Ad (3):
$C_n = \{\{L_1, \ldots, L_n\} \mid L_i \in \{p_i, \neg p_i\}, 1 \leq i \leq n\}$
$\Phi_n = C_n \cup \{\{p_i, \neg p_i\} \mid 1 \leq i \leq n\}$

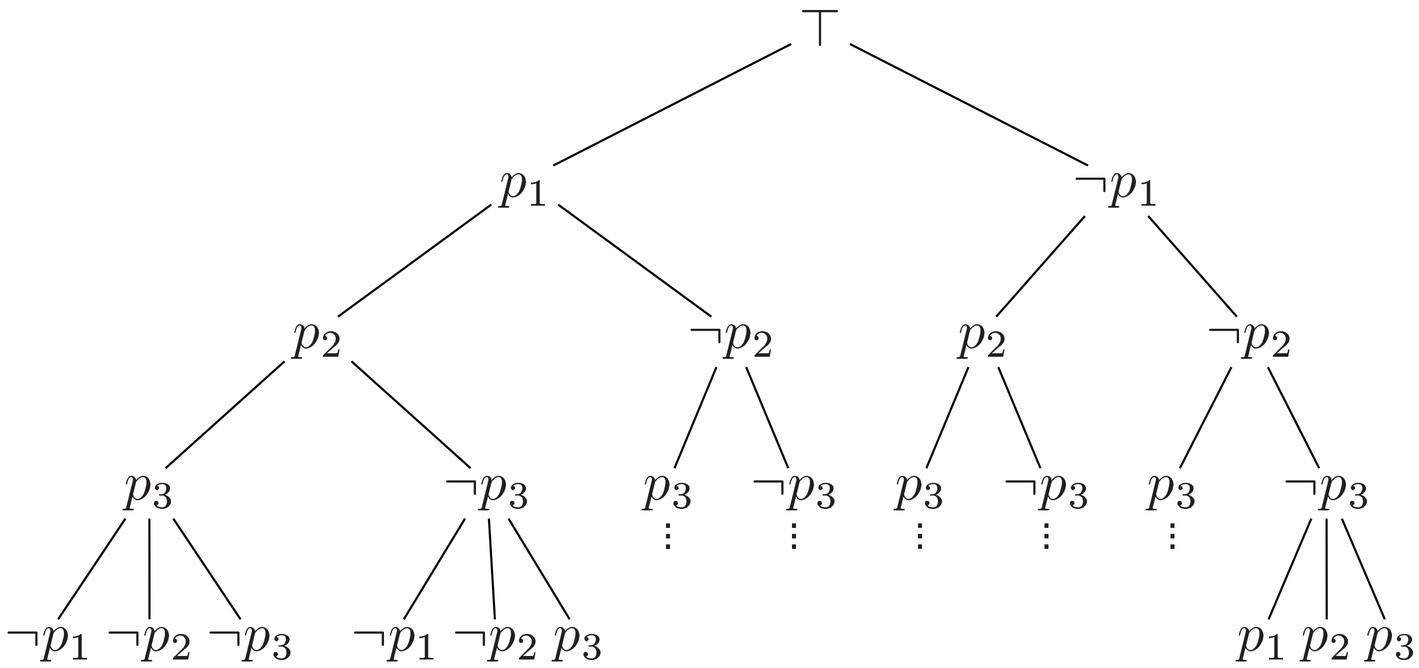
$$\begin{array}{llll}
\Phi_3 = \{ & \{p, q, r\}, & \{\neg p, q, r\}, & \{p, \neg p\}, \\
& \{p, q, \neg r\}, & \{\neg p, q, \neg r\}, & \{q, \neg q\} \\
& \{p, \neg q, r\}, & \{\neg p, \neg q, r\}, & \{r, \neg r\} \\
& \{p, \neg q, \neg r\}, & \{\neg p, \neg q, \neg r\} & \quad \}
\end{array}$$

$|\Phi_n| = n2^n + 2n$ (Anzahl Literale)

Kürzester Klauseltableaubeweis T_n für Φ_n
hat $n2^n + 2^{n+1} - 1$ Knoten,
also $|T_n| \in \mathcal{O}(|\Phi_n|)$.

