

Prädikatenlogik: Resolution

- Beweisverfahren, Widerlegungsverfahren für die Prädikatenlogik
 - Vorverarbeitung: Wandlung der eigentlichen Aufgabenstellung (z.B. Folgerungsprüfung) in eine Widerlegungsaufgabe
Bildung einer (erfüllbarkeitsäquivalenten) Skolemform zu der Widerlegungsaufgabe (vgl. Kap. 10)
 - Ausgangspunkt:
Beziehung zwischen aussagenlogischer und prädikatenlogischer Semantik
 - Grundresolution
 - Erweiterung der aussagenlogischen Resolution um die Behandlung von Variablen
 - Unifikation
 - prädikatenlogische Resolventenbildung
- In dieser Vorlesung wird das Verfahren der prädikatenlogischen Resolution vorgestellt, die semantischen Hintergründe (einschliesslich Korrektheits- und Vollständigkeitsbeweise) sind Gegenstand von spezialisierten Vorlesungen (insbesondere im Masterstudium).

Resolution: Von der Aussagenlogik zur Prädikatenlogik

Prädikatenlogik

- berücksichtigt die interne Struktur von Aussagen und erlaubt es, zusätzlich bestimmte Beziehungen zwischen ‘Objekten’ zum Ausdruck zu bringen.
 - $\text{Engl_ersch}(hp7) \approx$ Harry Potter 7 ist auf englisch erschienen.
 - $\text{Dt_ersch}(hp7) \approx$ Harry Potter 7 ist auf deutsch erschienen.
 - $\text{Dt_ersch}(hp7) \Rightarrow \text{Engl_ersch}(hp7) \approx$ Wenn Harry Potter 7 auf deutsch erschienen ist, dann ist Harry Potter 7 auf englisch erschienen.
- Sei $\mathbf{F} = \{\text{Dt_ersch}(hp7), \text{Dt_ersch}(hp7) \Rightarrow \text{Engl_ersch}(hp7)\}$
 - Gilt dann: $\mathbf{F} \models \text{Engl_ersch}(hp7)$? [Ja, ergibt sich direkt aus Def 9.8, 9.12.]
 - Gilt dann: $\mathbf{F} \vdash_{\text{MP}} \text{Engl_ersch}(hp7)$? [mit einer entsprechenden Erweiterung der Konzeption der ‘Anwendung von Inferenzregeln, Def 6.5]
 - Gilt dann: $\mathbf{F} \vdash_{\text{res}} \text{Engl_ersch}(hp7)$? [mit einer entsprechenden Erweiterung der Konzeption der Resolventenbildung / der Resolution]
 - Wie ist $\mathbf{F} \models \text{Engl_ersch}(hp7)$ unter Verwendung des Resolutionsverfahrens nachzuweisen?

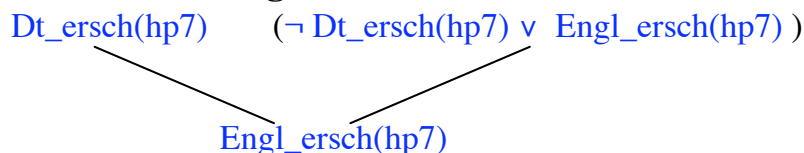
Eine intuitive Vorüberlegung zur Resolutionsableitung: Beispiel 1

Gegeben: $Dt_ersch(hp7) \wedge (Dt_ersch(hp7) \Rightarrow Engl_ersch(hp7))$

In KNF: $Dt_ersch(hp7) \wedge (\neg Dt_ersch(hp7) \vee Engl_ersch(hp7))$

Ableitbarkeit von $Engl_ersch(hp7)$ durch Resolution?

Resolutionsableitung



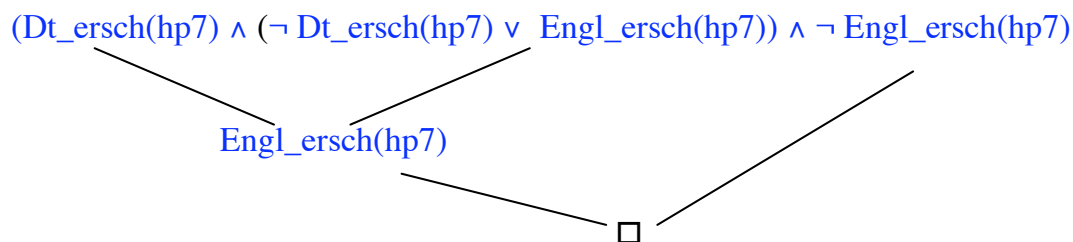
Unter der Auffassung, dass $Dt_ersch(hp7)$ und $\neg Dt_ersch(hp7)$ als komplementäre Literale angesehen werden können.

