) Heinz fuhr weiter und fuhr einen Fußgänger an.

(2) Heinz fuhr einen Fußgänger an und fuhr weiter.

(3) Wenn ich das Fenster öffne haben wir Frischluft.

(4) Wenn ich das Fenster öffne kreist die Erde um die Sonne.

(5) Wenn die Sonne um die Erde kreist dann haben wir Frischluft.

(6) Hannes arbeitet, oder er ist in der Kneipe.

(7) Euclid war Grieche oder Mathematiker.

- In (1), (2) haben wir eine implizite zeitliche Ordnung.
- (3) ist ein kausaler Zusammenhang. (4) und (5) kommen uns widersinnig vor, (5) sogar falsch; (5) könnte eine rhetorische Figur sein ("Wenn die Sonne um die Erde kreist dann bin ich Kaiser von China!").
- In (6) und (7) wird "oder" exklusiv gelesen; (6) erscheint richtig (es sei denn Hannes arbeitet in der Gastronomie), (7) falsch.

Zusammenfassend hat natürliche Sprache mehrere Probleme: sie ist *mehrdeutig* und enthälft oft versteckte (implizite) *Annahmen*.

1.3 Aussagenlogik

Aussagenlogik kombiniert atomare Aussage mit Konnektiven, die eine feste Bedeutung haben. Dabei betrachten wir keine zeitlichen Ablauf oder sonstige Modalitäten (das wäre dann Temporallogik oder Modallogik).

1.3.1 Sprache

Wir definieren erst unsere formale Sprache der Aussagenlogik:

Definition 1.3 *Gegeben eine abzählbar unendliche Menge P von atomaren Aussagen, dann ist $Prop$ die kleinste Menge so dass mit $\phi, \psi \in Prop$:*

$$\begin{aligned} P &\subseteq Prop \\ \bot &\in Prop \\ \neg\phi &\in Prop \\ \phi \wedge \psi &\in Prop \\ \phi \vee \psi &\in Prop \\ \phi \longrightarrow \psi &\in Prop \end{aligned}$$

Einfacher geschrieben:

$$p ::= q \in P \mid \bot \mid \neg p \mid p_1 \wedge p_2 \mid p_1 \vee p_2 \mid p_1 \longrightarrow p_2$$

Syntaktisch gelten folgende *Präzedenzen:* $\neg$ vor $\wedge$ vor $\vee$ vor $\longrightarrow$

Übung 1.1 *Wie sehen folgende Ausdrücke voll geklammert aus?*

- $A \wedge B \vee C \longrightarrow C \vee X$
- $\neg(B \vee C) \wedge X \vee Y$
- $A \longrightarrow B \vee \neg C \wedge D$

1.3.2 Bedeutung

Was ist die *Bedeutung* einer aussagenlogischen Formal $p \in Prop$? Sie ist entweder wahr oder falsch. Wenn wir wahr mit 1 kodieren und falsch mit 0, dann ist $\mathbb{B} = \{0,1\}$ die Menge aller boolschen Werte. (Offensichtlich ist $\mathbb{B} \subset \mathbb{N}$.) Die Semantik ist dann eine Abbildung $sem : Prop \to \mathbb{B}$ (die wir allerdings mit "semantischen Klammern" $[\![\cdot]\!]$ schreiben). Das ist nicht so ganz perfekt — was ist mit atomaren Aussagen? Deren Bedeutung können wir *festlegen*; sie sind quasi Parameter der Bedeutungsfunktion — wir wissen zwar, dass die Erde um die Sonne kreist, aber bspw. nicht ob Kalle oder Fieter größer ist. (Wir wissen ja nicht mal, wer die beiden sind.)

Wir definieren die Funktion *induktiv* über der Struktur von *Prop*. (Das ist wie in Haskell.) Dafür benötigen wir alle vier Konnektive eine *Bedeutung*.

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$\neg$	
0	1
1	0

$\longrightarrow$	0	1
0	1	1
1	0	1

Abbildung 1.1: Wahrheitstabellen für die vier Konnektive

Für $\wedge$ heißt das beispielsweise, dass wir die Bedeutung von $p \wedge q$ aus der Kombination der möglichen Bedeutungen von p und q definieren müssen. Für jeden der beiden ist dies entweder $\top$ oder $\bot$, was uns zu vier Möglichkeiten führt.

Die mögliche Bedeutung läßt sich schnell an der Aussage "Serge und Christoph sind in der Vorlesung." verdeutlichen:

Serge	Christoph	Serge und Christoph
da	da	da
nicht da	da	nicht da
da	nicht da	nicht da
nicht da	nicht da	nicht da

Für die Disjunktion ergibt sich schnell eine ähnliche Tabelle:

Serge	Christoph	Serge oder Christoph
da	da	da
nicht da	da	da
da	nicht da	da
nicht da	nicht da	nicht da

Die Negation überlassen wir dem geneigten Leser, aber was mit der Implikation? "Wenn Serge in der Vorlesung ist, ist Christoph auch da." Eine Wahrheitstabelle:

Serge	Christoph	Wenn Serge da ist, dann ist Christoph da
da	da	wahr
nicht da	da	?
da	nicht da	falsch
nicht da	nicht da	wahr

In allen Fällen ist der Wahrheitsgehalt evident, bis auf die zweite Zeile. Ist die Aussage jetzt wahr oder nicht? Wir *definieren* sie in dem Fall als wahr — das könnte man auch anders machen, aber so ist die resultierende Theorie eleganter (Ableitungsregeln, Äquivalenzen, *etc.*).

Diese Wahrheitstabellen stellt man traditionellerweise etwas kompakter dar (vgl. Abb. 1.1).

Der Wahrheitswert einer beliebigen Formel $\phi \in \mathit{Prop}$ ist abhängig von den Wahrheitswerten der atomaren Aussagen (*Atome*). Dazu müssen wir nicht alle Atome mit einem Wert belegen, sondern nur die in ϕ enthaltenen. Wir definieren:

Definition 1.4 *Sei $\phi \in Prop$ eine Aussage, dann ist die Menge der in ϕ enthaltenen Atome induktiv definiert als*

$$\begin{aligned}
\text{atoms}(q) &= \{q\} \quad (\textit{für } q \in P)\\
\text{atoms}(\bot) &= \emptyset\\
\text{atoms}(\neg\phi) &= \text{atoms}(\phi)\\
\text{atoms}(\phi \wedge \psi) &= \text{atoms}(\phi) \cup \text{atoms}(\psi)\\
\text{atoms}(\phi \vee \psi) &= \text{atoms}(\phi) \cup \text{atoms}(\psi)\\
\text{atoms}(\phi \longrightarrow \psi) &= \text{atoms}(\phi) \cup \text{atoms}(\psi)
\end{aligned}$$

Für eine Aussage $\phi \in Prop$ ist eine *Valuation* oder *Belegung* (der Variablen) eine Funktion $v : \text{atoms}(\phi) \to \mathbb{B}$.

Definition 1.5 *Für eine Formel $\phi \in Prop$ und eine Belegung $v : \text{atoms}(\phi) \to \mathbb{B}$ ist der Wert $[\![\phi]\!]_v$ von ϕ unter v induktiv definiert als*

$$\begin{aligned}
[\![p]\!]_v &= v(p) \quad (\textit{für } p \in P)\\
[\![\bot]\!]_v &= 0\\
[\![\neg\phi]\!]_v &= 1 - [\![\phi]\!]_v\\
[\![\phi \wedge \psi]\!]_v &= \min([\![\phi]\!]_v, [\![\psi]\!]_v)\\
[\![\phi \vee \psi]\!]_v &= \max([\![\phi]\!]_v, [\![\psi]\!]_v)\\
[\![\phi \longrightarrow \psi]\!]_v &= \begin{cases} 0 & \textit{wenn } [\![\phi]\!]_v = 1 \textit{ und } [\![\psi]\!]_v = 0\\ 1 & \textit{sonst} \end{cases}
\end{aligned}$$

Für eine gegebene Belegung v ist die Semantik eindeutig definiert, *i.e.* wenn v und w zwei Belegungen für ϕ sind und $v(p) = w(p)$ für alle $p \in \text{atoms}(\phi)$, dann ist $[\![\phi]\!]_v = [\![\phi]\!]_w$. Der Beweis erfolgt durch Induktion über die Struktur von ϕ.

Vorlesung vom 18.04.23: Aussagenlogik II

Übung 1.2 (Aufwärmübung) *Formalisiert folgende Aussagen. Welches sind die Atome, und wie sieht die Formel $\phi \in Prop$ aus?*

> *"Wenn ich alle Übungsblätter abgebe und die Prüfung bestehe bekomme ich den Schein. Wenn ich keinen Schein habe, habe ich also entweder nicht alle Übungsblätter abgegeben oder die Prüfung nicht bestanden."*

> *"Wenn es regnet, werde ich naß. Wenn ich aber einen Schirm habe, dann werde ich nicht naß, wenn es regnet."*

> *"Wenn gegessen habe, bin ich satt, aber ich bin hungrig, also habe ich nicht gegessen."*

Sind diese Aussagen wahr? Was heißt das?

1.3.3 Tautologien

Damit können wir jetzt darüber reden, ob eine Formel (immer) "wahr" ist:

Definition 1.6 *Gegeben eine Formel $\phi \in Prop$, dann ist ϕ eine* Tautologie*, wenn $[\![\phi]\!]_v = 1$ für alle Belegungen v. Wir schreiben dafür auch $\models \phi$, und sagen ϕ ist* semantisch gültig.

Ist eine Formel ϕ für *keine* Belegung erfüllt, ist sie *unerfüllbar* (oder ein Widerspruch). Wie man leicht sieht, ist ϕ unerfüllbar gdw. $\neg\phi$ semantisch gültig ist (weil $[\![\neg\phi]\!]_v = 1$ gdw. $[\![phi]\!]_v = 0$). Darüber hinaus gibt es noch Formeln, die für einige, aber nicht alle Belegungen, erfüllbar sind.

Wie finden wir heraus, ob ϕ eine Tautologie ist? Das wird uns den Rest der Veranstaltung beschäftigen... aber eine erste, einfache Möglichkeit sind *Wahrheitstabellen*: wir zählen einfach alle möglichen Belegungen auf.

Beispiele:

- $A \wedge B \longrightarrow A$

A	B	A	$\wedge$	B	$\longrightarrow$	A
0	0	0	0	0	1	0
0	1	0	0	1	1	0
1	0	1	0	0	1	1
0	1	1	1	1	1	1

- $\bot \longrightarrow A$ (*Ex falso quodlibet*)

A	0	$\longrightarrow$	$\neg A$
0	0	1	1
1	0	1	0

- Gegenbeispiel: $A \vee B \longrightarrow A$

A	B	A	$\vee$	B	$\longrightarrow$	A
0	0	0	0	0	1	0
0	1	0	1	1	0	0
1	0	1	1	0	1	1
0	1	1	1	1	1	1

- Längeres Beispiel: $(A \longrightarrow (B \longrightarrow C) \longrightarrow (A \wedge B \longrightarrow C)$ (und umgekehrt)

A	B	C	$(A$	$\longrightarrow$	$(B$	$\longrightarrow$	$C))$	$\longrightarrow$	$(A$	$\wedge$	B	$\longrightarrow$	$C)$
0	0	0	0	1	0	1	0	1	0	0	0	1	0
0	0	1	0	1	0	1	1	1	0	0	0	1	1
0	1	0	0	1	1	0	0	1	0	0	1	1	0
0	1	1	0	1	1	1	1	1	0	0	1	1	1
1	0	0	1	1	0	1	0	1	1	0	0	1	0
1	0	1	1	1	0	1	1	1	1	0	0	1	1
1	1	0	1	0	1	0	0	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1

Übung 1.3 *Weitere Beispiele als Aufgabe:*

(1) $A \vee \neg A$ (Tertium non datur, Satz des ausgeschlossenen Dritten)

(2) $(\neg A \vee B) \longrightarrow (A \longrightarrow B)$ *und umgekehrt.*

Diese Art von Beweisen skaliert ganz klar nicht: die Anzahl der möglichen Belegungen für eine Formel $\phi \in Prop$ ist 2^n mit $n = |\text{atoms}(\phi)|$, d.h. die Anzahl der Atome in ϕ. Exponentielles Wachstum fällt immer unter "schlechte Nachrichten", was irgendeine Art von Automatisierung im Größeren betrifft. (Wir werden später noch weiter über Aufwand reden.)

1.3.4 Semantische Folgerung

Definition 1.7 (Semantische Folgerung) *Sei* $\Gamma = \{\phi_1, \ldots, \phi_n\}$ *eine Menge von Aussagen, und* ψ *eine Aussage, dann ist* $\Gamma \models \psi$ *gdw. folgendes gilt: für alle Valuationen v mit* $[\![\phi_1]\!]_v = 1, \ldots, [\![\phi_n]\!]_v = 1$ *ist auch* $[\![\psi]\!]_v = 1$.

Das folgende Theorem zeigt, dass semantische Folgerung und synaktische Implikation übereinstimmen. Das ist eine erste Aussage über den Zusammenhang von Syntax und Semantik:

Theorem 1.1 (Deduktionstheorem)

(1) $\phi \models \psi$ *gdw.* $\models \phi \longrightarrow \psi$

(2) $\models \phi \wedge \psi$ *gdw.* $\models \phi$ *und* $\models \psi$

(3) $\Gamma \models \psi$ *gdw.* $\models \phi_1 \wedge \ldots \wedge \phi_n \longrightarrow \psi$

Beweis. (1) und (2) folgen direkt aus der Definition von $\models$. Wir zeigen (1):

$$\begin{aligned}\phi \models \psi &\Longleftrightarrow \text{für alle } v\text{: wenn } [\![\phi]\!]_v = 1 \text{ dann } [\![\psi]\!]_v = 1\\ &\Longleftrightarrow \text{für alle v: } [\![\phi \longrightarrow \psi]\!]_v\\ &\Longleftrightarrow \models \phi \longrightarrow \psi\end{aligned}$$

Aus (1) und (2) folgt direkt (3). □

Vorlesung vom 20.04.2023: Aussagenlogik III

Übung 1.4 (Aufwärmübung) *Wir haben über das exklusive Oder schon gesprochen. Welche Wahrheitstabelle kann der entsprechende Operator* $\dot\vee$ *haben? Und wie können wir* $\phi \dot\vee \psi$ *mit den anderen Operatoren ausdrücken?*

Uns fällt auf, dass beispielsweise $A \longrightarrow B$ und $\neg A \longrightarrow B$ die gleiche Wahrheitstabelle haben. (Das gilt für viele andere Aussagen.) Aus dieser Beobachtungen folgen mehrere Fragen:

- Was heißt das genau, insbesondere syntaktisch? Können wir einen syntaktischen Äquivalenzoperator definieren?
- Können wir diesen Operator vielleicht benutzen, um durch *Umformen* wie in der Algebra Beweise zu führen?
- Können wir die Implikation mit Hilfe der Negation und Disjunktion definieren, und welches sind die wenigsten Operatoren, die wir eigentlich benötigen?

1.3.5 Beweis durch Umformen

In der Algebra beweisen wir oft durch wiederholtes Umformen von Ausdrücken, wie man es in der Schule lernt. Hier ein typischer Beweis:

$$\begin{aligned}
& x^2 - 8\cdot x + 12 = 0 \\
\iff\quad & x^2 - 2\cdot 4\cdot x + 16 - 4 = 0 \\
\iff\quad & (x-4)^2 - 4 = 0 \\
\iff\quad & (x-4)^2 = 4 \\
\iff\quad & (x-4) = 2 \vee (x-4) = -2 \\
\iff\quad & x = 6 \vee x = 2
\end{aligned}$$

Wann funktioniert dieser Beweis?

- Der wesentliche Schluss ist, das $x = 6$ oder $x = 2$ die Anfangsgleichung erfüllen. Damit das gilt, muss die Relation $\iff$ *transitiv* sein.
- Wir rechnen vorwärts (von der Anfangsgleichung ausgehend), aber sind am Schluss rückwärts interessiert (wenn x die berechneten Werte hat, dann ist es eine Lösung); dazu muss die Relation $\iff$ zwingend *symmetrisch* sein.
- Wir ersetzen *innerhalb* der Gleichungen, bspw. in dem $12 = 16 - 4$ nutzen, die Assoziativität der Addition, die binomische Formel *etc.*. Dazu muss die Relation $\iff$ *substitutiv* sein, d.h. wenn $A \iff B$ dann gilt auch $A' \iff B'$ wenn ich in A und B gleiches durch gleiches ersetzen.

1.3.6 Äquivalenz

Zuerst einmal definieren wir, was Äquivalenz überhaupt heißt. Dazu führen wir den Operator $\longleftrightarrow$ ein:

Definition 1.8 (Äquivalenz) *Die Äquivalenz ist definiert als*

$$A \longleftrightarrow B \; gdw \; (A \longrightarrow B) \wedge (B \longrightarrow A)$$

Mit der Definition ergibt sich folgende Wahrheitstabelle:

$\longleftrightarrow$	0	1
0	1	0
1	0	1

Wir wollen jetzt zeigen, dass die Äquivalenz semantisch der Gleichheit entspricht.

Lemma 1.2 *Für alle Belegungen v gilt:*

$$[\![\phi \longleftrightarrow \psi]\!]_v = 1 \; gdw. \; [\![\phi]\!]_v = [\![\psi]\!]_v \tag{1.1}$$

Beweis. Zum Beweis nutzen wir folgende Gleichung. Für alle Belegungen v gilt nämlich auch:

$$[\![\phi \longrightarrow \psi]\!]_v = 1 \text{ gdw. } [\![\phi]\!]_v \leq [\![\psi]\!]_v \tag{1.2}$$

Diese Gleichung folgt direkt aus der Definition von $[\![\phi \longrightarrow \psi]\!]_v$; man kann sie sogar als Definition dafür nehmen. Damit folgt jetzt: wenn $[\![\phi \longleftrightarrow \psi]\!]_v = 1$, dann $[\![\phi \longrightarrow \psi]\!]_v = 1$ und $[\![\phi \longrightarrow \psi]\!]_v = 1$ (nach Definition von $\longleftrightarrow$), mit (1.2) dann $[\![\phi]\!]_v \leq [\![\psi]\!]_v$ und $[\![\psi]\!]_v \leq [\![\phi]\!]_v$, und damit $[\![\phi]\!]_v = [\![\psi]\!]_v$ wie gewünscht. □

Lemma 1.2 zeigt, dass die syntaktische Äquivalenz auch eine semantische Gleichheit ist: wenn zwei Formeln äquivalent sind, haben sie die gleiche Semantik. Jetzt können wir die *semantische Äquivalenz* (auf Formeln) definieren:

Definition 1.9 *$A \approx B$ gdw. $\models A \longleftrightarrow B$*

Um damit rechnen zu können, müssen wir (s.o.) Transitivität und Symmetrie zeigen, mit anderen Worten, dass die Äquivalenz eine *Äquivalenzrelation* ist:

$$\phi \approx \phi \tag{1.3}$$

$$\text{Wenn } \phi \approx \psi \text{ dann } \psi \approx \phi \tag{1.4}$$

$$\text{Wenn } \phi \approx \psi \text{ und } \psi \approx \sigma \text{ dann } \psi \approx \sigma \tag{1.5}$$

$$\tag{1.6}$$

Beweis. Aus der Definition von $\approx$ folgt dass $A \approx B$ gdw. $[\![\phi]\!]_v = [\![\psi]\!]_v$ für alle Belegungen v. Da die Gleichheit $=$ transitiv, reflexiv und symmetrisch ist, folgt dies auch für $\approx$. □

Jetzt fehlt noch die *Substitutivität*, *i.e.* wir müssen in einer aussagenlogischen Formel gleiches durch gleiches ersetzen. Aber was heißt überhaupt ersetzen?

1.3.7 Ersetzung

Wir können atomare Aussagen als Variablen auffassen, die wir durch andere *Aussagen* ersetzen. Wir schreiben die Substitution als $\phi[\psi/p]$ für ein Atom p, gelesen als "in ϕ ersetzen wir ψ für p". Syntaktisch bindet sie schwächer als alle Operatoren.

Definition 1.10 (Substitution) *Für $\phi \in Prop$, $q \in P$ und $\psi \in Prop$ definieren wir die Ersetzung von x in ϕ durch ψ, geschrieben $\phi[\psi/p]$, rekursiv über der Struktur von ϕ wie folgt:*

$$p[\psi/q] = \begin{cases} \psi & p = q \quad (p \in P) \\ p & p \neq q \quad (p \in P) \end{cases}$$

$$\begin{aligned}
\bot[\psi/q] &= \bot \\
(\neg\phi)[\psi/q] &= \neg(\phi[\psi/q]) \\
(\phi_1 \wedge \phi_2)[\psi/q] &= (\phi_1[\psi/q]) \wedge (\phi_2[\psi/q]) \\
(\phi_1 \vee \phi_2)[\psi/q] &= (\phi_1[\psi/q]) \vee (\phi_2[\psi/q]) \\
(\phi_1 \longrightarrow \phi_2)[\psi/q] &= (\phi_1[\psi/q]) \longrightarrow (\phi_2[\psi/q]) \\
(\phi_1 \longleftrightarrow \phi_2)[\psi/q] &= (\phi_1[\psi/q]) \longleftrightarrow (\phi_2[\psi/q])
\end{aligned}$$

Warum fordern wir eigentlich nicht, dass $p \in \text{atoms}(\phi)$? Es stellt sich heraus, dass das gar nicht notwendig ist. Es gilt:

$$p \notin \text{atoms}(\phi) \Longleftrightarrow \phi[\psi/p] = \phi \text{ für } \psi \in Prop, \psi \neq p$$

Wenn p also nicht in ϕ vorkommt, dann ist die Substitition von p in ϕ die Identität. Der Beweis ist eine Induktion über der Struktur von ϕ.

So wie die Gleichheit auf $\mathbb{B}$ das semantische Gegenstück für die Äquivalenz $\longleftrightarrow$ ist, können wir auch für die Substitution ein semantisches Gegenstück angeben. Die Semantik einer Formel $\phi[p/\psi]$ in der das Atom p durch eine Formel *psi* ersetzt wird, ist die Semantik der Formel ϕ unter einer Belegung v', in der p auf die Semantik von ψ unter der Belegung v belegt wird.

Für eine partielle Funktion $f : A \rightharpoonup B$ und $a \in A$, $b \in B$ schreiben wir $f[a \mapsto b]$ für die Funktion, welche a auf b und alles andere wie f abbildet:

$$(f[a \mapsto b])(a_0) = \begin{cases} b & a = a_0 \\ f(a_0) & a \neq a_0 \end{cases}$$

Bevor wir unsere wesentlichen Ergebnisse formulieren, üben wir diese Konzepte anzuwenden:

Übung 1.5 *Gegeben die Formel* $\rho = (A \longrightarrow \neg B \wedge C) \vee (C \vee \neg A)$*, die Belegung* $v = \langle A \mapsto 1, B \mapsto 0, C \mapsto 1\rangle$ *und* $\sigma = (A \longrightarrow B \longleftrightarrow \neg B \wedge C)$.

(i) *Berechne* $\phi = \rho[\sigma/A]$.

(ii) *Berechne* $t_1 = [\![\phi]\!]_v$.

(iii) *Berechne* $s = [\![\sigma]\!]_v$.

(iv) *Berechne* $w = v[A \mapsto s]$.

(v) *Berechne* $t_2 = [\![\rho]\!]_w$.

Jetzt können wir unser Lemma formulieren:

Lemma 1.3 (Substitutionslemma) *Für alle Formeln* $\phi, \psi \in Prop$ *und Belegungen* v *gilt:*

$$[\![\phi[\psi/p]]\!]_v = [\![\phi]\!]_{v[p \mapsto [\![\psi]\!]_v]}$$

Beweis. Durch Induktion über der Struktur von ϕ.

Die Induktionsbasis sind die zwei Fälle $\phi = q$ mit $q \in P$ und $\phi = \bot$. Für $\phi = q$ gibt es zwei Fälle, nämlich $p = q$ oder $p \neq q$. Für den ersten Fall haben wir

$$[\![p[\psi/p]]\!]_v = [\![\psi]\!]_v = v[p \mapsto [\![\psi]\!]_v](p) = [\![p]\!]_{v[p \mapsto [\![\psi]\!]_v]},$$

und für den zweiten Fall

$$[\![q[\psi/p]]\!]_v = [\![q]\!]_v = v(q) = v[p \mapsto [\![\psi]\!]_v](q) = [\![q]\!]_{v[p \mapsto [\![\psi]\!]_v]}.$$

Für $\phi = \bot$ gilt einfacher $[\![\bot]\!]_w = 0$ für alle Belegungen w, und $[\![\bot[\psi/q]]\!]_v = [\![\bot]\!]_v$.

Wir zeigen den Induktionsschritt für die Negation, die anderen Fälle sind analog:

$$[\![\neg\phi[\psi/p]]\!]_v = 1 - [\![\phi[\psi/p]]\!]_v = 1 - [\![\phi]\!]_{v[p \mapsto [\![\psi]\!]_v]} = [\![\neg\phi]\!]_{v[p \mapsto [\![\psi]\!]_v]}$$

Der zweite Schritt ist hier die Induktionsvoraussetzung. □

Damit können wir folgendes Theorem zeigen:

Theorem 1.4 (Substitutionstheorem) *Wenn* $\models \phi_1 \longleftrightarrow \phi_2$, *dann* $\models \psi[\phi_1/p] \longleftrightarrow \psi[\phi_2/p]$.