Kürzester Konnektionstableaubeweis T'_n für Φ_n
hat $\geq 2(n-1)!$ Knoten,
also nicht polynomiell in $|\Phi_n|$.

Kürzester Klauseltableaubeweis für Φ_n:



Andere Kalküle als Varianten von Klauseltableaus

Konnektionstableaus spezifizieren keine Auswahlregel `select_branch`.

Verschiedene Festlegungen ergeben bekannte Kalküle in Tableauformulierung.

Die Konnektionsmethode Bibel (1981, 1987)

Wir präsentieren die AL Variante für Formeln in KNF. Die allgemeine Form der Konnektionsmethode ist für beliebige Formeln erster Stufe definiert.

Eine Formel in KNF der Form

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

wird in *Matrixschreibweise* notiert:

$$\begin{bmatrix} L_{1,1} & L_{2,1} & \cdots & L_{n,1} \\ \vdots & \vdots & & \vdots \\ L_{1,m_1} & L_{2,m_2} & \cdots & L_{n,m_n} \end{bmatrix}$$

Definition. Ein **Pfad** durch eine Matrix M ist eine Menge von Literalen der Gestalt

$$\{L_{1,i_1}, \ldots, L_{n,i_n}\}$$

wobei $1 \leq i_j \leq m_j$ für alle $1 \leq j \leq n$.

Ein **Pfad** heißt **komplementär**, falls er ein komplementäres Paar $\{p, \neg p\}$ als Teilpfad enthält.

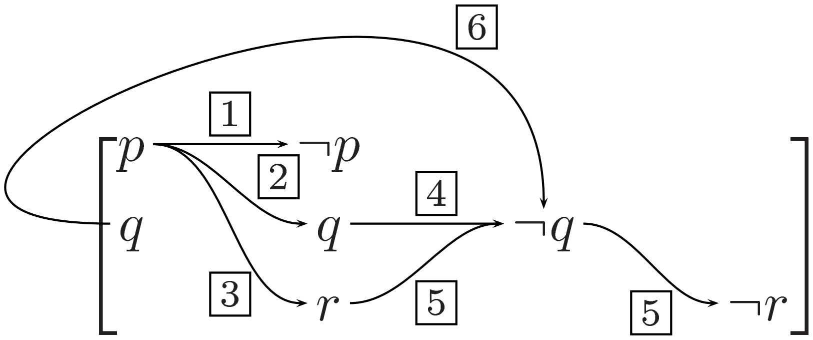
Eine **Matrix** heißt **komplementär**, falls alle Pfade durch sie komplementär sind.

Insbesondere ist jede Matrix für eine Klauselmenge die $\{\}$ enthält, komplementär. ■

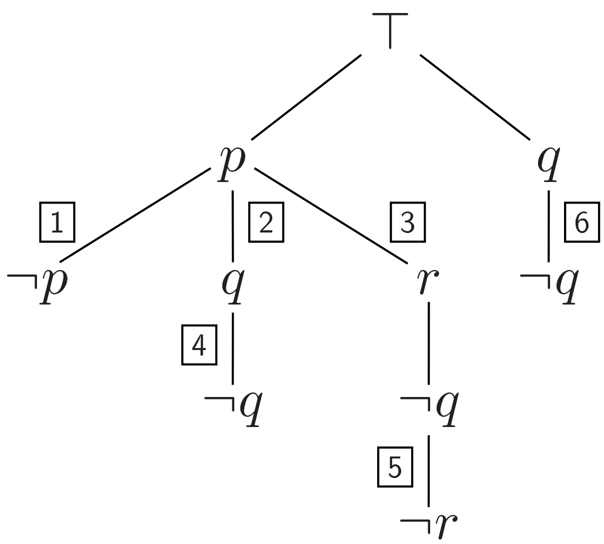
Satz. Die Matrix einer AL Klauselmenge S ist komplementär gdw S unerfüllbar ist. ■

Die Konnektionsmethode besteht in einer systematischen Prozedur für die Feststellung der Komplementarität einer Matrix.