Folgerbarkeitstest durch Resolutionsableitung: Beispiel 2

Gegeben: $F^1 = Dt_ersch(hp7) \wedge (\neg Dt_ersch(hp7) \vee Engl_ersch(hp7))$

Negation von $Engl_ersch(hp7)$: $\neg Engl_ersch(hp7)$

Unerfüllbarkeitstest durch Resolution von



Unter der Auffassung, dass

$Dt_ersch(hp7)$ und $\neg Dt_ersch(hp7)$ sowie

$Engl_ersch(hp7)$ und $\neg Engl_ersch(hp7)$

als komplementäre Literale angesehen werden können.

Warum sind

- $Engl_ersch(hp6) \approx$ Harry Potter 6 ist auf englisch erschienen.
- $Jp_ersch(hp7) \approx$ Harry Potter 7 ist auf japanisch erschienen.

nicht auf entsprechende Weise als aus F nachweisbar ?

Folgerbarkeitstest durch Resolutionsableitung: Beispiel 3

Eine andere Implikationsbeziehung: $\forall x (Dt_ersch(x) \Rightarrow Engl_ersch(x))$

Umformung: $\forall x (\neg Dt_ersch(x) \vee Engl_ersch(x))$

Klauselnormalform: $(\neg Dt_ersch(x) \vee Engl_ersch(x))$

Gegeben: $F^2 = Dt_ersch(hp7) \wedge (\neg Dt_ersch(x) \vee Engl_ersch(x))$

Negation von $Engl_ersch(hp7)$: $\neg Engl_ersch(hp7)$

Unerfüllbarkeitstest durch Resolution von

$(Dt_ersch(hp7) \wedge (\neg Dt_ersch(x) \vee Engl_ersch(x))) \wedge \neg Engl_ersch(hp7)$

$Engl_ersch(hp7)$

□

Unter der Auffassung, dass hier – unter gewissen Annahmen/Bedingungen – Komplementarität erreicht werden kann:

$Dt_ersch(hp7)$ und $\neg Dt_ersch(x)$ unter der "Bedingung": $hp7 \leftarrow x$

Folgerbarkeitstest durch Resolutionsableitung: Beispiel 4

Gegeben: $F = Dt_ersch(hp7) \wedge (\neg Dt_ersch(hp7) \vee Engl_ersch(hp7))$

Auf Folgerbarkeit zu testende Formel: $\exists x Engl_ersch(x)$

Negation: $\neg \exists x Engl_ersch(x) \equiv \forall x \neg Engl_ersch(x)$

Klauselnormalform: $\neg Engl_ersch(x)$

Unerfüllbarkeitstest durch Resolution von

$(Dt_ersch(hp7) \wedge (\neg Dt_ersch(hp7) \vee Engl_ersch(hp7))) \wedge \neg Engl_ersch(x)$

$Engl_ersch(hp7)$

□

Unter der Auffassung, dass hier – unter gewissen Annahmen/Bedingungen – Komplementarität erreicht werden kann:

$Engl_ersch(hp7)$ und $\neg Engl_ersch(x)$ unter der "Bedingung": $hp7 \leftarrow x$

Definition 11.1

Es sei F eine Formel in Skolemform mit Matrix F^* .

- Eine Substitution, die alle freien Variablen in F^* durch geschlossene (d.h. variablenfreie) Terme ersetzt, heißt **Grundsubstitution**.
- Wenn alle freien Variablen in F^* durch eine Grundsubstitution ersetzt werden, heißt die resultierende Formel eine **Grundinstanz** von F bzw. F^* .
- [Anm.: Wenn freie Variablen in F^* durch Substitution ersetzt werden, heißt die resultierende Formel eine **Instanz** von F bzw. F^*]

Beobachtung zu 11.1

Ist F eine Formel in Skolemform und G eine (Grund-)Instanz von F , dann folgt G aus F .

-
- Die Menge der Grundinstanzen von F spielen eine besondere Rolle für die prädikatenlogische Resolution.

Von der Modelltheorie zur prädikatenlogischen Beweistheorie: Herbrands Konzeption

- Die Konzeption von Herbrand stellt spezifische Strukturen und Modelle bereit, mittels derer Formeln in Mengen von Zeichenketten interpretiert werden.
 - Auf dieser Grundlage kann die Erfüllbarkeit / Unerfüllbarkeit von Klauselnormalformen über Erfüllbarkeit / Unerfüllbarkeit von aussagenlogischen Formelmengen geprüft werden.
 - Ein Test auf Erfüllbarkeit / Unerfüllbarkeit einer prädikatenlogischen Formel erfordert damit eine gezielte Suche einer endlichen unerfüllbaren Teilmenge in einer unendlichen AL-Formelmenge.
 - Die Herbrand-Expansion $E(F)$ einer Formel F ist eine Menge von geschlossenen und quantorenfreien Formeln. Sie korrespondiert zur Menge der Grundinstanzen von F .
- Die zentrale Frage ist somit:
Welche Grundsubstitutionen führen zu unerfüllbaren Teilmengen von $E(F)$?
- Zur Herbrandkonzeption:
- Schönig Kapitel 2.4 und/oder Spies Kapitel 10.5
 - Vorlesungen im Masterstudium

Mengendarstellung von Klauselnormalformen

Definition 11.2

Auch in der Prädikatenlogik werden

- atomare Formeln und ihre Negationen als **Literale**
- und Disjunktionen von Literalen als **Klauseln** bezeichnet.