Beweis. $\models \psi[\phi_1/p] \longleftrightarrow \psi[\phi_2/p]$ gdw. $[\![\psi[\phi_1/p]]\!]_v = [\![\psi[\phi_2/p]]\!]_v$; ferner wenn $\models \phi_1 \longleftrightarrow \phi_2$ dann $[\![\phi_1]\!]_v = [\![\phi_2]\!]_v$ (jeweils für beliebige v). Jetzt können wir rechnen:

$$[\![\psi[\phi_1/p]]\!]_v = [\![\psi]\!]_{v[p\mapsto[\![\phi_1]\!]_v]} = [\![\psi]\!]_{v[p\mapsto[\![\phi_2]\!]_v]} = [\![\psi[\phi_2/p]]\!]_v$$

Der erste und letzte Schritt folgen aus dem Substitutionslemma. □

1.3.8 Boolsche Algebren

Jetzt können wir rechnen. Dazu brauchen wir aber erstmal eine Handvoll von Äquivalenzen, von denen wir ausgehen können. Ein Beispiel in der Algebra sind die Gruppenaxiome, aus denen sich dann eine reichhaltige Theorie ergibt; leider sind Aussagen keine Gruppen, sondern sogenannte Boolsche Algebren.

Theorem 1.5 *Es gelten folgende Äquivalenzen für* $\wedge, \vee$ *und* $\neg$*:*

$(\phi \wedge \psi) \wedge \sigma \approx \phi \wedge (\psi \wedge \sigma)$	$(\phi \vee \psi) \vee \sigma \approx \phi \vee (\psi \vee \sigma)$	*(Assoziativität)*
$\phi \wedge \psi \approx \psi \wedge \phi$	$\phi \vee \psi \approx \psi \vee \phi$	*(Kommutativität)*
$\phi \wedge (\psi \vee \sigma) \approx (\phi \wedge \psi) \vee (\phi \wedge \sigma)$	$\phi \vee (\psi \wedge \sigma) \approx (\phi \vee \psi) \wedge (\phi \vee \sigma)$	*(Distributivität)*
$\neg(\phi \wedge \psi) \approx \neg\phi \vee \neg\psi$	$\neg(\phi \vee \psi) \approx \neg\phi \wedge \neg\psi$	*(De Morgan)*
$\phi \wedge \phi \approx \phi$	$\phi \vee \phi \approx \phi$	*(Idempotenz)*
	$\neg\neg\phi \approx \phi$	*(Doppelnegation)*
$\phi \wedge \bot \approx \bot$	$\phi \vee \bot \approx \phi$	*(Falsum)*

Die meisten dieser Eigenschaften folgen aus entsprechenden Eigenschaften der semantischen Funktionen. So gilt $(\phi \wedge \psi) \wedge \sigma \approx \phi \wedge (\psi \wedge \sigma)$ weil $\min(x, \min(y,z)) = \min(\min(x,y),z)$ oder $\neg\neg\phi \approx \phi$, weil $1-(1-x)) = x$. Etwas schwieriger sind die Distributivitätsgesetze, hier muss man zeigen dass $\min(x, \max(y,z)) = \max(\min(x,y), \min(x,z))$ (durch eine Unterscheidung aller sechs Fälle). Die letzten beiden folgen aus $\min(x,0) = 0$ und $\max(x,0) = x$, für $x \geq 0$.

Wie man sieht, fehlt hier noch ein neutrales Element für $\wedge$. Was kann das sein, und wie definieren wir es?

Definition 1.11 (Verum) *Wir definieren True ("Verum") als*

$$\top \longleftrightarrow \neg\bot$$

Damit folgt $[\![\top]\!]_v = 1$, und es gilt

$$\phi \wedge \top \approx \phi \qquad\qquad \phi \vee \top = \top$$

$$\models \phi \text{ gdw } \phi \approx \top$$

Die ersten beiden Äquivalenzen folgen aus $\min(x,1) = x$ und $\max(x,1) = 1$ für $x \leq 1$. Die dritte Gleichung folgt aus $[\![\phi]\!]_v = 1$ wenn $\models \phi$.

Übung 1.6 *Zeige durch Umformen, dass* $\phi \wedge (\phi \vee \psi)$ *eine Tautologie ist.*

Weitere Gleichungen ([2, Theorem 1.3.4] zeigen, wie wir Operatoren aus anderen definieren können:

$$(\phi \longleftrightarrow \psi) \approx (\phi \longrightarrow \psi) \wedge (\psi \longrightarrow \phi) \tag{1.7}$$
$$(\phi \longrightarrow \psi) \approx (\neg\phi \vee \psi) \tag{1.8}$$
$$\phi \vee \psi \approx \neg(\neg\phi \longrightarrow \psi) \tag{1.9}$$
$$\phi \wedge \psi \approx \neg(\neg\phi \vee \neg\psi) \tag{1.10}$$
$$\phi \vee \psi \approx \neg(\neg\phi \wedge \neg\psi) \tag{1.11}$$
$$\neg\phi \approx (\phi \longrightarrow \bot) \tag{1.12}$$
$$\bot \approx (\phi \wedge \neg\phi) \tag{1.13}$$

Beweise:

- (1.7): nach Definition von $\longleftrightarrow$.
- (1.8): nach Wahrheitstabellen beider Seiten.
- (1.9): $\phi \vee \psi \approx \neg(\neg\phi) \vee \psi \approx \neg\phi \longrightarrow \psi$
- (1.10): $\phi \wedge \psi \approx \neg(\neg(\phi \wedge \psi)) \approx \neg(\neg\phi \vee \neg\psi)$
- (1.11): $\phi \vee \psi \approx \neg(\neg(\phi \vee \psi)) \approx \neg(\neg\phi \wedge \neg\psi)$
- (1.12): $\neg\phi \approx \neg(\phi \wedge \top) \approx \neg\phi \vee \neg\top \approx \neg\phi \vee \bot \approx \phi \longrightarrow \bot$
- (1.13): $\bot \approx \neg\top \approx \neg(\neg\phi \vee \phi) \approx \neg\neg\phi \wedge \neg\phi \approx \phi \wedge \neg\phi$

□

Wie man hier sieht haben wir auf der linken Seite der Äquivalenz eine Formal $\phi \square \psi$, wobei $\square$ auf der rechten Seite der Äquivalenz nicht auftaucht. Durch wiederholte Anwendung dieser Umformung können wir jede Formel in eine überführen, in der der Operator $\square$ nicht mehr enthalten ist. ($\square$ kann hier für $\longleftrightarrow$, $\longrightarrow$, $\wedge$, $\vee$ stehen.)

1.3.9 Beispiele für Umformungen

Wir können $\models \sigma \longleftrightarrow \tau$ zeigen, in dem wir $\sigma \approx \tau$ durch Umformung zeigen, und wir können $\models \sigma$ zeigen, in dem wir $\sigma \approx \top$ durch Umformung zeigen.

- $\models \phi \longrightarrow \psi \longleftrightarrow \neg\psi \longrightarrow \neg\phi$:

$$\begin{aligned}\phi \longrightarrow \psi &\approx \neg\phi \vee \psi\\ &\approx \psi \vee \neg\phi\\ &\approx \neg(\neg\psi) \vee \neg\phi\\ &\approx \neg\psi \longrightarrow \neg\phi\end{aligned}$$

- $\models \phi \longrightarrow \psi \longrightarrow \phi$

$$\begin{aligned}\phi \longrightarrow \psi \longrightarrow \phi &\approx \neg\phi \vee (\neg\psi \vee \phi)\\ &\approx (\neg\phi \vee \phi) \vee \neg\psi\\ &\approx \top \vee \neg\psi \approx \top\end{aligned}$$

Übung 1.7 *Für die ausschließende Disjunktion gibt es zwei mögliche Formulierungen durch andere Operatoren:*

$$\phi \dot{\vee} \psi \longleftrightarrow (\neg(\phi \longleftrightarrow \psi)) \tag{1.14}$$
$$\phi \dot{\vee} \psi \longleftrightarrow (\phi \vee \psi) \wedge (\neg\phi \vee \neg\psi) \tag{1.15}$$

Zeige durch Umformen, dass diese äquivalent sind.

Vorlesung vom 25.04.2023: Aussagenlogik IV

1.3.10 Kernsprachen

Übung 1.8 (Aufwärmübung) *Zeige durch Umformen:*

$$\models \phi \longrightarrow (\psi \longrightarrow \sigma) \longleftrightarrow \phi \wedge \psi \longrightarrow \sigma$$

Aus den Gleichungen (1.7) bis (1.13) undTheorem 1.5 folgt, dass sich die Aussagenlogik nur mit folgenden Mengen von Operatoren definieren lässt:

$$\{\vee, \neg\} \quad \{\longrightarrow, \neg\} \quad \{\wedge, \neg\} \quad \{\longrightarrow, \bot\} \tag{1.16}$$

Die anderen Operatoren werden dann jeweils als abgeleitete Konnektive definiert, wie wir das mit Äquivalenzoperator gemacht haben. Warum machen wir das nicht? Es würde das Beweisen sehr erschweren, weil wir immer erst alle Operatoren durch die repräsentierenden ersetzen müssen, dadurch werden die Terme (und der Suchraum) sehr groß. (Schließlich programmieren wir ja auch nicht in minimalen Programmiersprachen wie SUBLEQ.)

Wie beweisen wir das? Man kann es sich einfach machen: “einfach die Äquivalenzen solange anwenden wie es geht”. Das ist allerdings nicht präzise genug. Man muss dazu dieses “Anwenden solange es geht” präziser machen. Wir zeigen eine etwas vereinfachte Ausage, die wir später brauchen:

Lemma 1.6 *Zu jeder aussagenlogischen Formel $\phi \in Prop$ gibt es eine äquivalente Formal $\phi' \in Prop$, die nur Atome und die Operatoren $\wedge, \vee, \neg$ enthält.*

Beweis. Sei $Prop'$ die Menge aller aussagenlogischen Formlen, die nur Atome und die Operatoren $\wedge, \vee, \neg$ enthalten. Der Beweis ist über der *Struktur* von ϕ:

- $\phi \equiv p \in P$: Induktionsbasis, $p \in Prop'$.
- $\phi \equiv \neg\phi_1$: Nach Induktionsvoraussetzung gibt es $\phi_1' \in Prop'$ mit $\phi_1 \approx \phi_1'$, dann ist $\phi' \stackrel{def}{=} \neg\phi_1' \in Prop'$ und $\phi' \equiv \phi$.
- $\wedge, \vee$: Analog.
- $\phi \equiv \phi_1 \longrightarrow \phi_2$: Nach Induktionsvoraussetzung gibt es $\phi_1', \phi_2' \in Prop'$ mit $\phi_1 \approx \phi_1', \phi_2 \approx \phi_2'$. Dann ist $\phi' \stackrel{def}{=} \neg\phi_1' \vee \phi_2' \approx \phi_1' \longrightarrow \phi_2' \approx \phi_1 \longrightarrow \phi_2 \approx \phi$ wie gefordert.
- $\phi \equiv \phi_1 \longleftrightarrow \phi_2$: Analog.

□

1.3.11 Der Sheffer-Strich

Man braucht nicht unbedingt zwei Operatoren, es reicht ein einziger: die Wahrheitstabelle des Sheffer-Strich ist definiert als

$\mid$	0	1
0	1	0
1	0	0

Offensichtlich ist das die negierte Konjunktion (NAND), und es gilt

$$\models \phi \mid \psi \longleftrightarrow \neg(\phi \wedge \psi)$$
$$\models \neg\phi \longleftrightarrow \phi \mid \phi$$
$$\models \phi \wedge \psi \longleftrightarrow (\phi \mid \psi) \mid (\phi \mid \psi)$$

Da wir wissen, dass $\{\neg, \wedge\}$ alle anderen Operatoren ausdrücken kann (1.16), reicht also auch der Sheffer-Strich alleine.

Wir können uns beliebig weitere Operatoren ausdenken und anhand ihrer Wahrheitstabelle definieren. Können wir diese auch mit den Operatoren aus (1.16), oder äquivalent dem Sheffer-Strich, ausdrücken? Ja, das geht, aber wir müssen das erstmal beweisen.

Theorem 1.7 (Functional completeness) *Sei $\$$ ein n-stelliger Operator, definiert durch die Auswertungsfunktion $f_\$$ (bspw. notiert als Wahrheitstabelle), i.e. $[\![\$(p_1,\ldots,p_n)]\!]_v = f_\$(v(p_1),\ldots,v(p_n))$ (mit Atomen $p_1,\ldots,p_n$). Dann gibt es eine Aussage $\tau \in Prop$, die lediglich Atome $p_1\ldots,p_n$, $\wedge$ und $\neg$ enthält, so dass $\models \tau \longleftrightarrow \$(p_1,\ldots,p_n)$.*

Beweis. Induktion über n. Siehe [2, Theorem 1.3.6] (dort wird die funktionale Vollständigkeit von $\vee$, $\neg$ gezeigt, aber das ist äquivalent zu unserer Behauptung, da wird $\vee$ durch $\wedge$ und $\neg$ ausdrücken können). □

Wir sagen, dass der Sheffer-Strich, die Menge $\{\vee, \neg\}$ oder die anderen Mengen von Operatoren aus (1.16) *funktional vollständig* (functionally complete) sind.

Übung 1.9 *Wir haben bereits gesehen, wie wir $\neg$ und $\wedge$ mit dem Sheffer-Strich ausdrücken; drücke auch alle anderen Operatoren damit aus.*

1.4 Beweisverfahren

Dieses ganze Umformerei oben ist ja etwas planlos. Wir haben eine ungefähre Idee, wie wir eine *gegebene* Formel beweisen, aber das ist noch keine allgemeines Rezept.

1.4.1 Normalformen

Kommen wir noch einmal auf die quadratischen Gleichungen vom Anfang zurück. Wie lösen wir eine *beliebige* quadratische Gleichungen wie

$$(2x+7)x+19 = 15x+13?$$

Die Antwort ist, wir bringen die Gleichung in eine Normalform:

$$\begin{aligned}(2x+7)x-19 &= 15x-9\\ 2x^2+7x-15x-19+9 &= 0\\ 2x^2-8x-10 &= 0\\ x^2-4x-5 &= 0\end{aligned}$$

Diese Normalform können wir dann schematisch lösen, und erhalten $x = -\frac{4}{2} \pm \sqrt{\frac{p^2}{4}+5} = 2 \pm 3$, also $x = -1 \vee x = 5$.

Normalformen für quadratische (und allgemeiner polynomiale) Gleichungen haben die Form $ax^2+bx+c=0$ (oder $\sum_{i=0}^{n} a_i x^i = 0$). Für Aussagenlogik gibt es etwas ähnliches. Wir können nämlich Lemma 1.6 anwenden, und unsere Aussagen in eine ähnliche Normalenform bringen.

Definition 1.12 (Literale und Normalformen) *Ein* Literal *hat die Form p oder $\neg p$ für $p \in P$.*

Ein Ausdruck $\phi \in Prop$ ist in konjunktiver Normalform *(conjunctive normal form, CNF) wenn ϕ die Form*

$$\phi = \phi_1 \wedge \phi_2 \wedge \cdots \wedge \phi_n$$

und jedes ϕ_i hat die Form

$$\phi_i = \psi_{i,1} \vee \psi_{i,2} \vee \cdots \vee \psi_{i,m_i}$$

hat, wobei $\psi_{i,j}$ ein Literal ist.

Ein Ausdruck $\phi \in Prop$ ist in konjunktiver Normalform *(conjunctive normal form, CNF) wenn ϕ die Form*

$$\phi = \phi_1 \vee \phi_2 \wedge \cdots \vee \phi_n$$

und jedes ϕ_i hat die Form

$$\phi_i = \psi_{i,1} \wedge \psi_{i,2} \wedge \cdots \wedge \psi_{i,m_i}$$

hat, wobei $\psi_{i,j}$ ein Literal ist.

Mit anderen Worten, konjunktive Normalformen sind Konjunktionen von Disjunktionen von Literalen, und disjunktive Normalformen sind Disjunktionen von Konjunktionen von Literalen.

Für die verallgemeinerte Konjunktion und Disjunktion nutzen wir folgende Schreibeweisen:

$$\bigwedge_{i=1,\ldots,n} \phi_i = \phi_1 \wedge \phi_2 \wedge \cdots \wedge \phi_n$$
$$\bigvee_{i=1,\ldots,n} \phi_i = \phi_1 \vee \phi_2 \wedge \cdots \vee \phi_n$$

Theorem 1.8 *Zu jeder aussagenlogischen Formel $\phi \in Prop$ gibt es eine Formel $\rho \in Prop$ in CNF, so dass $\models \phi \longleftrightarrow \rho$, und $\sigma \in Prop$ in DNF so das $\models \phi \longleftrightarrow \sigma$.*

Beweis. Der Beweis ist ähnlich wie Lemma 1.6 durch strukturelle Induktion über ϕ. Für die Negation ist es wichtig, dass man beide Teile des Theorems gleichzeitig beweist, nicht separat, weil aus der Induktionsvoraussetzung "es gibt eine CNF" der Schluss "es gibt eine DNF" und andersherum folgt. □

Übung 1.10 *Ist die CNF/DNF eindeutig?*

Ein Ausdruck wird also durch wiederholte Anwendung der Regeln in CNF/DNF überführt. Leider ist das sehr aufwändig, und die Formel wird sehr groß. Eine andere Möglichkeit ist, die CNF/DNF aus der Wahrheitstabelle abzulesen. Wir stellen das an einem Beispiel dar.

Gegeben die Formel $A = A_3 \vee \neg A_1) \longrightarrow (A_2 \longleftrightarrow A_3)$ mit folgender Wahrheitstabelle:

A_1	A_2	A_3	ϕ
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Hier können wir die DNF direkt ablesen. Die Formel ist wahr gdw. einer der Zeilen in denen der letzte Eintrag 1 ist zutrifft:

$$D = (\neg A_1 \wedge \neg A_2 \wedge \neg A_3) \vee (A_1 \wedge \neg A_2 \wedge \neg A_3) \vee (\neg A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge \neg A_3) \vee (A_1 \wedge A_2 \wedge A_3)$$

Die CNF ergibt sich aus den anderen Zeilen. Die Formel ist wahr, wenn sie nicht falsch ist, d.h. wenn keine der Zeilen in denen der letzte Eintrag 0 ist, zutrifft, und dafür muss mindestens eine der Variabel vorne mit 0 belegt sein:

$$C = (A_1 \vee A_2 \vee \neg A_3) \wedge (A_1 \vee \neg A_2 \vee A_3) \wedge (\neg A_1 \vee A_2 \vee \neg A_3)$$

Vorlesung vom 27.04.2023: Aussagenlogik V

1.4.2 Erfüllbarkeit und SAT

Die Frage, ob eine gegebene aussagenlogische Formel $\phi \in \mathit{Prop}$ erfüllbar ist (d.h. gibt es eine Belegung v der Atome in ϕ, so dass $[\![\phi]\!]_v = 1$, ist das Erfüllbarkeitsproblem (SAT). Es ist aus zwei Gründen prominent: zum einen lassen sich viele praktische Probleme aus dem Schaltkreisentwurf und der Programmverifikation auf Erfüllbarkeitsprobleme abbilden und damit lösen, und zum anderen war das Erfüllbarbeitsproblem das erste (soweit ich weiß), dessen NP-Vollständigkeit nachgewiesen wurde:

Theorem 1.9 (Cook) *Das Erfüllbarkeitsproblem für aussagenlogische Formeln ist NP-vollständig.*

NP-vollständig heißt zum einen, dass SAT NP-hart ist, *i.e.* nicht besser als NP lösbar, und darüber hinaus, dass jedes Problem in NP kann in polynomialier Zeit in ein SAT-Problem überführt werden kann. Das macht SAT zu einem beliebten Theorem, um NP-Vollständigkeit zu beweisen, indem man das Problem zu SAT reduziert; man muss dazunur die Problemstellung in Aussagenlogik formulieren (was wesentlich einfacher ist als es direkt in das Wortproblem einer Turingmaschine zu übersetzen).

Heutzutage gibt es eine Reihe von Werkzeugen, die SAT lösen (SAT-Solver, wie Minisat oder Chaff). Richtig mächtig sind Kombinationen von SAT mit sogenannten Hintegrundtheorien, wie beispielsweise

lineare Ungleichheiten über natürlichen Zahlen (*satisfiability modulo theory*, SMT), wie sie Werkzeugen wie Z3, Yices, CVC oder Alt-Ergo implementiert sind. Moderne SAT-Solver können problemlos Instanzen mit mehreren Millionen Klauseln lösen.

Das typische Eingabeformat für einen SAT-Solver ist CNF, kodiert im DIMACS-Format; SMT-Solver lesen Eingaben im smtlib-Format, und übersetzen intern in CNF (wenn nötig).

SAT-Solving geht von einer Formel ϕ in CNF aus:

$$\phi = \bigwedge_{i=1}^{n} \bigvee_{i=1}^{m_n} L_{i,j}$$

Die einzelnen Glieder der Konjunktionen nennen wir *Klauseln*. Im folgenden wird sowohl die Konjunktion als eine Menge von Klauseln, und die einzelnen Klauseln als eine Menge von Literalen betrachtet (weil die Reihenfolge hier Banane ist). Das erlaubt uns Dinge zu schreiben wie $L \in \phi_i$ wenn eine Klausel ϕ_i das Literal L enthält, oder $\phi_1 \cup \phi_2$ für die Vereinigung von Klauselmengen (was der Konjunktion entspricht). Eine Klausel mit einem einzigen Literal ist eine *Unit-Klausel*.

Um eine Formel ϕ wie oben zu erfüllen, müssen alle Klauseln erfüllt sein. Das machen wir, indem wir Stück für Stück alle Atome $a \in \text{atoms}(\phi)$ belegen. Das Problem ist, da gibt es sehr viele Möglichkeiten — nämlich exponentiell viele.

Trotzdem ist SAT NP-vollständig, also besser als exponentiell. Es hat die "klassische" Struktur von NP-vollständigen Problemen: einen sehr großen (expontiellen) Suchraum, aber eine effiziente Funktion zur Überprüfung (polynomial oder besser; hier ist das die Auswertung der Klauselmenge unter einer gegebenen Belegung).[1]

Der klassische Algorithmus nach Davis-Putnam-Logemann-Loveland (DPLL) ist ein Backtracking-Algorithmus und funktioniert wie folgt: wähle ein beliebiges Atom $a \in \phi$. Wähle eine Belegung $a \mapsto 1$ und propagiere die Belegung durch alle Klauseln in ϕ. Wenn dabei eine unerfüllbare Klausel entsteht, brechen wir diesen Versuch ab, ansonsten geht der Versuch mit dem nächsten Literal weiter. Beim Abbruch macht das Backtracking mit der Belegung $a \mapsto 0$ weiter.

Eine einfache Optimierung besteht darin, vor der Auswahl eines Literals alle Unit-Klauseln zu propagieren. Dazu wird für jede Unit-Klausel p die Belegung $p \mapsto 1$, und für $\neg p$ die Belegung $p \mapsto 0$ auf die Klauselmenge angewandt. Wenn durch die Belegung ein Literal zu 1 auswertet, dann wird diese Klausel gestrichen (weil $1 \vee A \approx 1$ ist), und wenn in einer Klausel ein Literal zu $\bot$ auswertet, wird dieses aus der Klausel gestrichen (weil $0 \vee A \approx A$).

Hier ist ein einfaches Beispiel. Gegeben folgende Formel in CNF:

$$\phi = (A \vee B) \wedge (B \vee C \vee \neg D) \wedge (\neg A \vee B \vee C) \wedge (A \vee \neg B)$$

[1] Eine andere wichtige Meta-Eigenschaft von NP-vollständigen Problemen, die aus dieser Struktur folgt, ist dass wir sie mit guten Heuristiken oft erstaunlich gut lösen können. Das bedeutet auch, dass für ein interessantes NP-vollständiges Problem die Suche nach guten Heuristiken ein lohnenswertes Unterfangen ist.

Wir berechnen eine Belegung für die Atome A, B, C, D:

	$A \vee B$	$B \vee C \vee \neg D$	$\neg A \vee B \vee C$	$A \vee \neg B$
$A \mapsto 0$	$0 \vee B$	$B \vee C \vee \neg D$	$0 \vee B \vee C$	$0 \vee \neg B$
	B	$B \vee C \vee \neg D$	$B \vee C$	$\neg B$
$UP : B \mapsto 1$	1	$1 \vee C \vee \neg D$	$1 \vee C$	0↯
$A \mapsto 1$	$1 \vee B$	$B \vee C \vee \neg D$	$0 \vee B \vee C$	$1 \vee \neg B$
	1	$B \vee C \vee \neg D$	$B \vee C$	1
$B \mapsto 0$	1	$0 \vee C \vee \neg D$	$0 \vee C$	1
	1	$C \vee \neg D$	C	1
$UP : C \mapsto 1$	1	$1 \vee \neg D$	1	1
	1	1	1	1

Die Belegung ist also $\sigma = \langle A \mapsto 1, B \mapsto 0, C \mapsto 0 \rangle$. D wird hier gar nicht belegt; das bedeutet, die Formel wird mit σ unter allen (beiden) Belegungen von D wahr.

Man beachte, wie die *unit propagation* (UP) die Berechnung deutlich beschleunigt.

