
Logik & Semantik

4. Vorlesung

Aussagenlogik

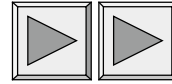
Ableitungs- und Beweisverfahren

- Verschiedene Ansätze: Exkurse
 - Die Idee der Uniformen Notation
 - Normalformenerstellung
-

Definition eines logischen Systems: Generelles Schema

Spezifikation

- einer formalen Sprache (zur Repräsentation)
- von Evaluations- / Interpretationsprinzipien
- semantischer Kategorisierungen und Beziehungen
- **Ableitungs-, Beweisverfahren**



Semantik – Beweisverfahren

Semantik / Modelltheorie

X ist allgemeingültig / eine Tautologie	$\models X$	$\neg X \models \perp$
X ist unerfüllbar / eine Kontradiktion	$X \models$	$X \models \perp$
X folgt aus S	$S \models X$	$\langle S \rangle \wedge \neg X \models \perp$

Beweistheorie

X ist beweisbar / ein Theorem	$\vdash X$	$\neg X \vdash \perp$
X ist widersprüchlich	$X \vdash \perp$	$X \vdash \perp$
X ist aus S ableitbar	$S \vdash X$	$\langle S \rangle \wedge \neg X \vdash \perp$

- Für die logischen Verfahren ist S bei Folgerung oder Ableitung einfach eine Formelmenge, auf deren Inhalt es weniger ankommt als auf deren Struktur. (Mit 'Struktur' ist hier gemeint, ob Teilformeln mehrfach vorkommen und mit welchen Junktoren sie verknüpft sind.)
- Für Anwendungen kommt dem Inhalt von S eine wesentliche Bedeutung zu. S kann zum Beispiel die Fakten einer Datenbank, ihre Konsistenzregeln und Regeln, die die Bedeutung der Datenfelder Beschreiben zusammenfassen.
- Im Bereich der KI spricht man auch von 'Wissensbasen', die neben Fakten auch Regeln enthalten.

Ableitung, Beweis

Basierend auf der syntaktischen Struktur der Sprache

Klassifikation von Formeln / Formelmengen

- Beweisbarkeit, Theorem
- Konsistenz
- Inkonsistenz, Widerspruch

Beziehung zwischen Formeln und Formelmengen

- Ableitbarkeit

Dabei sollte erreicht werden

- Korrektheit: „Das Verfahren darf nichts falsch machen !“
- Vollständigkeit: „Das Verfahren soll alles können.“

- Korrektheit
 - Nur Tautologien sind beweisbar / Theoreme.
 - Nur Kontradiktionen sind inkonsistent, widersprüchlich.
 - Nur folgerbare Formeln sind ableitbar.
- Vollständigkeit
 - Alle Tautologien sind beweisbar / Theoreme.
 - Alle Kontradiktionen sind inkonsistent, widersprüchlich.
 - Alle folgerbare Formeln sind ableitbar.

Ableitungs- und Beweisverfahren

können semantische Verfahren ersetzen

- also semantische Eigenschaften von Formeln und Formelmengen bzw. semantische Relationen bestimmen

sind syntaxorientiert

- jede Abfrage und jeder Arbeitsschritt betrachtet nichts als die Formelstruktur

sind semantisch motiviert

- die Rechtfertigung für die Arbeitsschritte basiert auf der Formelbedeutung
- einheitliche Behandlung beruht auf einheitlicher Bedeutung

- In der Aussagenlogik scheinen Beweisverfahren überflüssig zu sein, da man im Prinzip alles mit Wahrheitstafeln machen kann.
- Genauer gesagt: Gegeben eine aussagenlogische Formel oder eine endliche Menge von aussagenlogischen Formeln. Dann sind die Wahrheitstafeln der Formeln endlich (und in endlicher Zeit zu berechnen) und die Prüfung der semantischen Klassifikationen und Relationen (Tautologie, erfüllbar, Folgerung, ...) ist in endlicher Zeit möglich.
- Die Wahrheitstafelmethode funktioniert aber auch nur für die Aussagenlogik und nur für endliche Formelmengen (bzw. für Formelmengen mit endlich vielen Aussagesymbolen). Für andere Logiken, insbesondere die Prädikatenlogik, gibt es kein entsprechend einfaches Verfahren, da dort im Prinzip stets unendliche viele unterscheidbare Interpretationen in Betracht kommen.
- Beweisverfahren sind daher insbesondere für die komplexeren Logiken erforderlich. Da aber die Aussagenlogik so schön einfach ist und Aussagenlogik auch Teil jeder anderen Logik ist, werden die verschiedenen Verfahren auch immer an der Aussagenlogik demonstriert.

Typen von Ableitungs- und Beweisverfahren

„direkter Beweis“

- Aufbau von Formelfolgen, die von Axiomen und Annahmen zur Zielformel führen
 - z.B. Hilbert-Verfahren, Kalkül des natürlichen Schließens, Gentzen-Verfahren

Umformungsverfahren

- für Formeln
 - z.B. Äquivalenzumformungen zur Erstellung von Normalformen

Widerlegungsverfahren (*refutation*)

- Aufbau von (strukturierten) Formelmengen nach dem Prinzip der Erfüllbarkeitsäquivalenz
 - z.B. Resolutionsverfahren, Tableau-Verfahren



- „direkter Beweis“ entspricht den „deductive proofs“ bei Ben-Ari (2001)
- s. z.B. Def. 3.1 „deductive system“ und vergleiche dies mit dem Hilbert-System oder dem Gentzen-System.
- „Verfahren“ ist für vieles was jetzt folgt eigentlich zu hoch gegriffen, denn es werden hier viele wichtige Punkte, die z.B. für eine Implementation wichtig sind, außer acht gelassen.
- Hier wird auf die strukturellen Restriktionen fokussiert, die sich in den dargestellten Kalkülen fassen lassen.

Exkurs: Hilbert-Kalkül

Hilbert-Axiome

(H1) $P \supset (Q \supset P)$

(H2) $(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R))$

(H3) $(\neg P \supset \neg Q) \supset (Q \supset P)$

MP: $\frac{P, (P \supset Q)}{Q}$

Inferenzregel: Modus Ponens

Ein ‚Beweis‘

- Gegeben: Menge von Annahmen (Formeln), Datenbasis
- Eine Ableitung ist eine Folge von Formeln F_1, F_2, \dots, F_n wobei
 - F_i eine Annahme,
 - F_i durch uniforme Ersetzung aus einem Axiom hervorgegangen,
 - oder F_i Resultat der Anwendung des Modus Ponens auf zwei vorhergehende Glieder der Folge ist

- Mehr Details finden sich z.B. in den Unterlagen zu F1, bei Fitting (1996, p.77 ff) oder bei Ben-Ari (2001, p.48 ff)

Beispiel: Beweis im Hilbert-Ansatz

1.	$(P \supset ((P \supset P) \supset P)) \supset ((P \supset (P \supset P)) \supset (P \supset P))$	H2
2.	$P \supset ((P \supset P) \supset P)$	H1
3.	$((P \supset (P \supset P)) \supset (P \supset P))$	1, 2, MP
4.	$P \supset (P \supset P)$	H1
5.	$P \supset P$	3, 4, MP

Beweisführung im Hilbertansatz

- Erfordert vielfach die Betrachtung von Formeln, die viel komplexer als die zu beweisende Formel sind.

Weitere Junktoren

- Definition unter Verwendung von Implikation und Negation oder
- weitere Axiome und Regeln.

Exkurs: Kalkül des natürlichen Schließens



Grundidee

- keine Axiome, dafür viele Regeln (,Junktoren-Einführung', ,Junktoren-Löschung')
- eingebettete Beweise ermöglichen (erste Zeile: auch ungerechtfertigte Annahme)

	\perp	$\frac{X}{\neg X}$	$\frac{X}{X}$	$\frac{X \wedge Y}{X}$	$\frac{X \wedge Y}{Y}$	$\frac{X \quad Y}{X \wedge Y}$	$\frac{X \quad \neg Y}{X \supset Y}$	$\frac{\neg Y}{X \supset Y}$
\top	X	\perp	X	X	Y	$X \wedge Y$	Y	$\neg X$

X	$\neg X$	X	$\neg Y$
⋮	⋮	⋮	⋮
\perp	\perp	Y	$\neg X$
$\neg X$	X	$X \supset Y$	$X \supset Y$

Ch. Habel / C. Eschenbach: Logik & Semantik

4 – 9

- Durch die Kästen werden Unter-Beweise mit ihren ersten und letzten Zeilen signalisiert. Die erste Zeile ist dabei immer eine beliebige Annahme, die nicht gerechtfertigt werden muss. Die Zeile unter dem Kasten ist durch den Kasteninhalt gerechtfertigt. In nachfolgenden Ableitungsschritten darf dann nicht mehr auf den Inhalt des Kastens zugegriffen werden: Der Kasten wird abgeschlossen. Allerdings darf bei Beweisschritten, die in einem Kasten erfolgen, auf die Formel außerhalb des Kastens (bzw. vor/über dem Kasten) zugegriffen werden. Bei den Regeln ohne Kasten, ist die Reihenfolge der Formeln über dem Strich irrelevant.
- Annahmen aus der Datenbasis dürfen an beliebiger Stelle hinzugenommen werden.
- s.a. Fitting (1996, 86 ff), Ben-Ari (2001, 63)

Beispiel: Beweis im Kalkül des natürlichen Schließens

1.	$P \supset (Q \supset R)$	Annahme
2.	Q	Annahme
3.	P	Annahme
4.	$Q \supset R$	Impl. L. 1, 3
5.	R	Impl. L. 4, 2
6.	$P \supset R$	Impl. E. 3, 5
7.	$Q \supset (P \supset R)$	Impl. E. 2, 6
	$((P \supset (Q \supset R)) \supset (Q \supset (P \supset R)))$	Impl. E. 1, 7



Weitere Junktoren erfordern

- weitere Regeln

- Übungsaufgabe
- Beweisen Sie die drei Hilbert-Axiome im Kalkül des natürlichen Schließens.

Exkurs: Gentzen-Kalkül



ein Vorläufer der Tableau-Systeme

- Basiseinheiten sind Sequenzen (Paare von Formellisten, abgekürzt durch Γ und Δ , verbunden mit \rightarrow bzw. \vdash):
 $X_1, \dots, X_n \rightarrow Y_1, \dots, Y_m$ bzw. $X_1, \dots, X_n \vdash Y_1, \dots, Y_m$
- Intendierte Interpretation: die X_i rechtfertigen gemeinsam die Disjunktion der Y_j .
- Die leere Formelliste links entspricht \top , die leere Formelliste rechts entspricht \perp .