Ist K eine (prädikatenlogische) Klausel, mit

$$K = \left(\bigvee_{k=1}^m L_k \right), \text{ dann nennen wir}$$

$$K = \{L_1, \dots, L_m\} \text{ die } \textbf{Mengendarstellung} \text{ von } K.$$

Ist F eine Klauselnormalform mit der Matrix F^* , wobei

$$F^* = \left(\bigwedge_{i=1}^n \left(\bigvee_{k=1}^{m_i} L_{i,k} \right) \right), \text{ dann nennen wir}$$

$$F = \{ \{L_{1,1}, \dots, L_{1,m_1}\}, \dots, \{L_{n,1}, \dots, L_{n,m_n}\} \} \text{ die } \textbf{Mengendarstellung} \text{ von } F.$$

In der Mengendarstellung einer Klauselnormalform sind die Quantoren nicht mehr explizit.

Mengendarstellung von Klauselnormalformen (2)

Da in Klauselnormalformen

- keine freien Variablen und
- keine Existenzquantoren vorkommen,
- alle Allquantoren im Präfix stehen und
- die Reihenfolge von Allquantoren im Präfix semantisch unwesentlich ist,

werden alle Variablen in der Mengendarstellung behandelt, als wenn sie durch Allquantoren mit maximalem Skopus gebunden sind.

Wahrheitswertberechnung

für die Mengendarstellung von Klauselnormalformen

Klauseln

$$\mathcal{A}(K) = \text{Maximum}(\{ \mathcal{A}(L) \mid L \in K \})$$

Klauselnormalformen

$$\mathcal{A}(F) = \text{Minimum}(\{ \mathcal{A}'(K) \mid K \in F \text{ und } \mathcal{A}' \text{ ist eine Struktur, die sich von } \mathcal{A} \text{ nur durch die Interpretation der Variablen unterscheidet} \})$$

Grundresolution

Definition 11.3

Seien K_1 , K_2 und R prädikatenlogische Klauseln von Grundinstanzen in Mengendarstellung.

R heißt (Grund-)Resolvente von K_1 und K_2 , falls gilt:

1. Es gibt ein Literal mit $L \in K_1$, so dass $\bar{L} \in K_2$ und
2. $R = (K_1 - \{L\}) \cup (K_2 - \{\bar{L}\})$

Beispiel: $F^0 = \forall x \forall y ((\neg P(x) \vee Q(y)) \wedge P(a) \wedge \neg Q(b))$

Skolemform $F = \forall x \forall y ((\neg P(x) \vee Q(y)) \wedge P(a) \wedge \neg Q(b))$

Matrix $F^* = ((\neg P(x) \vee Q(y)) \wedge P(a) \wedge \neg Q(b))$

Grundinstanzen aus $E(F)$ mit Instanziierungen über $\{a, b\}$, den Konstanten in F :

$$F_1 = ((\neg P(a) \vee Q(a)) \wedge P(a) \wedge \neg Q(b))$$

$$F_2 = ((\neg P(a) \vee Q(b)) \wedge P(a) \wedge \neg Q(b))$$

$$F_3 = ((\neg P(b) \vee Q(a)) \wedge P(a) \wedge \neg Q(b))$$

$$F_4 = ((\neg P(b) \vee Q(b)) \wedge P(a) \wedge \neg Q(b))$$

Grundresolution [Forts. des Beispiels]

$$F_1 = \{ \{ \neg P(a), Q(a) \}, \{ P(a) \}, \{ \neg Q(b) \} \}$$

$$\{ Q(a) \}$$

$$F_2 = \{ \{ \neg P(a), Q(b) \}, \{ P(a) \}, \{ \neg Q(b) \} \}$$

$$\{ \neg P(a) \}$$

□

$$F_3 = \{ \{ \neg P(b), Q(a) \}, \{ P(a) \}, \{ \neg Q(b) \} \}$$

$$F_4 = \{ \{ \neg P(b), Q(b) \}, \{ P(a) \}, \{ \neg Q(b) \} \}$$

$$\{ \neg P(b) \}$$

Zum Selbststudium: Grundresolutionsalgorithmus (vgl. Schöning Kap 2.5)

Algorithmus

Eingabe: Eine geschlossene Formel F in Skolemform mit der Matrix F^* in KNF:

$[F_1, F_2, \dots]$ sei eine Aufzählung von $E(F)$.

$i := 0; \quad \mathbf{M} := \emptyset;$

repeat

$i := i + 1; \quad \mathbf{M} := \mathbf{M} \cup \{F_i\}; \quad \mathbf{M} := \text{Res}^*(\mathbf{M});$

until $\square \in \mathbf{M}$

Ausgabe: Gib „unerfüllbar“ aus und stoppe.

