2. Sitzung

Systeme des Natürlichen Schliessens

Gegenüberstellung von Systemen Natürlichen Schliessen und Axiomatischen Systemen

Natürliches Schliessen:

Axiome: $\varnothing$

Regeln: viele, abhängig von den logischen Konstanten

Gut für das praktische Arbeiten

Natürliche Schlussregeln

Gute Strategien für die Konstruktion von Derivationen

Weniger praktisch für metatheoretische Untersuchungen

Axiomatische Systeme:

Axiome (bzw. Axiomenschemata): wenige (3-10)

Regeln: wenige (1-3)

Schlecht für das praktische Arbeiten

Unnatürlich (wir schliessen nicht aus Axiomen)

Kaum gute Strategien für die Konstruktion von Derivationen vorhanden

Gut für metatheoretische Untersuchungen

Meine Motivation für diese Sitzung: Fertigkeiten im Natürlichen Schliessen hilfreich für spätere Beweise von Metatheoremen.

Implikationslogik

Einziges Konnektiv: $\supset$

$S_{\supset}$: keine Axiome

4 Regeln: hyp, reit, $\supset$intro, $\supset$elim

Derivation

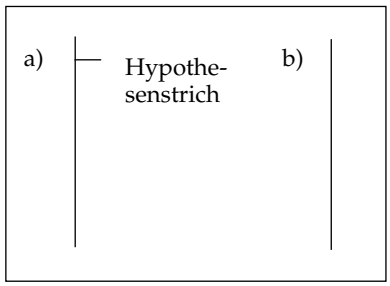
Die Objekte, die in $S_{\supset}$ konstruiert werden, heissen *Derivationen*. Eine Derivation besteht aus einer *Derivationslinie* mit einer endlichen Anzahl von *Gegenständen* (engl. *items*) unmittelbar rechts von der Derivationslinie.

Derivationslinie

Es gibt zwei Arten von Derivationslinien:

a) mit Hypothesenstrich

b) ohne Hypothesenstrich.



(Der Hypothesenstrich dient dazu, die durch Hyp eingeführten Aussagen von den durch andere Regeln gewonnenen Aussagen optisch abzugrenzen.)

Gegenstände einer Derivation

Es gibt weiterhin zwei Arten von Gegenständen (*items*) einer Derivation:

a) Formeln

b) Derivationen

(wobei Derivationen, die selbst wieder Gegenstände einer ("grösseren") Derivation sind, unmittelbare *Subderivationen* genannt werden. (Eine Derivation kann also eine Struktur von ineinandergeschachtelten Derivationen und Formeln sein.)

Subderivation

Auf solch einer Struktur kann eine Beziehung $\mathcal{D}$ ist Subderivation von $\mathcal{D}'$ rekursiv wie folgt definiert werden:

1) Jede Derivation $\mathcal{D}$ ist Subderivation von sich selbst ($\mathcal{D}$ sub $\mathcal{D}$).

2) Wenn $\mathcal{D}$ Subderivation von $\mathcal{D}'$ ist und $\mathcal{D}'$ ist ein Gegenstand von $\mathcal{D}''$, dann ist $\mathcal{D}$ auch Subderivation von $\mathcal{D}''$.

3) Sonst steht nichts in der Beziehung der Subderivation.

Ein einfacher Test zur Feststellung, ob zwischen zwei Derivationen die Relation der Subderivation besteht, geht wie folgt.

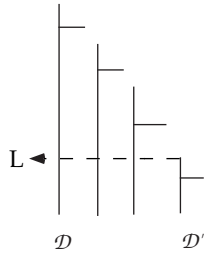
Frage: Ist $\mathcal{D}'$ Subderivation von $\mathcal{D}$ (d.h. $\mathcal{D}'$ sub $\mathcal{D}$)?

Methode:

1) Ziehe vom oberen Ende der Derivationslinie von $\mathcal{D}'$ eine horizontale Linie L nach links ins "Unendliche".

2) Stelle fest, ob L die Derivationslinie von $\mathcal{D}$ schneidet.

3) Wenn ja, denn ist $\mathcal{D}'$ sub $\mathcal{D}$; wenn nein, dann gilt nicht-($\mathcal{D}'$ sub $\mathcal{D}$).

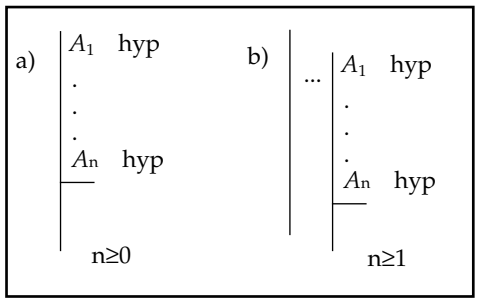


Die Regeln von $S_{\supset}$

Die Regel der Hypotheseneinführung (hyp):

