Korrektheit der obigen Umformungen:
Seien $M$ und $M'$ Mengen von wff in clause-form, wobei $M'$ durch obige Umformungsregeln aus $M$ gewonnen wird. Die Menge von wff, die logisch aus $M$ folgen, ist identisch mit der Menge von wff, die logisch aus $M'$ folgen. (Ohne Beweis)

Bisher sind bei der Anwendung der Resolutionsregel keine Variablen aufgetreten. I.A. sind diese jedoch zugelassen.

**Beispiel:**

$$\neg\ \mathrm{ABOVE}(B,\mathrm{TABLE})$$
$$\forall x \forall y\ (\mathrm{ON}(x,y)\ \Rightarrow \mathrm{ABOVE}\ (x,y))$$

Klausel-Form und Resolution:

$$\frac{\begin{array}{l}\neg\ \mathrm{ABOVE}(B,\mathrm{TABLE})\\ \neg\ \mathrm{ON}(u,v) \vee \mathrm{ABOVE}\ (u,v)\end{array}}{\neg\ \mathrm{ON}(B,\mathrm{TABLE})}$$

Beim Auftreten von Variablen sind sog. **Substitutionen** erforderlich. Eine Substitution ist eine Menge $S = \{t_1/v_1, \ldots, t_n/v_n\}$, wobei $t_i/v_i$ die Ersetzung der Variablen $v_i$ durch den Term $t_i$ bedeutet, $i = 1, \ldots, n$. Vorausgesetzung: alle $v_i$ in $S$ sind verschieden und $v_i$ tritt nicht in $t_i$ auf.

Im letzten Beispiel wurde die Substitution $\{B/u, TABLE/v\}$ angewendet.

Durch Anwendung einer Substitution $S$ auf ein Literal $\alpha$ erhält man eine *Instanz* von $\alpha$. Schreibweise: $\{\alpha\}S$.

**Beispiel:**

$$\begin{array}{lll} \alpha = Q(x,y,z), & & \beta = P(x,f(z),A) \\ S_1 = \{y/x, y/z\} & & \\ S_2 = \{f(y)/x, A/z\} & & \\ \{\alpha\}S_1 = \{Q(y,y,y)\} & & \{\beta\}S_1 = \{P(y,f(y),A)\} \\ \{\alpha\}S_2 = \{Q(f(y),y,A)\} & & \{\beta\}S_2 = \{P(f(y),f(A),A) \end{array}$$

Die Hintereinanderschaltung von Substitutionen ist möglich. Schreibweise: $\{\alpha\}S_1S_2 = (\{\alpha\}S_1)S_2$.

Die Anwendung einer Substitution auf eine Menge von Literalen ist ebenfalls möglich. Schreibweise: $\{\alpha_1, \ldots, \alpha_n\}S$ beziehungsweise $MS$ mit $M = \{\alpha_1, \ldots, \alpha_n\}$.

Ist $MS$ einelementig bei einer $n$-elementigen Menge $M$, $n \geq 1$, so heißt die Substitution $S$ **Unifikator** der Menge $M$.

**Beispiel:**

$$\begin{array}{rcl} M & = & \{P(x,x), P(y,z)\} \\ S & = & \{x/y, x/z\} \\ MS & = & \{P(x,x)\} \end{array} \qquad \begin{array}{rcl} S' & = & \{A/x, A/y, A/z\} \\ MS' & = & \{P(A,A)\} \end{array}$$

I.A. existieren mehrere Unifikatoren einer Menge $M$ von Literalen, falls $M$ überhaupt unifizierbar ist. $S$ ist der **allgemeinste Unifikator** von $M$ g.d.w. für jeden anderen Unifikator $S'$ ein weiterer Unifikator $S''$ existiert, mit $MS' = MSS''$.

Anschaulich bedeutet dies, dass $S$ so wenig Variablen wie möglich durch Konstanten oder Funktionen ersetzt.