- Die Vorschrift (der Berechnungsschritt) $\mathbf{M} := \text{Res}^*(\mathbf{M})$ terminiert für jedes i , da die Resolution einer endlichen Menge von Grundinstanzen terminiert. (Aussagenlogischer Resolutionssatz)

Satz 11.4

Bei Eingabe einer Formel F in Klauselnormalform stoppt der Grundresolutionsalgorithmus genau dann nach endlich vielen Schritten mit der Ausgabe „unerfüllbar“, wenn F unerfüllbar ist.

Gültigkeitsprüfung mit Hilfe von Grundresolution

Ist $H = \neg \exists y \forall z [P(z, y) \Leftrightarrow \neg \exists x (P(z, x) \wedge P(x, z))]$ gültig?
 \rightarrow ist $\neg H \equiv F = \exists y \forall z [P(z, y) \Leftrightarrow \neg \exists x (P(z, x) \wedge P(x, z))]$ unerfüllbar?

Umformungen zur Erstellung einer Klauselnormalform für F :

$$\begin{aligned}
 F &= \exists y \forall z [P(z, y) \Leftrightarrow \neg \exists x (P(z, x) \wedge P(x, z))] \\
 &\equiv \exists y \forall z [(\neg P(z, y) \vee \neg \exists x (P(z, x) \wedge P(x, z))) \wedge (P(z, y) \vee \exists x (P(z, x) \wedge P(x, z)))] && \text{Biiimplikation eliminieren} \\
 &\equiv \exists y \forall z [(\neg P(z, y) \vee \forall x (\neg P(z, x) \vee \neg P(x, z))) \wedge (P(z, y) \vee \exists x (P(z, x) \wedge P(x, z)))] && \text{Negation nach innen} \\
 &\equiv \exists y \forall z [(\neg P(z, y) \vee \forall x (\neg P(z, x) \vee \neg P(x, z))) \wedge (P(z, y) \vee \exists w (P(z, w) \wedge P(w, z)))] && [x/w] \text{ Umbenennung} \\
 &\equiv \exists y \forall z \exists w \forall x [(\neg P(z, y) \vee (\neg P(z, x) \vee \neg P(x, z))) \wedge (P(z, y) \vee (P(z, w) \wedge P(w, z)))] && \text{Pränexform} \\
 &\text{erfäqui} \quad \forall z \exists w \forall x [(\neg P(z, a) \vee (\neg P(z, x) \vee \neg P(x, z))) \wedge (P(z, a) \vee (P(z, w) \wedge P(w, z)))] && \text{Skolemisierung [y/a]} \\
 &\text{erfäqui} \quad \forall z \forall x [(\neg P(z, a) \vee (\neg P(z, x) \vee \neg P(x, z))) \wedge (P(z, a) \vee (P(z, f(z)) \wedge P(f(z), z)))] && \text{Skolemisierung [w/f(z)]} \\
 &\equiv \forall z \forall x [(\neg P(z, a) \vee \neg P(z, x) \vee \neg P(x, z)) \wedge (P(z, a) \vee P(z, f(z))) \wedge (P(z, a) \vee P(f(z), z))] && \text{KNF-Erstellung}
 \end{aligned}$$

Gültigkeitsprüfung mit Hilfe von Grundresolution [Fortsetzung]

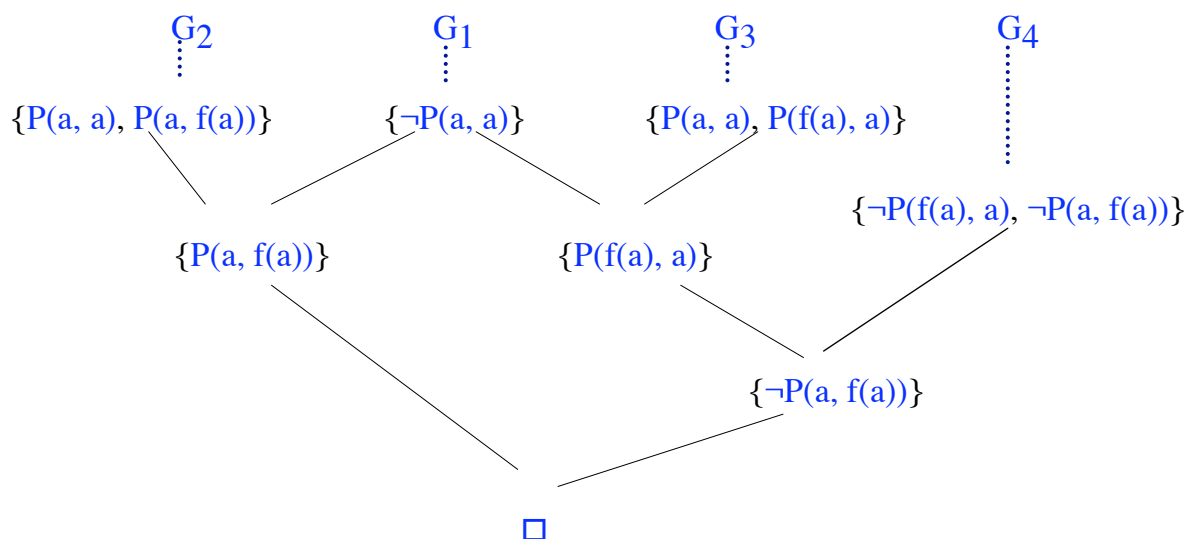
Matrix: $F^* = C_1 \wedge C_2 \wedge C_3$ $C_1 = (\neg P(z, a) \vee \neg P(z, x) \vee \neg P(x, z))$
 $C_2 = (P(z, a) \vee P(z, f(z)))$
 $C_3 = (P(z, a) \vee P(f(z), z))$

[Bei der Bildung der Grundinstanzen können die Klauseln individuell betrachtet werden, da die Konjunktion bei der Mengendarstellung sowieso aufgelöst wird.]