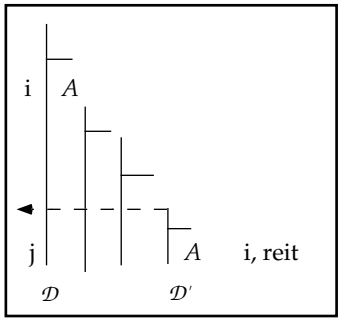
a) Um eine Derivation zu beginnen, kann man n ≥ 0 Hypothesen einführen, indem man eine Derivationslinie eröffnet und die n Formeln untereinander und unmittelbar rechts von der Derivationslinie und oberhalb des Hypothesenstrichs schreibt. Wenn n = 0, dann ist eine Derivationslinie ohne Hypothesenstrich zu verwenden.

b) In einer schon angefangenen Derivation kann man jederzeit n ≥ 1 Hypothesen einführen, indem man eine neue Derivationslinie eröffnet und die n Formeln untereinander und unmittelbar rechts von der Derivationslinie und oberhalb des Hypothesenstrichs schreibt.



Die Regel der Reiteration (reit):

Man kann eine in einer Derivation $\mathcal{D}$ schon gewonnene Formel A zu einem späteren Zeitpunkt in einer Subderivation $\mathcal{D}'$ von $\mathcal{D}$ wiederholen.



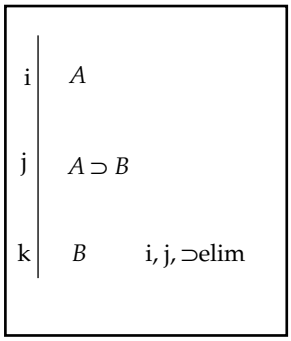
Für jede logische Konstante (hier: Konnektive) gibt es sogenannte *intelim*-Regeln, die den deduktiven Umgang mit dieser Konstanten festlegen. (Intelim für *introduction* und *elimination*.)

Regeln für den Konditional

Die Eliminierungsregel für den Konditional ist unser altbekannter Modus Ponens, während die Introduktionsregel dem Konditionalen Beweis entspricht:

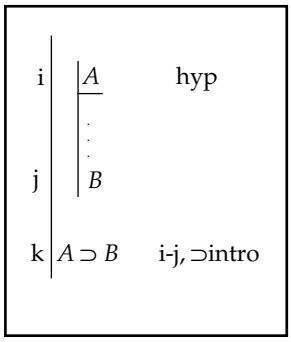
⊃Eliminierung (⊃elim):

Wenn man in einer Derivation einen Konditional, $A \supset B$, hat und dessen Antezedens, A, dann kann man zu einem späteren Zeitpunkt das Konsequens des Konditionals, B, hinschreiben.



⊃Introduktion (⊃intro):

Wenn man in einer Derivation eine (Sub-)Derivation hat, deren einzige Hypothese A und deren letzte Zeile B ist, dann kann man zu einem späteren Zeitpunkt den Konditional $A \supset B$ hinschreiben.



Beachte die Notation "i-j" in der Rechtfertigung von ⊃intro, anstelle von "i, j": die Bindestrich-Notation soll anzeigen, dass es sich bei der Prämisse um eine Derivation handelt, die sich von Zeile i bis zur Zeile j erstreckt. (Dies ist also eine 1-Prämissen-Regel!)

Beweis von A := eine (kategorische) Derivation mit 0 (äusseren) Hypothesen, deren letzter item die Formel A ist.

Ableitung von A aus Γ := eine (hypothetische) Derivation mit den Formeln aus Γ als einzigen (äusseren) Hypothesen und der Formel A als letzter item.

A ist beweisbar in $S_{\supset}$:= es gibt einen Beweis von A im System $S_{\supset}$. ($\vdash A$)

A ist in $S_{\supset}$ *ableitbar aus* Γ := es gibt eine Ableitung von A aus Γ im System $S_{\supset}$. ($\Gamma \vdash A$)

Beispiele:

Beweise der Axiome 1 und 2 von System S_0 aus Sitzung

(in Vorlesung gemacht)

Versuch eines Beweises von $((p \supset q) \supset p) \supset p$!

Soll diese Formel überhaupt in $S_{\supset}$ beweisbar sein?

Nur relativ zu einer Semantik von $S_{\supset}$ zu beantworten.

Wenn wir die übliche Interpretation von $\supset$ annehmen, dann ist die obige Formel logisch wahr; und wenn der Beweisbarkeitsbegriff das beweistheoretische Äquivalent des Begriffs der logischen Wahrheit sein soll, dann sollte diese Formel beweisbar sein.

Sonst wäre das System $S_{\supset}$ unvollständig!

Das System $S_{\supset\sim}$

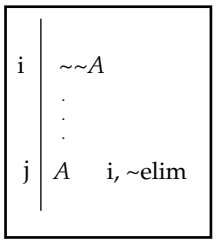
Keine Axiome, 6 Regeln: hyp, reit, ⊃intro, ⊃elim, ~intro, ~elim

Regeln für die Negation

Die Eliminierungsregel für die Negation ist die "Doppelte Negation", während die Introduktionsregel unser altbekannter Indirekter Beweis (Reductio ad absurdum) ist.