**Beispiel:** $M, S, S'$ wie im Beispiel oben

$$S'' = \{A/x\}$$

$$MSS'' = MS'$$

$MS$ ist bis auf Variablenumbenennung eindeutig bestimmt, falls $S$ allgemeinster Unifikator ist. (Ohne Beweis)

Es sind Algorithmen zur Bestimmung des allgemeinsten Unifikators bekannt.

Nicht jede Menge ist unifizierbar. Z.B.

$$\{P(x), Q(x,y)\}, \qquad \{P(A), P(B)\}$$

Die Vorschrift zur Anwendung der Resolutionsregel bei Klauseln mit Variablen lautet:

$$\begin{array}{lr}
P_1 \vee P_2 \vee \cdots \vee P_n & (\alpha_1) \\
\neg P_1 \vee Q_2 \vee \cdots \vee Q_m & (\alpha_2) \\
\hline
P_2 S \vee \cdots \vee P_n S \vee Q_2 S \vee \cdots \vee Q_m S & (\alpha)
\end{array}$$

- alle $P_i$ und $Q_i$ sind Literale, möglicherweise mit Variablen
- $P_1$ in $(\alpha_1)$ und $P_1$ in $(\alpha_2)$ seien unifizierbar mit dem allgemeinsten Unifikator $S$.
- $(\alpha)$ ergibt sich aus $P_2 \vee \cdots \vee P_n$ in $(\alpha_1)$ und $Q_2 \vee \cdots \vee Q_m$ in $(\alpha_2)$ durch Anwendung von $S$.

**Beispiel:**

$$\begin{array}{ll}
\neg\ \text{ABOVE}(B,\text{TABLE}) & \\
\neg\ \text{ON}(u,v) \vee \text{ABOVE}\ (u,v) & \\
\hline
\neg\ \text{ON}(B,\text{TABLE}) & \text{S} = \{B/u,\ \text{TABLE}/v\ \}
\end{array}$$

## 7.4 Schlussfolgern mit Hilfe der Resolutionsregel

Gegeben sei eine Menge $A$ von wff (Axiome), die Wissen über einen Problemkreis darstellen. Ferner sei eine wff $\alpha$ gegeben. Die Frage lautet, ob $\alpha$ aus $A$ logisch folgt.

Verfahren zur Feststellung, ob $\alpha$ aus $A$ logisch folgt:

1. Nimm $\neg\,\alpha$ in die Menge $A$ der Axiome auf; Ergebnis $A'$.
2. Umformung von $A'$ in Klausel-Form.
3. Wende die Resolutionsregel an, solange bis Widerspruch NIL abgeleitet wird.
4. Falls NIL abgeleitet werden kann, folgt wff $\alpha$ logisch aus der Menge $A$ der Axiome.

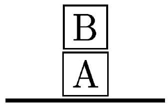
Dieses Verfahren entspricht einem Beweis durch Widerspruch (resolution refutation). Es besitzt folgende Eigenschaften:

- Vollständigkeit und Widerspruchsfreiheit
- Die Frage, ob $\alpha$ aus $A$ logisch folgt, ist nur partiell entscheidbar.

Interessant für praktische Anwendungen ist die Tatsache, dass nur eine einzige Inferenzregel benötigt wird.

## Beispiel:

$$\left.\begin{array}{l}\text{On(B,A)}\\ \text{On(A,TABLE)}\end{array}\right\}\ \text{Fakten:}$$




$$\left.\begin{array}{l}\forall x \forall y[\text{On}(x,y) \Rightarrow \text{Above}(x,y)]\\ \forall x \forall y \forall z[\text{Above}(x,y) \wedge \text{Above}(y,z) \Rightarrow \text{Above}(x,z)]\end{array}\right\}\ \text{Regeln}$$

zu zeigen: Above(B,Table)