Übung 1.11 *Gegeben folgende Formelmenge:*

$$\Psi = \{A \wedge B \longrightarrow C, D \longrightarrow B, D \longrightarrow B \vee C\}$$

- *Berechne die Klauselmenge für Ψ: bilde die Konjunkt aller Elemente, $\phi = \bigwedge \Psi$, und berechne CNF.*
- *Berechne eine erfüllende Belegung mit dem Algorithmus von oben.*

1.4.3 Resolution

Erfüllbarkeit ist bis jetzt eine rein semantische Eigenschaft. Intuitiv sollte eine Formelmenge ϕ erfüllbar sein, (genau dann) wenn sie konsistent ist, oder andersherum: eine Formelmenge Ψ ist inkonsistent, wenn aus Ψ der Widerspruch folgt, $\Psi \models \bot$; und eine Formelmenge sollte nicht erfüllbar sein gdw sie inkonsistent ist.

Wir werden jetzt einen syntaktischen Kalkül einführen, der es uns erlaubt aus einer Formelmenge andere Formeln zu schließen, mit dem Ziel, den Widerspruch herzuleiten. Damit hätte man dann gezeigt, dass $\Psi \models \bot$ wie oben. Damit kann man auch zeigen, ob die Formel ϕ eine Tautologie ist (indem man $\neg\phi$ zum Widerspruch führt), oder ob ϕ aus Ψ folgt, $\Psi \models \phi$ (indem man $\Psi \wedge \neg\phi$ zum Widerspruch führt).

Resolution basiert auf folgender Idee: wenn wir Klauseln $A \vee P$ und $B \vee \neg P$ haben, dann ist die Belegung von P eigentlich irrelevant, weil beide gelten, wenn $A \vee B$ gilt. Mit anderen Worten, Literale die in einer Klausul positiv (nicht negiert) und einer negiert auftreten, können wir elminieren.

Wir fassen das formal. Vorab etwas Notation: für ein Literal L (das entweder die Form p oder $\neg p$ für ein Atom $p \in P$ hat) ist $\bar{L}$ definiert als $\bar{(p)} = \neg p$ und $\bar{(\neg p)} = p$.

Gegeben drei Klauseln ϕ_1, ϕ_2, ϕ_3, dann ist ϕ_3 die *Resolvente* von ϕ_1 und ϕ_2, wenn

- es ein Literal L gibt, so dass $L \in \phi_1$ und $\bar{L} \in \phi_2$, und
- ϕ_3 von der Form $\phi_1 \setminus \{L\} \cup \phi_2 \setminus \{\bar{L}\}$ ist.[2]

[2] Diese Mengennotation ist korrekt, weil wir die Klauseln als Mengen (von Literalen) betrachten.

Wir resolven immer über einem Literal L. Gibt es mehrere $L_1, \ldots, L_N$ in ϕ_1, ϕ_2 welche die Voraussetzung erfüllen, gibt es auch mehrere Resolventen. Welches sind alle Resolventen der Formeln $\phi_1 = A \vee \neg B \vee X \vee D$ und $\phi_2 = \neg A \vee B \vee Y \vee \neg D$?

Lemma 1.10 (Resolutionslemma) *Sei F eine Klauselmenge, mit $\phi_1, \phi_2 \in F$ und ϕ_3 eine Resolvente von ϕ_1 und ϕ_2. Dann sind F und $F \cup \{\phi_3\}$ äquivalent.*

Beweis. Siehe [1, Kapitel 1.5, S. 40]. □

Der Resultionsalgorithmus fügt solange Resolventen hinzu, bis keine neuen mehr zu finden sind. Formal: für eine Klauselmenge F

$$\mathrm{Res}(F) = F \cup \{\phi_3 \mid \phi_1, \phi_2 \in F, \phi_3 \text{ ist Resolvente von } \phi_1 \text{ und } \phi_2\} \tag{1.17}$$

$$\mathrm{Res}^0(F) = F \tag{1.18}$$

$$\mathrm{Res}^{n+1}(F) = \mathrm{Res}(\mathrm{Res}^n(F) \tag{1.19}$$

$$\mathrm{Res}^*(F) = \bigcup_{0 \leq i} \mathrm{Res}^i(F) \tag{1.20}$$

Wichtig ist folgende Eigenschaft des Algorithmus: wenn wir bei der Resolution irgendwann durch Resolution eine leere Klausel finden, dann ist F nicht erfüllbar.

Theorem 1.11 (Resolutionssatz) *Eine Klauselmenge F ist unerfüllbar genau dann wenn $\emptyset \in \mathrm{Res}^*(F)$.*

Beweis. Dieser Satz ist nicht ganz so trivial zu beweisen, insbesondere die Vollständigkeit (Richtung von links nach rechts); d.h. wenn die Formelmenge unerfüllbar ist finden wir auch immer eine leere Klausel. Siehe [1, Kapitel 1.5, S. 41] □

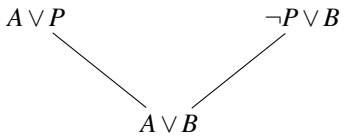
Resolution ist zum einen ein mächtiges Beweisverfahren, welches auch für die Prädikatenlogik funktioniert (im Gegensatz zum SAT-Solving, welches nur für die Aussagenlogik funktioniert), und zum anderen dient es als Ausführungsmodell bei der Programmiersprache Prolog. In Prolog sind Programme Klauseln in spezieller Form: eine *Hornklausel* ist eine Klausel $\phi = \{L_1, \ldots, L_n\}$ wobei es höchstens ein positives L_i mit $L_i = p_i$ (und alle anderen $L_j = \neg p_j$) gibt. Eine Hornklausel $\{\neg p_1, \ldots, \neg p_n, q\}$ ist offensichtlich äquivalent zu einer Implikation der Form $p_1 \wedge \ldots p_n \longrightarrow q$.

Übung 1.12 *Betrachte folgende Klauselmenge:*

$$F = \{\neg B \wedge A, B \vee A, \neg A \vee \neg C, \neg B \vee C, B \vee C\}$$

Können wir die Unerfüllbarkeit zeigen, indem wir mittels Resolution die leere Klausel herleiten?

Resolution wird gerne graphisch dargestellt:



Das wird bei längeren Resolutionsbeweisen schwierig, deshalb nutzen wir eine lineare Schreibweise, in der wir die Klauseln untereinander schreiben, und jeder eine fortlaufende Nummer geben. Bei neuen Resolventen wird notiert, welche Klausel darüber resolviert wurden.

Hier ist ein ganz einfaches Beispiel. Gegeben folgende Klauselmenge:

$$F = \{A, \neg A \vee B, \neg B\}$$

Res^0	1:	A	
	2:	$\neg A \vee B$	
	3:	$\neg B$	
Res^1	4:	B	$\text{Res}(1,3)$
	5:	$\neg A$	$\text{Res}(2,3)$
Res^2	6:	$\bot$	$\text{Res}(1,5)$
	7:	$\bot$	$\text{Res}(3,5)$

Im nullten Schritt ist die Klauselmenge Res^0 die initiale Klauselmenge (hier F). Danach resolvieren wir alle Klauseln miteinander, und erhalten hier zwei neue Klauseln 4 und 5. Im nächsten Schritte versuchen wir alle "alten" Klauseln mit den neuen zu resolvieren, und erhalten weitere neue Klauseln— in diesem Fall zwei leere Klauseln, und wir können die Resolution beenden, weil die Inkonsistenz nachgewiesen wurde. Hinter jeden neu hinzu gekommenen Klausel vermerken wir, von welchen Klauseln sie die Resolvente ist; falls das nicht eindeutig ist (s. unten), dann vermerken wir auch das Literal, über dem resolviert wurde.

Etwas allgemeiner müssen wir initial (Stufe 1) alle Klauseln in Res^0 miteinander resolvieren; die entstehenden Resolventen bilden dann die Menge Res^1. Im $i+1$-ten Schritt ($i \geq 1$) resolvieren wir dann die Klauseln aus Res^{i-1} mit denen in Res^i neu hinzugekommenen.

Nicht jede Resolvente ist neu; mit zunehmender Dauer des Algorithmus werden immer mehr Resolventen erzeugt, die der Algorithmus schon "gesehen" hat. Hier ist ein längeres Beispiel für eine Resolution. Gegeben seien folgende Klauselmenge:

$$F = \{\neg X \vee A, \neg A \vee Y, \neg X \vee Y, Y \vee X, X \vee A\}$$

Der Resolutionsalgorithmus berechnet Res^i für $i = 1,2,3$ wie in Tabelle 1.1; im vierten Schritt erhalten wir direkt einen Widerspruch und terminieren.

Eine Beobachtung aus Tabelle 1.1 ist, dass eine Resolution mit Klauseln der Form $P \vee \neg P$ keine neuen Klauseln hervorbringt: sei die andere Klausel $P \vee l$ (mit l einem Literal), dann ist die Resolvente wieder $\{P, \neg P\} \setminus \{\neg P\} \cup \{P, l\} \setminus \{P\} = \{P, l\}$ (analog für Resolution mit $\neg P \vee l$); diese Klauseln brauchen wir also eigentlich nicht weiter zu betrachten.

Vorlesung vom 02.05.2023: Natürliches Schließen I

1.5 Natürliches Schließen

Übung 1.13 (Aufwärmübung) *Formalisiert folgende Aussagen:*

> *"Wenn ihr intelligent seid, und für die Prüfung lernt, dann werdet ihr auch die Prüfung bestehen. Wenn ihr diesen Kurs gewählt habt, dann seit ihr schon mal intelligent. Wenn ihr hier seit, dann habt ihr wohl diesen Kurs gewählt, und ihr seit offensichtlich hier. Lernen tut ihr auch. Daraus folgt, dass ihr die Prüfung bestehen werdet."*

Die Atome sollten sein:

Res^0	1	:	$\neg X \vee A$	
	2	:	$\neg A \vee Y$	
	3	:	$\neg X \vee Y$	
	4	:	$Y \vee X$	
	5	:	$X \vee A$	
Res^1	6	:	$X \vee \neg Y$	$\text{Res}(1,2)$
	7	:	$A \vee Y$	$\text{Res}(1,4)$
	–	:	$X \vee \neg Y$	$\text{Res}(2,5)$, redundant
	8	:	Y	$\text{Res}(3,4)$
Res^2	10	:	$A \vee \neg Y$	$\text{Res}(1,6)$
	11	:	$Y \vee \neg Y$	$\text{Res}(2,7)$ über A
	12	:	$A \vee \neg A$	$\text{Res}(2,7)$ über Y
	13	:	$\neg A$	$\text{Res}(2,8)$
	–	:	$Y \vee \neg Y$	$\text{Res}(3,6)$ über X, redundant
	14	:	$X \vee \neg X$	$\text{Res}(3,6)$ über Y
	15	:	X	$\text{Res}(4,6)$
	–	:	$X \vee A$	$\text{Res}(6,7)$, redundant
	–	:	X	$\text{Res}(6,9)$, redundant
Res^3	–	:	$\neg X \vee A$	$\text{Res}(1,12)$, redundant
	–	:	$\neg X \vee A$	$\text{Res}(1,14)$, redundant
	16	:	A	$\text{Res}(1,15)$
	–	:	A	$\text{Res}(1,16)$, redundant
	17	:	$\neg Y$	$\text{Res}(2,10)$
	–	:	$A \vee \neg Y$	$\text{Res}(2,11)$, redundant
	–	:	$\neg A \vee \neg Y$	$\text{Res}(2,12)$, redundant
	–	:	$\neg Y$	$\text{Res}(2,16)$, redundant
	–	:	$\neg X \vee A$	$\text{Res}(3,10)$, redundant
	–	:	$\neg X \vee Y$	$\text{Res}(3,12)$, redundant
	–	:	$\neg X \vee Y$	$\text{Res}(3,14)$, redundant
	–	:	Y	$\text{Res}(3,15)$, redundant
	–	:	$Y \vee A$	$\text{Res}(3,16)$, redundant
	–	:	$X \vee A$	$\text{Res}(4,10)$, redundant
	–	:	$Y \vee X$	$\text{Res}(4,11)$, redundant
	–	:	$Y \vee X$	$\text{Res}(4,14)$, redundant
	–	:	$X \vee A$	$\text{Res}(4,12)$, redundant
	–	:	X	$\text{Res}(4,13)$, redundant
	–	:	$X \vee A$	$\text{Res}(4,14)$, redundant
	–	:	$X \vee Y$	$\text{Res}(6,11)$, redundant
	–	:	$X \vee \neg Y$	$\text{Res}(6,14)$, redundant
	–	:	A	$\text{Res}(7,10)$, redundant
	–	:	$A \vee Y$	$\text{Res}(7,11)$, redundant
	–	:	$A \vee Y$	$\text{Res}(7,12)$, redundant
	–	:	Y	$\text{Res}(7,13)$, redundant
	–	:	A	$\text{Res}(8,10)$, redundant
	–	:	Y	$\text{Res}(8,11)$, redundant
Res^4	18	:	$\bot$	$\text{Res}(8,17)$
	19	:	$\bot$	$\text{Res}(13,16)$

Tabelle 1.1: Resolution für die Klauselmenge $F = \{\neg X \vee A, \neg A \vee Y, \neg X \vee Y, Y \vee X, X \vee A\}$

- I — Ihr seit intelligent
- L — Ihr lernt für die Prüfung
- K — Ihr habt diesen Kurs gewählt
- A — Ihr seit hier (anwesend)
- P — Ihr besteht die Prüfung

Wir formalisieren den Sachverhalt in eine Menge Γ von Voraussetzungen, und eine Folgerung P — nämlich, dass ihr die Prüfung besteht:

$$\Gamma = \{I \wedge L \longrightarrow P, K \longrightarrow I, A \longrightarrow K, A, L\}$$

Zu zeigen ist $\Gamma \models P$ oder

$$(I \wedge L \longrightarrow P) \wedge (K \longrightarrow I) \wedge (A \longrightarrow K) \wedge A \wedge L \longrightarrow P$$

Wir könnten das mit den aus dem letzten Abschnitt bekannten Mitteln zeigen, die auf Wahrheitstabellen beruhen, aber auf die Dauer ist das nicht befriedigend, weil es auf einer externen Sicht von Wahrheit beruht. Was genau bedeutet es, wenn ein Atom "wahr" ist, oder nicht? In dem Beispiel oben, was bedeutet "intelligent"? Oder ein Beispiel aus der Mathematik, aus den Wahrheitstabellen folgt semantisch dass

$$\models R \vee \neg R$$

was für eine beliebige Aussage R heißt, dass eines von beiden gilt — aber was, wenn R nicht entscheidbar ist? Oder wir nicht wissen, welches von beiden gilt (*e.g.* für $P = NP$)?

Dieses Problem wird noch drängender, wenn unsere Aussagen nicht mehr atomar sind, sondern strukturiert, so dass wir beispielsweise mathematische Aussagen formalisieren können. Wir können dann einzelnen Aussagen nicht mehr einfach Wahrheitswerte zuweisen, sondern müssen strukturierte Beweise führen. Ein Beispiel hierfür sind wieder die Lösung quadratischer Gleichungen — um diese Art von Rechnung zu formalisieren reicht es sicherlich nicht, jeden Term $a = b$ als atomare Aussage zu behandeln.

Darüber hinaus zeigen die semantisch basierten Beweisverfahren aus den vorherigen Abschnitten für realistische Probleme (wenn wir über mehr reden als nur atomare Aussagen) eine überexponentielle Komplexität, oder sind gar nicht mehr entscheidbar. Um auch bei diesen Problemen — und dazu zählen typischerweise Probleme, die bei der Programmverifikation entstehen – behandeln zu können, müssen wir über *Beweise* reden.

Was bedeutet das alles? Anstatt eine Wahrheitstabelle für die oberen Aussagen zu machen, und zu folgern, dass P in allen Fällen gelten muss (wir überlassen das den geneigten Lesern als Übung), wollen wir folgende Beweisführung in Logik gießen — und zwar so, dass wir es hinterher mechanisch[3] überprüfen können:

1. Wir sind hier, und wenn wir hier sind, dann haben wir diesen Kurs gewählt, also haben wir den Kurs gewählt.
2. Wir haben den Kurs gewählt, und wenn man den Kurs wählt, ist man intelligent, also sind wir intelligent.

[3] Idealerweise mit einem Computer — aber dazu später.

3. Wir haben für die Prüfung gelernt.
4. Wir sind also intelligent *und* haben für die Prüfung gelernt.
5. Wenn wir intelligent sind und für die Prüfung gelernt haben, bestehen wird die Prüfung.
6. Also bestehen wir die Prüfung.

Noch einmal mit Atomen:

1. Wir haben A und es gilt $A \longrightarrow K$, also haben wir K;
2. Wir haben K und es gilt $K \longrightarrow I$, also haben wir I;
3. Wir haben L.
4. Wir haben I und L, also haben wir $I \wedge L$.
5. Wir haben $I \wedge L$, und es gilt $I \wedge L \longrightarrow P$, also haben wir P.

Wir sehen, dass unser Beweis aus einer Folge von Schlüssen besteht, die eine oder mehrere Voraussetzungen (Prämissen) haben, und eine Schlussfolgerung (Konklusion). Generell ist also eine Schlussregel von der Form

$$R \in Prop^n \times Prop$$

Wir können zwei Schlussregeln R und S komponieren, wenn die Konklusion von R einer Prämisse von S entspricht (ggf. müssen wir die Atome in R und S geeignet substituieren); dann entsteht eine neue Regel, deren Prämissen die von R und den von S übrigbleibenden Prämissen sind, deren Schlußfolgerung die von S ist.

Beispiel 1.1 (Eine Semantik-Freie Sprache) *Gegeben sei eine Sprache* $\mathscr{L} = \{\clubsuit, \spadesuit, \heartsuit, \diamondsuit\}$ *mit folgenden Schlußregeln:*

$$\frac{\diamondsuit}{\clubsuit}\,\alpha \qquad \frac{\diamondsuit}{\spadesuit}\,\beta \qquad \frac{\clubsuit \quad \spadesuit}{\heartsuit}\,\gamma \qquad \frac{}{\diamondsuit}\,\delta$$

Wie können wir jetzt $\heartsuit$ *ableiten?*

Wenn wir δ *mit* α *komponieren, erhalten wir eine Ableitung von* $\clubsuit$*; analog für* δ *und* β *eine Ableitung von* $\spadesuit$*. Diese beiden können wir mit* γ *komponieren, und erhalten eine Ableitung von* $\heartsuit$*.*

Graphisch stellen wir diese Ableitung wie folgt dar:

$$\dfrac{\dfrac{\dfrac{}{\diamondsuit}\,\delta}{\clubsuit}\,\alpha \quad \dfrac{\dfrac{}{\diamondsuit}\,\delta}{\spadesuit}\,\beta}{\heartsuit}\,\gamma$$

Für das natürliche Schließen führen wir folgende Regeln für Konjunktion und Implikation ein. Diese Regeln sind *Axiome*, sie sind die Basis aller Ableitungen, und wir müssen sie — zu einem gewissen Grad — einfach glauben, genau wie wir die Wahrheitstafeln als gegeben hinnehmen mussten.[4]

$$\frac{A \quad B}{A \wedge B}\,\wedge I \qquad\qquad \frac{A \quad A \longrightarrow B}{B}\,mp$$

[4] Wir können allerdings zeigen, dass diese Regeln und die Wahrheitstafeln zueinander passen — mehr dazu später.

Damit können wir jetzt unseren Beweis formalisieren:

$$\dfrac{\dfrac{\dfrac{A \quad A \longrightarrow K}{K}\, mp \qquad K \longrightarrow I}{I}\, mp \qquad L}{I \wedge L}\, \wedge I \qquad I \wedge L \longrightarrow P}{P}\, mp$$

Wie wir sehen, können wir Konjunktionen zeigen und mit Implikationen arbeiten, aber können wir mit Konjunktionen arbieten, und beispielsweise aus $I \wedge L\, I$ folgern? Dazu brauchen wir eine weitere Regel — genauer gesagt zwei:

$$\dfrac{A \wedge B}{A}\, \wedge E_L \qquad\qquad \dfrac{A \wedge B}{B}\, \wedge E_R$$

Wir beginnen ein Muster zu erkennen: es gibt für $\wedge$ eine Regel mit $\wedge$ in der Konklusion, und zwei (weil $\wedge$ zwei Argumente hat) mit $\wedge$ in der Prämisse. Regeln der ersten Art heißen *Einführungsregeln*, Regeln der zweiten Art *Eliminationsregeln*.

Für die Implikation hatten wir schon eine Regel mit $\longrightarrow$ in der Konklusion, also eine Eliminationsregel für $\longrightarrow$, der gute alte Modus Ponens (hier aus Gründen der Konsistenz *mp* genannt). Wie sieht hier die Einführungsregeln aus? Das führt uns zum wesentlichen Feature des natürlichen Schließens, nämlich der Umgang mit Annahmen. Die Regel ist

$$\dfrac{\begin{array}{c}[A]\\ \vdots \\ B\end{array}}{A \longrightarrow B}\, \longrightarrow I$$

Wenn wir aus der Annahme A die Schlussfolgerung B herleiten können, dann haben wir die Implikation $A \longrightarrow B$ gezeigt (und können die Annahme A *schließen*).

Mit den Axiomen von oben erlaubt uns diese Regel die Schlußfolgerung “Wenn ihr hier, seit ihr intelligent” (Glückwunsch!):

$$\dfrac{\dfrac{\dfrac{[A]^1 \quad A \longrightarrow K}{K}\, mp \qquad K \longrightarrow I}{I}\, mp}{A \longrightarrow I}\, \longrightarrow I_1$$

Jede Anwendung der $\longrightarrow I$-Regel *kann* eine oder *mehrere* Annahmen schließen. Damit wir wissen, welche Annahme geschlossen wird, notieren wir an der Regel und an der Annahme einen Index (hier 1).

Wir betrachten zwei weitere Beispiele ([2, S. 32]):

- Das erste ist $A \wedge B \longrightarrow B \wedge A$:

$$\dfrac{\dfrac{\dfrac{[A \wedge B]^1}{B}\, \wedge E_R \qquad \dfrac{[A \wedge B]^1}{A}\, \wedge E_L}{B \wedge A}\, \wedge I}{A \wedge B \longrightarrow B \wedge A}\, \longrightarrow I_1$$

- Das zweite ist $(P \longrightarrow (Q \longrightarrow R)) \longrightarrow (P \wedge Q \longrightarrow R)$:

$$\dfrac{\dfrac{\dfrac{[P \wedge Q]^2}{Q}\,\wedge E_R \qquad \dfrac{\dfrac{[P \wedge Q]^2}{P}\,\wedge E_L \quad [P \longrightarrow (Q \longrightarrow R)]^1}{Q \longrightarrow R}\,mp}{R}\,mp}{P \wedge Q \longrightarrow R}\,\longrightarrow I_2}{(P \longrightarrow (Q \longrightarrow R)) \longrightarrow (P \wedge Q \longrightarrow R)}\,\longrightarrow I_1$$

Übung 1.14 *Auf dem Landgut von Sir Archibald Fortescue Lord Netherworth-Middlington (seine Freunde nannten ihn Neddles) sich ein schreckliches Verbrechen ereignet: der Lord wurde morgens ermordet in Gewächshaus gefunden, wo er seine seltenen Rosen züchtet. Detective Constable Brian Quickthink von Scotland Yard ermittelt.*

Außer dem Lord waren zum Tatzeitpunkt (über Nacht) auf dem Landgut nur noch sein Butler und der Gärtner.[5] *Wenn es der Butler war, muss er im Gewächshaus gewesen sein. Allerdings war der Butler nicht im Gewächshaus (an seinen Schuhen war keine Spur von Dreck zu finden, und es hatte in der Nacht stark geregnet).*

Formalisiert den Sachverhalt in Aussagenlogik.

Wir haben Atome G (der Mörder war der Gärtner), B (der Mörder war der Butler), und C (der Butler war im Gewächshaus). Damit ist

$$\Gamma = \{G \vee B, B \longrightarrow C, \neg C\}$$

Wie kann DC Quickthink den Mörder überführen? Uns fehlen noch Regeln, mit Negation und Disjunktion umzugehen!