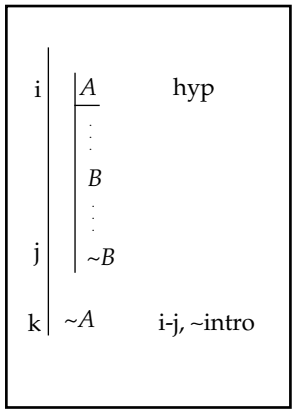
~Eliminierung (~elim):

Wenn man in einer Derivation eine doppelte Negation - also eine Formel der Form $\sim\sim A$ - hat, so kann man zu einem späteren Zeitpunkt die Formel A hinschreiben.



~Introduktion (~intro):

Wenn man in einer Derivation $\mathcal{D}$ eine (Sub-)Derivation hat, deren einzige Hypothese A und die zwei Zeilen B und $\sim B$ enthält, dann kann man zu einem späteren Zeitpunkt in $\mathcal{D}$ (oder einer Subderivation von $\mathcal{D}$) die Formel $\sim A$ hinschreiben.



In $S_{\supset\sim}$ ist die Peirce'sche Formel beweisbar! (Dieses System ist vollständig!)

An sich können mit den zwei Konnektiven von $S_{\supset\sim}$ sämtliche Wahrheitsfunktionen ausgedrückt werden (d.h. die Menge $\{\supset, \sim\}$ ist ausdrucksmässig vollständig).

Für das praktische Arbeiten mit der Logik ist es jedoch oft nützlich, wenn man noch andere Konnektive, z.B. die für *und*, *oder*, *gdw*, einführt.

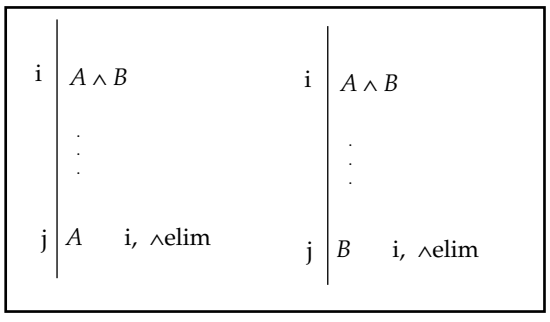
Dies führt uns zum System S_S, der vollen Satzlogik.

S_S hat, neben den Regeln von $S_{\supset\sim}$, noch 6 weitere intelim-Regeln.

Die Regeln für die Konjunktion

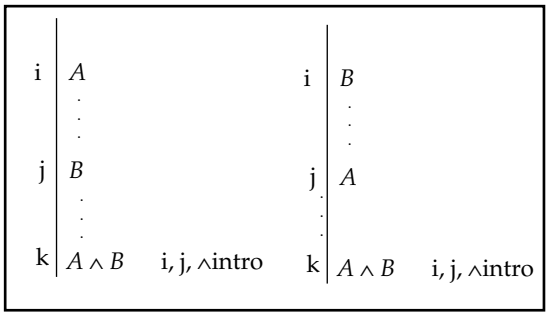
∧Eliminierung (∧elim):

Wenn man in einer Derivation $\mathcal{D}$ eine Konjunktion A & B hat, dann kann man zu einem späteren Zeitpunkt (in einer Subderivation $\mathcal{D}'$ von $\mathcal{D}$) eines der beiden Konjunkte, A oder B, hinschreiben.



∧Introduktion (∧intro):

Wenn man in einer Derivation $\mathcal{D}$ eine Formel A und in einer Derivation $\mathcal{D}'$ eine Formel B hat, so kann man zu einem späteren Zeitpunkt die Konjunktion von A und B, A & B, in einer Derivation $\mathcal{D}''$ hinschreiben, die sowohl zu $\mathcal{D}$ als auch zu $\mathcal{D}'$ subordiniert ist.

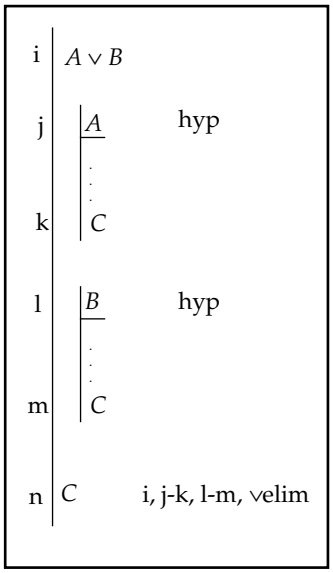


Regeln für die Disjunktion

Die Eliminierungsregel für die Disjunktion ist die altbekannte Fallunterscheidung oder das Dilemma. Die Introduktionsregel heisst auch oft Addition.

∨Eliminierung (∨elim):

Wenn man in einer Derivation eine Disjunktion $A \vee B$ hat und zwei (Sub-)Derivationen, deren eine als einzige Hypothese A und als letzte Zeile C hat und deren andere als einzige Hypothese B und als letzte Zeile C hat, so kann man zu einem späteren Zeitpunkt die Formel C hinschreiben.



∨Introduktion (∨intro):