Axiome

$$X \rightarrow X$$

$$\perp \rightarrow$$

$$\rightarrow \top$$

Strukturelle Regel (Ausdünnung)

Wenn $\Gamma_1 \subseteq \Gamma_2$ und $\Delta_1 \subseteq \Delta_2$

$$\frac{\Gamma_1 \rightarrow \Delta_1}{\Gamma_2 \rightarrow \Delta_2}$$

- Wir verwenden (wie Gentzen) im Weitere \rightarrow . Das Symbol \vdash stellt eine notationelle Alternative dar.
- s. a. Fitting (1996, p92 ff), Ben-Ari (2001, 62)

Gentzen-Regeln

Negation

$$\frac{\Gamma \rightarrow \Delta, X}{\Gamma, \neg X \rightarrow \Delta} \quad \frac{\Gamma, X \rightarrow \Delta}{\Gamma \rightarrow \Delta, \neg X}$$



Konjunktion

$$\frac{\Gamma, X, Y \rightarrow \Delta}{\Gamma, X \wedge Y \rightarrow \Delta} \quad \frac{\Gamma \rightarrow \Delta, X \quad \Gamma \rightarrow \Delta, Y}{\Gamma \rightarrow \Delta, X \wedge Y}$$

Disjunktion

$$\frac{\Gamma \rightarrow \Delta, X, Y}{\Gamma \rightarrow \Delta, X \vee Y} \quad \frac{\Gamma, X \rightarrow \Delta \quad \Gamma, Y \rightarrow \Delta}{\Gamma, X \vee Y \rightarrow \Delta}$$

Implikation

$$\frac{\Gamma, X \rightarrow \Delta, Y}{\Gamma \rightarrow \Delta, X \supset Y} \quad \frac{\Gamma \rightarrow \Delta, X \quad \Gamma, Y \rightarrow \Delta}{\Gamma, X \supset Y \rightarrow \Delta}$$





Beweis mit dem Gentzen-System

Ein ‚Beweis‘

- Eine **Ableitung** ist ein Baum von Sequenzen $\Sigma_1, \Sigma_2, \dots, \Sigma_n$ wobei
 - Σ_i durch uniforme Ersetzung aus einem Axiom hervorgegangen ist
 - oder Σ_i Resultat der Anwendung einer Regel auf vorhergehende Glieder des Baumes ist
- Die Sequenz $\Gamma \rightarrow \Delta$ ist **ableitbar**, wenn sie Wurzel eines Ableitungsbaums ist.
- Formel F ist **beweisbar**, wenn die Sequenz $\rightarrow F$ ableitbar ist. (F bedarf also keiner Rechtfertigung.)
- Formel F ist **aus Datenbasis S ableitbar**, wenn die Sequenz $S \rightarrow F$ ableitbar ist. (S rechtfertigt F).

- Ableitung ist hier also fast genauso definiert wie bei Hilbert, mit dem Unterschied, dass bei Hilbert eine Folge von Formeln gebildet wird, hier aber ein Baum von Sequenzen.

Beispiel-Beweis im Gentzenkalkül

$P \rightarrow P$	$Q \rightarrow Q$	Axiom	
$P, Q \rightarrow P$	$P, Q \rightarrow Q$	Ausdünnung	
$P, Q \rightarrow (P \wedge Q)$		Konj.Einf.R.	
$Q \rightarrow (P \wedge Q), \neg P$		Neg.Einf.R.	
$\rightarrow (P \wedge Q), \neg P, \neg Q$		Neg.Einf.R.	
$\neg(P \wedge Q) \rightarrow \neg P, \neg Q$		Neg.Einf.L.	
$\neg(P \wedge Q) \rightarrow (\neg P \vee \neg Q)$		Disj.Einf.R.	
$\rightarrow \neg(P \wedge Q) \supset (\neg P \vee \neg Q)$		Impl.Einf.R.	

Weitere Junktoren erfordern

- Weitere Regeln

- ‚von oben nach unten‘ erscheint der Aufbau des Beweises wie Magie: Wie soll man auf diesen Ablauf kommen?
- ‚von unten nach oben‘ ist das schon viel plausibler, wenn man beachtet, welche Regeln überhaupt zur Verfügung stehen.

Präsenzaufgabe 4-1

Führen Sie einen Beweis im Gentzen-Kalkül für

$$(\neg P \vee \neg Q) \supset \neg(P \wedge Q)$$

Tipp: Bauen Sie den Beweis ‚von unten‘ auf.



- Übungsaufgabe
- Beweisen Sie die drei Hilbert-Axiome mit dem Gentzen-Kalkül

Die Junktorenmengen in den vorgestellten Kalkülen

- Hilbert: \neg, \supset
- Natürliches Schließen: \neg, \supset, \wedge
- Gentzen: $\neg, \vee, \supset, \wedge$

Weitere Junktoren

- als Abkürzungen
 - $(\neg P \vee \neg Q) \supset \neg(P \wedge Q)$ ist Abk. für $(\neg\neg P \supset \neg\neg Q) \supset \neg\neg(P \supset \neg Q)$
- weitere Axiome (Hilbert)
- weitere Regeln (natürliches Schließen, Gentzen)

Uniforme Notation (Raymond Smullyan, 1968)



Einerseits: Vielfalt der Junktoren nutzen

- Es stehen alle (8) primären Junktoren zur Verfügung.

Andererseits

- Gemeinsamkeiten im „Wahrheitswertverhalten“ der Junktoren sollen berücksichtigt werden.
- Primäre Junktoren haben bei den resultierenden Wahrheitswerten ein Verhältnis von 3:1 oder von 1:3
- Zwei (duale) Typen von primären Junktoren: Konjunktive Junktoren (3 falsch, 1 wahr) und Disjunktive Junktoren (1 falsch, 3 wahr)

- Uniforme Notation geht auf Smullyan, R. M. (1968). *First-Order Logic*. Berlin: Springer-Verlag. (repr. by New York: Dover, 1995.)

Die sekundären Junktoren wurden von Smullyan nicht direkt in den Rahmen der uniformen Notation eingebettet. Sollen sie genutzt werden, dann erfordern sie ebenso wie die Negation oder die logischen Konstanten eine individuelle Behandlung. Zur Vereinfachung werden wir im folgenden davon ausgehen, dass die Formeln keine sekundären Junktoren enthalten, wenn die uniforme Notation eine Rolle spielt.

Steht auf den folgenden Folien doch noch einmal $(X \equiv Y)$, dann ist dies als Abkürzung für $((X \supset Y) \wedge (Y \supset X))$ zu verstehen.

**Uniforme Notation Primäre Junktoren:
 α -Formeln – β -Formeln**

Konjunktiv			Disjunktiv		
α	α_1	α_2	β	β_1	β_2
$X \wedge Y$	X	Y			
$X \supset Y$	X	$\neg Y$			
$X \not\supset Y$	$\neg X$	Y			
$X \downarrow Y$	$\neg X$	$\neg Y$			
			$X \uparrow Y$	$\neg X$	$\neg Y$
			$X \supset Y$	$\neg X$	Y
			$X \subset Y$	X	$\neg Y$
			$X \vee Y$	X	Y

Uniforme Notation Primäre Junktoren: α -Formeln – β -Formeln

Konjunktiv			Disjunktiv		
α	α_1	α_2	β	β_1	β_2
$X \wedge Y$	X	Y	$\neg(X \wedge Y)$	$\neg X$	$\neg Y$
$X \supset Y$	X	$\neg Y$	$\neg(X \supset Y)$	$\neg X$	Y
$X \not\subset Y$	$\neg X$	Y	$\neg(X \not\subset Y)$	X	$\neg Y$
$X \downarrow Y$	$\neg X$	$\neg Y$	$\neg(X \downarrow Y)$	X	Y
$\neg(X \uparrow Y)$	X	Y	$X \uparrow Y$	$\neg X$	$\neg Y$
$\neg(X \supset Y)$	X	$\neg Y$	$X \supset Y$	$\neg X$	Y
$\neg(X \subset Y)$	$\neg X$	Y	$X \subset Y$	X	$\neg Y$
$\neg(X \vee Y)$	$\neg X$	$\neg Y$	$X \vee Y$	X	Y



- Die Motivation und Rechtfertigung für die Bezeichnungen „Konjunktiv“ und „Disjunktiv“ ergibt sich aus Theorem 2.6.1 (nächste Folie).
- Wir können die Tabelle in zwei Weisen lesen: ausgehend von α bzw. β zur Bestimmung der zugehörigen α_1 und α_2 bzw. β_1 und β_2 oder umgekehrt ausgehend von zwei Formeln X und Y auf der Suche nach komplexen Formeln des α - oder β - Typs.

Aufgabe ohne Nummer (vgl. Folie 2-29)

Stellen Sie fest, wie sich die Dualität der Junktoren in dieser Tabelle widerspiegelt.

Welche Paare von Junktoren sind dual zueinander?

Uniforme Notation – Boolesche Auswertung

Theorem 2.6.1*

Für alle α -Formeln α und die zugehörigen α_1 und α_2 gilt:

- α und $(\alpha_1 \wedge \alpha_2)$ sind äquivalent

Für alle β -Formeln β und die zugehörigen β_1 und β_2 gilt:

- β und $(\beta_1 \vee \beta_2)$ sind äquivalent

Theorem 2.6.1: (So wie Fitting es formuliert)

Für alle Booleschen Bewertungen ν und alle α - und β -Formeln gilt:

- $\nu(\alpha) = \nu(\alpha_1) \wedge \nu(\alpha_2)$
 - $\nu(\beta) = \nu(\beta_1) \vee \nu(\beta_2)$
- Aufgabe (ohne Nummer).
Beweisen Sie Theorem 2.6.1.

Aufgabe (ohne Nummer)

Schlagen Sie vor, wie α - und β -Formeln für die sekundären Junktoren gewählt werden könnten, so dass auch für diese Fälle Theorem 2.6.1 zutrifft.

Welche Gründe gibt es wohl, diese Möglichkeit nicht in die uniforme Notation aufzunehmen?

Nutzen der uniformen Notation für Beweisverfahren

Die Formulierung der Regeln

- Muss sich nicht auf die einzelnen Junktoren beziehen sondern kann die Klassifikation als α -Formel oder β -Formel und die Zuordnung der α_1 und α_2 bzw. β_1 und β_2 voraussetzen.