---

Fakten und Regeln in clause-form:

| | |
|---|---|
| $\neg$ On(u,v) $\vee$ Above(u,v) | (1) |
| $\neg$ Above(x,y) $\vee$ $\neg$ Above(y,z) $\vee$ Above(x,z) | (2) |
| On(B,A) | (3) |
| On(A,Table) | (4) |
| $\neg$ Above(B,Table) | (5) |

---

| durch Resolution gewonnen: | | | aus: |
|---|---|---|---|
| $\neg$ Above(B,y) $\vee$ $\neg$ Above(y,Table) | (6) | B/x ; Table/z | (2)+(5) |
| $\neg$ On(y,Table) $\vee$ $\neg$ Above(B,y) | (7) | y/u ; Table/v | (1)+(6) |
| $\neg$ On(B,y) $\vee$ $\neg$ On(y,Table) | (8) | B/u ; y/v | (1)+(7) |
| $\neg$ On(A,Table) | (9) | A/y | (3)+(8) |
| Nil | (10) | | (4)+(9) |

### 7.4.1 Eliminationsstrategien

Bei der obigen Beschreibung des Ableitungsverfahrens bleibt offen, welche Klauseln, welche Literale und welche Substitutionen in welcher Reihenfolge zu wählen sind, um in möglichst wenigen Schritten NIL abzuleiten. Wir betrachten zunächst einige Vereinfachungsstrategien, mit deren Hilfe die Menge der zu betrachtenden Klauseln reduziert werden kann.

## Elimination reiner Literale:

Ein Literal heißt *rein* genau dann, wenn es nicht gleichzeitig negiert und unnegiert in einer Menge von Klauseln auftritt.

Beobachtung: Eine Klausel, welche ein reines Literal $P$ enthält, kann nie zur Ableitung von NIL beitragen, da $P$ nie beseitigt werden kann.

Konsequenz: Klauseln, welche ein oder mehrere reine Literale enthalten, können aus der Menge der betrachteten Klauseln gestrichen werden, ohne dass sich die Menge der ableitbaren Theoreme ändert.

Im Laufe des Resolutionsprozesses können keine neuen reinen Literale generiert werden. Somit braucht die Elimination nur einmal vor Beginn des Resolutionsverfahrens durchgeführt werden.

**Beispiel:** Elimination reiner Literale

(1) $\neg P \vee \neg Q \vee R$

(2) $\neg P \vee S$

(3) $\neg Q \vee S$

(4) $P$

(5) $Q$

(6') $\neg R$

Axiome: (1) - (5)
Negation des zu beweisenden Theorems: (6')

Anwendung der Resolutionsregel liefert:

(7) aus (1), (6'): $\neg P \vee \neg Q$

(8) aus (7), (4), (5): NIL

Die Klauseln (2) und (3), welche das reine Literal S enthalten, werden für die Ableitung von NIL nicht benötigt!

**Elimination von Tautologien:**

Eine *Tautologie* ist eine Klausel, welche ein Literal in negierter und unnegierter Form enthält. Z.B.

$$P(f(A)) \lor \neg P(f(A)), \qquad P(x) \lor Q(y) \lor R(Z) \lor \neg Q(y)$$

Beobachtung: Die Menge von Theoremen, die sich aus einer Menge $M$ von Klauseln ableiten lassen, ändert sich nicht, indem in $M$ Tautologien hinzugefügt oder gestrichen werden

Begründung: Betrachte folgende Klauseln

1. $P(x) \lor \neg P(x) \lor \ldots X \ldots$
2. $P(x) \lor \ldots Y \ldots$
3. $\neg P(x) \lor \ldots Z \ldots$

Soll die Tautologie (1) zur Ableitung von NIL beitragen, so ist (2) und (3) nötig. Aus (1), (2) und (3) ergibt sich

4. $\ldots X \ldots \lor \ldots Y \ldots \lor \ldots Z \ldots$

Nach Streichen von (1) lässt sich direkt aus (2) und (3)

5. $\ldots Y \ldots \lor \ldots Z \ldots$

ableiten, woraus NIL einfacher gewonnen werden kann als aus (4).