Einige Möglichkeiten der Bildung von Grundinstanzen zu den Klauseln (auf der Basis der Konstanten von F^*)

Klausel	Grundsubstitution	Grundinstanz der Klausel	Mengendarstellung
C_1	$[x/a], [z/a]$	$G_1 = \neg P(a, a) \vee \neg P(a, a) \vee \neg P(a, a)$	$\{\neg P(a, a)\}$
C_2	$[z/a]$	$G_2 = P(a, a) \vee P(a, f(a))$	$\{P(a, a), P(a, f(a))\}$
C_3	$[z/a]$	$G_3 = P(a, a) \vee P(f(a), a)$	$\{P(a, a), P(f(a), a)\}$
C_1	$[x/a], [z/f(a)]$	$G_4 = \neg P(f(a), a) \vee \neg P(f(a), a) \vee \neg P(a, f(a))$	$\{\neg P(f(a), a), \neg P(a, f(a))\}$

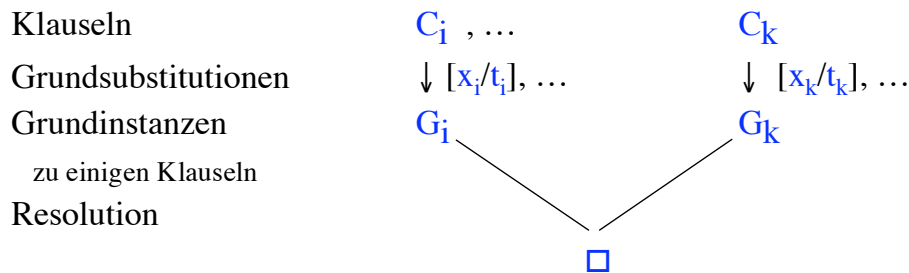
Gültigkeitsprüfung mit Hilfe von Grundresolution [Fortsetzung]



- ➔ F ist unerfüllbar.
- ➔ H ist gültig.

Grundresolution: Zwischenfazit

Das Schema der Grundresolution



- Zu zwei aussagenlogischen Klauseln gibt es stets nur endlich viele Resolventen.
- Zu zwei prädikatenlogischen Klauseln kann es unendlich viele Grundinstanzen geben, die die Bildung von (unendlich vielen) Resolventen erlauben.
- Der Aufwand des Grundresolutionsverfahrens ergibt sich durch den Aufwand bei der **Suche nach geeigneten Grundinstanzen**.
- Wünschenswerte Eigenschaft prädikatenlogischer Resolution:
Für jedes Klauselpaar: Beschränkung auf eine endliche Menge von Resolventen, so dass im Gesamtverfahren die Ableitbarkeit der leeren Klausel gegenüber der Grundresolution nicht eingeschränkt wird.

Von der aussagenlogischen Resolventenbildung zur prädikatenlogischen Resolventenbildung

- Grundresolutionsverfahren der Prädikatenlogik separiert zwei Aspekte
 - Bildung der Grundinstanzen:
reduziert prädikatenlogische Ausdrücke zu aussagenlogischen Ausdrücken
 - Resolution: ist ein Verfahren für aussagenlogische Ausdrücke
- Der Kern des Resolutionsprinzips:
Resolvieren von Klauseln mit komplementären Literalen:
 - Aussagenlogik: z.B. $(\neg P \vee Q) \wedge P \wedge \neg Q \vdash_{\text{res}} Q \wedge \neg Q \vdash_{\text{res}} \square$
 - Prädikatenlogik: z.B. $\forall x \forall y ((\neg P(x) \vee Q(y)) \wedge P(a) \wedge \neg Q(b))$
 - In welchem Sinne sind $\neg P(x)$ und $P(a)$ komplementär?
 - Wie können etwa $\neg P(x)$ und $P(a)$ komplementär gemacht werden?
→ z.B. durch geeignete Grundsubstitution.
- Prädikatenlogik allgemein:
 - In welchem Sinne sind $\neg P(x)$ und $P(y)$, bzw. $\neg Q(x)$ und $Q(f(y))$ komplementär?
 - Was ist die **Resolvente zweier prädikatenlogischen Klauseln**?