Beispiel: Kalkül des natürlichen Schließens



Regeln auf der Basis der Uniformen Notation

- Primäre Junktoren tauchen nicht explizit auf
- Semantisch einheitliches Verhalten resultiert in einheitlicher Nutzung bei den Beweisen.

	\perp	$\frac{X}{\neg X}$	$\frac{X}{X}$	$\frac{\alpha}{\alpha}$	$\frac{\alpha}{\alpha}$	$\frac{\alpha_1}{\alpha_2}$	$\frac{\neg \beta_1}{\beta}$	$\frac{\neg \beta_2}{\beta}$
\top	X	\perp	X	α_1	α_2	α	β_2	β_1
X	$\neg X$	$\neg \beta_1$	$\neg \beta_2$					
\vdots	\vdots	\vdots	\vdots					
\perp	\perp	β_2	β_1					
$\neg X$	X	β	β					

- s. Fitting (1996, p. 88)

Gentzen-Regeln auf der Basis der Uniformen Notation

Negation

$$\frac{\Gamma \rightarrow \Delta, X}{\Gamma, \neg X \rightarrow \Delta} \quad \frac{\Gamma, X \rightarrow \Delta}{\Gamma \rightarrow \Delta, \neg X}$$



α -Regeln

$$\frac{\Gamma, \alpha_1, \alpha_2 \rightarrow \Delta}{\Gamma, \alpha \rightarrow \Delta} \quad \frac{\Gamma \rightarrow \Delta, \alpha_1 \quad \Gamma \rightarrow \Delta, \alpha_2}{\Gamma \rightarrow \Delta, \alpha}$$

β -Regeln

$$\frac{\Gamma \rightarrow \Delta, \beta_1, \beta_2}{\Gamma \rightarrow \Delta, \beta} \quad \frac{\Gamma, \beta_1 \rightarrow \Delta \quad \Gamma, \beta_2 \rightarrow \Delta}{\Gamma, \beta \rightarrow \Delta}$$

- Die Implikation ist mit der Negations- und der β -Regel schon erfasst.

- Ben-Ari (2001, p. 45ff) verwendet eine Variante des Gentzen-systems, bei der die Formelmenge auf der linken Seite immer leer ist.

Prinzip der Strukturellen Induktion [Uniforme Notation]



Theorem 2.6.3*

Wenn für eine Eigenschaft Q die Bedingungen [V1]–[V4] gelten, dann gilt Q für alle Formeln der Aussagenlogik, die keine sekundären Junktoren enthalten.

Induktionsbasis_{UN}

- [V1] Jede atomare Formel X und ihre Negation $\neg X$ besitzen die Eigenschaft Q .

Induktionsschritt_{UN}

- [V2] Falls X die Eigenschaft Q hat, dann auch $\neg\neg X$.
- [V3] Falls α_1 und α_2 die Eigenschaft Q haben, dann auch α .
- [V4] Falls β_1 und β_2 die Eigenschaft Q haben, dann auch β .

- Auf den ersten Blick wirkt diese Version des Prinzips der Strukturellen Induktion komplexer als die Standardvariante (Theorem 2.2.3). Der Vorteil liegt jedoch darin, dass Eigenschaften von Verfahren, die auf der uniformen Notation basieren (z.B. Tableau-Verfahren), mit Hilfe dieser Version des Prinzips besser untersucht werden können.
- Wichtig ist es, hier festzuhalten, dass es mehr als ein Induktionsprinzip geben kann, das die Strukturierung von Beweisen unterstützt. Das hier vorgestellte Verfahren setzt z.B. voraus, dass die Formeln keinen sekundären Junktor enthalten. Grundsätzlich hängt es von der zu beweisenden Behauptung ab, welches Prinzip sinnvoller einzusetzen ist.

Zum Selbststudium: Beweis des Prinzips der Strukturellen Induktion [UN]

Voraussetzung

Q sei eine Eigenschaft, die die Bedingungen [V1]–[V4] erfüllt.

[V1] Jede atomare Formel X und ihre Negation $\neg X$ besitzen die Eigenschaft Q .

[V2] Falls X die Eigenschaft Q hat, dann hat auch $\neg\neg X$.

[V3] Falls α_1 und α_2 die Eigenschaft Q haben, dann auch α .

[V4] Falls β_1 und β_2 die Eigenschaft Q haben, dann auch β .

Arbeitsdefinition

- Eine Formel X heißt *gut*, falls sowohl X als auch $\neg X$ die Eigenschaft Q haben.
- Wenn wir zeigen, dass alle Formel *gut* sind, dann ist damit die neue Version des Prinzips, d.h. das Theorem 2.6.3, bewiesen.

Zum Selbststudium:
Prinzip der Strukturellen Induktion [UN] [Beweis – Forts.]

- Die ursprüngliche Version des Induktionsprinzips wird verwendet, um die UN-Version zu beweisen.

Induktionsbasis

- Wenn X atomar ist, so besitzen X und $\neg X$ die Eigenschaft Q nach [V1], also ist X *gut*.

Induktionsannahme

- X und Y sind *gut*.

Induktionsschritt

- X und $\neg X$ haben die Eigenschaft Q . [Ind. Annahme, def. *gut*]
- $\neg\neg X$ hat die Eigenschaft Q . [V2]
- Also haben $\neg X$ und $\neg\neg X$ die Eigenschaft Q und $\neg X$ ist *gut*. [def. *gut*]
- ➔ Menge *guter* Formeln ist **abgeschlossen unter Negation**.

Zum Selbststudium:
Prinzip der Strukturellen Induktion [UN] [Beweis – 2. Forts.]

Induktionsschritt [Fortsetzung]

- Sei \odot ein binärer primärer Junktor.
- $(X \odot Y)$ ist eine α -Formel oder β -Formel.
- Wenn $(X \odot Y)$ α -Formel, so ist $\neg(X \odot Y)$ eine β -Formel.
Wenn $(X \odot Y)$ β -Formel, so ist $\neg(X \odot Y)$ eine α -Formel. α - β -Tabelle
- $\{\alpha_1, \beta_1\} = \{X, \neg X\}$ und $\{\alpha_2, \beta_2\} = \{Y, \neg Y\}$
- $X, \neg X, Y, \neg Y$ haben die Eigenschaft **Q**. [Ind. Annahme][def. gut]
- Also haben $(X \odot Y)$ und $\neg(X \odot Y)$ die Eigenschaft **Q**. [V3], [V4]
- Also ist $(X \odot Y)$ gut.

Resümee

- Nach dem Prinzip der Strukturellen Induktion ist also jede Formel der AL ohne sekundären Junktor gut und hat die Eigenschaft **Q**.

Aufgabe (ohne Nummer)

Betrachten Sie nochmal Ihren Vorschlag zur Behandlung der Biimplikation (und Exklusiv-oder) im Rahmen der Uniformen Notation (Folie 4-19).

Ist der oben stehende Beweis auch für diese Erweiterung der Notation korrekt?

Wie muss er ggf. ergänzt oder modifiziert werden, um die Formeln mit sekundären Junktoren auch abzudecken?

Prinzip der Strukturellen Rekursion [Uniforme Notation]

Theorem 2.6.4

Es existiert genau eine Funktion f definiert über der Menge \mathcal{L}_{AL} aller Formeln der Aussagenlogik, die keinen sekundären Junktor enthalten, so dass gilt

Rekursionsbasis_{UN}

Der Wert von f ist explizit spezifiziert auf den atomaren Formeln und ihren Negationen.

Rekursionsschritt_{UN}

- Der Wert $f(\neg\neg X)$ ist spezifiziert über den Wert von $f(X)$.
- Der Wert $f(\alpha)$ ist spezifiziert über $f(\alpha_1)$ und $f(\alpha_2)$.
- Der Wert $f(\beta)$ ist spezifiziert über $f(\beta_1)$ und $f(\beta_2)$.

Anmerkung zur Beziehung zwischen der Standardvariante des Prinzips der Strukturellen Rekursion (2.2.4) und der UN-Version:

- Während in Theorem 2.2.4 explizit die verschiedenen Junktoren im Hinblick auf ihren Beitrag zur Berechnung des Wertes von f berücksichtigt werden müssen, geschieht dies in Theorem 2.6.4 implizit über ihren Beitrag die verschiedenen Typen eine α - bzw. β -Formeln bilden zu können.

Auch hier gilt, dass es von der zu definierenden Funktion abhängt, welches Rekursionsprinzip besser zu nutzen ist.

Beispiel: Strukturelle Rekursion_{UN}: Rang einer Formel

Definition 2.6.5

Der **Rang** r einer Formel ist definiert durch:

$r: \mathcal{L}_{AL} \rightarrow \mathbb{N}_0$

- Rekursionsbasis_{UN}

Ist P ein Aussagensymbol, so ist $r(P) = r(\neg P) = 0$

- Rekursionsschritt_{UN}

- $r(\top) = r(\perp) = 0$

$$r(\neg\top) = r(\neg\perp) = 1$$

- $r(\neg\neg X) = r(X) + 1$

- $r(\alpha) = r(\alpha_1) + r(\alpha_2) + 1$

$$r(\beta) = r(\beta_1) + r(\beta_2) + 1$$

Die Funktion Rang wird später nützlich sein, wenn wir Verfahren behandeln, die einerseits auf der uniformen Notation beruhen und andererseits eine Bewertung benötigen, welche Formeln einfacher als andere sind.

Im Rahmen der uniformen Notation können wir auch sagen, dass eine Formel mit einem niedrigeren Rang einfacher ist als eine Formel mit höherem Rang. Wie man sieht, ist in diesem Sinne die Negation eines Aussagensymbols nicht komplexer als das Aussagensymbol selbst. Literale haben also eine einheitliche Komplexität.

Insgesamt wird das Negationssymbol eigentlich nur gerechnet, wenn es doppelt oder vor einer logischen Konstante auftaucht. Das heißt dann aber auch, dass die Negation von Formeln mit gleichem Rang nicht unbedingt zu Formeln mit gleichem Rang führt.

Aufgabe (ohne Nummer)

Betrachten Sie nochmal Ihren Vorschlag zur Behandlung der Bimplikation (und Exklusiv-oder) im Rahmen der Uniformen Notation (Folie 4-19).

Ist die Funktion Rang auch für Formeln mit diesen Junktoren sinnvoll (und eindeutig) definiert?

Präsenzaufgabe 4-2

Welchen Rang r besitzt die Formel

$$\neg ((P_1 \supset P_2) \vee ((P_3 \wedge P_2) \supset \top)) ?$$

Tipp

- erstmal klären, wie wir am besten vorgehen.