Wir führen diese ein:

$$\dfrac{A}{A \vee B}\,\vee I_L \qquad \dfrac{B}{A \vee B}\,\vee I_R \qquad \dfrac{A \vee B \qquad \begin{array}{c}[A]\\ \vdots\\ C\end{array} \qquad \begin{array}{c}[B]\\ \vdots\\ C\end{array}}{C}\,\vee E$$

$$\dfrac{\begin{array}{c}[A]\\ \vdots\\ \bot\end{array}}{\neg A}\,\neg I \qquad \dfrac{A \quad \neg A}{\bot}\,\neg E$$

$$\dfrac{\bot}{A}\,false \qquad \dfrac{\begin{array}{c}[\neg A]\\ \vdots\\ \bot\end{array}}{A}\,raa$$

Die beiden Einführungsregeln für die Disjunktion sind (hoffentlich) offensichtlich; wenn A gilt, dann auch $A \vee B$. Die Eliminationsregel müssen wir als *Fallunterscheidung* begreifen: um aus $A \vee B$ etwas zu folgern, müssen wir zum einen A und zum anderen B annehmen (also die beiden Fälle unterscheiden), und

[5] Es ist ja heute nicht mehr so einfach Personal zu finden…

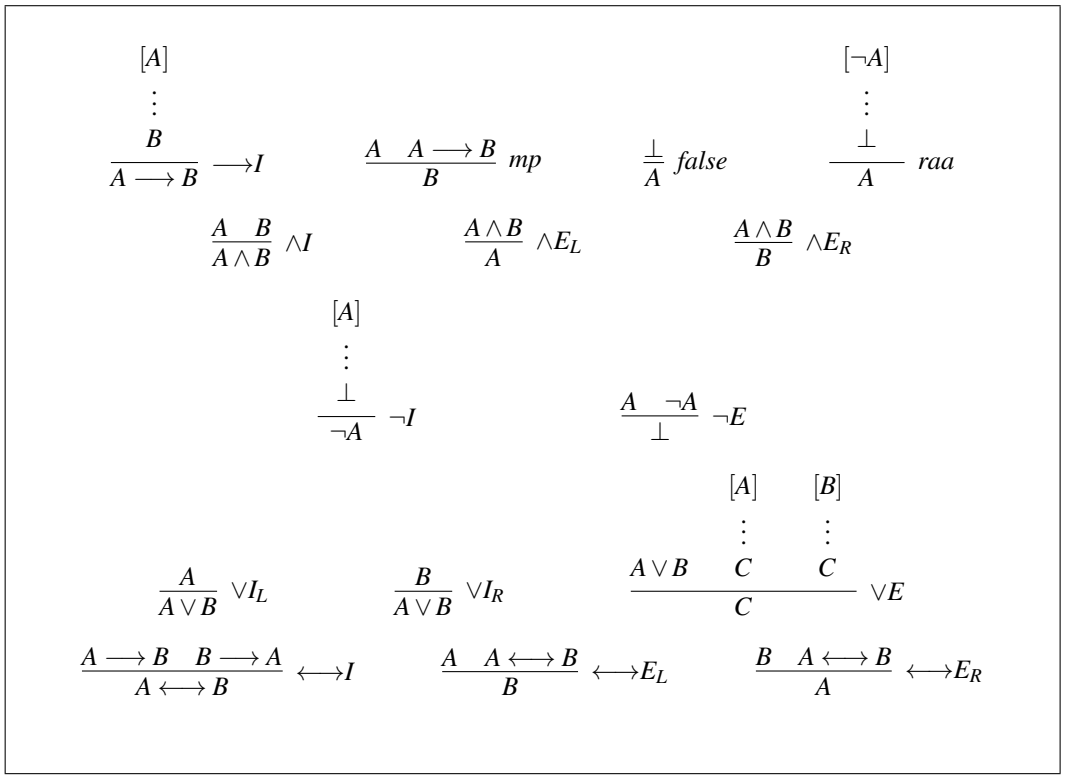


Tabelle 1.2: Regeln für das natürliche Schließen

aus beiden jeweils das gleiche folgern, dann gilt es auch für die Disjunktion. Die Regeln für die Negation werden schnell klar, wenn wir die Negation als Abkürzung $\neg A \equiv A \longrightarrow \bot$ einführen. Die Regeln für Falsum sind etwas erklärungsbedürftig. Die erste (eine Eliminationsregel für $\bot$) besagt semantisch "ex falso quodlibet", also wenn ich erst einmal einen Widerspruch habe, kann ich alles zeigen; die zweite Regel ist der klassische Widerspruchsbeweis: um A zu zeigen, führe ich $\neg A$ zum Widerspruch.

Mit diesen Regel können wir den Mörder überführen:

$$\dfrac{B \vee G \qquad \dfrac{\dfrac{[B]^1 \quad B \longrightarrow C}{C}\,mp \qquad \neg C}{\dfrac{\bot}{G}\,false}\,\neg E \qquad [G]^1}{G}\,\vee E_1$$

Der Mörder war also der Gärtner. Verhaften Sie die üblichen Verdächtigen.

Vorlesung vom 04.05.2023: Natürliches Schließen II

Es fehlen noch Regeln für die Äquivalenz ($\longleftrightarrow$), die aber recht offensichtlich sein sollten. Tabelle 1.2 zeigt die Regeln für das natürliche Schließen mit allen Konnektiven.

Damit können wir jetzt die syntaktische Ableitbarkeit definieren:

Definition 1.13 (Ableitbarkeit) *Seit Γ eine Menge von Aussagen und $\phi \in Prop$ eine Aussagen, dann ist*

ϕ aus Γ ableitbar, geschrieben

$$\Gamma \vdash \phi$$

genau dann wenn es eine Ableitung mit den Regeln aus Tabelle 1.2 gibt, so dass die offenen Hypothesen der Ableitung alle in Γ enthalten sind.

Für $\emptyset \vdash \phi$ schreiben wir $\vdash \phi$.

Aus Gründen der Lesbarkeit verzichten wir an dieser Stelle auf eine präzise mathematische Definition von Ableitung ([2, Abschnitt 1.5] hat hier die technischen Details); die obige Definition ist für unsere Zwecke präzise genug.

Natürlich stellt sich sofort die Frage, wie $\Gamma \vdash \phi$ und $\Gamma \models \phi$ (syntaktische Herleitbarkeit und semantische Gültigkeit) zusammenhängen — tatsächlich ist das einer der Kernfragen der Logik. Wenn die syntaktische Herleitbarkeit die semantische Gültigkeit impliziert, sagen wir die Logik ist *korrekt*; wenn die semantische Gültigkeit die syntaktische Herleitbarkeit impliziert, wenn also alle semantisch gültigen Aussagen syntaktisch herleitbar sind, dann is die Logik *vollständig*. Für die Aussagenlogik, soviel sei schon mal verraten, sind die beiden äquivalent; wir zeigen das im Detail später, erstmal betrachten wir die syntaktische Herleitbarkeit etwas mehr im Detail.

Noch ein einfaches Beispiel: $\vdash (A \wedge B \longrightarrow C) \longrightarrow (A \longrightarrow (B \longrightarrow C))$

$$\dfrac{\dfrac{\dfrac{[A]^2 \quad [B]^3}{A \wedge B}\,\wedge I \qquad [A \wedge B \longrightarrow C]^1}{C}\,mp}{\dfrac{B \longrightarrow C}{A \longrightarrow (B \longrightarrow C)}\,\longrightarrow^3 I^2}{(A \wedge B \longrightarrow C) \longrightarrow (A \longrightarrow (B \longrightarrow C))}\,\longrightarrow I^1$$

Dieses Beispiel demonstriert, wie wir ganz schematisch beweisen können. Das geht nicht immer so einfach, besonders wenn man Negation oder RAA benutzen muss.

Übung 1.15 *Leite folgende Theoreme her:*

$$\vdash (A \longrightarrow B) \longrightarrow ((B \longrightarrow C) \longrightarrow (A \longrightarrow C)) \tag{1.21}$$
$$\vdash (A \longrightarrow B) \wedge \neg B \longrightarrow \neg A \tag{1.22}$$
$$\vdash A \longrightarrow (B \longrightarrow A) \tag{1.23}$$
$$\vdash A \longrightarrow (\neg A \longrightarrow B) \tag{1.24}$$
$$\vdash \neg\neg A \longleftrightarrow A \tag{1.25}$$
$$\vdash (A \longrightarrow (B \longrightarrow C)) \longleftrightarrow (A \wedge B \longrightarrow C) \tag{1.26}$$
$$\vdash \bot \longleftrightarrow (A \wedge \neg A) \tag{1.27}$$

Vorlesung vom 09.05.2023: Natürliches Schließen III

1.5.1 Formalisierung eines mathematischen Beweises

Hier ist der Beweis, dass die Quadratwurzel von 2 keine rationale Zahl ist: ein klassischer Widerspruchsbeweis.

Zu zeigen: $\sqrt{2}$ ist irrational.

Beweis.

	Annahme: $\sqrt{2}$ ist rational	$\neg A$
$\Longleftrightarrow$	$\sqrt{2} = \frac{p}{q}$ und p,q teilerfremd	$B_1 \wedge C$
$\Longleftrightarrow$	$2 = \frac{p^2}{q^2}$	B_2
$\Longleftrightarrow$	$p^2 = 2q^2$ (p^2 gerade)	B_3
$\Longleftrightarrow$	$p = 2r$ (p gerade)	B_4
$\Longleftrightarrow$	$p^2 = 4r^2$	B_5
$\Longleftrightarrow$	$2q^2 = 4r^2$	B_6
$\Longleftrightarrow$	$q^2 = 2r^2$ (q^2 gerade)	B_7
$\Longleftrightarrow$	q gerade	B_8
$\Longleftrightarrow$	p und q gerade, also sind p und q nicht teilerfremd	$\neg C$
	Widerspruch ↯	$\bot$
	Also ist $\sqrt{2}$ irrational	A

□

Wir wollen diesen Beweis jetzt in natürlichem Schließen und Aussagenlogik modellieren. Dabei sind unsere Axiome die Umformungen oben:

$$\begin{aligned}\Gamma = \{\ & \neg A \longleftrightarrow B_1 \wedge C,\\ & B_1 \longleftrightarrow B_2, B_2 \longleftrightarrow B_3, B_3 \longleftrightarrow B_4, B_4 \longleftrightarrow B_5,\\ & B_3 \wedge B_5 \longrightarrow B_6,\\ & B_6 \longleftrightarrow B_7, B_7 \longleftrightarrow B_8,\\ & B_4 \wedge B_8 \longrightarrow \neg C\}\end{aligned}$$

Damit ergibt sich folgender Ableitungsbaum (aus Gründen der Übersichtlichkeit mussten wir ihn in meh-

rere aufspalten):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\neg A]^1 \quad \neg A \longleftrightarrow B_1 \wedge C}{B_1 \wedge C}\longleftrightarrow E_L}{B_1}\wedge E_L \quad B_1 \longleftrightarrow B_2}{B_2}\longleftrightarrow E_L \quad B_2 \longleftrightarrow B_3}{B_3}\longleftrightarrow E_L \tag{1.28}$$

$$\dfrac{\dfrac{\dfrac{(1.28)}{B_3} \quad B_3 \longleftrightarrow B_4}{B_4}\longleftrightarrow E_L \quad B_4 \longleftrightarrow B_5}{B_5}\longleftrightarrow E_L \tag{1.29}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(1.28)}{B_3} \quad \dfrac{(1.29)}{B_5}}{B_3 \wedge B_5}\wedge I \quad B_3 \wedge B_5 \longrightarrow B_6}{B_6}\,mp \quad B_6 \longleftrightarrow B_7}{B_7}\longleftrightarrow E_L \quad B_7 \longleftrightarrow B_8}{B_8}\longleftrightarrow E_L \tag{1.30}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(1.28)}{B_3} \quad B_3 \longleftrightarrow B_4}{B_4}\longleftrightarrow E_L \quad \dfrac{(1.30)}{B_8}}{B_2 \wedge B_8}\wedge I \quad B_4 \wedge B_8 \longrightarrow \neg C}{\neg C}\,mp \quad \dfrac{\dfrac{[\neg A]^1 \quad \neg A \longleftrightarrow B_1 \wedge C}{B_1 \wedge C}\wedge E_R}{C}\neg E}{\dfrac{\bot}{A}\,raa} \tag{1.31}$$

Was sehen wir hier?

- Auch ein einfacher Beweis wird ziemlich lang.
- Was hier definitiv fehlt ist die Möglichkeit, Aussagen über ganzen Zahlen formulieren zu können — wir würden gerne die Atome der Aussagen weiter zerlegen können (Kernspaltung?). Diese Trennung zwischen einer Welt der logschen Aussagen und der Terme (Dinge *über* die wir reden) führt uns in die Welt der Prädikatenlogik—mehr dazu nach Himmelfahrt.
- Die Aussagenlogik repräsentiert sehr gut den *logischen* Kern, die Argumentation des Beweises.

Ein anderes interessantes Beispiel ist $\vdash (A \wedge B \longrightarrow C) \longleftrightarrow (B \wedge A \longrightarrow C)$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[B \wedge A]^1}{A}\wedge E_L \quad \dfrac{[B \wedge A]^1}{B}\wedge E_R}{A \wedge B}\wedge I \quad [A \wedge B \longrightarrow C]^2}{C}\,mp}{B \wedge A \longrightarrow C}\longrightarrow I^1}{(A \wedge B \longrightarrow C) \longrightarrow (B \wedge A \longrightarrow C)}\longrightarrow I^2 \quad \dfrac{\vdots}{(B \wedge A \longrightarrow C) \longrightarrow (A \wedge B \longrightarrow C)}\longrightarrow I^3}{(A \wedge B \longrightarrow C) \longleftrightarrow (B \wedge A \longrightarrow C)}\longleftrightarrow I$$

Diese Beispiel zeigt, wie wir Unterformeln umformen (hier die Prämisse der Implikation), genau wie die Substitutivität der semantischen Äquivalenz. Tatsächlich gibt es auch für die syntaktische Ableitbarkeit einen Substitutionstheorem, analog zu Theorem 1.4:

Theorem 1.12 (Substitutionstheorem) *Für Aussagen $\phi_1, \phi_2, \psi \in Prop$ und ein Atom $p \in P$ gilt:*

$$\vdash (\phi_1 \longleftrightarrow \phi_2) \longrightarrow (\psi[\phi_1/p] \longleftrightarrow \psi[\phi_2/p])$$

Beweis. Das Theorem folgt aus Theorem 1.4 mit der Vollständigkeit (*i.e.* $\models \phi$ gdw $\vdash \phi$), läßt sich aber auch unabhängig durch Induktion über ψ beweisen. □

1.5.2 Minimale Systeme für das Natürliche Schließen

Die Regeln in Tabelle 1.2 sind nicht zufällig oder aus ästhetischen Gründen so angeordnet; tatsächlich ist es so, dass die unteren Zeilen sich immer aus denen darüber herleiten lassen, wenn wir die Konnektive als Abkürzungen begreifen (mit den Äquivalenzen aus (1.7)-(1.13)). Für die Negation haben wir das mit $\neg A \equiv A \longrightarrow \bot$ schon gesehen. Die Regeln für die Äquivalenz folgen leicht mit $A \longleftrightarrow B \equiv (A \longrightarrow B) \wedge (B \longrightarrow A)$.

Für die Disjunktion wählen wir $A \vee B \equiv \neg(\neg A \wedge \neg B)$. Damit ergeben sich die Regeln für $\vee$ wie folgt:

- Einführungsregel links $\vee I_L$ ($\vee I_R$ ist analog):

$$\dfrac{\dfrac{\dfrac{[\neg A \vee \neg B]^2}{\neg A}\wedge E_L \quad A}{\bot}}{A \vee B \equiv \neg(\neg A \vee \neg B)}\neg I^1$$

- Eliminationsregel $\vee E$:

$$\dfrac{A \vee B \equiv \neg(\neg A \wedge \neg B) \qquad \dfrac{\dfrac{\dfrac{[A]^2}{\vdots}}{C} \quad \dfrac{[\neg C]^1}{\bot}\neg E}{\neg A}\neg I^2 \qquad \dfrac{\dfrac{\dfrac{[B]^3}{\vdots}}{C} \quad \dfrac{[\neg C]^1}{\bot}\neg E}{\neg B}\neg I^3}{\neg A \wedge \neg B}\wedge I}{\dfrac{\bot}{C}\, raa^1}\neg E$$

Was bedeutet das jetzt? Das bedeutet, dass wir jeden Ableitungsbaum, der den Operator $\vee$ und die Regeln $\vee E, \vee I_L, \vee I_R$ enthält, durch einen Ableitungsbaum ersetzen können, der weder $\vee$ noch diese Regeln enthält. Hier ein konkretes Beispiel: der Beweis für $\vdash (P \vee Q) \wedge \neg P \longrightarrow Q$. Mit $\vee$ sieht der Baum so aus:

$$\dfrac{\dfrac{[(P \vee Q) \wedge \neg P]^1}{P \vee Q}\wedge E_L \qquad \dfrac{\dfrac{[P]^2 \quad \dfrac{[(P \vee Q) \wedge \neg P]^1}{\neg P}\wedge E_R}{\bot}\neg E}{Q}\bot \qquad [Q]^2}{(P \vee Q) \wedge \neg P \longrightarrow Q}\vee E^2 \longrightarrow I^1$$

Jetzt ersetzen wir $P \vee Q$ durch $\neg(\neg P \wedge \neg Q)$ und die Regel (hier $\vee E$) durch den entsprechenden Regelbaum. Dabei müssen wir (in dem Baum) A durch P und B durch Q ersetzen. Wir erhalten folgenden Baum, in dem wir der Deutlichkeit halber den eingesetzten Regelbaum blau markiert haben:

$$\dfrac{\dfrac{\dfrac{[\neg(\neg P\wedge\neg Q)\wedge\neg P]^1}{\neg P}\wedge E_R \quad [P]^3}{\dfrac{\bot}{Q}\,\bot}\neg E \qquad \dfrac{[\neg Q]^2}{\dfrac{\bot}{\neg P}\,\neg I^3}\neg E \qquad \dfrac{[Q]^4\quad [\neg Q]^2}{\dfrac{\bot}{\neg Q}\,\neg I^4}\neg E}{\dfrac{[\neg(\neg P\wedge\neg Q)\wedge\neg P]^1}{\neg(\neg P\wedge\neg Q)}\wedge E_L \qquad \neg P\wedge\neg Q}\wedge I \qquad \neg E$$
$$\dfrac{\dfrac{\bot}{Q}\,raa^2}{\neg(\neg P\wedge\neg Q)\wedge\neg P\longrightarrow Q}\longrightarrow I^1$$

Damit können wir also die Regeln für $\vee$ aus den Regeln für $\wedge$, $\longrightarrow$, $\bot$, $\neg$ herleiten. Etwas überraschend ist vielleicht, dass wir die Regeln für $\wedge$ aus den Regeln für $\longrightarrow$ und $\bot$ herleiten können. Dazu setzen wir $A\wedge B\equiv\neg(A\longrightarrow\neg B)$, und erhalten:

- Einführungsregel $\wedge I$:
$$\dfrac{\dfrac{A\quad [A\longrightarrow\neg B]^1}{\neg B}\quad B}{\dfrac{\bot}{A\wedge B\equiv\neg(A\longrightarrow\neg B)}}\neg I^1$$
- Eliminationsregel links $\wedge E_L$:
$$\dfrac{A\wedge B\equiv\neg(A\longrightarrow\neg B)\quad \dfrac{[\neg B]^1}{A\longrightarrow\neg B}\longrightarrow I^2}{\dfrac{\bot}{B}\,raa^1}\neg E$$
- Eliminationsregel rechts $\wedge E_R$:
$$\dfrac{A\wedge B\equiv\neg(A\longrightarrow\neg B)\quad \dfrac{\dfrac{[A]^2\quad [\neg A]^1}{\dfrac{\bot}{\neg B}\,\neg I^3}\neg E}{A\longrightarrow\neg B}\longrightarrow I^2}{\dfrac{\bot}{A}\,raa^1}\neg E$$

Zusammenfassend können wir also die Operatoren und ihre Regeln für das natürliche Schließen in vier Schritten aus zwei gegebenen Operatoren wie folgt aufbauen:

	Operator	Definition	Regeln
Gegeben	$\longrightarrow,\bot$		$mp,\longrightarrow I,raa,false$
(1)	$\neg$	$\neg A\equiv A\longrightarrow\bot$	$\neg I,\neg E$
(2)	$\wedge$	$A\wedge B\equiv\neg(A\longrightarrow\neg B)$	$\wedge I,\wedge E_L,\wedge E_R$
(3)	$\vee$	$A\vee B\equiv\neg(\neg A\wedge\neg B)$	$\vee I_L,\vee I_R,\vee E$
(4)	$\longleftrightarrow$	$A\longleftrightarrow B\equiv(A\longrightarrow B)\wedge(B\longrightarrow A)$	$\longleftrightarrow I,\longleftrightarrow E_L,\longleftrightarrow E_R$

Theorem 1.13 $\{\longrightarrow,\bot\}$ *und die Regeln* $mp,\longrightarrow I,raa,falsee$ *sind ein minimales System für die Aussagenlogik mit natürlichem Schließen.*

Vorlesung vom 11.05.2023: Korrektheit und Vollständigkeit I

1.6 Korrektheit und Vollständigkeit

Übung 1.16 (Aufwärmübung) *Ein Student schreibt:*

```
Ich hab eine Frage zum aktuellen Übungsblatt.

Dort sollen wir durch natürliches Schließen zeigen, dass:

(A -> B) <-> (!A -> !B)

ein Theorem ist.

Nach meinen Verständnis ist dies jedoch kein Theorem. Die Belegung
v=[A |->1, B|->0] ergibt einen Widerspruch.
```

- *Warum ist das richtig, aber die Argumentation nach unserem jetzigen Kenntnisstand lückenhaft?*
- *Was ist der Unterschied zwischen einer lückenhaften Argumentation und einer falschen Behauptung?*

Wir fassen mal zusammen:

- Wir haben syntaktisch die Menge *Prop* aller möglichen Aussagen definiert (basierend auf einer Menge P von atomaren Aussagen, die auch als Variablen fungieren). Wenn wir die atomaren Aussagen mit Werten belegen, können wir eine Aussage auswerten: sie ist dann gültig ($[\![phi]\!]_v = 1$) oder falsch ($[\![\phi]\!]_v = 0$).
- Die semantische Gültigkeit teilt die Menge *Prop* in drei Teilmengen:
  (1) Unter allen möglichen Variablenbelegungen gültig (Tautologien);
  (2) Unter manchen Variablenbelegungen gültig (erfüllbar);
  (3) Unter keiner Variablenbedingung gültig (unerfüllbar).
- Basierend darauf haben wir semantische Gültigkeit $\models \phi$ und semantische Folgerung $\Gamma \models \phi$ definiert.
- Dann haben wir syntaktische *Beweisbarkeit* betrachtet; dazu haben wir das *natürliche Schließen* eingeführt. Wir haben die syntaktische Folgerung als $\Gamma \vdash \phi$ definiert (es läßt sich ein Beweisbaum konstruieren mit der Konklusion ϕ und den offenen Blättern in Γ), und $\vdash \phi$.

Es stellt sich natürlich die Frage, wie $\models \phi$ und $\vdash \phi$ zusammenhängen—die Frage nach der Korrektheit (aus $\vdash \phi$ folgt $\models \phi$, alles was ich herleiten kann ist auch semantisch wahr) und Vollständigkeit (alles was semantisch wahr ist läßt sich auch herleiten).

1.6.1 Korrektheit

Als erstes zeigen wir die Korrekheit. Dazu benötigen wir ein Beweisprinzip über Ableitungen (derivations). Da diese induktiv aufgebaut sind (letzten Endes sind es ja gelabelte Bäume), ist das Beweisprinzip die *Induktion über der Herleitung*.

Übung 1.17 *Warum reicht strukturelle Induktion über Prop — bspw. der Konklusion — nicht aus?*

Strukturelle Induktion über Herleitungen funktioniert wie folgt. Jede Ableitung hat einen darunterliegenden Baum, dessen Knoten mit Aussagen $\phi \in Prop$ markiert sind. Die Blätter des Baumes sind die Hypothesen der Herleitung (sie sind offen, wenn sie nicht durch Anwendung bestimmter Regeln geschlossen werden). Die Knoten des Baumes müssen Regelanwendungen entsprechen, wobei die Markierungen der Kinderknoten die Prämissen der Regel sind, und die Markierung des Elternknoten die Konklusion. Die Wurzel des Baumes[6] ist die Konklusion.

Wir folgen der Notation von [2]: eine Ableitung mit der Konklusion ψ schreiben wir als $\overset{\mathscr{D}}{\psi}$, und die Anwendung einer Regel $\dfrac{\psi}{\phi}$ oder $\dfrac{\psi_1 \quad \psi_2}{\phi}$ auf diese Ableitung schrieben wir als $\dfrac{\overset{\mathscr{D}}{\psi}}{\phi}$ bzw. $\dfrac{\overset{\mathscr{D}}{\psi_1} \quad \overset{\mathscr{D}}{\psi_2}}{\phi}$.

Übung 1.18 *Wofür steht der Baum, der nur aus einem Blatt ϕ besteht? Ist das eine gültige Ableitung?*

Jetzt können wir Induktion über Ableitungen definieren. Der Trick ist dabei, dass die Induktion über die Regelanwendung funktioniert, nicht über die Größe des Bäumes, weil die Regeln die Menge der gültigen Ableitungen bestimmen (nicht alle Bäume sind korrekte Ableitungen).

Lemma 1.14 (Induktionsprinzip über Ableitungen) *Sei A eine Eigenschaft über Ableitungen. Dann gilt $A(D)$ für alle Ableitungen, wenn:*

- *Induktionsbasis: $A(X)$ gilt für alle $X \in Prop$.*
- *Induktionsschritt $\longrightarrow I$: Wenn $A(D)$ für $D = \overset{\mathscr{D}}{\psi}$ gilt, dann gilt auch $A(D)$ für $D = \dfrac{\overset{[\phi]}{\overset{\mathscr{D}}{\psi}}}{\phi \longrightarrow \psi}$.*
- *Induktionsschritt mp: Wenn $A(D_1)$ für $D_1 = \overset{\mathscr{D}_1}{\phi}$ und $A(D_2)$ für $D_2 = \overset{\mathscr{D}_2}{\phi \longrightarrow \psi}$ gilt, dann gilt auch $A(D)$ für $D = \dfrac{\overset{\mathscr{D}_1}{\phi} \quad \overset{\mathscr{D}_2}{\phi \longrightarrow \psi}}{\psi}$.*
- *Induktionsschritt $\bot$: Wenn $A(D)$ für $D = \overset{\mathscr{D}}{\bot}$ gilt, dann gilt auch $A(D)$ für $D = \dfrac{\overset{\mathscr{D}}{\bot}}{\phi}$.*
- *Induktionsschritt raa: Wenn $A(D)$ für $D = \overset{\mathscr{D}}{\bot}$ gilt, dann gilt auch $A(D)$ für $D = \dfrac{\overset{[\neg\phi]}{\overset{\mathscr{D}}{\psi}}}{\phi}$.*

Hier ist ein einfaches Beispiel für eine Induktion über der Ableitung:

[6] Ganz untypisch für Informatiker zeichnen wir die Bäume hier tatsächlich mit der Wurzel unten— genau wie diese Holzdinger, die im Wald stehen.