Von der Grundresolution zur prädikatenlogischen Resolution

- Statt Grundsubstitutionen andere Arten geeigneter Substitutionen, denn
 - Grundsubstitutionen sind sehr speziell (Festlegung auf individuelle geschlossene Terme)
 - Grundsubstitutionen erzeugen viele Grundinstanzen, die später in der Resolution nicht verwendet werden.
- Ziel: „zurückhaltende“ Substitutionen:

Beispiel: $\{P(x), \neg Q(g(x))\}$ $\{\neg P(f(y))\}$
 $\downarrow [x / f(y)]$
 $\{P(f(y)), \neg Q(g(f(y)))\}$ $\{\neg P(f(y))\}$
 \downarrow (prädikatenlogische) Resolution
 $\{\neg Q(g(f(y)))\}$

- keine Festlegung im Hinblick auf y .
- **Unifikation** / *unifizieren*: vereinigen, zusammenführen.

Unifikator, Unifizierbarkeit

Definition 11.5

Eine Substitution σ ist genau dann ein **Unifikator** einer endlichen Menge von Ausdrücken (Termen t_1, \dots, t_n , bzw. Formeln F_1, \dots, F_k), wenn durch σ alle Ausdrücke dieser Menge auf denselben Ausdruck abgebildet werden.

D.h. $t_1 \sigma = \dots = t_n \sigma$ bzw. $F_1 \sigma = \dots = F_k \sigma$

- Wichtiger Spezialfall: Die Unifikation von Literalen
- Terme sind nur mit Termen unifizierbar, Formeln nur mit Formeln

Eine Menge von Ausdrücken ist **unifizierbar**, falls es einen Unifikator für diese Menge gibt.

- Beispiel: $\{x, f(y)\}$ bzw. $\{P(x), P(f(y))\}$
 $\sigma = [x / f(y)]$
 $x \sigma = f(y) = f(y) \sigma$ bzw. $P(x) \sigma = P(x \sigma) = P(f(y)) = P(f(y)) \sigma$

- Anmerkung zur Definition bei U. Schöning:
Schöning fokussiert auf den Fall der Unifikation von Literalen.

Komposition von Substitutionen

Definition 11.6

- Es seien x und y Variablen und t_1 und t_2 Terme.
- $[x/t_1] [y/t_2]$ bezeichnet die Substitution, die zuerst jedes freie Vorkommen von x durch t_1 ersetzt und anschließend jedes freie Vorkommen von y durch t_2 ersetzt.
- Seien σ und τ Substitutionen. Die **Komposition** von σ und τ ist die Substitution $\sigma\tau$, für die gilt: $x(\sigma\tau) = (x\sigma)\tau$, d.h.: $\sigma\tau(x) = \tau(\sigma(x))$, für alle Variablen x .

Beispiel

$$F = P(x) \wedge Q(f(y))$$

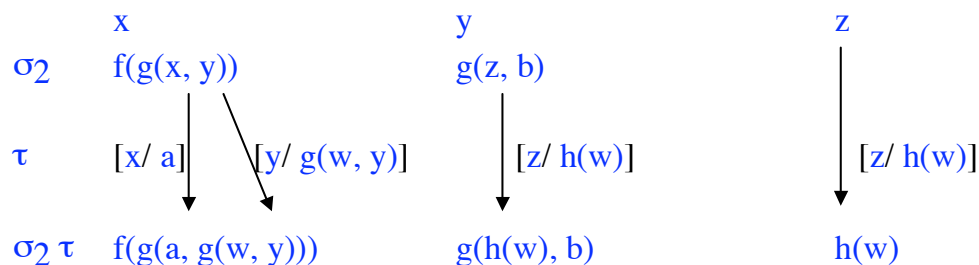
$$\begin{array}{l}
 F_{[x/a]} = P(a) \wedge Q(f(y)) \quad \left| \quad F_{[x/y]} = P(y) \wedge Q(f(y)) \quad \left| \quad F_{[x/g(y)]} = P(g(y)) \wedge Q(f(y)) \right. \\
 F_{[x/a][y/b]} = P(a) \wedge Q(f(b)) \quad \left| \quad F_{[x/y][y/a]} = P(a) \wedge Q(f(a)) \quad \left| \quad F_{[x/g(y)][y/a]} = P(g(a)) \wedge Q(f(a))
 \end{array}$$

Die Relation *allgemeiner als* zwischen Substitutionen (Unifikatoren)

Definition 11.7 Seien σ_1 und σ_2 zwei Substitutionen. σ_2 ist **allgemeiner als** σ_1 , falls es eine Substitution τ gibt, für die gilt: $\sigma_1 = \sigma_2 \tau$.

- **Beispiel:** $\sigma_1 = [x/ f(g(a, g(w, y)))] [y/ g(h(w), b)] [z/ h(w)]$
 $\sigma_2 = [x/ f(g(x, y))] [y/ g(z, b)]$
 $\tau = [x/ a] [y/ g(w, y)] [z/ h(w)]$

denn: $\sigma_2 \tau = [x/ f(g(x, y))] [y/ g(z, b)] [x/a] [y/ h(z)] [z/ h(w)]$



- ➔ σ_2 ist *allgemeiner als* σ_1 im folgenden Sinne: Es ist möglich erst σ_2 auszuführen und dann durch eine weitere Substitution, τ , den Effekt zu erreichen, den σ_1 in einem Schritt erzielt.