Typen von Ableitungs- und Beweisverfahren

„direkter Beweis“

- Aufbau von Formelfolgen, die von Axiomen und Annahmen zur Zielformel führen
 - z.B. Hilbert-Verfahren, Kalkül des natürlichen Schließens, Gentzen-Verfahren

Umformungsverfahren

- für Formeln
 - z.B. Äquivalenzumformungen zur Erstellung von Normalformen

Widerlegungsverfahren (*refutation*)

- Aufbau von (strukturierten) Formelmengen nach dem Prinzip der Erfüllbarkeitsäquivalenz
 - z.B. Resolutionsverfahren, Tableau-Verfahren



- ‚Verfahren‘ ist für vieles was jetzt folgt eigentlich zu hoch gegriffen, denn es werden hier viele wichtige Punkte, die z.B. für eine Implementation wichtig sind, außer acht gelassen.
- Hier wird auf die strukturellen Restriktionen fokussiert, die sich in den dargestellten Kalkülen fassen lassen.

Generalisierte Disjunktion - Generalisierte Konjunktion

Definition 2.8.1

Sei X_1, X_2, \dots, X_n eine – eventuell leere – Liste von Formeln. Es werden zwei neue Typen von Formeln definiert, und zwar

- $[X_1, X_2, \dots, X_n]$, die *generalisierte Disjunktion* von X_1, X_2, \dots, X_n
- $\langle X_1, X_2, \dots, X_n \rangle$, die *generalisierte Konjunktion* von X_1, X_2, \dots, X_n

Für Booleschen Bewertungen v wird festgelegt:

- $v([X_1, X_2, \dots, X_n]) = \mathbf{t}$ falls v mindestens ein X_i zu \mathbf{t} auswertet, anderenfalls $v([X_1, X_2, \dots, X_n]) = \mathbf{f}$,
- $v(\langle X_1, X_2, \dots, X_n \rangle) = \mathbf{f}$ falls v mindestens ein X_i zu \mathbf{f} auswertet, anderenfalls $v(\langle X_1, X_2, \dots, X_n \rangle) = \mathbf{t}$.

Erweiterung von Rang für generalisierte Disjunktion bzw. Konjunktion:

$$\bullet r([X_1, X_2, \dots, X_n]) = \sum_{i=1}^n r(X_i) \qquad r(\langle X_1, X_2, \dots, X_n \rangle) = \sum_{i=1}^n r(X_i)$$

- Die Einführung von generalisierter Konjunktion bzw. Disjunktion stellt eine Erweiterung der Syntax dar. Wir haben es jetzt – im strikten Sinne – mit einer neuen Sprache der Aussagenlogik zu tun. Daher muss auch die Semantik erweitert werden.

Aufgabe (ohne Nummer)

Da generalisierte Konjunktion und Disjunktion auch für eine leere Liste von Formeln definiert sind, muss auch für diesen Fall das Verhalten der Booleschen Bewertungen festgelegt sein.

Dies geschieht in folgender Weise:

$v([\])$ = \mathbf{f} für beliebige v .

- Argumentieren Sie dafür, warum diese Festlegung sinnvoll ist.
- Überlegen Sie, wie $v(\langle \ \rangle)$ charakterisiert werden soll.

Wenn \mathbf{S} eine endliche Formelmengende ist, werde ich später auch die Kurzschreibweise $\langle \mathbf{S} \rangle$ verwenden, um mich auf die Konjunktion der Formeln aus \mathbf{S} zu beziehen.

Warum ist es dafür wichtig, dass \mathbf{S} endlich ist?

Literale und Klauseln

Definition 2.8.2

Aussagensymbole, ihre Negationen und die beiden logischen Konstanten \perp , \top bilden die Menge der *Literale*.

Definition 2.8.3

Eine *Klausel* ist ein generalisierte Disjunktion $[X_1, X_2, \dots, X_n]$, in der alle Elemente Literale sind.

Eine *duale Klausel* ist ein generalisierte Konjunktion $\langle X_1, X_2, \dots, X_n \rangle$, in der alle Elemente Literale sind.

Eine Formel ist in *konjunktiver Normalform (Klauselnormalform)*, wenn sie eine Konjunktion $\langle C_1, C_2, \dots, C_n \rangle$ von Klauseln C_i ist.

Eine Formel ist in *disjunktiver Normalform (dualer Klauselnormalform)*, wenn sie eine Disjunktion $[D_1, D_2, \dots, D_n]$ von dualen Klauseln D_i ist.

Normalformen

Definition

- Eine Formel X ist genau dann eine **konjunktive Normalform von einer Formel Y** , wenn X in konjunktiver Normalform ist und X und Y äquivalent sind.
- Eine Formel X ist genau dann eine **disjunktiver Normalform von einer Formel Y** , wenn X in disjunktiver Normalform ist und X und Y äquivalent sind.

Theorem 2.8.4

- Es gibt Algorithmen, die zu jeder aussagenlogischen Formel eine konjunktive bzw. disjunktive Normalform berechnen.

Semantische Beobachtungen

Eine Konjunktion $\langle X_1, X_2, \dots, X_n \rangle$

- ist genau dann eine Tautologie, wenn alle X_i Tautologien sind

Eine Disjunktion $[X_1, X_2, \dots, X_n]$

- ist genau dann unerfüllbar, wenn alle X_i unerfüllbar sind.

Eine Konjunktion von Literalen $\langle X_1, X_2, \dots, X_n \rangle$

- ist genau dann unerfüllbar, wenn für ein i $X_i = \perp$ oder für ein i und ein j gilt: $X_i = \neg X_j$

Eine Disjunktion von Literalen $[X_1, X_2, \dots, X_n]$

- ist genau dann eine Tautologie, wenn für ein i $X_i = \top$ oder für ein i und ein j gilt: $X_i = \neg X_j$

Mögliche Verfahren

zur Tautologieprüfung einer Formel F

- Bilde die Konjunktive Normalform $\langle C_1, C_2, \dots, C_n \rangle$ zu F
- Prüfe, ob es eine Klausel C_i gibt, die weder \top noch ein Paar $X, \neg X$ enthält.
- F ist genau dann eine Tautologie, wenn die Überprüfung fehlschlägt.

zur Erfüllbarkeitsprüfung einer Formel F

- Bilde die Disjunktive Normalform $[D_1, D_2, \dots, D_n]$ zu F
- Prüfe, ob es eine duale Klausel D_i gibt, die weder \perp noch ein Paar $X, \neg X$ enthält.
- F ist genau dann unerfüllbar, wenn die Überprüfung fehlschlägt.

- So, wie oben dargestellt, sind die Verfahren blind und ineffizienter als erforderlich. Prüfungen können bereits an Zwischenergebnissen der Umformung vorgenommen werden, so dass in vielen Fällen Antworten auch gegeben werden können, ohne dass die Normalformen gebildet werden.

Mögliche Verfahren (2)

zur Tautologieprüfung einer Formel F

- Bilde die Konjunktive Normalform $\langle C_1, C_2, \dots, C_n \rangle$ zu F
- Lösche alle Klauseln C_i , die \top oder ein Paar $X, \neg X$ enthalten.
- F ist genau dann eine Tautologie, wenn das Resultat $\langle \rangle$ ist.

zur Erfüllbarkeitsprüfung einer Formel F

- Bilde die Disjunktive Normalform $[D_1, D_2, \dots, D_n]$ zu F
- Lösche alle dualen Klauseln D_i , die \perp oder ein Paar $X, \neg X$ enthalten.
- F ist genau dann unerfüllbar, wenn das Resultat $[]$ ist.

➤ Dies ist die Grundidee des aussagenlogischen Tableau-Verfahrens

Verfahren basierend auf Uniformer Notation: Erstellung einer konjunktive Normalform

Eingabe

- X eine Formel der Aussagenlogik

Schritt 1

- Beginne mit $\langle [X] \rangle$

Resultat nach Schritt n

- sei $\langle D_1, D_2, \dots, D_n \rangle$, wobei die D_i Disjunktionen sind. Wenn dies noch keine konjunktive Normalform ist, dann ist eines der D_i keine Klausel. Es ist folgendes zu tun:

Schritt n+1

- Wähle ein Element D_i , das keine Klausel ist. D_i enthält eine Formel N , die kein Literal ist.
- Führe folgende Ersetzung durch:

Ersetzung in der Normalformenerstellung : Konjunktive Normalform



- Falls $N = \neg\top$, so ersetze N durch \perp .
- Falls $N = \neg\perp$, so ersetze N durch \top .
- Falls $N = \neg\neg Z$, so ersetze N durch Z .
- Falls N eine β -Formel, so ersetze N durch die Sequenz β_1, β_2 .
- Falls N eine α -Formel, so ersetze die Disjunktion D_i durch die Sequenz von Disjunktionen D_{i1} und D_{i2} , wobei D_{i1} aus D_i hervorgeht, indem N durch α_1 ersetzt wird und D_{i2} aus D_i hervorgeht, indem N durch α_2 ersetzt wird.

Beispiel

Der β -Fall	$\langle [R, P \supset Q] \rangle$	$\langle [R, \neg P, Q] \rangle$
Der α -Fall	$\langle [R, \neg(P \supset Q)] \rangle$	$\langle [R, P], [R, \neg Q] \rangle$

Aufgabe (ohne Nummer)

- Machen Sie einen Vorschlag zur Behandlung der sekundären Junktoren bei der Normalformenerstellung.
- Übersetzen Sie die Beispiele in die Standard-Schreibweise ohne generalisierte Konjunktionen und Disjunktionen
- Führen Sie – zur Übung – einige Normalformenerstellungen durch.