Konsequenz: Tautologien können aus der Menge der betrachteten Klauseln gestrichen werden, ohne dass sich die Menge der ableitbaren Theoreme ändert.

Da Tautologien während des Resolutionsprozesses entstehen können, ist neben einer anfänglichen Prüfung auch eine Prüfung jeder neu generierten Klausel nötig, wenn die Vereinfachungsstrategie konsequent angewandt werden soll.

Man beachte ferner, dass die betrachteten Literale zeichenweise bis auf die Negation übereinstimmen müssen. Andernfalls liegt keine Tautologie vor.

**Beispiel:**

(1) $\neg P(A) \vee P(x)$
(2) $P(A)$
(3) $\neg P(B)$
Aus (1) - (3) ist NIL ableitbar.

(1) ist keine Tautologie!
Würde man (1) streichen, so liesse sich aus (2) und (3) NIL nicht mehr ableiten.

**Elimination von Subsumptionen:**

Eine Klausel $\alpha$ *subsumiert* eine Klausel $\beta$ g.d.w. eine Substitution $S$ existiert mit $\alpha S \subseteq \beta$, wobei

$$\alpha = \alpha_1 \vee \alpha_2 \vee \ldots \vee \alpha_n = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \text{ und}$$
$$\beta = \beta_1 \vee \beta_2 \vee \ldots \vee \beta_m = \{\beta_1, \beta_2, \ldots, \beta_m\};\ n, m \geq 1.$$

Z.B.

$$\alpha = \{P(x), Q(y)\}$$
$$\beta = \{P(A), Q(v), R(w)\}.$$

$\alpha$ subsumiert $\beta$, da mit $S = \{A/x, v/y\}$ gilt $\alpha S \subseteq \beta$.

Beobachtung: Falls eine Klausel $\beta$ von einer Klausel $\alpha$ subsumiert wird, kann $\beta$ eliminiert werden, ohne dass sich die Menge der ableitbaren Theoreme ändert.

Informelle Begründung: Die subsumierende Klausel $\alpha$ ist bei der Anwendung der Resolutionsregel universeller einsetzbar als $\beta$. D.h., wenn immer $\beta$ bei der Anwendung der Resolutionsregel verwendbar ist, ist auch $\alpha$ verwendbar. Im allgemeinen ist $\alpha$ noch geeigneter für die Ableitung von NIL als $\beta$, da weniger Literale bei der Anwendung der Resolutionsregel entstehen.

Konsequenz: Subsumierte Klauseln können eliminiert werden, ohne dass sich die Menge der ableitbaren Theoreme ändert.

Subsumierte Klauseln können während des Resolutionsprozesses entstehen. Somit ist, neben einer initialen Überprüfung, jede neu generierte Klausel zu testen, falls diese Strategie konsequent angewandt werden soll.

### 7.4.2 Resolutionsstrategien

Trotz der Effizienzsteigerung durch die Eliminationsstrategien können bei einer unkontrollierten Anwendung der Resolutionsregel viele überflüssige Klauseln, die nichts zur Gewinnung von NIL beitragen, entstehen. Im folgenden werden einige Strategien vorgestellt, mit denen sich die Anzahl der für die Resolution in Frage kommenden Klauseln verringern lässt. Als Ausgangspunkt dient die Breitensuche, bei der die Resolutionsregel systematisch auf alle vorhandenen Klauseln angewandt wird.

## Breitensuche:

Die Menge $A$ der ursprünglich gegebenen Axiome sowie die zu beweisende Formel $\alpha$ bilden die Klauselmenge der Stufe 0. Es werden sukzessive alle möglichen Klauseln der Stufe 1,2,3,... solange generiert, bis NIL auftritt. Zur Gewinnung einer Klausel der Stufe $i \geq 1$ verwendet man eine Vorgängerklausel der Stufe $i-1$ zusammen mit einer weiteren Vorgängerklausel der Stufe $0 \leq j \leq i-1$.