Satz 11.8

- (i) Seien σ_1 , σ_2 und σ_3 Substitutionen. Wenn σ_3 allgemeiner ist als σ_2 und σ_2 allgemeiner ist als σ_1 , dann ist σ_3 allgemeiner als σ_1 . [Transitivität]
- (ii) Für alle Substitutionen σ gilt: σ ist allgemeiner als σ . [Reflexivität]
- (iii) Seien σ_1 und σ_2 Substitutionen. Wenn σ_1 allgemeiner ist als σ_2 und σ_2 allgemeiner ist als σ_1 , dann gilt für alle Variablen x : $x\sigma_1$ und $x\sigma_2$ unterscheiden sich nur in der Wahl von Variablen.

Beispiel für 11.8.iii

$$\sigma_1 = [x/f(y)][z/g(b, y)]$$

$$\sigma_2 = [x/f(u)][z/g(b, u)]$$

mit $\tau_1 = [y/u]$ und $\tau_2 = [u/y]$ gilt $\sigma_1 = \sigma_2 \tau_2$ und $\sigma_2 = \sigma_1 \tau_1$

Allgemeinster Unifikator

Definition 11.9

Ein Unifikator σ einer endlichen Menge von Ausdrücken (Termen t_1, \dots, t_n , bzw. Formeln F_1, \dots, F_k) ist genau dann ein **allgemeinster Unifikator**, wenn er allgemeiner ist als alle Unifikatoren dieser Menge.

- Allgemeinster Unifikator σ einer Menge von Ausdrücken M :

$$\begin{array}{ccc} M & \xrightarrow{\sigma} & M\sigma \\ & \searrow \sigma' & \downarrow \tau \\ & & M\sigma\tau = M\sigma' \end{array}$$

- allgemeinster Unifikator – **most general unifier** – MGU
- Wenn σ ein allgemeinster Unifikator einer Menge $L = \{L_1, \dots, L_k\}$ von Literalen ist, dann werden alle Literale aus L durch σ auf $L_i\sigma$ abgebildet. Derartige $L_i\sigma$ sind die Formeln, die Ausgangspunkt der prädikatenlogischen Resolution sind.

Unifikationsatz

Satz 11.10

Jede endliche unifizierbare Menge von Literalen besitzt einen allgemeinsten Unifikator.

Konstruktiver Beweis

- Durch die Angabe eines **Unifikationsalgorithmus**
Die Idee:
Sukzessive Konstruktion eines Unifikators
 - Ausgangspunkt: $L = \{ L_1, \dots, L_n \}$
 - L ist noch nicht unifiziert, wenn $L\sigma = \{ L_1\sigma, \dots, L_n\sigma \}$ mehr als ein Element besitzt:
 - $|L\sigma| = 1$ bedeutet L wird durch σ unifiziert.
 - $|L\sigma| > 1$ bedeutet L wird durch σ nicht unifiziert.
 - Sukzessive Konstruktion von Substitutionen σ_i :
 - Jede zusätzliche Substitution unifiziert eine Abweichung zwischen Termen
 - $|L\sigma_1 \dots \sigma_i|$ wird immer kleiner. Ziel: $|L\sigma_1 \dots \sigma_i| = 1$.
- Unifikation bezieht sich stets auf endliche Mengen.

Unifikationsalgorithmus

Eingabe: Eine nicht-leere (endliche) Menge L von Literalen

$i := 0$; $\sigma_0 := []$; (\approx leere Substitution, jede Variable wird auf sich selbst abgebildet)

while $|L\sigma_i| > 1$ **do**

begin Suche in $L\sigma_i$ (von links nach rechts) die erste Position, in der sich mindestens zwei Literale L_j und L_k **unterscheiden**.

if Der Unterschied wird durch Junktor (Negation) oder Prädikatensymbole hervorgerufen

then stoppe mit der Ausgabe „nicht unifizierbar“

else if keiner der unterschiedlichen Terme ist eine Variable

then stoppe mit der Ausgabe „nicht unifizierbar“

else begin Sei x die Variable und t der andere Term

if x kommt in t (als echter Teilterm) vor

then stoppe mit der Ausgabe „nicht unifizierbar“

else $\sigma_{i+1} := \sigma_i [x/t]$; $i := i+1$;

end;

end;

Gib σ_i als allgemeinsten Unifikator aus!