Erstellung von Klauselmengen (Konjunktive Normalform)

Ersetzungsregeln basierend auf der Uniformen Notation

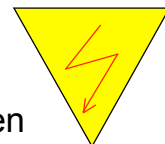
- spezifische Formelmanipulationsregeln
- durch logische Äquivalenzen semantisch gerechtfertigt

$$\frac{\neg T}{\perp} \quad \frac{\neg \perp}{T} \quad \frac{\neg \neg Z}{Z} \quad \frac{\alpha}{\alpha_1 \mid \alpha_2} \quad \frac{\beta}{\beta_1 \mid \beta_2}$$

Ch. Habel / C. Eschenbach: Logik & Semantik

4 – 40

- Diese Ersetzungsregeln haben die generelle Form von Zeichenersetzungsregeln über Zeichenketten.
- Der semantischer Gehalt der Regeln ergibt sich dadurch, dass – aufgrund des Ersetzungstheorems – Umformungen vermittels dieser Regeln zu äquivalenten Formeln führen.
[Zur Erinnerung: Zwei Formeln X und Y sind – logisch – äquivalent, wenn sie denselben Wahrheitswertverlauf haben.]
Dass dies gewährleistet ist, ist u.a. für die beiden letzten Umformungsregeln in Theorem 2.6.2 nachgewiesen.
- Derartige Ersetzungsregeln werden unter gewissen Zielen, d.h. für die Lösung spezifischer Aufgaben, verwendet.
 - Die Ersetzungsregeln für die Erstellung der disjunktiven Normalform unterscheiden sich im Hinblick auf die Umformungen von α - und β -Formeln.
 - Die Ersetzungsregeln für die Erstellung von Normalformen werden auch in Verfahren des automatischen Beweisens Anwendung.



Korrektheit der Konjunktiven Ersetzung

Theorem 2.8.5* („Konjunktive Ersetzung“)

Wenn S eine Konjunktion von Disjunktionen und S^* das Ergebnis der Anwendung einer Klauselmengen-Reduktionsregel auf S ist, so sind S und S^* äquivalent ($\models S \equiv S^*$).

$\frac{\neg \perp}{\perp}$	$\frac{\neg \top}{\top}$	$\frac{\neg \neg Z}{Z}$	$\frac{\alpha}{\alpha_1 \mid \alpha_2}$	$\frac{\beta}{\beta_1 \mid \beta_2}$
----------------------------	--------------------------	-------------------------	---	--------------------------------------

- Für den Beweis von 2.8.5 wird auf das Ersetzungstheorem 2.6.2 zurückgegriffen. Relevante Fälle sind dabei die Äquivalenzen, die es in der Aufgabe unten zu beweisen gilt.

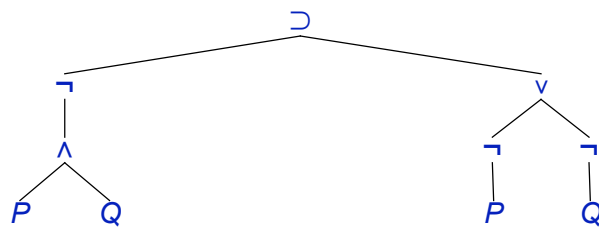
Aufgabe (ohne Nummer).

Zeige, dass die folgenden Formelpaare äquivalent sind.

- $[C_1, \dots, C_{k-1}, (U \vee V), C_{k+1}, \dots, C_n]$ und $[C_1, \dots, C_{k-1}, U, V, C_{k+1}, \dots, C_n]$
- $[C_1, \dots, C_{k-1}, (U \wedge V), C_{k+1}, \dots, C_n]$ und $[C_1, \dots, C_{k-1}, U, C_{k+1}, \dots, C_n] \wedge [C_1, \dots, C_{k-1}, V, C_{k+1}, \dots, C_n]$
- $\langle C_1, \dots, C_{k-1}, (U \wedge V), C_{k+1}, \dots, C_n \rangle$ und $\langle C_1, \dots, C_{k-1}, U, V, C_{k+1}, \dots, C_n \rangle$
- $\langle C_1, \dots, C_{k-1}, (U \vee V), C_{k+1}, \dots, C_n \rangle$ und $\langle C_1, \dots, C_{k-1}, U, C_{k+1}, \dots, C_n \rangle \vee \langle C_1, \dots, C_{k-1}, V, C_{k+1}, \dots, C_n \rangle$

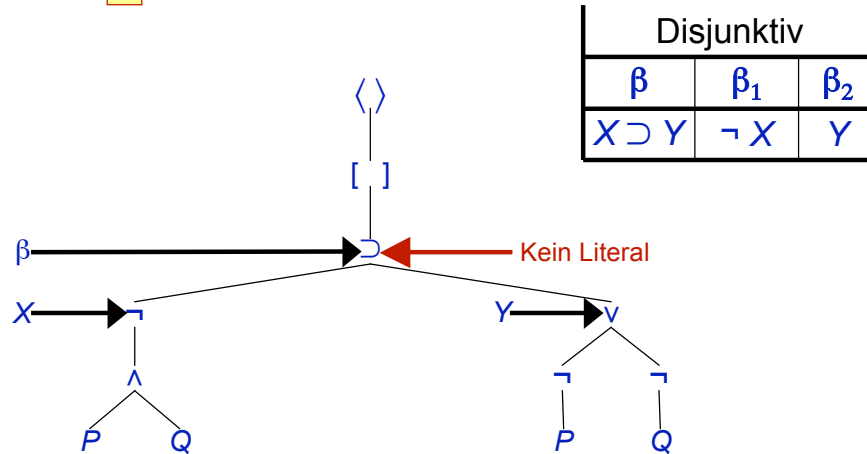
Strukturbaum einer Formel

$$\neg(P \wedge Q) \supset (\neg P \vee \neg Q)$$



Konjunktive Normalform: 1

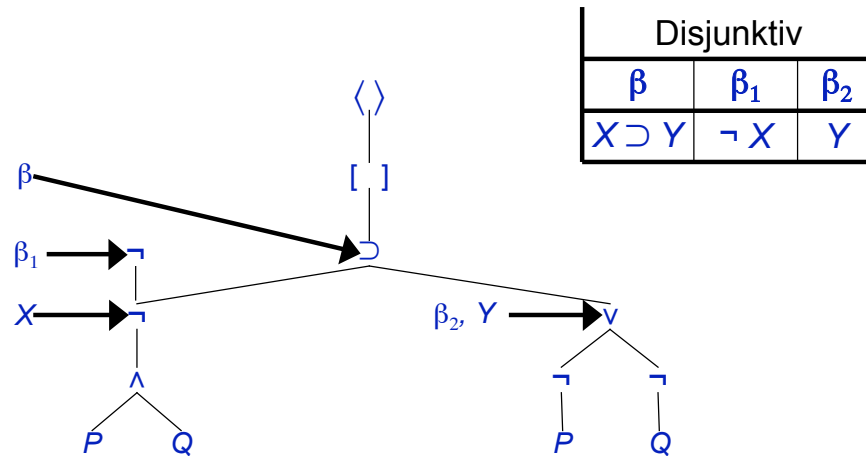
$$\langle [\neg(P \wedge Q) \supset (\neg P \vee \neg Q)] \rangle$$



- Falls N eine β -Formel, so ersetze N durch die Sequenz β_1, β_2 .

Konjunktive Normalform: 1

$\langle [\neg(P \wedge Q) \supset (\neg P \vee \neg Q)] \rangle$



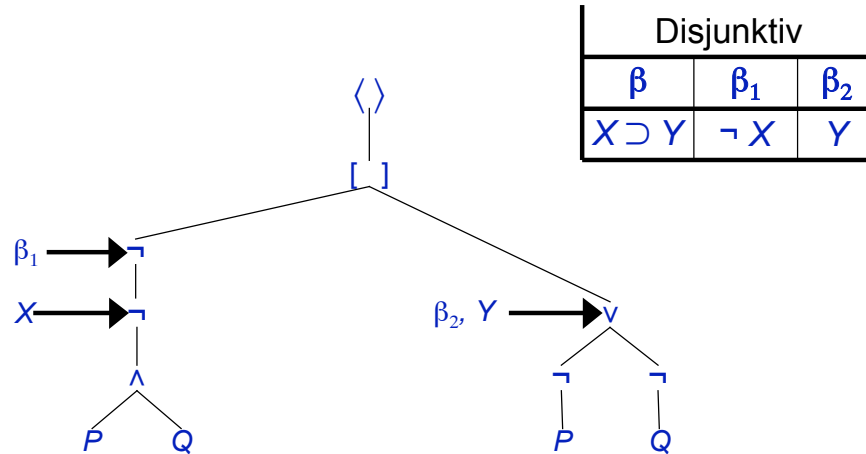
Ch. Habel / C. Eschenbach: Logik & Semantik

4 - 44

- Falls N eine β -Formel, so ersetze N durch die Sequenz β_1, β_2 .

Konjunktive Normalform: 2

$\langle [\neg \neg (P \wedge Q) , (\neg P \vee \neg Q)] \rangle$

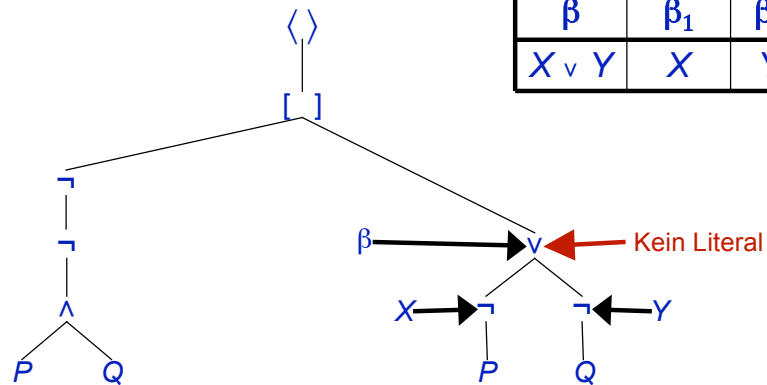


- Falls N eine β -Formel, so ersetze N durch die Sequenz β_1, β_2 .

Konjunktive Normalform: 2

$\langle [\neg \neg (P \wedge Q) , (\neg P \vee \neg Q)] \rangle$

Disjunktiv		
β	β_1	β_2
$X \vee Y$	X	Y

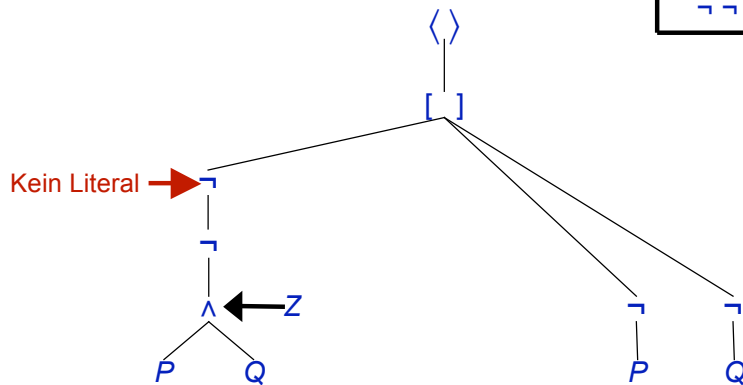


- Falls N eine β -Formel, so ersetze N durch die Sequenz β_1, β_2 .