Dieses Verfahren ist vollständig.

## **Beispiel:** Breitensuche

(1) Whoever can read is literate.
$(\forall x)[R(x) \to L(x)]$

(2) Dolphins are not literate.
$(\forall x)[D(x) \to \neg L(x)]$

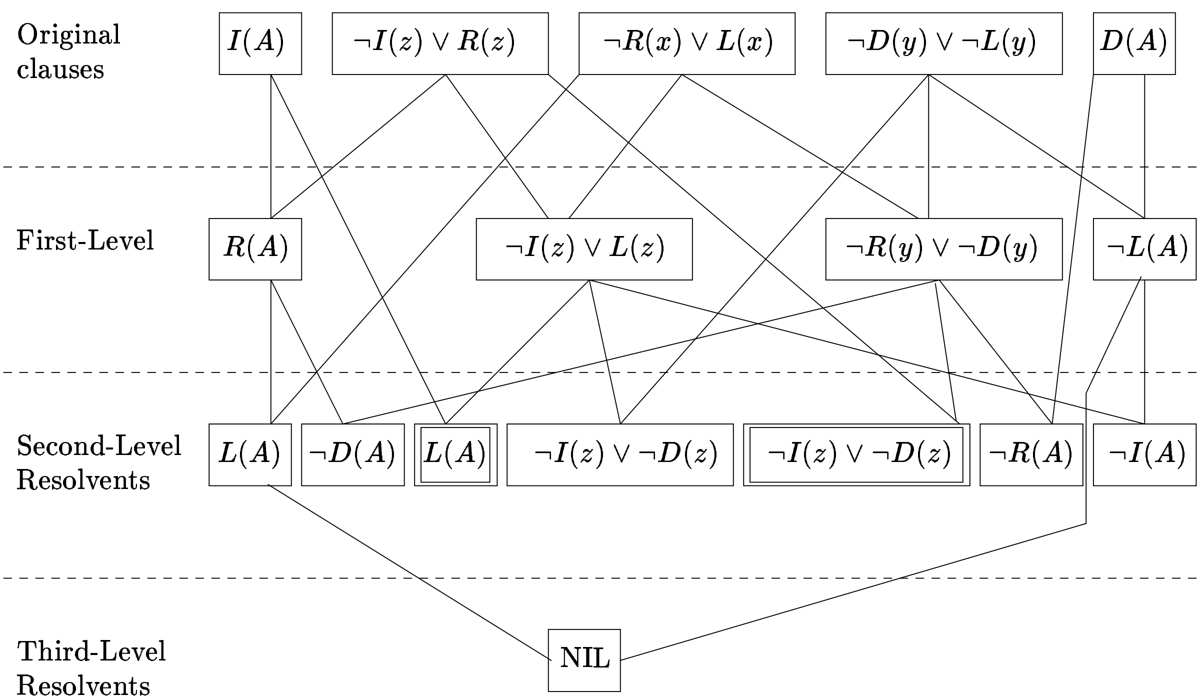
(3) Some Dolphins are intelligent.
$(\exists x)[D(x) \wedge I(x)]$

Axiome

(4) Some who are intelligent cannot read.
$(\exists x)[I(x) \wedge \neg R(x)]$

zu beweisendes Theorem

| | |
|---|---|
| (1) | $\neg R(x) \vee L(x)$ |
| (2) | $\neg D(y) \vee \neg L(y)$ |
| (3a) | $D(A)$ |
| (3b) | $I(A)$ |
| (4') | $\neg I(z) \vee R(z)$ |