Beispiel: $S = \{\{p, q\}, \{\neg p, q, r\}, \{\neg q\}, \{\neg r\}\}$.



(Teil-)Pfade in obiger Matrix entsprechen (Teil-)Zweigen in einem Klauseltableau für S:



Modellelimination
Loveland (1969)

Modellelimination ist eine Familie von Beweisprozeduren.

Die Ziele beim Entwurf waren:

- kleiner Suchraum
- geringer Platzverbrauch
- mögliche Interpretation als systematisches Ausschließen von Modellen

Zunächst leicht vereinfachte Version für AL:

Definition. Eine **Kette** ist ein endliche Folge $\langle L_1, \ldots, L_n \rangle$ von Literalen.

Jedes Literal einer Kette hat **Typ** A oder **Typ** B (Typ A Literale durch Umrandung gekennzeichnet).

Zu einer Klausel C gehört die Kette, die aus den Literalen in C als Typ-B-Literalen besteht. ■

Definition. Die **zulässigen Ketten** über einer endlichen Klauselmenge S sind wie folgt induktiv definiert:

Init: Jede zu $C \in S$ gehörige Kette ist zulässig.

Extend: Ist $\langle L_1, \ldots, L_n \rangle$ eine zulässige Kette mit

- L_n vom Typ B,
- $\langle M_1, \ldots, M_m \rangle$ die zu $C \in S$ gehörige Kette,
- L_n komplementär zu einem M_j $(1 \leq j \leq m)$,

dann auch

$$\langle L_1, \ldots, L_{n-1}, \boxed{L_n}, M_1, \ldots, M_{j-1}, M_{j+1}, \ldots, M_m \rangle$$

Reduce: Ist $\langle L_1, \ldots, L_n \rangle$ ein zulässige Kette mit

- $\boxed{L_i}$ vom Typ A,
- $L_j, \ldots, L_n$ vom Typ B für ein $j > i$,
- L_i komplementär zu L_j,

dann auch

$$\langle L_1, \ldots, L_{j-1}, L_{j+1}, \ldots, L_n \rangle$$

Contract: Ist $\langle L_1, \ldots, L_n \rangle$ eine zulässige Kette mit

- $\boxed{L_j}, \ldots, \boxed{L_n}$ vom Typ A,

dann auch

$$\langle L_1, \ldots, L_{j-1} \rangle$$

■

Definition. Ein **Beweis mit Modellelimination** für S ist Folge von zulässigen Ketten über S, die $\langle\rangle$ enthält. ■

Satz. Eine AL Klauselmenge S ist unerfüllbar gdw es einen Beweis mit Modellelimination für S gibt. ■

Beispiel:

$$S = \{\{p\}, \{r, \neg p, q\}, \{s, \neg q\}, \{\neg q, \neg s\}, \{\neg q, \neg r\}, \{\neg r, q\}\}$$