Konjunktive Normalform: 3

$\langle [\neg \neg (P \wedge Q), \neg P, \neg Q] \rangle$

Doppel Negation	
$\neg \neg Z$	Z



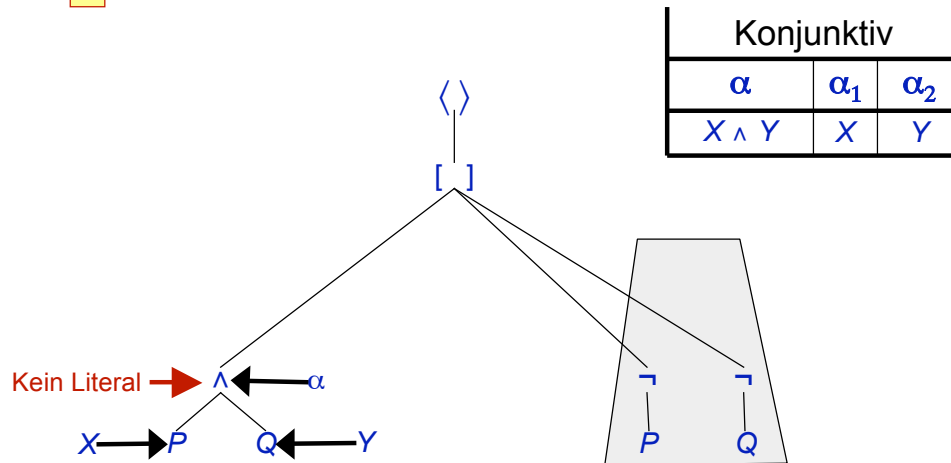
Ch. Habel / C. Eschenbach: Logik & Semantik

4 - 47

- Falls $N = \neg \neg Z$, so ersetze N durch Z .

Konjunktive Normalform: 4

$\langle [(P \wedge Q), \neg P, \neg Q] \rangle$



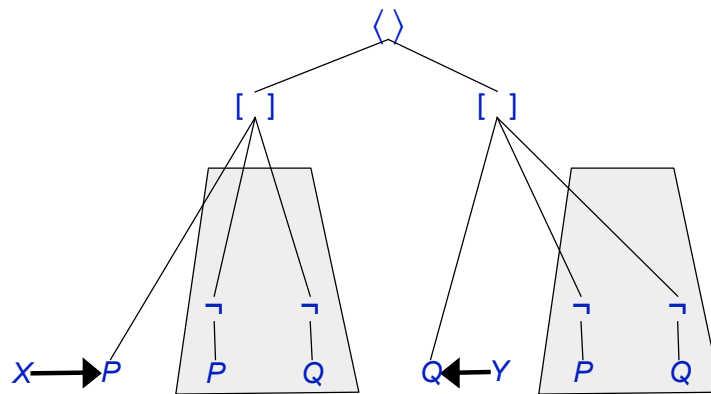
Ch. Habel / C. Eschenbach: Logik & Semantik

4 – 48

- Falls N eine α -Formel, so ersetze die Disjunktion D_i durch die **Sequenz** von Disjunktionen D_{i1} und D_{i2} , wobei D_{i1} aus D_i hervorgeht, indem N durch α_1 ersetzt wird und D_{i2} aus D_i hervorgeht, indem N durch α_2 ersetzt wird.

Konjunktive Normalform: 4

$\langle [P, \neg P, \neg Q], [Q, \neg P, \neg Q] \rangle$



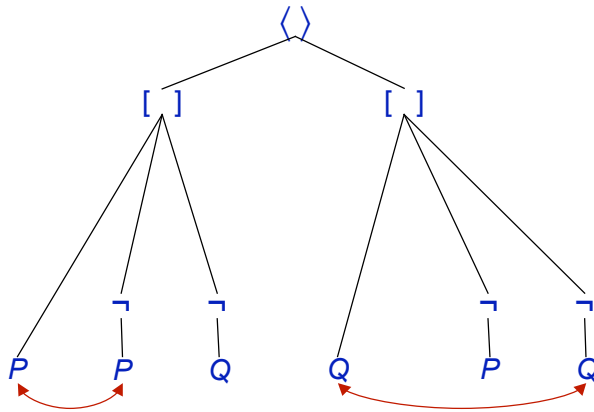
Ch. Habel / C. Eschenbach: Logik & Semantik

4 – 49

- Falls N eine α -Formel, so ersetze die Disjunktion D_i durch die Sequenz von Disjunktionen D_{i1} und D_{i2} , wobei D_{i1} aus D_i hervorgeht, indem N durch α_1 ersetzt wird und D_{i2} aus D_i hervorgeht, indem N durch α_2 ersetzt wird.

Tautologieprüfung

$\langle [P, \neg P, \neg Q], [Q, \neg P, \neg Q] \rangle$



Ch. Habel / C. Eschenbach: Logik & Semantik

4 – 50

- Prüfe, ob jede Klausel \top oder ein Paar $X, \neg X$ enthält.

Klauselform-Algorithmus

Eingabe: X eine Formel der Aussagenlogik

$S := \langle [X] \rangle$

• (** Z1 **)

WHILE S enthält eine Nicht-Klausel DO

• Wähle ein Element D aus S , das keine Klausel ist.

• Wähle ein Nicht-Literal N aus D

• Wende die entsprechende Klauselmengen-Reduktionsregel bzgl. N in D auf S an. Resultat S^*

• (** Z2 **)

• $S := S^*$

• (** Z3 **)

END

- Im folgenden wird (vergleichsweise informell) auf die Korrektheit und Termination des Verfahrens eingegangen. Im zweiten Teil der Vorlesung ('Semantik') werden die formalen Hilfsmittel eingeführt, die es erlauben, solche Beweise formell zu führen.

Korrektheit des Klauselformalalgorithmus

Korrektheit der Verfahrens

Wenn der Algorithmus für den Eingabewert X terminiert, dann ist das Resultat eine Klauselnormalform S , für die gilt: $\models S \equiv X$

- Zusicherung (Schleifeninvariante):
 $Z : S$ ist eine Konjunktion von Disjunktionen und $\models S \equiv X$
- (** Z1 **) Z ist wahr für den Anfangswert $S = \langle [X] \rangle$
- (** Z2 **) Z ist wahr für S^* , das Resultat einer Umformung, denn
 - $\models S^* \equiv S$ wegen Theorem 2.8.5
 - S^* ist eine Konjunktion von Disjunktionen
- (** Z3 **) Z ist wahr für S
- Z ist wahr bei Termination. Da bei Termination S bis auf Literale reduziert ist, liegt eine Klauselnormalform vor.

Die Frage der Termination des Verfahrens wird im Anschluss an den Korrektheitsbeweis behandelt.

Termination des Klauselform-Algorithmus

Eingabe: X eine Formel der Aussagenlogik

$S := \langle [X] \rangle$

WHILE S enthält eine Nicht-Klausel DO

- Wähle ein Element D aus S , das keine Klausel ist.
- Wähle ein Nicht-Literal N aus D
- Wende die entsprechende Klauselmengen-Reduktionsregel bzgl. N in D auf S an. Resultat S^*
- $S := S^*$

END

$\neg T$	$\neg \perp$	$\neg \neg Z$	α	β
\perp	\top	Z	$\alpha_1 \mid \alpha_2$	β_1 β_2

- In Bezug auf die Termination stellt sich das Problem, dass durch die Ersetzungsvorgänge die Formeln immer komplexer werden, in dem Sinne, dass es immer mehr Klauseln in der umfassenden (generalisierten) Disjunktion gibt und diese (generalisierten) Konjunktionen immer mehr Formeln enthalten.
- Um Termination zu zeigen muss man deshalb ein geeignetes Maß finden, mit dem man dann zeigen kann, dass die Formeln (im Sinne dieses Maßes) immer einfacher werden.
- Ein zweites Problem ist die fehlende Eindeutigkeit der Formulierung des Algorithmus.

Exkurs: *Termination* Nichtdeterministischer Algorithmen

Nichtdeterministischer Algorithmus

- Algorithmus mit „Wahlmöglichkeiten“, d.h. es gibt verschiedenen „Verlaufsmöglichkeiten“

Schwache Termination

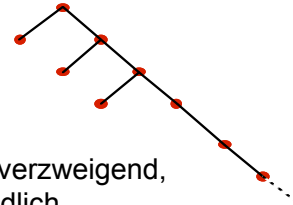
- eine Folge von Auswahlen (Verläufen) führt zur Termination, d.h. die richtige Auswahl garantiert das Terminieren.

Starke Termination

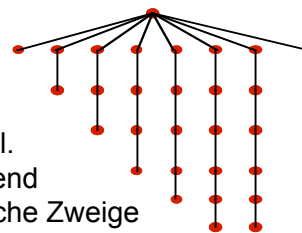
- alle Folgen von Auswahlen (Verläufen) führen zur Termination aber nicht unbedingt zum gleichen Ergebnis

Exkurs: Endliche & nicht-endliche Bäume – Königs Lemma

- Ein Baum ist *endlich*, falls er eine endliche Anzahl von Knoten enthält; anderenfalls ist er *nicht-endlich*.
- Ein Zweig ist *endlich*, falls er eine endliche Anzahl von Knoten enthält; anderenfalls ist er *nicht-endlich*.
- Ein Baum ist *endlich verzweigend*, falls jeder Knoten eine endliche Anzahl von direkten Nachfolgern (Töchter) besitzt.



Endlich verzweigend,
Nicht endlich



Nicht-endl.
verzweigend
Nur endliche Zweige

- In der folgenden Darstellungen interessieren nur Bäume mit ausgezeichneter Wurzel, bei denen die Kanten als von der Wurzel weg gerichtet aufgefasst werden können.
- Ein Zweig in so einem Baum ist eine (maximale) Folge von Knoten (v_0, v_1, \dots) , so dass v_0 die Wurzel des Baumes ist und aufeinanderfolgende Knoten v_i, v_{i+1} im Baum durch eine (gerichtete) Kante verbunden sind. Wenn ein Zweig ein Blatt des Baumes enthält, dann ist dies der letzte Knoten des Zweiges. In endlichen Bäumen können wir daher Zweige auch als die Pfade von der Wurzel zu den Blättern auffassen.

Königs Lemma

Theorem 2.7.2

Ein endlich-verzweigender nicht-endlicher Baum besitzt einen nicht-endlichen Ast.