Lemma 1.15 *Sei* $\Gamma \vdash \phi$*, dann gibt es eine* endliche *Menge* $\Delta \subseteq \Gamma$*, so dass* $\Delta \vdash \phi$.

Wichtig ist hier: wenn $\Gamma \vdash \phi$ dann müssen die offenen Blätter der Ableitung in Γ enthalten sein, aber Γ kann auch mehr Aussagen enthalten (die gar nicht in der Ableitung auftauchen); diese etwas liberale Definition ist später nützlich.

Damit können wir jetzt zum Hauptergebnis dieses Abschnitts schreiten:

Theorem 1.16 (Korrektheit des natürlichen Schließens) *Wenn* $\Gamma \vdash \phi$*, dann* $\Gamma \models \phi$.

Beweis. Beweis per Induktion über der Ableitung $\mathscr{D}$ von $\Gamma \vdash \phi$.

- Basis: Wenn $\mathscr{D}$ nur ein Element enthält, dann ist $\phi \in \Gamma$, damit offensichtlich $\Gamma \models \phi$.
- Schritt $(\longrightarrow I)$:

  Induktionsvoraussetzung: Wenn Γ alle Hypothesen von $\overset{\mathscr{D}}{\psi}$ enthält, dann $\Gamma \models \psi$.

  Zu zeigen: wenn Γ' alle Hypothesen von $\dfrac{\overset{[\phi]}{\overset{\mathscr{D}}{\psi}}}{\phi}$ enthält, dann $\Gamma' \models \phi \longrightarrow \psi$.

  Beweis: $\Gamma' \cup \{\phi\}$ enthält alle Hypothesen von $\overset{\mathscr{D}}{\psi}$, also erfüllt es die Induktionsvoraussetzung. Damit gilt also wenn $[\![\phi]\!]_v = 1$ und $[\![\rho]\!]_v = 1$ für alle $\rho \in \Gamma'$), dann $[\![\psi]\!]_v = 1$. Wann ist $[\![\phi \longrightarrow \psi]\!]_v = 1$? Wenn $[\![\phi]\!]_v = 0$ ist der Wert von $[\![\psi]\!]_v$ egal, oder wenn $[\![\phi]\!]_v = 1$ dann muss $[\![\psi]\!]_v = 1$, aber das haben wir gerade gezeigt; deshalb also $\Gamma' \models \phi \longrightarrow \psi$.
- Schritt (mp):

  Induktionsvoraussetzung: Wenn Γ alle Hypothesen von $\overset{\mathscr{D}}{\phi}$ enthält, dann $\Gamma \models \phi$; und wenn Γ alle Hypothesen von $\overset{\mathscr{D}}{\phi \longrightarrow \psi}$ enthält, dann $\Gamma \models \phi \longrightarrow \psi$.

  Zu zeigen: wenn Γ' alle Hypothesen von $\dfrac{\overset{\mathscr{D}}{\phi} \qquad \overset{\mathscr{D}}{\phi \longrightarrow \psi}}{\psi}$ enthält, dann $\Gamma' \models \psi$.

  Beweis: Offensichtlich erfüllt Γ' die Induktionsvoraussetzungen, also $\Gamma' \models \phi$ und $\Gamma' \models \phi \longrightarrow \psi$. Sei also v eine Belegung mit $[\![\rho]\!]_v = 1$ für alle $\rho \in \Gamma'$, dann $[\![\phi]\!]_v = 1$ und $[\![\phi \longrightarrow \psi]\!]_v = 1$; also muss nach der Wahrheitstabelle von $\longrightarrow$ auch $[\![\psi]\!]_v = 1$ gelten, also $\Gamma' \models \psi$.
- Schritt $(\bot)$:

  Induktionsvoraussetzung: Wenn Γ alle Hypothesen von $\overset{\mathscr{D}}{\bot}$ enthält, dann $\Gamma \models \bot$. Da $[\![\bot]\!]_v = 0$ für alle v ist dies gleichbedeutend mit der Tatsache, dass es kein v gibt so dass $[\![\psi]\!]_v = 1$ für alle $\psi \in \Gamma$.

  Zu zeigen: Wenn Γ' alle Hypothesen von $\dfrac{\overset{\mathscr{D}}{\bot}}{\phi}$ enthält, dann $\Gamma' \models \phi$.

  Beweis: Wenn Γ' alle Hypothesen von $\dfrac{\overset{\mathscr{D}}{\bot}}{\phi}$ enthält, dann insbesondere auch die von $\overset{\mathscr{D}}{\bot}$, also erfüllt Γ' die Induktionsvoraussetzung. Es kann also kein v geben, so dass $[\![\psi]\!]_v = 1$ für alle $\psi \in \Gamma'$, also $\Gamma' \models \phi$.
- Schritt (raa):

  Induktionsvoraussetzung: Wenn Γ alle Hypothesen von $\overset{\mathscr{D}}{\bot}$ enthält, dann $\Gamma \models \bot$. Da $[\![\bot]\!]_v = 0$ für alle v ist dies gleichbedeutend mit der Tatsache, dass es kein v gibt so dass $[\![\psi]\!]_v = 1$ für alle $\psi \in \Gamma$.

Zu zeigen: Wenn Γ' alle Hypothesen von $\dfrac{\begin{array}{c}[\neg\phi]\\ \mathscr{D}\\ \bot\end{array}}{\phi}$ enthält, dann $\Gamma' \models \phi$.

Beweis: Nehmen wir an, dass Γ' alle Hypothesen von $\dfrac{\begin{array}{c}[\neg\phi]\\ \mathscr{D}\\ \bot\end{array}}{\phi}$ enthält, und dass $\Gamma' \not\models \phi$. Dann gäbe es eine Valuation v mit $[\![\psi]\!]_v = 1$ für alle $\psi \in \Gamma'$ und $[\![\phi]\!]_v = 0$, also $[\![\neg\phi]\!]_v = 1$. Sei $\Gamma'' = \Gamma' \cup \{\neg\phi\}$, dann erfüllt Γ'' die Induktionsvoraussetzung, also kann es kein v geben so dass $[\![\psi]\!]_v = 1$ für alle $\psi \in \Gamma' \cup \{\neg\phi\}$, insbesondere $\psi \in \Gamma'$; also muss $\Gamma' \models \phi$ gelten.

□

Außer der tiefen Befriedigung, dass unsere Ableitungen alle auf semantische Weise "richtig" sind, hat dieses Lemma auch eine praktische Bedeutung, um nämlich Widersprüche aufzuzeigen. Dazu nutzen wir die Kontraposition:

Korollar 1.17 *Wenn $\Gamma \not\models \phi$, dann $\Gamma \nvdash \phi$*

Jetzt können wir mit semantischen Methoden wie Resolution und Erfüllbarkeit $\Gamma \not\models \phi$ zeigen, und haben damit gezeigt, dass wir gar keinen Beweis von $\Gamma \vdash \phi$ finden können (das wäre sonst nämlich mit unseren Methoden gar nicht möglich).

Vorlesung vom 16.05.2023: Korrektheit und Vollständigkeit II

1.6.2 Konsistenz

Übung 1.19 (Aufwärmübung) *Formalisiert folgenden Beweis:*

> *Wenn Hugo Kaffee trinkt, isst Hugo auch Kuchen. Wenn Hugo Kuchen isst, wird er dick.*
> *Hugo trinkt Kaffee und ist schlank.*
> *Also studiert Hugo Informatik.*

Mit folgenden Atomen:

- *A — Hugo trinkt Kaffe*
- *B — Hugo isst Kuchen*
- *C — Hugo ist dick, $\neg C$ — Hugo ist schlank*
- *D — Hugo studiert Informatik.*

Warum ist der Beweis richtig? Und warum ist er unsinnig?

Konsistenz ist ein Schlüsselbegriff der symbolischen Logik. Intuitiv ist eine Logik konsistent, wenn sich kein Blödsinn damit anstellen läßt (wie in der Aufwärmübung oben). Präziser formuliert: ich kann nichts gegensätzliches ableiten, also nicht *sowohl A als auch* $\neg A$. Es stellt sich heraus, das ist dasselbe als wenn ich $\bot$ ableiten kann. Wir fassen das mal etwas formaler:

Definition 1.14 (Konsistenz) *Eine Menge* Γ *von Aussagen ist* konsistent, *wenn* $\Gamma \nvdash \bot$, *und* inkonsistent, *wenn* $\Gamma \vdash \bot$.

Lemma 1.18 *Die folgenden drei Aussagen sind äquivalent:*

(i) Γ *ist inkonsistent;*

(ii) Es gibt ein ϕ *so dass* $\Gamma \vdash \phi$ *und* $\Gamma \vdash \neg\phi$*;*

(iii) $\Gamma \vdash \phi$ *für alle* ϕ.

Beweis. (i)$\Longrightarrow$(iii): Wenn $\Gamma \vdash \bot$ können wir mit der Regel *false* eine Ableitung $\bot \vdash \phi$ für ein beliebiges ϕ hinzufügen, so dass $\Gamma \vdash \phi$. (iii)$\Longrightarrow$(ii): Trivial. (ii)$\Longrightarrow$(i): Aus ϕ und $\neg\phi$ folgt mit der Regel $\neg E$ $\bot$, also $\Gamma \vdash \bot$. □

Klausel (iii) zeigt, warum inkonsistente Mengen von Aussagen ("Theorien") uninteressant sind: wenn ich alles beweisen kann, ist ein Beweis nichts wert. Dieses Lemma läßt sich auch genau anderherum formulieren:

Lemma 1.19 *Die folgende drei Aussagen sind äquivalent:*

(i) Γ *ist konsistent;*

(ii) Es gibt kein ϕ *so dass* $\Gamma \vdash \phi$ *und* $\Gamma \vdash \neg\phi$*;*

(iii) Es gibt mindest ein ϕ *so dass* $\Gamma \nvdash \phi$.

Lemma 1.20 *Es gilt:*

(i) Wenn $\Gamma \cup \{\neg\phi\}$ *inkonsistent ist, dann* $\Gamma \vdash \phi$.

(ii) Wenn $\Gamma \cup \{\phi\}$ *inkonsistent ist, dann* $\Gamma \vdash \neg\phi$.

Beweis.

(i) Die Voraussetzung gibt uns eine Herleitung für $\bot$, wobei die offenen Voraussetzungen entweder in Γ oder $\neg\phi$ sind. Diese können wir mit der Regel *raa* zu einer Herleitung ϕ verknüpfen, in der die Voraussetzung $\neg\phi$ geschlossen wird; damit gilt $\Gamma \vdash \phi$.

(ii) Analog mit $\neg I$.

□

Definition 1.15 (Maximal konsistent) *Eine Menge* Γ *von Aussagen ist* maximal konsistent *gdw.*

(i) Γ *ist konsistent, und*

(ii) wenn $\Gamma \subseteq \Delta$ *und* Δ *konsistent, dann* $\Delta = \Gamma$,

Man könnte das auch anders formulieren: wenn Γ eine echte Untermenge von Δ ist, dann ist Δ inkonsistent (*i.e.* jede weitere Aussage macht Γ inkonsistent).

Jede Menge von Axiomen ist in eine maximal konsistente Menge von Axiomen einbettbar. Jede? Nein, natürlich nur konsistente:

Lemma 1.21 *Jede konsistente Menge Γ von Axiomen ist in einer maximal konsistente Menge Δ enhalten.*

Beweis. Die Konstruktion nutzt die Aufzählbarkeit der Aussagen *Prop* mit einer Art "Aschenputtel-Prinzip" (die guten ins Töpfchen, die schlechten ins Kröpfchen), um alle mit Γ konsistenten herauszusortieren.

Etwas präziser ausgedrückt: Sei $\langle\phi_i\rangle_{i\in\mathbb{N}}$ eine Sequenz aller Aussagen. Wir konstruieren eine Sequenz Γ_i von Mengen von Aussagen wie folgt:

$$\begin{aligned}\Gamma_0 &= \Gamma\\ \Gamma_{i+1} &= \begin{cases}\Gamma_i \cup \{\phi_i\} & \Gamma_i \cup \{\phi_i\} \text{ konsistent}\\ \Gamma_i & \text{sonst}\end{cases}\\ \Delta &= \bigcup_{0\le i}\{\Gamma_i\}\end{aligned}$$

Wenn Γ konsistent ist, sind auch alle Γ_i konsistent. Damit ist auch Δ konsistent, denn wenn $\Delta \vdash \bot$ gäbe es eine Ableitung mit endlich vielen Hypothesen, die dann in einer der Γ_i wären.

Es bleibt zu zeigen, dass Δ maximal konsistent ist. Sei $\Delta \subseteq \Lambda$ und Λ konsistent. Für jedes $\psi \in \Lambda$ ist $\psi = \phi_m$ für ein $m \in \mathbb{N}$ (jede Aussage hat eine Aufzählungsnummer). Wir betrachten die korrespondiere Menge Γ_m: $\Gamma_m \subseteq \Delta \subseteq \Lambda$, und Λ konsistent, also ist $\Gamma_m \cup \{\phi_m\}$ auch konsistent. Deshalb ist $\Gamma_{m+1} = \Gamma_m \cup \{\phi_m\}$, und damit $\phi_m \in \Gamma_{m+1}$, und $\phi_m \in \Delta$, also ist $\Lambda \subseteq \Delta$ und damit $\Delta = \Lambda$. □

Übung 1.20 *Ist die maximal konsistente Menge Δ eindeutig bestimmt?*

Nein, es hängt davon ab, wie wir die Aussagen aufzählen. Betrachten wir ein triviales Beispiel:

$$\Gamma = \{A \longrightarrow B, B \longrightarrow C\}$$

Wenn unsere Aufzählung so ist, dass wir zuerst alle Atome aufzählen... dann werden wir nicht fertig (dann wir haben ja schon mal abzählbar unendliche viele Atome). Tatsächlich ist es recht instruktiv, sich zu überlegen, wie wir Aussagen effektiv aufzählen können; wir werden diesem Problem später als "Gödelisierung" wiederbegegnen (und eine Lösung von Kurt Gödel dazu kennenlernen).

Auf jeden Fall ist in $A \longrightarrow C \in \Delta$. Aber was ist mit A? Wenn $A \in \Delta$, dann auch $B, C \in \Delta$. Genausogut könnte $\neg A \in \Delta$ sein, dann aber auch $\neg B, \neg C \in \Delta$. Gleiches gilt für komplett andere Atome, wie D oder E; hier kann $D \in \Delta$ oder $\neg\mathscr{D} \in \Delta$ sein (natürlich nicht beides, aber tatsächlich mindest eines von beiden, wie wir gleich zeigen). Wir sehen also, dass die maximal konsistente Menge Δ von der Aufzählungsreihenfolge $\langle\phi_i\rangle$ abhängt — je nachdem, ob zuerst A oder $\neg A$ zuerst aufgezählt werden, kommen sie zu Γ_i (und damit Δ) hinzu.

Maximal konsistente Mengen haben zwei magische Eigenschaften, die für den Vollständigkeitsbeweis nötig sind:

Lemma 1.22 (Magische Eigenschaften) *Sei Δ maximal konsistent, dann:*

(i) Δ *ist unter Ableitbarkeit geschlossen, i.e. wenn* $\Delta \vdash \phi$ *dann* $\phi \in \Delta$*;*

(ii) für alle ϕ *ist entweder* $\phi \in \Delta$ *oder* $\neg\phi \in \Delta$*;*

(iii) für alle ϕ, ψ *gilt:* $\phi \longrightarrow \psi \in \Delta$ *gdw. (*$\phi \in \Delta$ *dann* $\psi \in \Delta$*).*

Beweis.

(i) Sei $\Delta \vdash \phi$ und $\phi \notin \Delta$; dann müsste $\Delta \cup \{\phi\}$ inkonsistent sein. Also $\Delta \vdash \neg\phi$, aber dann wäre Δ inkonsisten. ↯

(ii) Es können nicht beide $\phi, \neg\phi$ in Δ enthalten sein. Also betrachte $\Delta' = \Delta \cup \{\phi\}$; wenn das inkonsistent ist, dann (mit Lemma 1.20) und (i) $\neg\phi \in \Delta$; wenn es konsistent ist, dann ist wegen der Maximalität $\phi \in \Delta$.

(iii) Sei $\phi \longrightarrow \psi \in \Delta$ und $\phi \in \Delta$. Dann ist mit $\longrightarrow I$ $\Delta \vdash \psi$, und nach (i) $\psi \in \Delta$. In der anderen Richtung, nehmen wir an, dass wenn $\phi \in \Delta$ dann $\psi \in \Delta$ und zeigen dass $\phi \longrightarrow \psi \in \Delta$ nach Fallunterscheidung über ϕ: entweder $\phi \in \Delta$, dann $\psi \in \Delta$ und $\phi \longrightarrow \psi \in \Delta$; oder $\phi \notin \Delta$, dann $\neg\phi \in \Delta$ (nach (ii)), also $\Delta \vdash \neg\phi$ und damit $\Delta \vdash \phi \longrightarrow \psi$.

□

Aus Lemma 1.22(i) folgt, dass jede aus Γ ableitbare Formel in einer maximal abgeschlossenenen Menge Δ, die Γ enthält, enthalten sein muss. Ist eine Formel nicht aus Γ ableitbar, ist nach Lemma 1.20(i) $\Gamma \cup \{\neg\phi\}$ konsistent, damit gibt es mindestens ein maximal konsistentes Δ, was $\neg\phi$ und damit nicht ϕ enthält. Daraus können wir schließen, dass die Schnittmenge aller maximal konsistenten Mengen Δ welche Γ enthalten genau die Menge aller aus Γ ableitbaren Formeln ist (die *Theorie* von Γ).

1.6.3 Vollständigkeit

Wir wollen die Vollständigkeit zeigen, *i.e.* wenn $\Gamma \models \phi$ dann auch $\Gamma \vdash \phi$.

Hier ist die Beweisstrategie:

1. Wir zeigen die Kontraposition: wenn $\Gamma \nvdash \phi$, dann $\Gamma \not\models \phi$. Dazu müssen wir zeigen, dass es eine Valuation gibt, mit der alle Aussagen in Γ zu wahr auswerten, aber ϕ zu falsch.
2. Um das zu zeigen, betten wir Γ in ein maximal konsistentes Δ ein, dass ϕ nicht enthält.
3. Damit können wir eine Belegung konstruieren, die alle Aussagen in Δ zu wahr auswertet, aber nicht ϕ. Damit ist $\Gamma \not\models \phi$ gezeigt.

Schlüssel zu dem Beweis ist also, dass wir für eine konsistente Menge von Aussagen eine Belegung (der Atome) konstruieren können, mit der alle Aussagen wahr auswerten. Das ist unser erstes Lemma:

Lemma 1.23 *Für ein konsistentes* Γ *gibt es eine Belegung* v *so dass* $[\![\phi]\!]_v = 1$ *für alle* $\phi \in \Gamma$*.*

Beweis. Der Beweis erfolgt in drei Schritten:

1. Nach Lemma 1.21 ist Γ in einem maximal konsistenten Δ enthalten. Damit definieren wir
$$v(p) := \begin{cases} 1 & p \in \Delta \\ 0 & \text{sonst} \end{cases}$$
2. Jetzt zeigen wir: $[\![\phi]\!]_v = 1$ gdw. $\phi \in \Delta$ durch Induktion über ϕ:
   - Basisfall $\phi \equiv p \in P$: gilt nach Definition von v.
   - Basisfall $\phi \equiv \bot$: Trivial ($[\![\bot]\!]_v = 0$ genauso $\bot \notin \Delta$).
   - Induktionsschritt: $\phi \equiv \psi \longrightarrow \sigma$: Es gilt
$$\begin{aligned} & [\![\psi \longrightarrow \sigma]\!]_v = 0 \\ \text{gdw. } & [\![\psi]\!]_v = 1, [\![\sigma]\!]_v = 0 \\ \text{gdw. } & \psi \in \Delta, \sigma \notin \Delta \quad \text{nach I.v.} \\ \text{gdw. } & \psi \longrightarrow \sigma \notin \Delta \quad \text{nach Lemma 1.22(iii)} \end{aligned}$$
3. Da $\Gamma \subseteq \Delta$ haben wir $[\![\phi]\!]_v = 1$ für alle $\phi \in \Gamma$.

□

Lemma 1.24 $\Gamma \nvdash \phi$ [7] *gdw. es gibt eine Belegung v so dass $[\![\psi]\!]_v = 1$ für $\psi \in \Gamma$, und $[\![\phi]\!]_v = 0$.*

Beweis.

$\Gamma \nvdash \phi$ gdw. $\Gamma \cup \{\neg\phi\}$ konsistent (nach Lemma 1.20);
gdw. es gibt Belegung v so dass $[\![\psi]\!]_v = 1$ für $\psi \in \Gamma \cup \{\neg\phi\}$ (nach Lemma 1.23)
gdw. es gibt Belegung v so dass $[\![\psi]\!]_v = 1$ für $\psi \in \Gamma$ und $[\![\phi]\!]_v = 0$. □

Jetzt können wir unser Hauptresultat beweisen:

Theorem 1.25 (Vollständigkeit) *Wenn $\Gamma \models \phi$, dann $\Gamma \vdash \phi$.*

Beweis. Wir zeigen die Kontraposition: wenn $\Gamma \nvdash \phi$, dann $\Gamma \not\models \phi$. Nehmen wir an, dass $\Gamma \nvdash \phi$. Dann gibt es nach Lemma 1.24 eine Belegung v, so dass $[\![\psi]\!]_v = 1$ für alle $\psi \in \Gamma$, und $[\![\phi]\!]_v = 0$, *i.e.* $\Gamma \not\models \phi$. □

Die Vollständigkeit der Aussagenlogik hat nützliche Auswirkungen. Statt syntaktische Beweise zu konstruieren, können wir auf die semantischen Verfahren wie Erfüllbarkeit (oder einfach erschöpfende Suche) zurückgreifen. Daraus folgt auch direkt die *Entscheidbarkeit* der Aussagenlogik: für jede Aussage ϕ können wir zeigen, ob $\vdash \phi$ oder $\nvdash \phi$.

Was aus unserem Beweis der Vollständigkeit nicht folgt ist eine Konstruktion der Herleitung $\Gamma \vdash \phi$. Dazu könnte man beispielsweise auf die Resolution nutzen. Zuerst konstruieren wir einen Beweis, dass $\vdash \phi \longleftrightarrow \phi'$ mit ϕ' in KNF (*i.e.* wir überführen ϕ in KNF). Um $\vdash \phi$ zu zeigen, konstruieren wir eine Resolution, die aus $\neg\phi'$ die leere Klausel ableitet. Diesen Resolutionsbeweis können wir dann nutzen, um einen Beweis für ϕ im natürlichen Schließen zu konstruieren. Das ist jetzt keine leere Theorie — der Theorembeweiser Isabelle nutzt dieses Verfahren, um mit Resolutionsbeweisern Beweise im natürlichen Schließen zu konstruieren.

[7] Wir brauchen hier nicht vorauszusetzen, dass Γ konsistent ist, dass folgt aus $\Gamma \nvdash \phi$.

1.7 Zusammenfassung

Mit Korrektheit und Vollständigkeit schließen wir unsere Behandlung der Aussagenlogik. Wir haben die Aussagenlogik kennengelernt, und gesehen, wie wir Formeln der Aussagenlogik eine Bedeutung (via Wahrheitstabellen) zuweisen können. Damit können wir zwischen immer wahren Aussagen (Tautologien), erfüllbaren und unerfüllbaren Aussagen unterscheiden. Wir haben Beweisverfahren wie SAT-Solving und Resolution kennengelernt, die vollautomatisch arbeiten, und das natürliche Schließen, mit dem wir Beweise aus elementaren Regeln manuell konstruieren. Wir haben gezeigt, dass alle Formeln, die wir mit dem natürlichen Schließen beweisen können, auch semantisch wahr sind (*Korrektheit*), und das andererseits, dass alle semantisch wahren Formeln auch im natürlichen Schließen bewiesen werden können (*Vollständigkeit*). In einer Formel:

$$\Gamma \vdash \phi \Longleftrightarrow \Gamma \models \phi$$

Zusammenfassend kann man sagen:

- Aussagenlogik ist in seiner Ausdrucksmächtigkeit beschränkt, wie wir in vielen Beispielen gesehen haben, aber dafür kann ich viel automatisch beweisen. (Mit anderen Worten, der Computer kann für uns arbeiten. Kein Wunder, dass Aussagenlogik bei Informatikern so beliebt ist.)
- Das natürliche Schließen ist mühsam, skaliert aber problemlos auch für mächtigere Logiken wie den Prädikatenkalkül, den wir als nächstes kennenlernen werden.