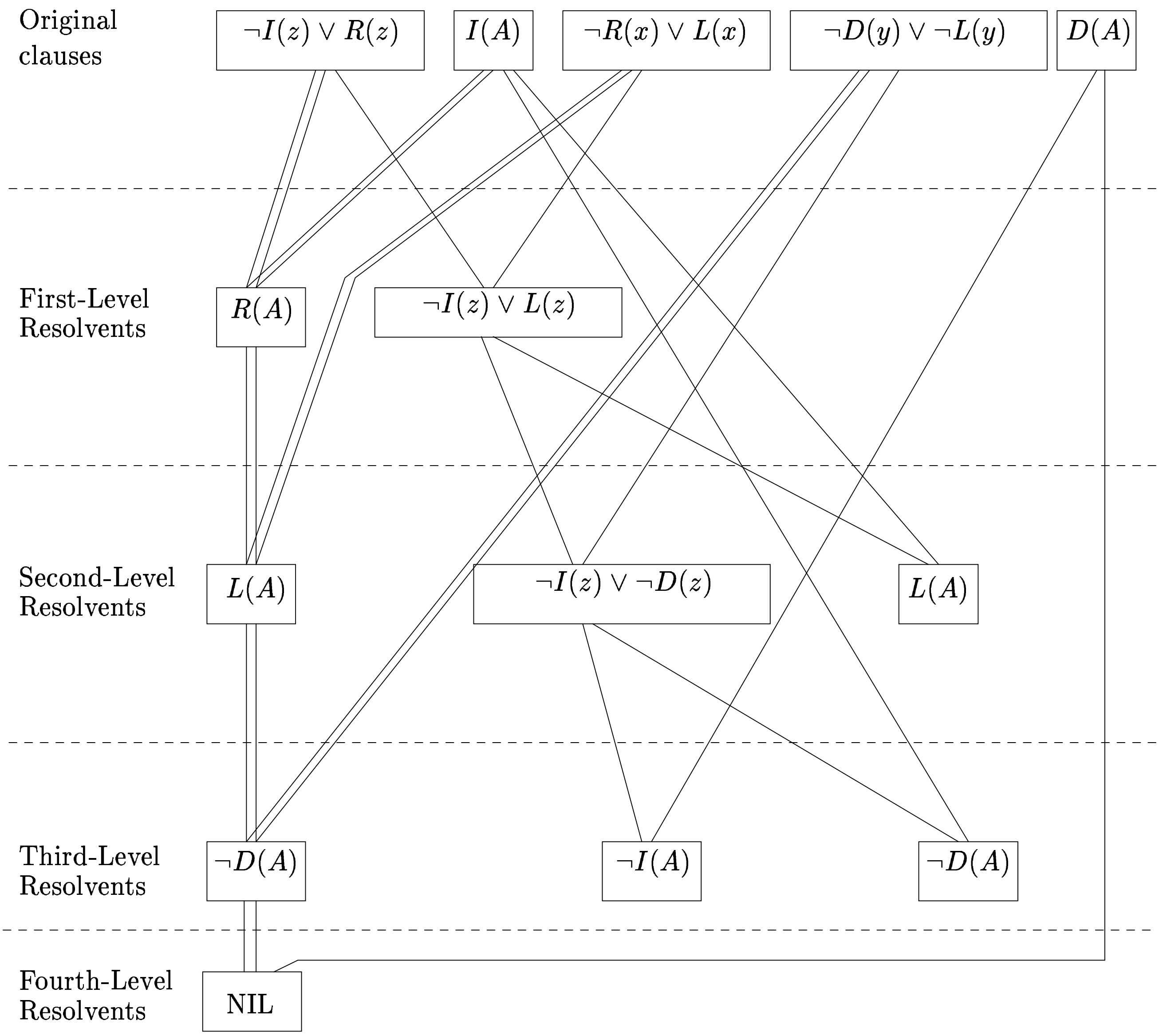


**Stützmengen-Resolution (set of support):**

Man führt nur solche Resolutionsschritte durch, bei denen die Negation von $\alpha$ oder eine daraus abgeleitete Klausel beteiligt ist. IA. werden hierbei weniger Klauseln produziert als bei einer Breitensuche. Begründung: Zur Ableitung eines Widerspruchs muss die Negation von $\alpha$ direkt oder indirekt beteiligt sein, falls die Axiome $A$ widerspruchsfrei sind.