Voraussetzungen

- T sei ein endlich-verzweigender nicht-endlicher Baum.
- Arbeitsdefinition
 - Ein Knoten N ist *gut*, falls er *unendlich viele Nachfolgeknoten* besitzt.
 - Damit gilt: $wurzel(T)$ ist *gut*, da T nicht-endlich.

Behauptung

- T hat einen nicht-endlichen Ast.

Königs Lemma (Beweis)

Beweis

Es sei N ein *guter* Knoten von T .

- Es gibt endlich viele Töchter C_1, \dots, C_n von N . [T ist endl. verzw.]
- Wenn C_i *nicht gut* ist, dann hat C_i endlich viele Nachfolger.
[Def. von *gut*.]
- Wenn alle C_i *nicht gut* wären, dann wäre N *nicht gut*.
- Also: es gibt eine *gute Tochter* zu N .

→ Jeder gute Knoten hat eine *gute Tochter*.

→ Die *Sequenz guter Töchter* bildet einen nicht-endlichen Zweig in T .

Königs Lemma: Smullyans Ballaufgabe

Gegeben

- ist eine Kiste und eine Menge von Bällen:
Jeder Ball ist mit einer natürlichen Zahl (≥ 0) gekennzeichnet.
- Verschiedene Bälle können gleich gekennzeichnet sein.
- Die Zahlen (auf den Bällen) können beliebig groß sein.

Startsituation: Wir beginnen mit einem Ball in der Kiste.

Die Spielregeln:

- Nimm **einen** Ball **aus** der Kiste. [markiert mit n]
- Lege **beliebig viele** Bälle **in** die Kiste, wobei alle Bälle mit einer (echt) kleineren Zahl als n markiert sein müssen.

Spielende: Das Spiel endet – stets –

- nach endlicher Zeit
- mit einer leeren Kiste.

Aufgabe 4-3

Beweisen Sie – unter Verwendung von Königs Lemma – dass Smullyans Ballaufgabe stets dazu führt, dass die Kiste gelehrt ist.

Termination des Klauselform-Algorithmus

Satz

Der Klauselform-Algorithmus terminiert stark.

Beweisskizze

Korrespondenz zwischen Klauselform-Erstellung und Smullyans Ballaufgabe.

- der Kiste entspricht die (generalisierte) Konjunktion
- den Bällen entsprechen die (generalisierten) Disjunktionen
- der Nummer auf den Bällen entspricht die Funktion Rang.
- Jede Reduktion führt zu „Bällen“ / Disjunktionen mit echt niedrigerer Nummer / Rang (aber es können durchaus mehr werden)

Aufgabe 4-4

- Arbeiten Sie die Beweisskizze für den Beweis der Termination des Klauselform-Algorithmus zu einem Beweis aus.
- Schlagen Sie eine Behandlung von Formeln mit sekundären Junktoren vor, so dass auch dafür die Termination beweisbar ist. (Wie ist ggf. die Definition von Rang anzupassen?)

Dual Clause Form – Disjunktive Normalform

Theorem 2.8.6* („Disjunktive Ersetzung“)

Wenn S eine Disjunktion von Konjunktionen und S^* das Ergebnis der Anwendung einer *Dualklauselmengen-Reduktionsregel* auf S ist, so ist $\models S \equiv S^*$.

$\frac{\neg T}{\perp}$	$\frac{\neg \perp}{T}$	$\frac{\neg\neg Z}{Z}$	$\frac{\beta}{\beta_1 \mid \beta_2}$	$\frac{\alpha}{\alpha_1 \mid \alpha_2}$
------------------------	------------------------	------------------------	--------------------------------------	---

- Aufgabe (ohne Nummer)
Machen Sie sich klar, was disjunktive Ersetzung und konjunktive Ersetzung voneinander unterscheidet.

Dualklauselform-Algorithmus



Eingabe: X eine Formel der Aussagenlogik

$S := [\langle X \rangle]$

WHILE S enthält eine Nicht-duale Klausel DO

- Wähle ein Element K aus S , das keine duale Klausel ist.
- Wähle ein Nicht-Literal N aus K .
- Wende die entsprechende Dualklauselmengen-Reduktionsregel bzgl. N in K auf S an. Resultat S^*
- $S := S^*$

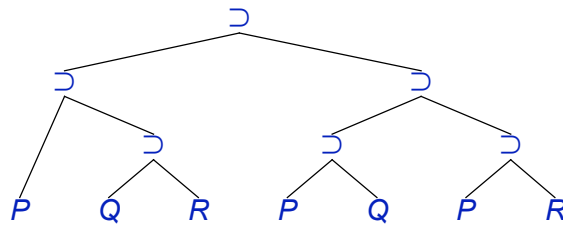
END

$\frac{\neg T}{\perp}$	$\frac{\neg \perp}{T}$	$\frac{\neg\neg Z}{Z}$	$\frac{\beta}{\beta_1 \mid \beta_2}$	$\frac{\alpha}{\alpha_1 \mid \alpha_2}$
------------------------	------------------------	------------------------	--------------------------------------	---

Noch ein Umformungsbeispiel

Strukturbaum einer Formel

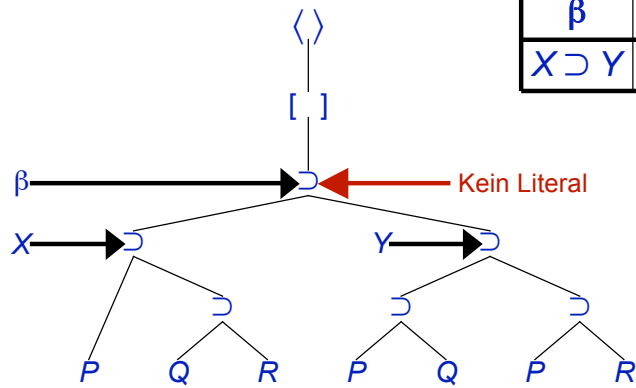
$(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R))$



Konjunktive Normalform: 1

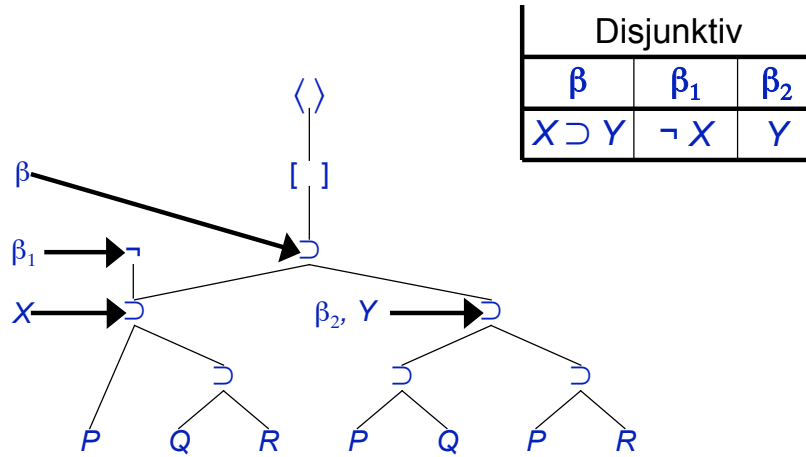
$$\langle [(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R))] \rangle$$

Disjunktiv		
β	β_1	β_2
$X \supset Y$	$\neg X$	Y



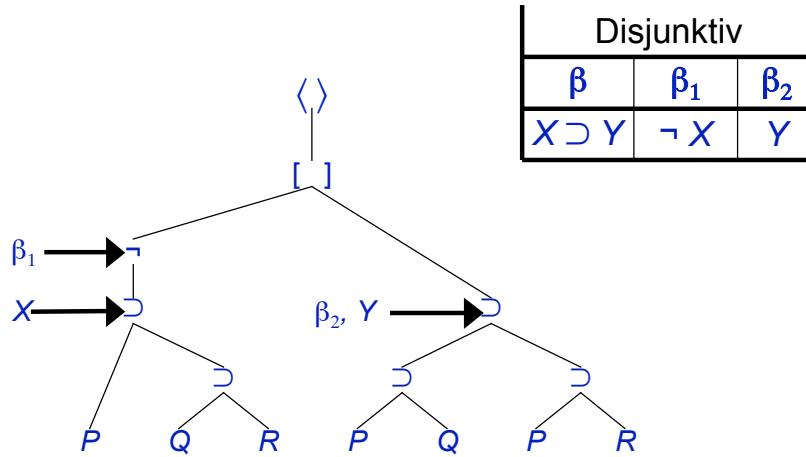
Konjunktive Normalform: 1

$\langle [(P \supset (Q \supset R)) \supset ((P \supset Q) \supset (P \supset R))] \rangle$



Konjunktive Normalform: 2

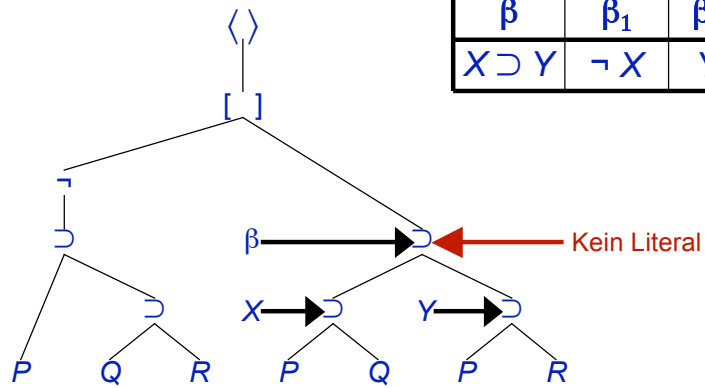
$\langle [\neg(P \supset (Q \supset R)), ((P \supset Q) \supset (P \supset R))] \rangle$



Konjunktive Normalform: 2

$\langle [\neg(P \supset (Q \supset R)), (P \supset Q) \supset (P \supset R)] \rangle$

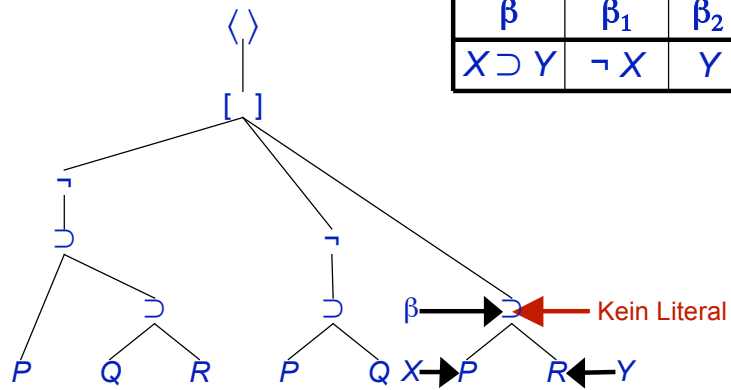
Disjunktiv		
β	β_1	β_2
$X \supset Y$	$\neg X$	Y