p Init

$\boxed{p}$ r q Ext

$\boxed{p}$ r $\boxed{q}$ s Ext

$\boxed{p}$ r $\boxed{q}$ $\boxed{s}$ $\neg q$ Ext

$\boxed{p}$ r $\boxed{q}$ $\boxed{s}$ Red

$\boxed{p}$ r Contr

$\boxed{p}$ $\boxed{r}$ q Ext

$\boxed{p}$ $\boxed{r}$ $\boxed{q}$ $\neg r$ Ext

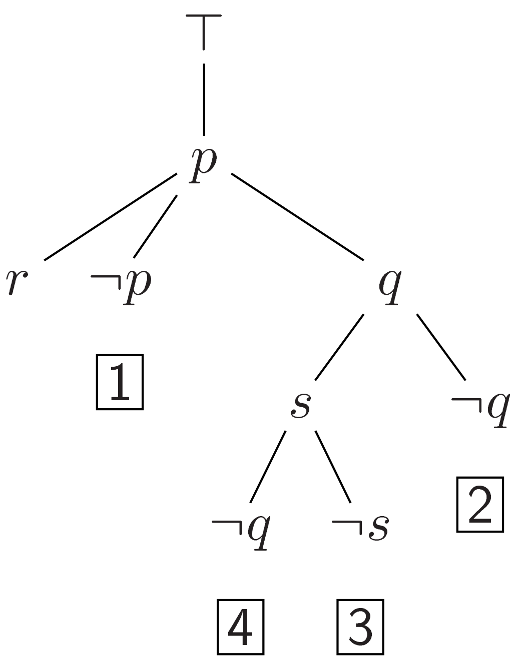
$\boxed{p}$ $\boxed{r}$ $\boxed{q}$ Red

Contr

Beispiel:

$$S = \{\{p\}, \{r, \neg p, q\}, \{s, \neg q\},$$
$$\{\neg q, \neg s\}, \{\neg q, \neg r\}, \{\neg r, q\}\}$$

Das partielle Konnektionstableau



entspricht der Folge von Ketten:

	p	Init				
1	$\boxed{p}$	r	q	Ext		
2	$\boxed{p}$	r	$\boxed{q}$	s	Ext	
3	$\boxed{p}$	r	$\boxed{q}$	$\boxed{s}$	$\neg q$	Ext
4	$\boxed{p}$	r	$\boxed{q}$	$\boxed{s}$	Red	
	$\boxed{p}$	r	Contr			

Zusammenfassung: ME & Tableaus

- B-Literale sind Blätter noch offener Äste.
  $\boxed{A}$-Literale sind Vorgängerliterale auf aktuellem Ast.

- Init-Regel der ME entspricht der Initialisierungsregel bei Konnektionstableaus.

- Extensionsregel entspricht Konnektionstableauerweiterung des am weitesten rechts stehenden Zweigs.

- Reduktionsregel entspricht Abschluß eines Zweigs mit Reduktion.

- Kontraktionsregel kann nur angewandt werden, falls Ende der Kette A-Literal, d.h., falls es keine offenen Zweige unter diesem Literal gibt. Entspricht dem Löschen geschlossener Zweige.

Also:

- Modellelimination entspricht Konnektionstableaus mit right_first Zweigauswahlregel.

- Vollständigkeit der Modellelimination folgt aus der Vollständigkeit der Konnektionstableaus.

Modellgenerierung (MG)
Manthey & Bry (1988)
Fujita & Hasegawa (1991)

Klauseln als Regeln geschrieben (wie bei Prolog):

$$p_1, \dots, p_n \to q_1, \dots, q_m$$

Definition. Eine PL Klauselmenge S heißt **bereichsbeschränkt**, falls für alle

$$C = p_1, \dots, p_n \to q_1, \dots, q_m \in S$$

gilt:

$$\mathsf{Var}(\{p_1, \dots, p_n\}) \supseteq \mathsf{Var}(\{q_1, \dots, q_m\}). \qquad \blacksquare$$

MG ist nur für bereichsbeschränkte Klauselmengen definiert.

Die Regeln definieren eine Menge $\mathcal{M}$ von Modellkandidaten $M \subseteq \mathsf{At}^0_\Sigma$:

Init $\mathcal{M} = \{\emptyset\}$.

Modellerweiterung Ist $C \to d_1, \ldots, d_k$ eine Klausel in S ($C = \emptyset$ möglich) und gibt es ein σ, so daß

- $C\sigma \subseteq M \in \mathcal{M}$,
- $D\sigma \cap M = \emptyset$,

dann setze

$$\mathcal{M} := (\mathcal{M} - \{M\}) \cup \{M \cup \{d_1\sigma\}, \ldots, M \cup \{d_k\sigma\}\}$$

Modell verwerfen Ist $C \to$ eine Klausel in S ($C \neq \emptyset$) und gibt es ein σ, so daß

- $C\sigma \subseteq M \in \mathcal{M}$,

dann setze

$$\mathcal{M} := \mathcal{M} - \{M\}.$$

Satz. Eine bereichsbeschränkte PL Klauselmenge S ist genau dann unerfüllbar, wenn mit ihr die leere Menge $\mathcal{M} = \emptyset$ von Modellkandidaten mit S hergeleitet werden kann.