Das Verfahren ist vollständig.

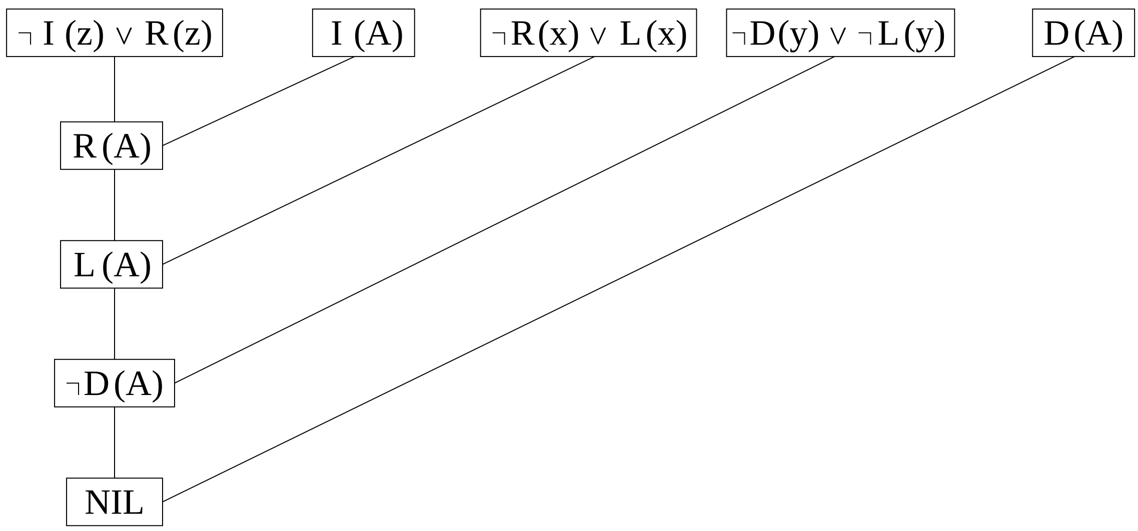
**Beispiel:**



**Einheitspriorität:**

Falls möglich, wird eine Klausel für einen Resolutionsschritt ausgewählt, welche nur aus einem Literal besteht.

**Beispiel:**




Unter dieser Strategie erfolgt ein sukzessives Verkürzen von Klauseln, was das Auffinden von NIL beschleunigt.

I.A. kombiniert man diese Strategie mit einer anderen Strategie, da unter Umständen gar keine Klauseln mit nur einem Literal existieren. Bei einer strikten Anwendung der Einheitspriorität geht die Eigenschaft der Vollständigkeit verloren.

## 7.5 Antwort-Generierung

Wir betrachten eine Menge $A$ von Axiomen und eine zu beweisende wff $\alpha$ der Form $\exists P(x)$, wobei $P(x)$ eine atomare Formel ist. Gewünscht ist nicht nur ein Beweis, dass $\alpha$ aus $A$ logisch folgt, sondern die explizite Angabe eines Elements $x$, für welches $P(x)$ gilt.

Als Faustregel gilt, dass Fragen "Wo…", "Wer…", "Wann…" u.s.w. durch Formeln mit existentiell quantifizierten Variablen repräsentiert werden.

Zur Gewinnung einer Antwort, d.h. der Angabe eines Elements $x$, welches $P(x)$ erfüllt, können offensichtlich die bei der Herleitung von NIL verwendeten Substitutionen herangezogen werden.

Verfahren zur Antwort-Generierung:

1. Erweitere die Klausel $\neg\alpha$ um den sog. Antwort-Term. Der Antwort-Term ist identisch mit $\alpha$.
2. Führe die gleichen Resolutionsschritte wie vorher durch.
3. Wird NIL bei der Resolution erhalten, so entspricht der Antwort-Term der gewünschten Antwort.