Kapitel 2

Prädikatenlogik

Vorlesung vom 22.05.2023: Prädikatenlogik Einführung

2.0.1 Motivation

Aussagenlogik hilft Argumentationsstruktur zu beschreiben, abstrahiert aber zu stark

$$\frac{\text{Alle Menschen sind sterblich}\quad \text{Sokrates ist ein Mensch}}{\text{Sokrates ist sterblich}} \qquad \frac{\text{Jedes } P \text{ ist auch ein } Q\quad s \text{ ist ein } P}{s \text{ ist } Q} \qquad \frac{\forall x.P(x) \longrightarrow Q(x)\quad P(s)}{Q(s)}$$

- $\forall n \in \mathbb{N}.\exists n' \in \mathbb{N}.n' = \mathrm{nf}(n)$
- $\forall n \in \mathbb{N}.\, \mathrm{nf}(n) \neq 0$
- $0 \in \mathbb{N}$

Definition 2.1 (Signatur) *Eine* Signatur $\tau = \langle \mathscr{F}, \mathscr{R} \rangle$ *besteht aus disjunkten Mengen von Funktionssymbolen* $\mathscr{F}$ *und Relationssymbolen* $\mathscr{R}$*. Jedes dieser Symbole hat eine feste endliche Stelligkeit, auch Arität genannt. Nullstellige Funktionssymbole nennen wir Konstantensymbole.*

Notation: In einer konkreten Signatur geben wir die Stelligkeit der Symbol als Superscript an. Zum Beispiel denotiert die Signatur

$$\langle \{a^0, f^1, g^2\}, \{R^0, P^2, S^1\} \rangle$$

eine Menge von Funktionssymbolen a hat Stelligkeit 0*, f hat Stelligkeit* 1 *und g hat Stelligkeit* 2

eine Menge von Relationssymbolen R hat Stelligkeit 0*, P hat Stelligkeit* 2 *und S hat Stelligkeit* 1

Vorlesung vom 14.06.2023: Prädikatenlogik VI

Der wesentliche Unterschied zwischen der Prädikatenlogik und der Aussagenlogik ist dass die Prädikatenlogik den Objekten, die vorher nur atomare Aussagen waren, eine Struktur gibt. Deshalb gibt es

syntaktisch auch zwei Kategorien: die *Terme*, welche diese Objekte beschreiben, und *Formeln*, welche den Aussagen der Aussagenlogik entsprechen.

Definition 2.2 (Terme) *Gegeben eine Signatur τ und eine Menge X von Objektvariablen, dann sind die* Terme *zu dieser Signatur die kleinste Menge $Term_\tau$ so dass*

$$\begin{aligned} X &\subseteq Term_\tau \\ f^n \in \tau, t_1, \ldots, t_n \in Term_\tau \text{ dann } f(t_1, \ldots, t_n) &\in Term_\tau \end{aligned}$$

Für die zweite Klausel behandeln wir den Fall $n = 0$ syntaktisch vereinfachend, indem wir c statt $c()$ schreiben; in diesem Fall ist c eine Konstante.

Definition 2.3 (Formeln) *Gegeben eine Signatur τ und eine Menge X von Objektvariablen, dann sind die* Formeln *über τ die kleinste Menge $Prop_\tau$ so dass mit $\phi, \psi \in Prop_\tau$:*

$$\begin{aligned} t_1, t_2 \in Term_\tau \text{ dann } s = t &\in Prop_\tau \\ P^n \in \tau, t_1, \ldots, t_n \in Term_\tau \text{ dann } P(t_1, \ldots, t_n) &\in Prop_\tau \\ \bot &\in Prop_\tau \\ \neg\phi &\in Prop_\tau \\ \phi \wedge \psi &\in Prop_\tau \\ \phi \vee \psi &\in Prop_\tau \\ \phi \longrightarrow \psi &\in Prop_\tau \\ \phi \longleftrightarrow \psi &\in Prop_\tau \\ x \in X \text{ dann } \forall x.\, \phi &\in Prop_\tau \\ x \in X \text{ dann } \exists x.\, \phi &\in Prop_\tau \end{aligned}$$

Variablen können frei oder gebunden sein. Für eine Formel $\phi \in Prop_\tau$ und eine Variable $x \in X$ ist

(i) x ist *bindend* in $\forall x.\phi$, $\exists x.\psi$;

(ii) für $\forall x.\phi$ und $\exists x.\phi$ ist x in allen Teilformeln von ϕ *gebunden*;

(iii) Ansonsten ist x *frei*.

Damit können wir die Menge $Var(\phi)$ aller *freien* Variablen für $\phi \in Prop_\tau$ wie folgt induktiv definieren:

$$\begin{aligned}
Var(x) &= \{x\} \\
Var(f(t_1,\ldots,t_n)) &= Var(t_1) \cup \ldots \cup Var(t_n) \\
Var(t_1 = t_n) &= Var(t_1) \cup Var(t_2) \\
Var(P(t_1,\ldots,t_n)) &= Var(t_1) \cup \ldots \cup Var(t_n) \\
Var(\bot) &= \emptyset \\
Var(\neg\phi) &= Var(\phi) \\
Var(\phi_1 \wedge \phi_2) &= Var(\phi_1) \cup Var(\phi_2) \\
Var(\phi_1 \vee \phi_2) &= Var(\phi_1) \cup Var(\phi_2) \\
Var(\phi_1 \longrightarrow \phi_2) &= Var(\phi_1) \cup Var(\phi_2) \\
Var(\phi_1 \longleftrightarrow \phi_2) &= Var(\phi_1) \cup Var(\phi_2) \\
Var(\forall x.\, \phi) &= Var(\phi) \setminus \{x\} \\
Var(\exists x.\, \phi) &= Var(\phi) \setminus \{x\}
\end{aligned}$$

Die Ersetzung von Variablen ist wie in der Aussagenlogik rekursiv über die Formelstruktur definiert, aber durch die Quantoren überraschend subtil. Zum einen ersetzen wir Variablen durch *Terme*, nicht durch Aussagen, weil wir keine atomaren Aussagen mehr haben. Wenn wir eine Variable x in einer Formel ϕ durch einen Term t ersetzen, geschrieben $\phi[t/x]$, dann gibt es dreierlei zu beachten:

1. Ist x bindend in ϕ, dann wird *nicht* ersetzt.

2. Ist eine andere Variable y bindend in ϕ, dan müssen wir bei der rekursiven Ersetzung von x in ϕ zum einen beachten, dass y nicht in den freien Variablen von t vorkommt (ansonsten würden diese freien Variablen inkorrekterweise gebunden).

3. Ist dies der Fall, dann müssen wir die bindende Variable y umbenennen, und zwar in eine neue Variable, die *frisch* ist, *i.e.* nicht in ϕ oder t vorkommt.

Die Definition der Substitution ist damit:

Definition 2.4 (Substitution) *Für $\phi \in Prop_\tau$, $x \in X$ und $t \in Term_\tau$ definieren wir die Ersetzung von x in*

ϕ durch t, geschrieben $\phi[t/x]$, rekursiv über der Struktur von ϕ wie folgt:

$$
\begin{aligned}
y[t/x] &= \begin{cases} t & x = y \\ y & x \neq y \end{cases} \\
f(t_1,\ldots,t_n)[t/x] &= f(t_1[t/x],\ldots,t_n[t/x]) \\
\bot[t/x] &= \bot \\
(\neg\phi)[t/x] &= \neg(\phi[t/x]) \\
(\phi_1 \wedge \phi_2)[t/x] &= (\phi_1[t/x]) \wedge (\phi_2[t/x]) \\
(\phi_1 \vee \phi_2)[t/x] &= (\phi_1[t/x]) \vee (\phi_2[t/x]) \\
(\phi_1 \longrightarrow \phi_2)[t/x] &= (\phi_1[t/x]) \longrightarrow (\phi_2[t/x]) \\
(\phi_1 \longleftrightarrow \phi_2)[t/x] &= (\phi_1[t/x]) \longleftrightarrow (\phi_2[t/x]) \\
P(\phi_1,\ldots,\phi_n)[t/x] &= P(\phi_1[t/x],\ldots,\phi_n[t/x]) \\
(\forall y.\,\phi)[t/x] &= \begin{cases} \forall y.\,\phi & x = y \\ \forall y.\,(\phi[t/x]) & x \neq y, y \notin Var(t) \\ \forall z.\,((\phi[z/y])[t/x]) & x \neq y, y \in Var(t), z \notin Var(t) \cup Var(\phi) \end{cases} \\
(\exists y.\,\phi)[t/x] &= \begin{cases} \exists y.\,\phi & x = y \\ \exists y.\,(\phi[t/x]) & x \neq y, y \notin Var(t) \\ \exists z.\,((\phi[z/y])[t/x]) & x \neq y, y \in Var(t), z \notin Var(t) \cup Var(\phi) \end{cases}
\end{aligned}
$$

Man beachte, dass sich durch Substitution gebundende Variablen plötzlich umbenennen. Folgende Übung zeigt das in Aktion:

Übung 2.1 *Berechne folgende Substitution:*

$$(\forall x.\,(\forall y.\,z = y) \wedge u = x)[y + x/z]$$

Vorlesung vom 22.05.2023: Prädikatenlogik VII

2.1 Natürliches Schließen

Für syntaktische Beweise in Prädikatenlogik erster Stufe können wir das System aus Abschnitt 1.5 erweitern. Aus den gleichen Gründen wie dort befassen wir uns erst mit einer *Kernsprache*. Wir haben in Abschnitt 1.3.10 gesehen, wie wir die Aussagenlogik aus wenigen Operatoren aufbauen können, und in Abschnitt 1.5.2, wie wir mit Regeln für wenige Operatoren anfangen, und die Regeln der restlichen Operatoren daraus herleiten können. Für die Prädikatenlogik gilt das gleiche: da die Prädikateligk die gleichen Konnektive wie die Aussagenlogik zuzüglich $\forall$ und $\exists$ hat, und wir entweder $\forall$ durch $\exists$ ausdrücken können, oder umgekehrt, reicht uns für die gesamte Prädikatenlogik die gleichen Kernsprachen wie für die Aussagenlogik zuzüglich entweder $\forall$ oder $\exists$.

Konkret wählen wir eine Kernsprache, welche nur die Konnektive $\wedge, \longrightarrow, \bot$ und $\forall$ enthält, und leiten die Regel für $\exists$ später ab.

2.1.1 Natürliches Schließen mit dem Allquantor

Die Regeln für die ersten drei Konnektive — die auch in der Aussagenlogik sind — bleiben genau gleich. (Streng genommen sind es natürlich etwas andere Regeln, weil sie jetzt über Formeln in $Prop_\tau$ und nicht mehr $Prop$ formuliert sind, aber wenn wir sie hinschreiben, sehen sie genauso aus.)

Für die Relationen $P^n \in \tau$ gibt es keine strukturellen Regeln (also Einführungs- und Eliminationsregeln wie wir sie für die Konnektive aus der Aussagenlogik kennen), es können lediglich Axiome formuliert werden.

Damit bleibt also nur der Allquantor. Für diesen gibt es eine Einführungs- und eine Eliminationsregel:

$$\frac{\phi}{\forall x.\,\phi}\ \forall I\ \ (*) \qquad\qquad \frac{\forall x.\,\phi}{\phi[t/x]}\ \forall E$$

Für die Regel $\forall I$ gilt hier die *Eigenvariablenbedingung:* x darf nicht *frei* in den offenen Vorbedingungen von ϕ sein. Bei der Regel $\forall E$ kann es durch die Substitution zu Umbenennungen kommen (in ϕ, nicht im Rest des Baumes).

Was bedeutet die Eigenvariablenbedingung? Die Idee hinter der Regel $\forall I$ ist, dass wenn wir ϕ für ein *beliebiges* x gezeigt haben, dann können wir über dem x quantifizieren, und es später (Regel $\forall E$) durch einen x-beliebigen Term t ersetzen. Beliebig heißt allerdings, dass wir keinerlei Annahmen über x in dem Beweis gemacht haben dürfen, was sich technisch darin niederschlagen würde, dass x in den offenen Annahmen des Beweisbaumes auftaucht.

Hier ist ein einfaches Gegenbeispiel, für eine Signatur mit einer Konstanten c^0:

$$\dfrac{\dfrac{\dfrac{\dfrac{[x=c]}{\forall x.\,x=c}}{x=c\longrightarrow\forall x.\,x=c}}{\forall x.\,x=c\longrightarrow\forall x.\,x=c}}{c=c\longrightarrow\forall x.\,x=c}$$

Jetzt gilt $c=c\longrightarrow\forall x.\,x=c$ aber sicherlich nicht in Strukturen mit zwei oder mehr Elementen; die Herleitung ist also falsch (eben genau wegen der verletzten Eigenvariablenbedingung in der zweiten Zeile).

Hier ist ein etwas banal anmutendes Beispiel:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\forall x.\,\forall y.\,\phi]^1}{\forall y.\,\phi}\ \forall E}{\phi}\ \forall E}{\forall x.\,\phi}\ \forall I}{\forall y.\,\forall x.\,\phi}\ \forall I}{\forall x.\,\forall y.\,\phi\longrightarrow\forall y.\,\forall x.\,\phi}\ \longrightarrow I^1$$

So ganz trivial ist das allerdings dann doch nicht, wenn wir müssen natürlich für die beiden Anwendungen von $\forall I$ die Eigenvariablenbedingung zeigen. Woher wissen wir denn jetzt genau, dass in den offenen Vorbedingungen x resp. y nicht frei auftreten? Welches sind überhaupt die offenen Vorbedingungen? Es hilft, sich den relevanten Teilbeweis zu vergegenwärtigen:

$$\dfrac{\dfrac{\dfrac{\dfrac{\forall x.\,\forall y.\,\phi}{\forall y.\,\phi}\ \forall E}{\phi}\ \forall E}{\forall x.\,\phi}\ \forall I}{\forall y.\,\forall x.\,\phi}\ \forall I$$

Die relevante offene Teilbedingung ist also $\forall x.\forall y.\phi$; und hier sind natürlich weder x noch y frei.

Übung 2.2 *Beweise folgende Theoreme durch natürliches Schließen:*

- $\vdash (\forall x.(\phi \wedge \psi)) \longrightarrow (\forall x.\phi) \wedge (\forall x.\psi)$
- $\vdash (\forall x.\phi) \wedge (\forall x.\psi) \longrightarrow (\forall x.(\phi \wedge \psi))$

Manchmal müssen wir die Eigenvariablenbeding explizit als Seitenbedingung annotieren, wie in diesem Beispiel:

$$\dfrac{\dfrac{\dfrac{[\phi]^2 \qquad \dfrac{[\forall x.\phi \longrightarrow \psi]^1}{\phi \longrightarrow \psi}\,\forall E}{\psi}\,mp}{\forall x.\psi}\,\forall I(*)}{\phi \longrightarrow (\forall x.\psi)}\,\longrightarrow I^2}{(\forall x.\phi \longrightarrow \psi) \longrightarrow (\phi \longrightarrow (\forall x.\psi))}\,\longrightarrow I^1$$

Hier ist für den Schritt $(*)$ die Bedingung $x \notin Var(\phi)$ nötig, *i.e.* wir zeigen

$$x \notin Var(\phi) \text{ dann } \vdash (\forall x.\phi \longrightarrow \psi) \longrightarrow (\phi \longrightarrow (\forall x.\psi))$$

2.1.2 Korrektheit

Wir zeigen zuerst einmal, dass der Kalkül korrekt ist, *i.e.* wir können nur semantisch gültige Formeln herleiten. Dazu benötigen wir folgende:

Definition 2.5 *Sei Γ eine Menge von Formeln und ϕ eine Forme, mit $Var(\Gamma) = \{Var(\psi) \mid \psi \in \Gamma\}$ und $Var(\Gamma) \cup Var(\phi) = \{x_1,\ldots,x_n\}$. Wenn $\mathbf{a} = \langle a_1,\ldots,a_n\rangle$ eine Sequenz von Elementen aus $\mathfrak{A}$, dann ist $Gamma(\mathbf{a})$ definiert als die gleichzeitige Ersetzung von x_i mit $\bar{a}_i$ (analog $\phi(\mathbf{a})$). Dann ist*

$$\begin{aligned}\mathfrak{A} \models \Gamma(\mathbf{a}) &gdw.\ \mathfrak{A} \models \psi \text{ für alle } \psi \in \Gamma(\mathbf{a})\\ \Gamma \models \phi\, &gdw\ \mathfrak{A} \models \Gamma(\mathbf{a}) \text{ dann } \mathfrak{A} \models \phi(\mathbf{a}) \text{ für alle } \mathfrak{A},\mathbf{a}.w\end{aligned}$$

Für den Fall, dass Γ und ϕ keine freien Variablen enthalten vereinfacht sich dass zu der bekannteren Definition

$$\Gamma \models \phi \text{ gdw } (\mathfrak{A} \models \Gamma \text{ dann } \mathfrak{A} \models \phi \text{ für alle } \mathfrak{A}$$

Jetzt können wir die Korrektheit (*soundness*) zeigen:

Theorem 2.1 (Korrektheit des natürlichen Schließens) *Wenn $\Gamma \vdash \phi$, dann $\Gamma \models \phi$.*

Beweis. Der Beweis ist eine Erweiterung des Beweises der Korrektheit des natürlichen Schließens für die Aussagenlogik (Theorem 1.16), da insbesondere die Definition der semantischen Gültigkeit und Folgerung in der Aussagenlogik als Sonderfall der semantischen Gültigkeit und Folgerung für die Prädikatenlogik begriffen werden kann.

Analog zu Theorem 1.16 erfolgt der Beweis durch Induktion über der Ableitung $\mathscr{D}$ von $\Gamma \vdash \phi$. Zu zeigen bleiben die Induktionsschritte für $\forall I$ und $\forall E$; wir zeigen hier nur den ersten.

Induktionsvoraussetzung: $\Gamma \models \phi$, *i.e.* wenn $\mathfrak{A} \models \Gamma(\mathbf{a})$ dann $\mathfrak{A} \models \phi(\mathbf{a})$ für alle $\mathfrak{A}, \mathbf{a}$.

Zu zeigen: $\Gamma \models \forall x.\, \phi$, *i.e.* wenn $\mathfrak{A} \models \Gamma(\mathbf{a})$ dann $\mathfrak{A} \models (\forall x.\, \phi)(\mathbf{a}')$ für alle $\mathfrak{A}, \mathbf{a}'$.

Beweis: $\mathfrak{A} \models (\forall x.\, \phi)(\mathbf{a}')$ ist definiert als $\mathfrak{A} \models \phi[\bar{a}/x](\mathbf{a}')$ für alle $a \in \mathfrak{A}$; mit $\mathbf{a} \stackrel{def}{=} \langle \bar{a}, \mathbf{a}' \rangle$ (*i.e.* die Sequenz $\mathbf{a}$ ist genau $\bar{a}$ verkettet mit $\mathbf{a}'$[1]) ist $\phi[\bar{a}/x](\mathbf{a}') = \phi(\mathbf{a})$, und damit folgt $\mathfrak{A} \models \phi[\bar{a}/x](\mathbf{a}')$ aus der Induktionsvoraussetzung $\mathfrak{A} \models \phi(\mathbf{a})$. □

Übung 2.3 *Wo bleibt die Eigenvariablenbedingung im Beweis oben?*

2.1.3 Der Existenzquantor

Für den Existenzquantor können wir zwei Regeln ableiten:

$$\frac{\phi[t/x]}{\exists x.\, \phi}\ \exists I \qquad\qquad \frac{\exists x.\, \phi \qquad \begin{array}{c}[\phi]\\ \vdots \\ \psi\end{array}}{\psi}\ \exists E(*)$$

Auch hier haben wir die Eigenvariablenbedingung (*): x ist nicht frei in ψ, oder in einer offenen Hypothese (außer natürlich ϕ) der Ableitung. Für die Ableitung der Regeln wählen wir $\exists x.\, \phi \equiv \neg(\forall x.\, \neg\phi)$.

- Einführungsregel:

$$\dfrac{\dfrac{\dfrac{[\forall x.\, \neg\phi]^1}{\neg\phi[t/x]}\ \forall E \qquad \phi[t/x]}{\bot}\ \neg E}{\neg(\forall x.\, \neg\phi)}\ \neg I^1$$

- Eliminationsregel:

$$\dfrac{\neg(\forall x.\, \neg\phi) \qquad \dfrac{\dfrac{\begin{array}{c}[\phi]\\ \vdots \\ \psi\end{array} \qquad [\neg\psi]^1}{\bot}\ \neg E}{\neg\phi}\ \neg I}{\forall x.\, \neg\phi}\ \forall I}{\dfrac{\bot}{\psi}\ raa^1}\ \neg E$$

  Hier folgt die Eigenbedingung für die abgeleitete Regel aus der Eigenvariablenbedingung bei der Anwendung der Regel $\forall I$; da $\neg\psi$ für diese Teilableitung eine offene Vorbedingung ist, darf x nicht in ψ frei sein.

[1] Hier nehmen wir an, dass x die erste freie Variable ist; wenn das nicht der Fall ist müssen wir entsprechend die i-Variable ersetzen.

2.1.4 Gleichheit

Für den Umgang mit Gleichungen in der Prädikatenlogik haben wir bis jetzt noch kein Mittel (genausowenig für andere Prädikate). Was für Regeln brauchen wir für die Gleichheit? Im wesentlichen müssen wir die Gleichheit *axiomatisieren*: sie ist eine Äquivalenzrelation (also reflexiv, symmetrisch und transitiv) und sie ist eine Kongruenzrelation (man kann gleiches durch gleiches ersetzen). In Regeln ausgedrückt:

$$\frac{}{s=s}\ \mathit{refl} \qquad \frac{s=r}{r=s}\ \mathit{sym} \qquad \frac{r=s \quad s=t}{r=t}\ \mathit{trans}$$

$$\frac{r=s}{t[r/z]=t[s/z]}\ \mathit{substt} \qquad \frac{r=s \quad \phi[r/z]}{\phi[s/z]}\ \mathit{subst}$$

Hier reichen die Regeln *refl* und *subst* als Axiome. Wir zeigen als Beispiel $\{\mathit{refl}, \mathit{subst}\} \vdash \mathit{sym}$:

$$\frac{s=r \quad \overline{s=s \equiv (z=s)[s/z]}\ \mathit{refl}}{r=s \equiv (z=s)[r/z]}\ \mathit{subst} \text{ mit } \phi \equiv z=s$$

Da sieht falsch aus, weil in der Voraussetzung $\phi[s/z]$ steht, aber in der Regel *subst* (wie in den anderen Regeln auch) stehen r und s für *beliebige* Terme; das entscheidende ist, dass in der Voraussetzung von *substt* die Variable z mit der *linken* Seite der anderen Voraussetzung (hier s) ersetzt wird, und in der Konklusion mit der rechten Seite (hier r).

Entsprechende Beweis für die Substitution in Termen und die Transitivität bleiben als Übung:

Übung 2.4 *Beweise* $\{\mathit{refl}, \mathit{subst}\} \vdash \mathit{substt}$ *und* $\{\mathit{refl}, \mathit{subst}\} \vdash \mathit{trans}$.

Vorlesung vom 29.06.2023: Prädikatenlogik VIII

2.2 Formalisierung eines mathematischen Beweises

Zu zeigen: $\sqrt{2}$ ist keine rationale Zahl.

Beweis:

1. Wir nehmen an, dass $\sqrt{2}$ rational ist, und leiten einen Widerspruch her.
2. Wenn $\sqrt{2}$ rational ist, dann ist $\sqrt{2} = \frac{p}{q}$, mit p und q teilerfremd.
3. Dann gilt:

$$\begin{aligned} \sqrt{2} &= \frac{p}{q} \\ 2 &= (\frac{p}{q})^2 = \frac{p^2}{q^2} \\ p^2 &= 2q^2 \end{aligned} \tag{2.1}$$

   Damit ist p^2 gerade.

4. Wenn p^2 gerade ist, ist auch p gerade.
5. Es gilt also (für ein festes r):

$$\begin{aligned} p &= 2 \cdot r && (2.2)\\ p^2 &= 4r^2 \\ 2 \cdot q^2 &= 4 \cdot r^2 && \text{wegen (2.1)} \\ q^2 &= 2 \cdot r^2 \end{aligned}$$

6. Damit ist q^2 gerade. Daraus folgt, dass q gerade ist.
7. Wenn p und q gerade sind, dann sind sie nicht teilerfremd. ↯

Bei der Formalisierung in Aussagenlogik haben wir die Struktur des Beweises formalisiert (bspw. den Beweis durch Widerspruch), jetzt geht es darum, auch die Umformungen beweisen zu können, indem wir sie auf einfachere Axiome über Multiplikation zurückführen. Dabei werden wir auch Dinge präzisieren, die in dem Beweis oben vage bleiben, bspw. dass p,q und r existenzquantifiziert sind, und fehlende Nebenbedingungen entdecken.