MG ist beweiskonfluent. ■

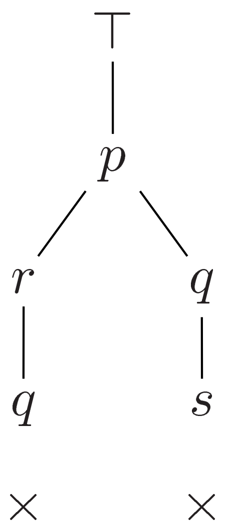
Beweis: Modifiziere Beweis für geordnete Tableaus. ■

Modellkandidaten entsprechen Ästen in Klauseltableaus.

Beispiel:

Klauseln in S:

$$
\begin{array}{rl}
\to & p \\
p \to & r, q \\
q \to & s \\
q, s \to & \\
q, r \to & \\
r \to & q
\end{array}
$$



MG ist Variante der regulären schwachen Konnektionstableaus mit Neustart, wobei:

1. Neustartklauseln sind genau die positiven Klauseln.
2. Alle negativen Literale der erweiternden Klausel müssen Link mit dem erweitertem Ast haben.

Anstatt MG auch: **positive Hypertableaus**

Bereichsbeschränktheit impliziert, daß alle Modellkandidaten (Äste) aus Grundliteralen bestehen.

Daher: Substitutionen müssen nicht auf $\mathcal{M}$ angewandt werden.

`select_pair` und `select_mode` sind deterministisch (Modell verwerfen bevorzugt).

Falls `select_clause` (Modellerweiterung) fair ist, ist Implementierung ohne Backtracking möglich

`select_branch` meist `left_first` oder *parallel*.

Verstärkungen von Klauseltableaus

Gegensatz:

Verfeinerungen $\rightarrow$ Einschränkung des Suchraums

Verstärkungen $\rightarrow$ Kürzere Beweise möglich

Motivation für Wunsch nach Verstärkung:
Klauselmengen Φ_n (Folie 33)

- Länge des kürzesten Tableaubeweises für Φ_n ist in $\mathcal{O}(|\Phi_n|)$.
- Länge des kürzesten Tableaubeweises für $C_n \subset \Phi_n$ (also ohne die tautologischen Klauseln) ist nicht einmal in $\mathcal{O}(|C_n|^k)$.
- Wahrheitstafeln für C_n haben Aufwand in $\mathcal{O}(|C_n|)$.

($|C_n| = n \cdot$Anzahl AL Interpretationen von C_n)

Ursache des Problems:
Klauseltableaus können AL Interpretationen nicht effizient aufzählen.

Mögliche Abhilfe:
Regel (atomarer Schnitt), die Tautologien einführt.

Tautologien $\{\{p_i, \neg p_i\} \mid 1 \leq i \leq n\}$
können durch atomare Schnitte erzeugt werden.

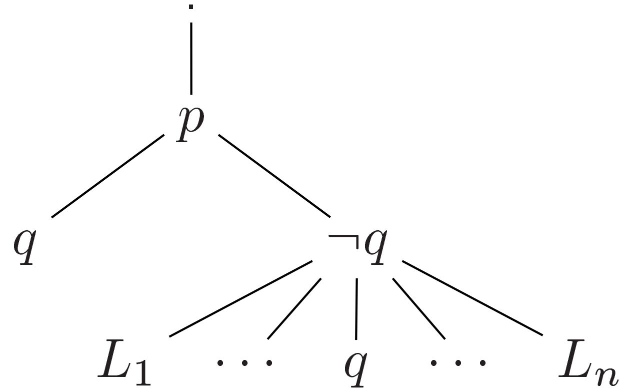
$$\frac{}{p \mid \neg p} \qquad \text{(Atomarer Schnitt)}$$

$$p \in \mathsf{At}_\Sigma$$

Korrektheit der Schnittregel ist offensichtlich.

Nachteil: Anwendung der Schnittregel unbeschränkt.

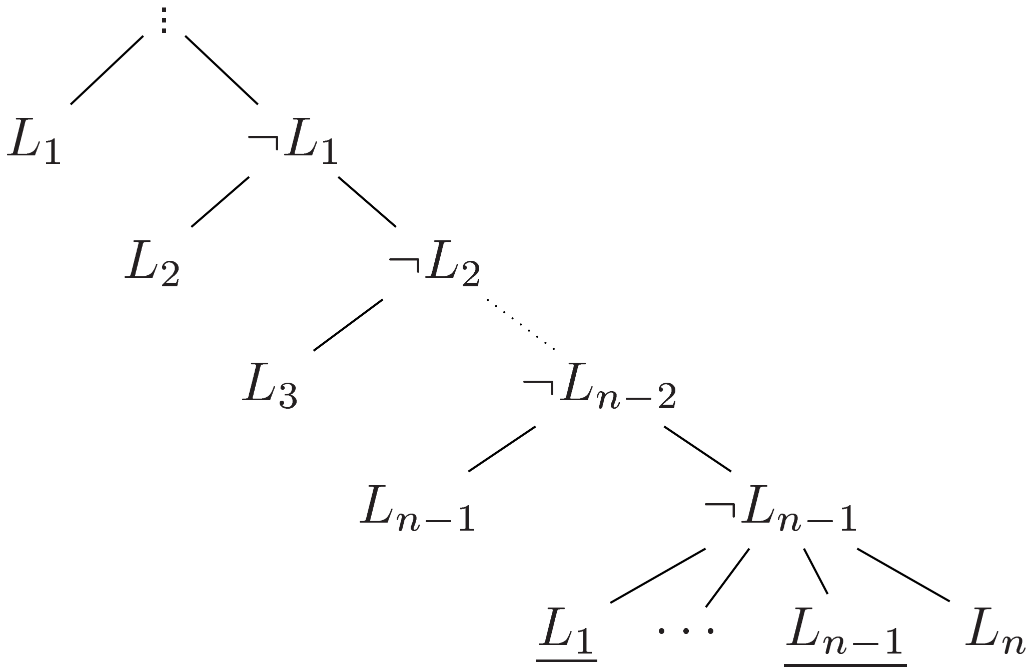
Sinnvolle Einschränkung: Schnittregel nur anwenden, wenn p oder $\neg p$ komplementäres Paar mit einem Literal aus der Klausel bilden, mit der direkt anschließend erweitert wird:



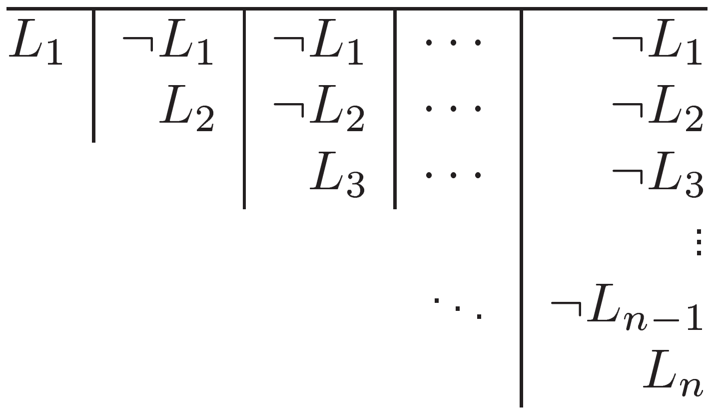
Zahl neuer Zweige wie ohne Schnitt.

Jedenfalls nicht schwächer als Klauseltableau.

Vor Tableauerweiterung mit $C = \{L_1, \ldots, L_n\} \in S$ wende Schnittregel $n-1$-mal auf $L_1, \ldots, L_{n-1}$ an:



Lösche geschlossenen Zweige:



$$\{L_1, \ldots, L_n\} \in S$$

Anstatt Schnittregel:
Klauseltableaus mit (lokalen) Lemmata.