Konjunktive Normalform: 3

$\langle [\neg(P \supset (Q \supset R)), \neg(P \supset Q), (P \supset R)] \rangle$

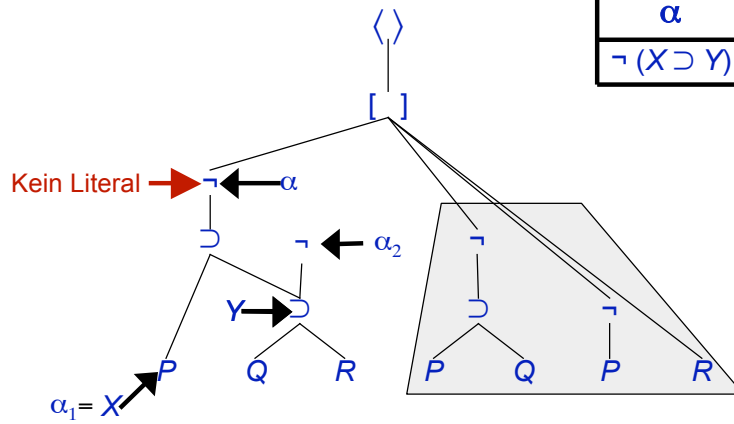
Disjunktiv		
β	β_1	β_2
$X \supset Y$	$\neg X$	Y



Konjunktive Normalform: 4

$\langle [\neg(P \supset (Q \supset R)), \neg(P \supset Q), \neg P, R] \rangle$

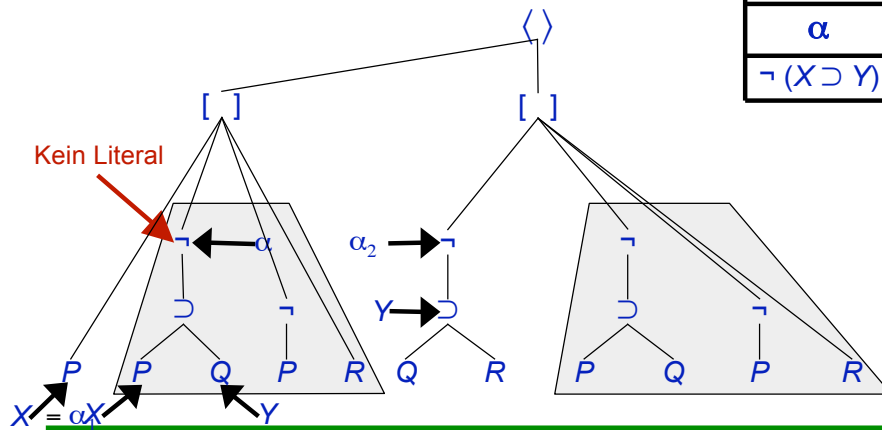
Konjunktiv		
α	α_1	α_2
$\neg(X \supset Y)$	X	$\neg Y$



Konjunktive Normalform: 5

$\langle [P, \neg(P \supset Q), \neg P, R], [\neg(Q \supset R), \neg(P \supset Q), \neg P, R] \rangle$

Konjunktiv		
α	α_1	α_2
$\neg(X \supset Y)$	X	$\neg Y$

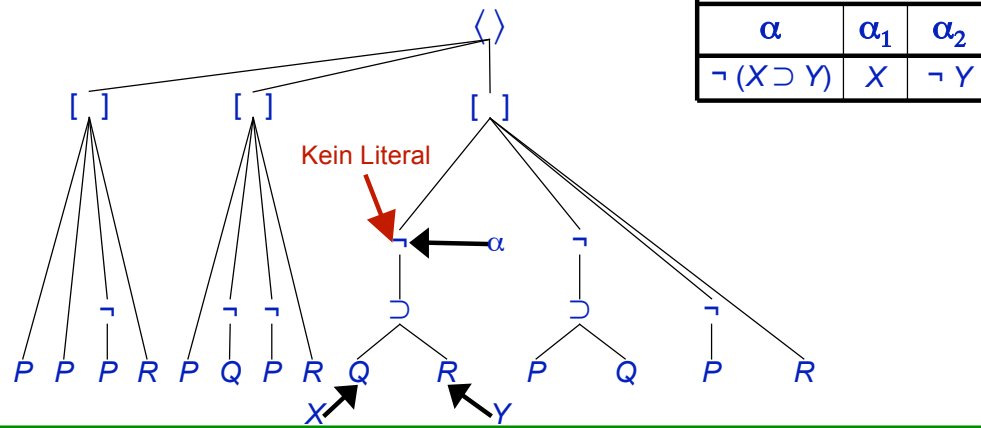


Ch. Habel / C. Eschenbach: Logik & Semantik

4 - 71

Konjunktive Normalform: 6

$\langle [P, P, \neg P, R], [P, \neg Q, \neg P, R], [\neg(Q \supset R), \neg(P \supset Q), \neg P, R] \rangle$

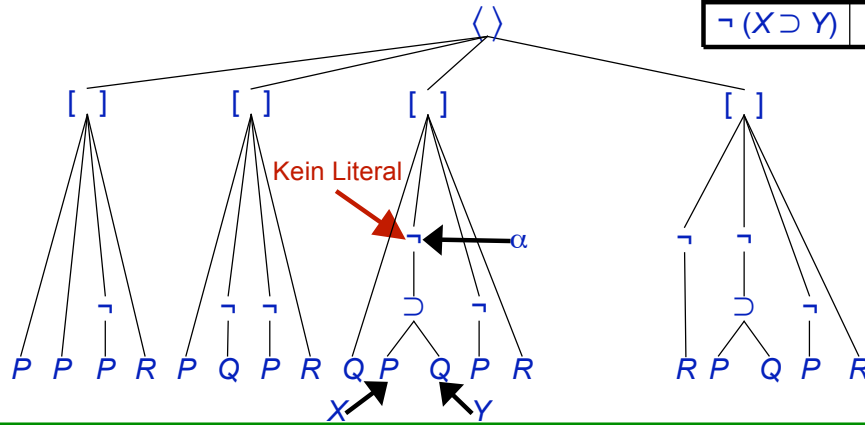


Konjunktiv		
α	α_1	α_2
$\neg(X \supset Y)$	X	$\neg Y$

Konjunktive Normalform: 7

$\langle [P, P, \neg P, R], [P, \neg Q, \neg P, R],$
 $[Q, \neg(P \supset Q), \neg P, R], [\neg R, \neg(P \supset Q), \neg P, R] \rangle$

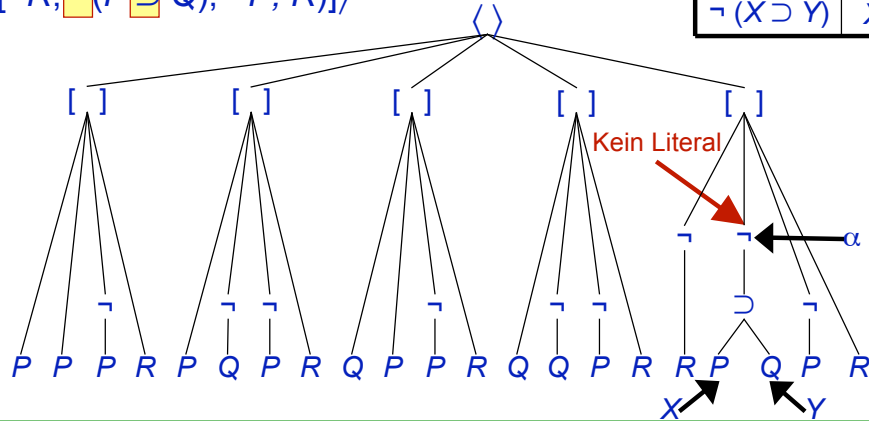
Konjunktiv		
α	α_1	α_2
$\neg(X \supset Y)$	X	$\neg Y$



Konjunktive Normalform: 8

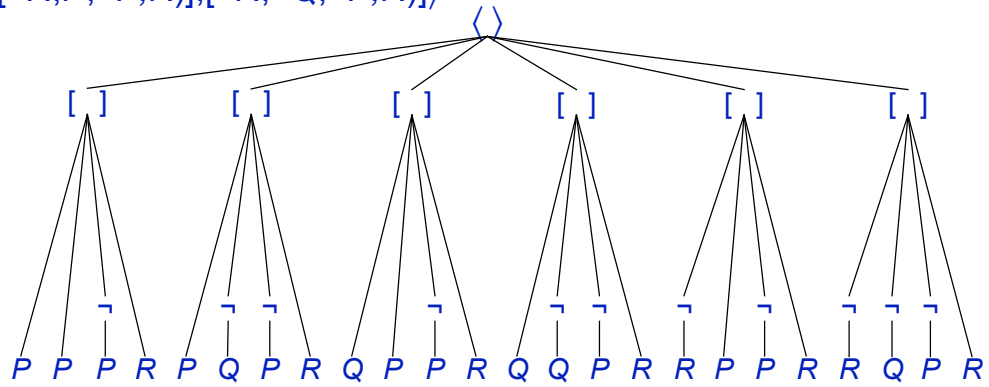
$\langle [P, P, \neg P, R], [P, \neg Q, \neg P, R],$
 $[Q, P, \neg P, R], [Q, \neg Q, \neg P, R],$
 $[\neg R, \neg(P \supset Q), \neg P, R] \rangle$

Konjunktiv		
α	α_1	α_2
$\neg(X \supset Y)$	X	$\neg Y$



Konjunktive Normalform: 9

$\langle [P, P, \neg P, R], [P, \neg Q, \neg P, R],$
 $[Q, P, \neg P, R], [Q, \neg Q, \neg P, R],$
 $[\neg R, P, \neg P, R], [\neg R, \neg Q, \neg P, R] \rangle$



Tautologieprüfung

$\langle [P, P, \neg P, R], [P, \neg Q, \neg P, R],$
 $[Q, P, \neg P, R], [Q, \neg Q, \neg P, R],$
 $[\neg R, P, \neg P, R], [\neg R, \neg Q, \neg P, R] \rangle$