Um über Zahlen, Brüche, Quadratwurzeln und Teiler sprechen zu können, müssen wir erst einmal eine Signatur mit geeignetem Vokabular definieren. In der Notation von oben ist dies:

$$\begin{aligned} \tau_{\mathcal{Q}} = \langle\ & \{0^0, 1^0, 2^0, 4^0, s^1, (\sqrt{-}^{-1}, (-^2)^1, (- \cdot -)^2, (- + -)^2, (- \mid -)^2\}, \\ & \{\mathrm{rat}^1, \mathrm{even}^1, \mathrm{odd}^1, \mathrm{cp}^2\}\rangle \end{aligned}$$

Wir haben hier eine an die Mathematik angelehnte Notation gewählt, die es uns erlaubt, Terme wie $\frac{p^2}{q^2}$ zu schreiben. Oben steht $-$ für das Argument, $\sqrt{-}$ ist also die Operation, die als $\sqrt{x}$ geschrieben wird. Etwas umständlich steht $(-^2)^1$ für die unäre Operation $-^2$ (der äußere Superskript ^{1} denotiert also die Stelligkeit, der innere ^{2} ist nur Syntax). Die binären Operatoren $\cdot$, $+$ und $\mid$ werden als Infix-Operatoren geschrieben (letzterer ist der Teilt-Operator, *i.e.* $p \mid q$ steht für p ist ein Teiler von q).

Wir könnten auch ganz normale Präfix-Operatoren nehmen, bpsw. $\mathrm{fract}^2, \mathrm{sqr}^1$, aber dann würde der Term oben lauten $\mathrm{fract}(\mathrm{sqr}(p), w\,\mathrm{sqr}(q))$, was der Leserlichkeit Abbruch täte. Wichtig ist aber, sich vor Augen zu führen, dass diese Operatoren alle reine *Syntax* sind, *i.e.* keine inhärente Bedeutung haben. Diese Bedeutung geben wir ihnen durch die Axiome, die wir über diesen Operatoren annehmen.

Fangen wir mal mit einigen Axiomen an:

$$\begin{aligned} \forall x, y.\, x \mid y &\longleftrightarrow \exists n.\, x \cdot n = y && (\textit{def-div}) \\ \forall x, y.\, \mathrm{cp}(x,y) &\longleftrightarrow \forall n.\, n \mid x \wedge n \mid y \longrightarrow n = 1 && (\textit{def-cp}) \\ \forall x.\, \mathrm{rat}(x) &\Longleftrightarrow \exists p, q.\, x = \frac{p}{q} \wedge \mathrm{cp}(x,y) \wedge q \neq 0 && (\textit{def-rat}) \\ \forall x.\, x^2 &= x \cdot x && (\textit{def-sqr}) \\ \forall x.\, (\sqrt{x})^2 &= x && (\textit{def-sqrt}) \end{aligned}$$

Was bedeuten diese Axiome?

- x teilt y, wenn es ein y ein Vielfaches von x ist (*def-div*).
- x und y sind teilerfremd (co-prim), wenn der einzige gemeinsame Teiler von x und y 1 ist.

- Eine rationale Zahl[2] ist eine, die sich als Bruch zweier teilerfremder Zahlen darstellen läßt (wobei die Zahl unter dem Bruchstrich nicht 0 ist).
- Die Zweierpotenz einer Zahl ist diese mit sich selbst multipliziert.
- Die Quadratwurzel ist das Inverse der Zweierpotenz.

Man beachte, dass alle Axiome *geschlossen* sind, *i.e.* keine freien Variablen enthalten. Dass muss so sein, weil wir natürlich im Vorhinein nicht wissen, mit welchen Termen wir das Axiom instantiieren wollen, und sonst Namenskonflikte entstehen können.

Die Arbeit mit Gleichngen in Natürlichem Schließen

Der Beweis oben arbeitet viel mit Gleichungen, *i.e.* gleichheitserhaltenden Umformungen. Deren Formalisierung in Prädikatenlogik und natürlichem Schließen ist länglich, und erfordert etwas umständlich anmutende Herleitungssequenzen. Betrachten wir einmal den Schritt oben von $\sqrt{2} = \frac{p}{q}$ zu $2 = \frac{p^2}{q^2}$. Als algebraische Herleitungskette formuliert:

$$\begin{aligned} & \sqrt{2} = \frac{p}{q} \\ \iff & (\sqrt{2})^2 = (\frac{p}{q})^2 \\ \iff & 2 = \frac{p^2}{q^2} \end{aligned}$$

Hier passieren einige Dinge implizit, bspw. indem Operationen gleichzeitig auf beiden Seiten angewandt werden oder Gleichungen wie $(\sqrt{x})^2 = x$ angewandt werden.

Wir formalisieren diesen Beweis erst einmal komplett (“from first principles”) in natürlichem Schließen. Wir benötigen dazu noch ein weiteres Axiom:

$$\forall x, y.\, (\frac{x}{y})^2 = \frac{x^2}{y^2} \qquad \textit{sqr-rat}$$

Der Beweis in seiner ganzen Schönheit (“A terrible beauty is born”):

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\forall x.\,(\sqrt{x})^2 = x}\,\textit{sqrt-inv}}{(\sqrt{2})^2 = 2}\,\forall E}{2 = (\sqrt{2})^2}\,\textit{sym} \qquad \dfrac{\sqrt{2} = \frac{p}{q} \qquad \dfrac{}{(\sqrt{2})^2 = (\sqrt{2})^2}\,\textit{refl}}{(\sqrt{2})^2 = (\frac{p}{q})^2}\,\textit{subst mit } \phi \mapsto (\sqrt{2})^2 = z^2 \qquad \dfrac{\dfrac{}{\forall x, y.\,(\frac{x}{y})^2 = \frac{x^2}{y^2}}\,\textit{sqr-rat}}{(\frac{p}{q})^2 = \frac{p^2}{q^2}}\,\forall E}{2 = (\frac{p}{q})^2}\,\textit{trans}}{2 = \frac{p^2}{q^2}}\,\textit{trans} \tag{2.3}$$

Es lohnt sich, hier die Anwendung der Regel *subst* genauer zu betrachten. Wir setzen hier den Term ϕ in der Regel zu $(\sqrt{2})^2 = z^2$. Damit wird mit $s \mapsto \frac{p}{q}$ die Konklusion $\phi[s/z] = (\sqrt{2})^2 = (\frac{p}{q})^2$ wie gefordert.

[2] Die Logik ist nicht *typisiert*, *i.e.* wir unterscheiden nicht zwischen Zahlen und anderen Dingen; der einzige Unterschied ist zwischen Termen und Prädikaten.

Für die Prämissen wird mit $r \mapsto \sqrt{2}$ die erste Prämisse $r = s$ oder $\sqrt{2} = \frac{p}{q}$, und für die zweite Prämisse $\phi[r/z] \equiv (\sqrt{2})^2 = (\sqrt{2})^2$. Diese Anwendung der *subst*-Regel ist typisch, wenn wir einen Subterm umformen wollen, oder wie hier eine Operation auf beiden Seiten der Gleichung anwenden wollen.

Da die Axiome alle geschlossene Ausdrücke sind, müssen wir jedesmal bevor wir ein Axiom anwenden die außen stehenden Allquantoren entfernen, und durch den tatsächlichen Term ersetzen.[3]

Typisch ist auch die Verwendung der Transitivitätsregel *trans*, um Sequenzen von Gleichungen zu modellieren, oder *sym*, um Gleichungen (gerade Axiome) umzuorientieren.

Dieser Detailgrad ist nicht hilfreich, denn er versperrt uns die Sicht auf die tatsächlich wichtigen Details (siehe unten). Deshalb vereinbaren wir folgende Vereinfachungen:

(1) Die Anwendungen der Regel $\forall E$ nach einem Axiom können wir weglassen.

(2) Die Anwendung der Regel *sym* bei einem Axiom kann implizit bleiben.

(3) Die Anwendung der Regel *trans* kann implizit bleiben.

(4) Bei der Anwendung der Regel *subst* kann die Anwendung der Regel *refl* in der zweiten Prämisse implizit bleiben.

Das erlaubt es uns, den Beweis (2.3) einfacher aufzuschreiben:

$$\dfrac{\dfrac{\dfrac{\sqrt{2} = \frac{p}{q}}{(\sqrt{2})^2 = (\frac{p}{q})^2}\ \textit{subst mit } \phi \mapsto (\sqrt{2})^2 = z^2}{2 = (\frac{p}{q})^2}\ \textit{sqrt-inv}}{2 = \frac{p^2}{q^2}}\ \textit{sqr-rat}$$

A Few Axioms More

Wir benötigen noch einige Axiome für den Beweis, die wir zuerst einführen, damit sie nicht so plötzlich kommen:

$$\forall x, y.\, y \neq 0 \land x = y \longrightarrow (\frac{x}{y}) \cdot y = x \qquad (\textit{cancel-rat-right})$$

$$\forall x, y.\, x \neq 0 \land x \cdot y = x \cdot z \longrightarrow y = z (\textit{cancel-left}) \forall x.\, 0 \cdot x = 0 (\textit{null-left})$$

$$\forall x, y, z.\, x \cdot (y \cdot z) = (x \cdot y) \cdot z \qquad (\textit{assoc})$$

$$\forall x, y.\, x \cdot y = y \cdot x \qquad (\textit{comm})$$

$$4 = 2 \cdot 2 \qquad (\textit{def-4})$$

$$2 \neq 0 \qquad (\textit{neq-zero})$$

$$\forall x, y.\, x \cdot y = 0 \longrightarrow x = 0 \lor y = 0 \qquad (\textit{mult-zero}^*)$$

$$\forall x, y.\, (x \cdot y)^2 = x^2 \cdot y^2 \qquad (\textit{sqr-mult}^*)$$

$$\forall x.\, x \neq 0 \longrightarrow x^2 \neq 0 \qquad (\textit{sqrt-zero}^*)$$

[3] Wir haben hier schon dahingehend vereinfacht, dass wir nur *eine* Anwendung von $\forall E$ aufschreiben, obwohl es eigentlich eine Regel pro Allquantor ist, also bspw. bei *sqr-rat* zwei.

Die ersten und zweiten Axiome sind interessant, weil sie eine nötige Vorbedingung haben. Ohne diese kommen wir schnell auf eine Inkonsistenz:

$$\dfrac{\dfrac{\dfrac{}{1=1}\,\mathit{refl}}{(\frac{1}{0})\cdot 0 = 1}\,\mathit{cancel\text{-}rat\text{-}right'}}{0=1}\,\mathit{null\text{-}left}$$

Die oben mit * markierten Axiome sind Theoreme — sie können mit Hilfe der anderen Axiome bewiesen werden. Dazu kommen noch gerade Zahlen und einige ihrer Eigenschaften:

$$\forall x.\ \mathrm{even}(x) \longleftrightarrow \exists r.\, x = 2\cdot r \qquad (\mathit{def\text{-}even})$$
$$\forall x.\ \mathrm{even}(x^2) \longrightarrow \mathrm{even}(x)(\mathit{even\text{-}sqr})^*$$
$$\forall x, y.\ \mathrm{even}(x) \wedge \mathrm{even}(q) \longrightarrow \neg \mathrm{cp}(x,y)(\mathit{even\text{-}not\text{-}cp}^*)$$

Formalisierung des Beweises

Wir zerlegen den Beweis in mehrere Teilbäume; das ist allerdings reine Notation, die Teilbäume sind *keine* eigenständigen Beweise. (Man sieht das daran, dass Annahmen in einem Teilbaum in einem anderen geschlossen werden!)

Wir fangen an mit Beweis $\mathbf{A}_1$:

$$\dfrac{\dfrac{\dfrac{\left[\begin{array}{l}\sqrt{2}=\frac{p}{q}\\ \wedge\,\mathrm{cp}(p,q)\\ \wedge\, q\neq 0\end{array}\right]^2}{q\neq 0}\,\wedge E_3}{q^2\neq 0}\,\mathit{sqrt\text{-}zero} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\left[\begin{array}{l}\sqrt{2}=\frac{p}{q}\\ \wedge\,\mathrm{cp}(p,q)\\ \wedge\, q\neq 0\end{array}\right]^2}{\sqrt{2}=\frac{p}{q}}\,\wedge E_2}{(\sqrt{2})^2=(\frac{p}{q})^2}\,\mathit{subst}, \phi \mapsto (\sqrt{2})^2 = z^2}{2=(\frac{p}{q})^2}\,\mathit{sqrt\text{-}inv}}{2=\frac{p^2}{q^2}}\,\mathit{sqr\text{-}rat}}{2\cdot q^2 = \frac{p^2}{q^2}\cdot q^2}\,\mathit{subst}, \phi \mapsto 2\cdot q^2 = z\cdot q^2}{2\cdot q^2 = p^2}\,\mathit{cancel\text{-}left} \tag{2.4}$$

Der zweite Teilbaum $\mathbf{A}_2$ ist etwas kürzer, und erweitert $\mathbf{A}_1$ zu einem Beweis, dass p gerade ist:

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{c}\mathbf{A}_1\\ \vdots\\ 2\cdot q^2 = p^2\end{array}}{p^2 = p\cdot q^2}\,\mathit{sym}}{\exists r.\, p^2 = 2\cdot r}\,\exists I}{\mathrm{even}(p^2)}\,\mathit{even\text{-}sqr}}{\mathrm{even}(p)} \tag{2.5}$$

Der dritte Teilbaum **B** ist der logische Kern des Beweises, in dem wir herleiten, dass q gerade ist:

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\mathbf{A}_2\\ \vdots\end{matrix}}{\mathrm{even}(p)}}{\exists r.\, p = 2\cdot r}\;\mathit{def\text{-}even} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\begin{matrix}\mathbf{A}_1\\ \vdots\end{matrix}}{p^2 = 2\cdot q^2}}{2\cdot q^2 = p^2}\;\mathit{sym} \quad \dfrac{\dfrac{\dfrac{[p = 2\cdot r]^3}{p^2 = (2\cdot r)^2}\;\mathit{subst}, \phi \mapsto p^2 = z^2}{p^2 = 2^2\cdot r^2}\;\mathit{sqr\text{-}mult}}{p^2 = 4\cdot r^2}\;\mathit{def\text{-}4}}{2\cdot q^2 = 4\cdot r^2}\;\mathit{trans}}{2\cdot q^2 = (2\cdot 2)\cdot r^2}\;\mathit{def\text{-}4}}{2\cdot q^2 = 2\cdot(2\cdot r^2)}\;\mathit{assoc}}{q^2 = 2\cdot r^2}\;\mathit{cancel\text{-}left}}{\exists s.\, q^2 = 2\cdot s}\;\exists I}{\mathrm{even}(q^2)}\;\mathit{def\text{-}even}}{\dfrac{\mathrm{even}(q^2)}{\mathrm{even}(q)}\;\mathit{even\text{-}sqr}}\;\exists E^3 \tag{2.6}$$

Und jetzt der *coup de grace*: wenn p und q gerade sind, dann haben wir unseren Widerspruch:

$$\dfrac{\dfrac{\dfrac{[\exists p, q.\, \sqrt{2} = \frac{p}{q} \wedge \mathrm{cp}(p,q) \wedge q \neq 0]^1 \qquad \dfrac{\dfrac{[\sqrt{2} = \frac{p}{q} \wedge \mathrm{cp}(p,q) \wedge q \neq 0]^2}{\mathrm{cp}(p,q)}\;\wedge E_2 \qquad \dfrac{\dfrac{\dfrac{\begin{matrix}\mathbf{A}_2\\ \vdots\end{matrix}}{\mathrm{even}(p)} \quad \dfrac{\begin{matrix}\mathbf{B}\\ \vdots\end{matrix}}{\mathrm{even}(q)}}{\mathrm{even}(p) \wedge \mathrm{even}(q)}\;\wedge I}{\neg\mathrm{cp}(p,q)}\;\mathit{even\text{-}not\text{-}cp}}{\bot}\;\neg I}{\bot}\;\exists E^2}{\neg(\exists p, q.\, \sqrt{2} = \frac{p}{q} \wedge \mathrm{cp}(p,q) \wedge q \neq 0)}\;\neg I^1}{\neg\mathrm{rat}(\sqrt{2})}\;\mathit{def\text{-}sqrt} \tag{2.7}$$

Man beachte, dass in $\mathbf{A}_1$ Annahmen geschlossen werden, die wir in (2.7) geöffnet haben. Das bedeutet auch, dass die existenzquantifizierten Variablen p und q in allen Teilbäumen als freie Variablen behandelt werden, und damit "beliebig aber fest" sind; wir können wie Annahme 2, die mit der Regel $\exists E$ in (2.7) geöffnet wird, nur mit einer Behauptung schließen, die p und q nicht mehr enhält (*i.e.* $\bot$); deshalb muss diese Regel soweit unten stehen.

Vorlesung vom 06.07.2023: Arithmetik

2.3 Elementare Arithmetik

Der Beweis der Irrationalität von $\sqrt{2}$ zeigt, dass wir nicht-triviale Aussagen über Zahlen in der Prädikatenlogik formalisieren können. Nicht befriedigend war dort allerdings, dass wir dort immer wieder axiomatische Aussagen über Zahlen hinzufügen mussten; wir haben gesehen, dass es sehr leicht ist, dabei Inkonsistenzen zu erzeugen (und wenn wir ehrlich sind haben wir das nur vermeiden, weil wir wussten, worauf wir achten müssen).

Deshalb wollen wir jetzt die elementare Arithmetik von Grund auf axiomatisieren. Wir fangen an mit den natürlichen Zahlen und der Addition. Wenig überraschend bestehen die natürlichen Zahlen aus der 0 und

der Nachfolgeoperation, zusammen mit einer rekursiven Definition der Addition, und einem *Induktionsschema*:

Definition 2.6 (Presburger-Arithmetik) *Die Presburger-Arithmetik besteht aus*

- *Signatur:* $\Sigma_{PR} \stackrel{def}{=} \{0^0, S^1, (+)^2\}$
- *Axiome:*Γ_{PR} *mit*

$$\begin{aligned} \forall x.\, 0 \neq S(x) && (PA1)\\ \forall x, y.\, S(x) = S(y) \longrightarrow x = y && (PA2)\\ \forall x.\, x + 0 = x && (PA3)\\ \forall x, y.\, x + S(y) = S(x + y) && (PA4)\\ \phi(0) \wedge (\forall x.\, \phi(x) \longrightarrow \phi(S(x))) \longrightarrow \forall x.\, \phi(x) && (ind) \end{aligned}$$

In (*ind*) ist ϕ eine Formel mit einer ausgezeichneten, sonst nicht frei vorkommenden Variablen, die wir dann mit 0 und x instantiieren (das ist unsere Notation $\phi(0), \phi(x)$). Man kann das formalisieren mit \$ als besondere Variable, und $\phi(t) = \phi[t/\$]$.

Die Presburger-Arithmetik ist alleine deshalb erwähnenswert, weil sie ein reales Beispiel für hyperexponentiellen Aufwand bietet:

Theorem 2.2 *Presburger-Arithmetik ist mit dem Aufwand* $2^{2^{cn}}$ *entscheidbar.*

Entscheidbar bedeutet hier, dass ich für ein gegebene Formel $\psi \in Prop_{\Sigma_{PR}}$ immer zeigen kann, ob $PR \vdash \psi$ oder nicht; das n ist hier die Anzahl der Symbole in ψ. Darüberhinaus ist die Presburger-Arithmetik deshalb interessant (und relevant), weil wir in *PR* schon die natürliche Ordnung ($<$) auf den natürlichen Zahlen definieren können (s.u.). Aussagen in *PR* sind damit lineare Ungleichungen über natürlichen Zahlen, die beispielsweise bei der Programmverifikation als Aussagen über die Zeigerarithmetik oder Gültigkeit von Array-Zugriffen eine große Rolle spielen.

Erweitern wir *PR* um Plutimikation erhalten wir die *Peano-Arithmetik*:

Definition 2.7 (Peano-Arithmetik) *Die Peano-Arithmetik besteht aus*

- *Signatur:* $\Sigma_{PA} \stackrel{def}{=} \{0^0, S^1, (+)^2, (\cdot)^2\}$
- *Axiome:* Γ_{PA} *mit*

$$\begin{aligned} \forall x.\, 0 \neq S(x) && (PA1)\\ \forall x, y.\, S(x) = S(y) \longrightarrow x = y && (PA2)\\ \forall x.\, x + 0 = x && (PA3)\\ \forall x, y.\, x + S(y) = S(x + y) && (PA4)\\ \phi(0) \wedge (\forall x.\, \phi(x) \longrightarrow \phi(S(x))) \longrightarrow \forall x.\, \phi(x) && (ind)\\ \forall x.\, x \cdot 0 = 0 && (PA5)\\ \forall x, y.\, x \cdot S(y) = x \cdot y + x && (PA6) \end{aligned}$$

Die Peano-Arithmetik ist *nicht* mehr entscheidbar; wir werden das später bei den Gödelschen Unvollständigkeitssätzen sehen.

2.3.1 Die Arbeit mit Zahlen

Wir können jetzt die wesentlichen Konzepte aus Section 2.2 konservativ herleiten. Die Axiome für |, cp, even, odd müssen wir an dieser Stelle nicht wiederholen. Wir können natürlich auch definieren, wann x eine Primzahl ist. Die Lehrbuchdefinition ist: n ist eine Primzahl, wenn n genau zwei Teiler (nämlich n und 1) hat. Diese Definition schließt implizit 1 als Primzahl aus. Wir können hier nicht über Mengen reden, deshalb sagen wir: n ist eine Primzahl, wenn es nur durch 1 und sich selbst teilbar ist, und nicht 1 ist.

$$\text{prime}(n) \Longleftrightarrow \forall x.x \mid n \longrightarrow (x = 1 \vee x = n) \wedge n \neq 1 \tag{2.8}$$

2.3.2 Primitive und Partiell Rekursive Funktionen

Wir haben verschiedene Funktionen und Prädikate auf Zahlen definiert, und darbei argumentiert, dass diese Definitionen konservativ sind. Dieses Argument war allerdings eher informell; wenn wir weitere Funktionen definieren wollen, müssen wir diese Argumentation formaler fassen.

Wir gehen an dieser Stelle nicht ins Detail, und verweisen auf [2, § 7.1]. In aller Kürze, primitiv rekursive Funktionen sind definiert als die kleinste Klasse aller Funktionen $\mathscr{F}$ so dass:

$$\begin{array}{rl}
C_m^k \in \mathscr{F} & C_m^k(n_0,\ldots,n_{k-1}) \stackrel{def}{=} m \\
S \in \mathscr{F} & S(n)\,\text{def}(n)+1 \\
P_i^k \in \mathscr{F} & P_i^k(n_0,\ldots,n_{k-1}) \stackrel{def}{=} n_i(i<k) \\
g,h_0,\ldots,h_{p-1} \in \mathscr{F} \text{ dann } f \in \mathscr{F} & f(\vec{n}) = g(h_0(\vec{n}),\ldots,h_{p-1}(\vec{n})) \\
g,h \in \mathscr{F} \text{ dann } f \in \mathscr{F} & \begin{cases} f(0,\vec{n}) & = g(\vec{n}) \\ f(m+1,\vec{n}) & = h(f(m,\vec{n}),\vec{n},m) \end{cases}
\end{array}$$

Hierbei ist C_m^k die *konstante* Funktion (die für k Argumente immer n zurückgibt), S natürlich die gute, alte *Nachfolgefunktion*, und P_i^k die *Projektion* der i-ten Komponente aus einem k-Tupel. Die beiden letzten Klauseln modellieren *Substitution* und *Rekursion*.

Damit können wir rekursive Funktionen wie die oben definieren, allerdings nicht alle berechenbaren Funktionen. (Man sieht das alleine daran, dass alle Klauseln nur totale Funktionen definieren, keine partiellen; oder aber daran, dass die Vorgängerfunktion nicht definierbar ist.) Um alle *berechenbaren* Funktionen definieren zu können, definieren wir ein rekursives Prädikat[4]

$$\{e\}(\vec{x}) \sim y$$

mit $e,y \in \mathbb{N},\vec{x} \in \mathbb{N}^n$, welche alle berechenbaren Funktionen abbildet. Der Trick hier ist, dass wir $\{e\}(x) \sim y$ erst als Relation definieren, und dann beweisen, dass diese Relation funktional (*i.e.* $\{e\}(\vec{x}) \sim y_1, \{\vec{e}\}(x) \sim y_2 \Longrightarrow y_1 = y_2$) aber partiell ist. Für die genaue, nicht ganz einfache Definition verweisen wir auf [2, §7.2]. Im wesentlichen kodiert die Definition von $\{e\}(x) \sim y$ ein abstraktes Berechnungsmodell auf Sequenzen von natürlichen Zahlen, ähnlich einer Turing-Maschine.

Partielle Funktionen können wir nicht direkt durch Funktionen in der Prädikatenlogik repräsentieren, da in der Prädikatenlogik alle Funktionen total sind; ein Term, der nicht auswertet ist schlicht und einfach nicht vorgesehen. Deshalb wird eine partielle Funktion $\{e\}(\vec{x})$ durch ein Prädikat $p(\vec{x},y)$ repräsentiert, so dass $p(\vec{x},y)$ gdw. $\{e\}(\vec{x}) \sim y$.

Vorlesung vom 11.07.2023: Unvollständigkeit I

[4] Rekursiv definierte Prädikate heißen induktiv, wahrscheinlich weil das vornehmer klingt.

Kapitel 3

Gödels Unvollständigkeitssatz

In diesem Abschnitt skizzieren wir den Beweis von Gödels ersten Unvollständigkeitssatz, einem der wichtigsten Resultate der formalen Logik des letzten Jahrhunderts. Ungenau formuliert (wir werden unten genauer) lautet der erste Unvollständigkeitssatz:

> Jede konsistente Theorie, die hinreichend expressiv ist, um die natürlichen Zahlen zu formalisieren, erlaubt die Formulierung von wahren Aussagen, die nicht beweisbar sind.