**Beispiel:** Antwort-Generierung

"Wenn Udo seinen Notebook PC immer bei sich hat und Udo sich im Hörsaal B7 aufhält, wo befindet sich dann sein PC?"

Darstellung in der Prädikatenlogik:

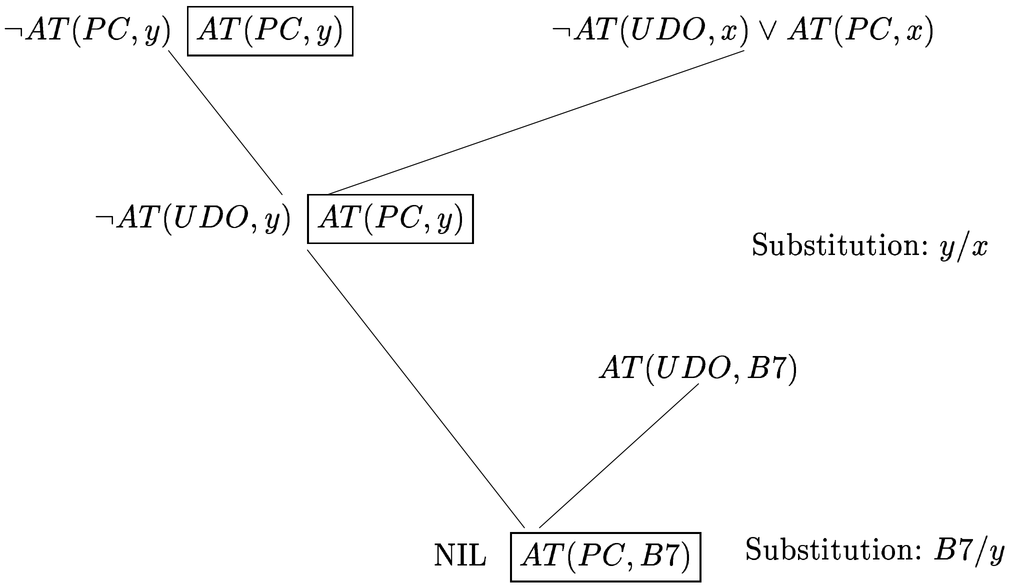
$$\left.\begin{array}{ll}(1) & \forall x[AT(UDO,x) \Rightarrow AT(PC,x)] \\ (2) & AT(UDO,B7)\end{array}\right\}\text{Axiome}$$

$$(3)\ \exists x[AT(PC,x)] \qquad \}\ \text{zu beweisendes Theorem}$$

Umformung in Klausel-Form und Negation des Theorems:

(1)$\neg AT(UDO,x) \vee AT(PC,x)$
(2)$AT(UDO,B7)$
(3)$\neg AT(PC,y)$

Anwendung der Resolutionsregel:



**Beispiel:** Antwort-Generierung

”Wenn x Vater von y und y Vater von z ist, so ist x Grossvater von z”
”Jeder hat einen Vater”

Frage: ”Gibt es Individuen x und y, so dass x Grossvater von y?”

Darstellung in der Prädikatenlogik:

$$\left.\begin{array}{l}(1)\forall x\ \forall y\ \forall z[F(x,y) \wedge F(y,z) \Rightarrow G(x,z)] \\ (2)\forall y\ \exists x\ F(x,y)\end{array}\right\}\text{Axiome}$$

$$(3)\exists x\ \exists y\ G(x,y) \qquad \}\,\text{zu beweisendes Theorem}$$

Umformung in Klausel-Form und Negation des Theorems:
(1) $\neg F(x,y) \vee \neg F(y,z) \vee G(x,z)$
(2) $F(f(w),w)$ [f ist Skolemfunktion, welche den Vater liefert]
(3′) $\neg G(u,v)$

Anwendung der Resolutionsregel: