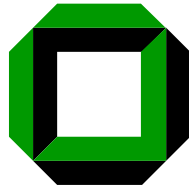


Formale Systeme

Prof. Dr. Bernhard Beckert

Fakultät für Informatik  
Universität Karlsruhe (TH)



Winter 2008/2009



- Widerlegungskalkül
- Voraussetzung: Alle Formeln sind in konjunktiver Normalform.
- Es gibt eine einzige Regel, die Resolutionsregel, eine Modifikation des Modus ponens.
- Es sind keine logischen Axiome mehr notwendig; insbesondere entfällt die Suche nach („langen“) passenden Instantiierungen der Axiomenschemata  $Ax1$ ,  $Ax2$ ,  $Ax3$ .



Syntax

Eine *Klausel* ist eine endliche Menge von Literalen.  $\square$  ist die *leere Klausel*.

$$\{P_1, P_2\} \quad \{P_1, \neg P_2\}$$

Eine Klausel lesen wir als die Disjunktion der ihr angehörenden Literale.

$$P_1 \vee P_2 \quad P_1 \vee \neg P_2$$



Syntax

Die grundlegende Datenstruktur, auf der wir operieren, sind die *Mengen von Klauseln*,

$$\{\{P_1, P_2\}, \{P_1, \neg P_2\}, \{\neg P_1, P_2\}, \{\neg P_1, \neg P_2\}\}$$

gelesen als Konjunktion der Klauseln.

$$(P_1 \vee P_2) \wedge (P_1 \vee \neg P_2) \wedge (\neg P_1 \vee P_2) \wedge (\neg P_1 \vee \neg P_2)$$

Bis auf die Reihenfolge der Disjunktions- bzw. Konjunktionsglieder sind also in umkehrbar eindeutiger Weise die Formeln in konjunktiver Normalform als Klauselmengen wiedergegeben.



## Semantik

Ist  $I : \Sigma \rightarrow \{W, F\}$  eine Interpretation der Atome, so hat man als zugehörige Auswertung

$val_I : \text{Mengen von Klauseln} \rightarrow \{W, F\} :$

$$val_I(M) = \begin{cases} W & \text{falls für alle } C \in M \text{ gilt:} \\ & val_I(\{C\}) = W \\ F & \text{sonst} \end{cases}$$

$$val_I(\{C\}) = \begin{cases} W & \text{falls ein } L \in C \text{ existiert mit} \\ & I(L) = W \\ F & \text{sonst} \end{cases}$$

für Klauseln  $C$ ,

Es folgt also

$$val_I(\{\square\}) = F,$$

$$val_I(\emptyset) = W$$



für die leere Klauselmeng  $\emptyset$ .

## Der Kalkül R0

### Definition

Die *aussagenlogische Resolutionsregel* ist die Regel

$$\frac{C_1 \cup \{P\}, C_2 \cup \{\neg P\}}{C_1 \cup C_2}$$

mit einer AL-Variablen  $P$  und Klauseln  $C_1, C_2$   
 $C_1 \cup C_2$  heisst *Resolvente* von  $C_1 \cup \{P\}, C_2 \cup \{\neg P\}$ .

Der Resolutionskalkül enthält die Resolutionsregel als einzige Regel.



## Beispiel

Gegeben sei die Klauselmeng

$$M = \{\{P_1, P_2\}, \{P_1, \neg P_2\}, \{\neg P_1, P_2\}, \{\neg P_1, \neg P_2\}\}$$

$$\frac{\{P_1, P_2\}, \{P_1, \neg P_2\}}{\{P_1\}}$$

$$\frac{\{\neg P_1, P_2\}, \{\neg P_1, \neg P_2\}}{\{\neg P_1\}}$$

$$\frac{\{P_1\}, \{\neg P_1\}}{\square}$$

Insgesamt:

$$M \vdash_{Res} \square$$



## Ein zweites Beispiel

Es soll gezeigt werden, dass

$$(A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C))$$

eine Tautologie ist.

Dazu werden wir zeigen, dass die Negation dieser Formel

$$\neg((A \rightarrow B) \rightarrow ((B \rightarrow C) \rightarrow (A \rightarrow C)))$$

nicht erfüllbar ist.

Klauselnormalform:

$$M = \{\{\neg A, B\}, \{\neg B, C\}, \{A\}, \{\neg C\}\}$$



## Ein zweites Beispiel Fortsetzung

$$M = \{\{\neg A, B\}, \{\neg B, C\}, \{A\}, \{\neg C\}\}$$

Ableitung der leeren Klausel aus  $M$ :

- (1)  $\square$        $\{\neg A, B\}$
- (2)  $\square$        $\{\neg B, C\}$
- (3)  $\square$        $\{A\}$
- (4)  $\square$        $\{\neg C\}$
- (5)  $[1, 3]$   $\{B\}$
- (6)  $[2, 5]$   $\{C\}$
- (7)  $[4, 6]$   $\square$



## Notwendigkeit der Mengenschreibweise

Die Menge von Formeln

$$E = \{P_1 \vee \neg P_2, \neg P_1 \vee P_2, \neg P_1 \vee \neg P_2, P_1 \vee P_2\}$$

ist sicherlich nicht erfüllbar.

Es gibt die folgenden Resolutionsmöglichkeiten:

$$\frac{\neg P_1 \vee \neg P_2, P_1 \vee \neg P_2}{\neg P_2 \vee \neg P_2} \quad \frac{\neg P_1 \vee \neg P_2, \neg P_1 \vee P_2}{\neg P_1 \vee \neg P_1}$$

$$\frac{\neg P_1 \vee \neg P_1, P_1 \vee \neg P_2}{\neg P_1 \vee \neg P_2} \quad \frac{\neg P_2 \vee \neg P_2, \neg P_1 \vee P_2}{\neg P_1 \vee \neg P_2}$$

Auf diese Weise ist  $\square$  nicht herleitbar.



## Korrektheit und Vollständigkeit

### Theorem

Für eine Menge  $M$  von Klauseln gilt

$$M \text{ unerfüllbar} \Leftrightarrow M \vdash_{R0} \square.$$

*Beweis: Korrektheit:* Übung

*Beweis: Vollständigkeit:*

- Aus  $M \not\vdash_{R0} \square$  beweisen wir die Erfüllbarkeit von  $M$ .
- Wir fixieren eine feste Reihenfolge der Atome,  $P_0, \dots, P_n, \dots$
- Mit  $M_0$  bezeichnen wir die Menge aller Klauseln  $C$ , für die  $M \vdash_{R0} C$  gilt. Es gilt also  $M \subseteq M_0$  und  $\square \notin M_0$ .
- Wir werden eine Interpretation  $I$  angeben, so dass für alle  $C \in M_0$   $val_I(C) = W$  gilt. Insbesondere ist dann  $M$  als erfüllbar nachgewiesen.
- Wir definieren  $I$  induktiv. **Rest des Beweises: Tafel/Skriptum**



## Beispiel einer Beweisaufgabe

$$\begin{array}{l} (p1h1 \rightarrow \neg p2h1) \\ \wedge (p1h1 \rightarrow \neg p3h1) \\ \wedge (p2h1 \rightarrow \neg p3h1) \\ \wedge (p1h2 \rightarrow \neg p2h2) \\ \wedge (p1h2 \rightarrow \neg p3h2) \\ \wedge (p2h2 \rightarrow \neg p3h2) \end{array} \vdash \begin{array}{l} (\neg p1h1 \wedge \neg p1h2) \\ \vee (\neg p2h1 \wedge \neg p2h2) \\ \vee (\neg p3h1 \wedge \neg p3h2) \end{array}$$



## Klauselmenge

### Voraussetzungen plus negierte Behauptung

$M =$

$$\left\{ \begin{array}{l} \{\neg p1h1, \neg p2h1\}, \quad \{\neg p1h1, \neg p3h1\}, \\ \{\neg p2h1, \neg p3h1\}, \quad \{\neg p1h2, \neg p2h2\}, \\ \{\neg p1h2, \neg p3h2\}, \quad \{\neg p2h2, \neg p3h2\}, \\ \\ \{p1h1, p1h2\}, \quad \{p2h1, p2h2\}, \\ \{p3h1, p3h2\} \end{array} \right\}$$


## Maschineller Resolutionbeweis

|          |                             |            |                             |
|----------|-----------------------------|------------|-----------------------------|
| 1 []     | $\neg p1h1 \vee \neg p2h1.$ | 20 [9,6]   | $p3h1 \vee \neg p2h2.$      |
| 2 []     | $\neg p1h1 \vee \neg p3h1.$ | 23 [10,4]  | $\neg p3h1 \vee \neg p2h2.$ |
| 3 []     | $\neg p2h1 \vee \neg p3h1.$ | 49 [23,20] | $\neg p2h2.$                |
| 4 []     | $\neg p1h2 \vee \neg p2h2.$ | 50 [23,14] | $\neg p3h1.$                |
| 5 []     | $\neg p1h2 \vee \neg p3h2.$ | 51 [49,15] | $\neg p1h1.$                |
| 6 []     | $\neg p2h2 \vee \neg p3h2.$ | 54 [50,9]  | $p3h2.$                     |
| 7 []     | $p1h1 \vee p1h2.$           | 55 [51,12] | $\neg p3h2.$                |
| 8 []     | $p2h1 \vee p2h2.$           | 56 [55,54] | .                           |
| 9 []     | $p3h1 \vee p3h2.$           |            |                             |
| 10 [7,2] | $p1h2 \vee \neg p3h1.$      |            |                             |
| 12 [7,5] | $p1h1 \vee \neg p3h2.$      |            |                             |
| 14 [8,3] | $p2h2 \vee \neg p3h1.$      |            |                             |
| 15 [8,1] | $p2h2 \vee \neg p1h1.$      |            |                             |

-----> UNIT CONFLICT .



## 1-Resolution

*unit resolution*

Die 1-Resolutionsregel ist ein Spezialfall der allgemeinen Resolutionsregel:

$$\frac{\{P\}, C_2 \cup \{\neg P\}}{C_2} \quad \frac{\{\neg P\}, C_2 \cup \{P\}}{C_2}$$

Der 1-Resolutionskalkül ist nicht vollständig.

Die Klauselmenge

$$E = \{\{P_1, P_2\}, \{P_1, \neg P_2\}, \{\neg P_1, P_2\}, \{\neg P_1, \neg P_2\}\}$$

ist nicht erfüllbar, aber mit 1-Resolution ist aus  $E$  nichts ableitbar, also auch nicht  $\emptyset$ .