Interessanterweise gibt es auch einen *Vollständkeitssatz* von Gödel, nämlich die Vollständigkeit der Prädikatenlogik erster Stufe — das war seine Doktorarbeit.

3.1 Beweisskizze

Der Beweis der Unvollständigkeitssätze ist eine faszinierende Mischung aus vielen umständlichen, technisch anspruchsvollen Konstruktionen, die dann in einer eleganten Diagonalisierung enden. Wenn man die umständlichen Konstruktionen glaubt, ist der Beweis recht einfach nachzuvollziehen.

Besage Konstruktionen drehen sich alle um die *Kodierung* der Prädikatenlogik (oder einer anderen Logik) in die Peano-Arithmetik **PA**. Das funktioniert so:

(i) Zu jeder Formel φ gibt es eine natürliche Zahl $\lceil\varphi\rceil$, die diese Formel eindeutig kodiert.

(ii) Zu jedem ND-Beweis D für φ gibt es eine natürliche Zahl $\lceil D\rceil$, die diesen Beweis eindeutig kodiert.

(iii) Die Beweisbarkeit von φ in $\mathbb{N}$ ist als rekursives Prädikat $\mathrm{Provable}(\lceil\varphi\rceil)$ in **PA** formalisierbar.

(iv) Da wir **PA**selber in Prädikatenlogik formalisieren können, können wir eine Formel φ mit der Aussage "Ich bin nicht beweisbar" konstruieren: $\varphi \iff \neg\,\mathrm{Prov}(\lceil\varphi\rceil)$

(v) Daraus folgt die Unvollständigkeit: die Formel muss wahr sein, aber nicht beweisbar.

Kleine Anmerkung noch zu **PA**: wenn wir **PA** sagen, meinen wir damit die in Prädikatenlogik formalisierbare Theorie der partiellen und primitiv rekursive Funktion aus Section 2.3.2. Wir werden diese Unterscheidung hier nicht weiter vertiefen; es ist für die logische Korrektheit der Argumentation allerdings wichtig, sich zu vergegenwärtigen, ob die Definitionen in **PA** (also natürliche Zahlen und rekursiven Funktionen) oder in der Prädikatenlogik stattfinden.

3.2 Kodierung von Termen und Formeln

Wir brauchen in **PA** jenseits der Funktionen auf natürlichen Zahlen, wie sie im vorherigen Kapitel definiert wurden, noch einige weitere. Wir werden diese in der etwas informellen rekursiven Schreibweise definieren. Etwas Notation dazu: für eine primitiv rekursive Relation R ist $\mu y.R(\vec{x},y)$ das *kleinste* y so dass $R(\vec{x},y)$ wahr ist, und $\mu y < n.R(\vec{x},y)$ das kleinste y bis n, so dass $R(\vec{x},y)$ wahr ist, oder n falls es kein solches y gibt. Damit definieren wir

$$\exp(x,y)\begin{cases}\exp(x,0) & = 1\\ \exp(x,y+1) & = \exp(x,y)\cdot x\end{cases}$$

$$\mathrm{p}(n)\begin{cases}\mathrm{p}(0) & = 2\\ \mathrm{p}(n+1) & = \mu x.\ \mathrm{prime}(x)\wedge \mathrm{p}(n) < x\end{cases}$$

Zentral für die Kodierung von Prädikatenlogik in **PA** (genannt Gödel-Kodierung oder auch Gödelisierung) sind *Sequenzen*. (Wenn wir **PA** als Programmiersprache auffassen, ist es klar, dass wir einen aggregierenden Datentyp benötigen. Für einfache Progammiersprachen sind dies meist zusammenhängende Speicherbereiche, also Felder, die wir auch als veränderliche Sequenzen fester Länge auffassen können.) Hierfür spielen Primzahlen eine zentrale Rolle. Zwar ist es einfach, Sequenzen von Zahlen als Zahl zu kodieren (einfach alle addieren?!), aber die Kodierung soll auch umkehrbar sein. Der Trick ist, dass wir in einer Sequenz $\langle a_1,a_2,a_3,a_4\rangle$ die i-te Position durch die a_i-fache Potenz der i-ten Primzahl darstellen. Damit ist die Dekodierung dann durch die (eindeutige) Primfaktorisierung gegeben. Hier ein Beispiel:

$$\langle 7,6,3\rangle = 2^{7+1}\cdot 3^{6+1}\cdot 5^{3+1} = 2^8\cdot 3^7\cdot 5^4 = 256\cdot 2187\cdot 625 = 349920000$$

Und ein Beispiel für die Rückrichtung:

$$\begin{aligned}&5\,317\,623\,862\,875\,450\,000\,000\\ =&2\cdot 2\cdot 2\cdot 2\cdot 2\cdot 2\cdot 2\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 3\cdot 5\cdot 5\cdot 5\cdot 5\cdot 5\cdot 5\cdot 5\cdot 7\cdot 7\cdot 7\cdot 7\cdot 7\cdot 7\cdot 7\\ =&2^7\cdot 3^{17}\cdot 5^8\cdot 7^7 = \langle 6,16,7,6\rangle\end{aligned}$$

Man beachte, dass nicht jede Zahl die Kodierung einer Sequenz ist:

$$2\,342\,100\,000 = 2\cdot 2\cdot 2\cdot 2\cdot 2\cdot 3\cdot 5\cdot 5\cdot 5\cdot 5\cdot 5\cdot 37\cdot 211 = 2^5\cdot 3^1\cdot 5^4\cdot 37\cdot 211$$

Hier fehlen die Potenzen der Primzahlen zwischen 5 und 37, und 37 und 211; eine Zahl ist also nur die Kodierung einer Sequenz, wenn die Primfaktoren aus aufeinander folgenden Primzahlen bestehen. Hier sind die formalen Definitionen:

$$\mathrm{Seq}(n) \stackrel{def}{=} \forall p,q.\ p<n\wedge q<n \longrightarrow (\mathrm{prime}(p)\wedge\mathrm{prime}(q)\wedge q<p\wedge p \mid n \longrightarrow q \mid n)\wedge n\neq 0 \tag{3.1}$$

$$\mathrm{len}(n) \stackrel{def}{=} \mu x \le n+1.\ \neg(p_x \mid n) \tag{3.2}$$

$$(n)_i \stackrel{def}{=} (\mu x \le n.\ p_i^x \mid n \wedge \neg p_{i+1}^x \mid n) - 1 \tag{3.3}$$

$$n +\!\!+ m \stackrel{def}{=} n\cdot \prod_{i=0}^{\mathrm{len}(m)-1} p_{\mathrm{len}(n)+1}^{(m)_i+1} \tag{3.4}$$

$\mathrm{Seq}(n)$ ist wahr, wenn n eine *Sequenzzahl* ist (also eine Zahl, die eine Sequenz definiert; wie oben gesehen, gilt das nicht für alle), $\mathrm{len}(n)$ ist die *Länge* der Sequenz (für eine Sequenzzahl, sonst einfach 1), $(n)_i$ ist die Projekten des i-ten Elements der Sequenz, und $+\!\!+$ ist die Konkatenation. Alles so ähnlich wie Haskell, nur ungetypt und mit primitiv rekursiven Funktionen auf natürlichen Zahlen.

Übung 3.1 *Ist* 1 *eine Sequenznummer?*

Entscheidend ist hier, dass wir die Operationen auf Sequenzen von natürlichen Zahlen direkt als primitiv rekursive Funktionen definieren; wir haben also keine direkte Repräsentation von Sequenzen in unserer Sprache, sondern nur implizit als Menge $\mathrm{Seq}(n)$. Im folgenden schreiben wir $\langle x_1, x_2, \ldots, x_n \rangle$ für die Sequenznummer, welche die Sequenz $x_1, \ldots, x_n$ repräsentiert.

3.3 Kodierung der Syntax

Jetzt können wir Terme und Prädikate kodieren. Wir beschränken uns dabei auf die funktional vollständige Menge $\bot, \wedge, \longrightarrow, \forall$ von logischen Konnektiven, und die für Peano-Arithmetik nötigen Funktionen und Prädikate:

$$\Sigma = \{0^0, S^1, +^2, \cdot^2, \exp^2, =^2\}$$

Wir nehmen ferner eine feste, beliebig große, aufzählbare Menge $X = \{x_1, \ldots, x_n, \ldots\}$ von Variablen an. Die Funktion $\lceil x \rceil$ (die Gödel-Nummer) ist die Kodierung der Syntax (der Termen und Formeln) in **PA**. Wir definieren zuerst

$\lceil \bot \rceil$	$\lceil \wedge \rceil$	$\lceil \longrightarrow \rceil$	$\lceil \forall \rceil$	$\lceil 0 \rceil$	$\lceil S \rceil$	$\lceil + \rceil$	$\lceil \cdot \rceil$	$\lceil \exp \rceil$	$\lceil = \rceil$	$\lceil x_i \rceil$
2	3	5	7	11	13	17	19	23	29	p_{9+i}

Damit kodieren wir Terme und Formeln (die Kodierung der Konstanten $0, \bot$ steht schon oben):

$$\begin{aligned}
\lceil S(t) \rceil &\stackrel{def}{=} \langle \lceil S \rceil, \lceil t \rceil \rangle \\
\lceil s + t \rceil &\stackrel{def}{=} \langle \lceil + \rceil, \lceil s \rceil, \lceil t \rceil \rangle \\
\lceil s \cdot t \rceil &\stackrel{def}{=} \langle \lceil \cdot \rceil, \lceil s \rceil, \lceil t \rceil \rangle \\
\lceil Exp(s,t) \rceil &\stackrel{def}{=} \langle \lceil \exp \rceil, \lceil s \rceil, \lceil t \rceil \rangle \\
\lceil t = s \rceil &\stackrel{def}{=} \langle \lceil = \rceil, \lceil t \rceil, \lceil s \rceil \rangle \\
\lceil p \wedge q \rceil &\stackrel{def}{=} \langle \lceil \wedge \rceil, \lceil p \rceil, \lceil q \rceil \rangle \\
\lceil p \longrightarrow q \rceil &\stackrel{def}{=} \langle \lceil \longrightarrow \rceil, \lceil p \rceil, \lceil q \rceil \rangle \\
\lceil \forall x_i . p \rceil &\stackrel{def}{=} \langle \lceil \forall \rceil, \lceil x_i \rceil, \lceil p \rceil \rangle
\end{aligned}$$

Hier ist die Kodierung des Axioms $(PA3)$ von Definition 2.7:

$$\begin{aligned}\lceil\forall x.\forall y.x+S(y)=S(x+y)\rceil &= \langle\lceil\forall\rceil,\lceil x\rceil,\lceil\forall\rceil,\lceil y\rceil,\lceil x+S(y)=S(x+y)\rceil\rangle\\ &= \langle\lceil\forall\rceil,\lceil x\rceil,\lceil\forall\rceil,\lceil y\rceil,\lceil=\rceil,\lceil+\rceil,\lceil x\rceil,\lceil S\rceil,\lceil y\rceil,\lceil S\rceil,\lceil+\rceil,\lceil x\rceil,\lceil y\rceil\rangle\\ &= \langle 7,31,7,37,29,17,31,13,37,13,17,31,39\rangle\\ &= 511\,484\,093\,774\,428\,751\,894\,175\,834\,105\,994\,806\,038\,555\,320\,421\,376\\ &\quad 593\,503\,973\,135\,131\,016\,562\,013\,355\,565\,044\,304\,630\,571\,681\,180\,419\\ &\quad 533\,586\,498\,479\,195\,825\,466\,276\,007\,800\,729\,403\,849\,001\,896\,614\,384\\ &\quad 023\,941\,390\,755\,441\,997\,247\,976\,133\,913\,151\,417\,408\,087\,525\,644\,645\\ &\quad 820\,434\,883\,035\,791\,562\,644\,380\,605\,006\,968\,416\,227\,856\,417\,764\,995\\ &\quad 652\,312\,586\,800\,395\,117\,606\,581\,260\,408\,699\,280\,121\,014\,656\,715\,233\\ &\quad 875\,327\,287\,549\,179\,040\,479\,602\,800\,742\,879\,706\,403\,340\,794\,725\,309\\ &\quad 582\,635\,387\,692\,900\,000\,000\end{aligned}$$

Wie wir sehen, werden die Gödelkodierungen auch von kleinen Formeln sehr schnell sehr groß.[1]

Wir können jetzt in **PA** primitiv rekursive Funktionen Term und Form definieren, welche Terme und Formeln charakterisieren, sowie Funktionen welche Variablen und Konstanten charakterisieren. Darüber hinaus können wir weitere primitiv rekursive Funktionen definieren, welche syntaktische Eigenschaften und Termmanipulationen definieren, darunter $\mathrm{Free}(x,y)$, welches wahr ist gdw. wenn x die Gödelnummer eines Terms ist, und y die Gödelnummer einer Variabeln, die frei in x vorkommt. Die Krönung der Syntax-Kodierung ist dann eine Funktion $\mathrm{subst}(x,y,z)$, welche die Substitution der (durch die Gödelnumme kodierten) Variablen y im (durch die Gödelnummer kodierten) Formel x mit dem (durch die Gödelnummer etc) Term z definiert, sprich

$$\mathrm{subst}(\lceil\varphi\rceil,\lceil x\rceil,\lceil t\rceil) = \lceil\varphi[t/x]\rceil \tag{3.5}$$

Man beachte, (3.5) ist die *Korrektheitseigenschaft* der Funktion subst, nicht die Definition.

3.3.1 Kodierung der Ableitbarkeit

Damit kommen wir jetzt unserem Ziel näher. An dieser Stelle führen wir noch etwas Syntax ein. $\overline{z}$ ist für eine Gödelnummer z die Formel (oder Term), welche zu z kodiert, *i.e.* $\overline{\lceil\varphi\rceil} \longleftrightarrow \varphi$. Ziel ist jetzt es, ein primitiv rekursives Programm $\mathrm{Der}(x,y,z)$ in **PA** zu definieren, welches wahr ist, wenn $\overline{x}$ eine Ableitung mit den Prämissen $\overline{y_0},\ldots,\overline{y_{\mathrm{len}\,y-1}}$ und der Konklusion $\overline{z}$ ist.

[1] Das ist umso bemerkenswerter, als dass zu Gödels Zeiten Speicher noch sehr teuer war.

Wir beginnen mit den Ableitungen:

$$\left\lceil \varphi \right\rceil \stackrel{def}{=} \langle 0, \lceil \varphi \rceil \rangle$$

$$\left\lceil \dfrac{\overset{D_1}{\phi} \quad \overset{D_2}{\psi}}{\phi \wedge \psi}\ \wedge I \right\rceil \stackrel{def}{=} \langle \langle 0, \lceil \wedge \rceil \rangle, \left\lceil \overset{D_1}{\phi} \right\rceil, \left\lceil \overset{D_2}{\psi} \right\rceil, \lceil \phi \wedge \psi \rceil \rangle$$

$$\left\lceil \dfrac{\overset{D}{\phi \wedge \psi}}{\phi}\ \wedge E_L \right\rceil \stackrel{def}{=} \langle \langle 1, \lceil \wedge \rceil \rangle, \left\lceil \phi \wedge \overset{D}{\psi} \right\rceil, \lceil \phi \rceil \rangle$$

$$\left\lceil \dfrac{\begin{array}{c} D \\ \psi \end{array}}{\phi \longrightarrow \psi}\ \longrightarrow I \right\rceil \stackrel{def}{=} \langle \langle 0, \lceil \longrightarrow \rceil \rangle, \left\lceil \begin{array}{c} D \\ \psi \end{array} \right\rceil, \lceil \phi \longrightarrow \psi \rceil \rangle$$

$$\left\lceil \dfrac{\overset{D_1}{\phi} \quad \phi \overset{D_2}{\longrightarrow} \psi}{\psi}\ mp \right\rceil \stackrel{def}{=} \langle \langle 1, \lceil \longrightarrow \rceil \rangle, \left\lceil \overset{D_1}{\phi} \right\rceil, \left\lceil \phi \overset{D_2}{\longrightarrow} \psi \right\rceil, \lceil \psi \rceil \rangle$$

Entsprechend für RAA, $\forall I$, $\forall E$ (siehe [2, S. 248]).

Jetzt definieren wir $\mathrm{Der}(p,h,z)$: $\overline{p}$ ist Beweis für $\overline{z}$ aus Hypothesen $\overline{h}$. Intuitiv gilt $\mathrm{Der}(p,h,z)$, wenn z eine Formel ist, h eine Sequenz von Formeln, und entweder z in den Hypothesen z enthalten ist, oder p die Herleitung von z aus h durch die Anwendung einer der Regeln oben kodiert. Wir zeigen die Definition auszugsweise, aber es sollte klar werden, wie das ganze funktioniert:

$$\begin{aligned}
\mathrm{Der}(p,h,z) \stackrel{def}{=} & \mathrm{Form}(z) \wedge \bigwedge_{i=0}^{\mathrm{len}(h)-1} \mathrm{Form}((h)_i) \wedge \\
& (\ (\exists i.\, z = (y)_i \wedge p = \langle 0, z \rangle) && \text{Hypothese} \\
& \vee \ \exists p_1, h_1, z_1, p_2, h_2, z_2. && \wedge I \\
& \quad \mathrm{Der}(p_1, h_1, z_1) \wedge \mathrm{Der}(p_2, h_2, z_2) \wedge \\
& \quad p = \langle \langle 0, \lceil \wedge \rceil \rangle, p_1, p_2, z \rangle \\
& \quad h = h_1 +\!\!+ h_2 \wedge \\
& \quad z = \langle \lceil \wedge \rceil, z_1, z_2 \rangle \\
& \vee \ \exists p_1, h_1, z_1, u. && \longrightarrow I \\
& \quad \mathrm{Der}(p_1, h_1, z_1) \wedge \\
& \quad p = \langle \langle 0, \lceil \longrightarrow \rceil \rangle, p_1, z_1 \rangle \wedge \\
& \quad (h = \mathrm{cancel}(u, h_1) \vee h = h_1) \wedge \\
& \quad z = \langle \lceil \longrightarrow \rceil, u, z_1 \rangle \\
& \ldots \\
&)
\end{aligned}$$

Wir sind an dieser Stelle etwas ungenau: was diese Definition primitiv rekursiv macht ist die Tatsache, dass alle Existenzquantoren nach oben eine strikte Schranke haben (beispielsweise $\mathrm{len}(h)$ für i in dem Hypothesen-Fall, oder p für $p_1, h_1, z_1, p_2, h_2, z_2$ im Fall $\wedge I$); wir habe das in der Formel elidiert, um die Übersichtlichkeit zu wahren. Weiterhin nutzen wir noch eine separat zu definierende, primitiv rekursive Funktion $\mathrm{cancel}(u,y)$ welche aus einer Sequenz y die Zahl u streicht. Sie wird dazu benötigt, um die durch die Anwendung der Einführungsregel $\longrightarrow I$ geschlossene Voraussetzung u aus den Hypothesen zu entfernen. Die Definition der weiteren Regeln ist analog; bei der Einführungsregel für den Allquantor muss die Eigenvariablenbedingung geprüft werden.

Das Endziel ist, die Beweisbarkeit einer Formel φ in **PA**zu definieren. Dazu brauchen wir noch ein Prädikat, welches die Axiome der Peano-Arithmetik (Definition 2.7), erweitert um die Definition der Funktion exp sowie die Axiome für die Gleichheit (siehe Section 2.1.4) charakterisiert. $\mathrm{Ax}(n)$ ist wahr, wenn n die Gödelnummer eines dieser Axiome ist:

$$\mathrm{Ax}(n) \stackrel{def}{=} n = \lceil PA1 \rceil \vee n = \lceil PA2 \rceil \vee n = \lceil PA3 \rceil \vee \ldots$$

Damit definieren wir $\mathrm{Prov}(p,f)$: p ist die Gödelnummer eines ND-Beweis für $\overline{f}$:

$$\mathrm{Prov}(p,f) \stackrel{def}{=} \exists h.\, \mathrm{Der}(p,h,f) \wedge \bigwedge_{i=0}^{\mathrm{len}(h)-1} \mathrm{Ax}((h)_i)$$

und $\mathrm{Thm}(f)$ für $\overline{f}$ ist ein Theorem in **PA**:

$$\mathrm{Thm}(f) \stackrel{def}{=} \exists p.\, a\mathrm{Prov}(p,f)$$

Entscheidend ist hierbei, dass Thm primitiv rekursiv ist und damit in **PA**definierbar ist.

Vorlesung vom 13.07.2023: Unvollständigkeit II

3.3.2 Unvollständigkeit

Im letzten Schritt zeigen wir jetzt die Unvollständigkeit unserer Logik (und jeder anderen Logik, welche die Peano-Arithmetik formalisiert).

Wir benötigen dazu das folgende

Theorem 3.1 (Fixpunkt-Theorem) *For jede Formel $\varphi(x)$ mit genau einer freien Variablen x gibt es eine Formel ψ so dass $\vdash \varphi(\overline{\lceil \psi \rceil}) \longleftrightarrow \psi$.*

Beweis. Sei $s(a,b) \stackrel{def}{=} \mathrm{subst}(a, \lceil x \rceil, b)$ die oben (nicht) in **PA** definierte Substitutionsfunktion subst für die feste Variable x, d.h. $s(a,b)$ ersetzt in der Formel a die Variable x durch den Term b (alles in **PA**, *i.e.* durch das Rechnen mit Zahlen). Es gilt dann mit (3.5) $s(\lceil \rho \rceil, \lceil t \rceil) = \lceil \rho[t/x] \rceil$.

Jetzt setzen wir $\theta(a) \stackrel{def}{=} \varphi(\overline{s(\lceil a \rceil, \lceil a \rceil)}), m \stackrel{def}{=} \lceil \theta(x) \rceil$ und $\psi \stackrel{def}{=} \theta(\overline{m})$, und erhalten:

$$\begin{aligned}
\psi \longleftrightarrow \theta(\overline{m}) &\longleftrightarrow \varphi(\overline{s(\lceil \overline{m} \rceil, \lceil \overline{m} \rceil)}) \\
&\longleftrightarrow \varphi(\overline{s(\lceil \overline{\lceil \theta(x) \rceil} \rceil, \lceil \overline{m} \rceil)}) \\
&\longleftrightarrow \varphi(\overline{s(\lceil \theta(x) \rceil, \lceil \overline{m} \rceil)}) \\
&\longleftrightarrow \varphi(\overline{\lceil \theta(\overline{m}) \rceil}) \\
&\longleftrightarrow \varphi(\overline{\lceil \psi \rceil})
\end{aligned}$$

□

So gerüstet können wir den ersten Unvollständigkeitssatz beweisen. Dazu noch eine technische Definition:

Definition 3.1 (ω-Konsistenz) *Eine Formalisierung T von **PA** ist ω-konsistent wenn es keine Formel $\phi(x)$ gibt so dass $T \vdash \exists x.\, \phi(x)$ und für alle n $T \vdash \neg\phi(\overline{n})$.*

Konkret bedeutet dies: in einer ω-konsistenten Theorie kann ich nicht eine existenzquantifizierte Formel beweisen, wenn alle Instanzen (aufgezählt anhand ihrer Gödelnummern) das Gegenteil beweisen.

Theorem 3.2 (Gödel I) *Wenn **PA** ω-konsistent ist, dann ist **PA** unvollständig.*

Beweis. Wir wenden das Fixpunkttheorem 3.1 auf das Prädikat $\neg\,\mathrm{Thm}(x)$ an, und erhalten φ (die "Gödel-Formel") mit der Eigenschaft

$$\varphi \longleftrightarrow \neg\,\mathrm{Thm}(\lceil\varphi\rceil)$$

φ bedeutet semantisch "Diese Formel ist nicht in **PA** beweisbar." Jetzt gibt es zwei Fälle:

1. $\vdash \varphi$: Dann gibt es n so dass $\vdash \mathrm{Prov}(\lceil\varphi\rceil, \overline{n})$, also $\vdash \exists y.\, \mathrm{Prov}(\lceil\varphi\rceil, y)$, also $\vdash \mathrm{Thm}(\lceil\varphi\rceil)$, also $\vdash \neg\phi$. Also ist **PA** inkonsistent.
2. $\vdash \neg\varphi$: Dann $\vdash \mathrm{Thm}(\lceil\varphi\rceil)$, also $\vdash \exists x.\, \mathrm{Prov}(\lceil\varphi\rceil, x)$. Wenn **PA** ω-konsistent wäre, dann auch konsistent, also kann nicht gleichzeit $\vdash \varphi$ und $\vdash \neg\varphi$ gelten, also gilt $\nvdash \varphi$, und damit $\vdash \neg\,\mathrm{Prov}(\lceil\varphi\rceil, n)$ für alle n; damit wäre **PA** aber ω-inkonsistent — Widerspruch. Also ist **PA** ω-inkonsistent.

Also kann weder ϕ noch $\neg\phi$ herleitbar sein. Damit ist ϕ wahr, aber eben nicht beweisbar. □

Literaturverzeichnis

[1] U. Schöning. *Logik für Informatiker*. Spektrum Akademischer Verlag, 2000.

[2] D. van Dalen. *Logic and Structure*. Springer Verlag, 2004. Vierte Auflage.