Unifikationsalgorithmus (Beispiel 1)

$$\mathbf{L} = \{Q(x, f(a)), Q(b, y)\} \quad \sigma_0 = [] \quad |\mathbf{L}\sigma_0| = 2$$

$$\begin{array}{ll} 1. & L_1 = Q(x, f(a)) \quad L_2 = Q(b, y) \\ & t_1 = x \quad \uparrow \quad t_2 = b \quad \uparrow \quad \text{Paar unterschiedlicher Terme} \\ & \sigma_1 = [x/b] \end{array}$$

$$\begin{array}{ll} 2. & L_1\sigma_1 = Q(b, f(a)) \quad L_2\sigma_1 = Q(b, y) \quad |\mathbf{L}\sigma_1| = 2 \\ & t_1 = f(a) \quad \uparrow \quad t_2 = y \quad \uparrow \quad \text{Paar unterschiedlicher Terme} \\ & \sigma_2 = [x/b] [y/f(a)] \end{array}$$

$$3. \quad L_1\sigma_2 = Q(b, f(a)) \quad L_2\sigma_2 = Q(b, f(a)) \quad |\mathbf{L}\sigma_2| = 1$$

- ➔ Die Literalmenge \mathbf{L} wurde erfolgreich unifiziert.
- ➔ allgemeinsten Unifikator: $[x/b] [y/f(a)]$

Unifikationsalgorithmus (Beispiel 2)

$$\begin{array}{ll} 1. & \mathbf{L} = \{Q(x, f(a)), P(b, y)\} \quad \sigma_0 = [] \quad |\mathbf{L}\sigma_0| = 2 \\ & L_1 = Q(x, f(a)) \quad L_2 = P(b, y) \\ & \quad \quad \quad \uparrow \quad \quad \quad \uparrow \quad \text{unterschiedliche Prädikatensymbole} \end{array}$$

- ➔ Die Literalmenge \mathbf{L} ist nicht unifizierbar.

$$\begin{array}{ll} 2. & \mathbf{L} = \{Q(x, f(a)), Q(b, f(x))\} \\ & L_1 = Q(x, f(a)) \quad L_2 = Q(b, f(x)) \quad |\mathbf{L}\sigma_0| = 2 \\ & t_1 = x \quad \uparrow \quad t_2 = b \quad \uparrow \quad \text{Paar unterschiedlicher Terme} \\ & \sigma_1 = [x/b] \\ & L_1\sigma_1 = Q(b, f(a)) \quad L_2\sigma_1 = Q(b, f(b)) \quad |\mathbf{L}\sigma_1| = 2 \\ & t_1 = a \quad \uparrow \quad t_2 = b \quad \uparrow \quad \text{Paar unterschiedlicher Terme} \\ & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{aber variablenfrei.} \end{array}$$

- ➔ Die Literalmenge \mathbf{L} ist nicht unifizierbar.

Unifikationsalgorithmus (Beispiel 3)

$$\mathbf{L} = \{Q(x, x), Q(y, f(y))\}$$

$$L_1 = Q(x, x)$$

$$t_1 = x \quad \uparrow$$

$$\sigma_1 = [x/y]$$

$$L_1\sigma_1 = Q(y, y)$$

$$t_1 = y \quad \uparrow$$

$$L_2 = Q(y, f(y))$$

$$t_2 = y \quad \uparrow \quad \text{Paar unterschiedlicher Terme}$$

$$L_2\sigma_1 = Q(y, f(y))$$

$$t_2 = f(y) \quad \uparrow \quad \text{Paar unterschiedlicher Terme}$$

$$|\mathbf{L}\sigma_0| = 2$$

$$|\mathbf{L}\sigma_1| = 2$$

y kommt als echter Teilterm in $f(y)$ vor.

→ Die Literalmenge \mathbf{L} ist nicht unifizierbar.

Zum Selbststudium: Unifikationsalgorithmus (Beispiel 4)

$$\mathbf{L} = \{\neg P(f(z, g(a, y)), h(z)), \neg P(f(f(u, v), w), h(f(a, b)))\}$$

$$\sigma_0 = []$$

$$|\mathbf{L}\sigma_0| = 2$$

1. $L_1 = \neg P(f(z, g(a, y)), h(z))$

$$t_1 = z \quad \uparrow$$

$$\sigma_1 = [z/f(u, v)]$$

$$L_2 = \neg P(f(f(u, v), w), h(f(a, b)))$$

$$t_2 = f(u, v) \quad \uparrow$$

2. $L_1\sigma_1 = \neg P(f(f(u, v), g(a, y)), h(f(u, v)))$

$$t_1 = g(a, y) \quad \uparrow$$

$$\sigma_2 = [z/f(u, v)] [w/g(a, y)]$$

$$L_2\sigma_1 = \neg P(f(f(u, v), w), h(f(a, b)))$$

$$t_2 = w \quad \uparrow$$

3. $L_1\sigma_2 = \neg P(f(f(u, v), g(a, y)), h(f(u, v)))$

$$t_1 = u \quad \uparrow$$

$$\sigma_3 = [z/f(u, v)] [w/g(a, y)] [u/a]$$

$$L_2\sigma_2 = \neg P(f(f(u, v), g(a, y)), h(f(a, b)))$$

$$t_2 = a \quad \uparrow$$

4. $L_1\sigma_3 = \neg P(f(f(a, v), g(a, y)), h(f(a, v)))$

$$t_1 = v \quad \uparrow$$

$$\sigma_4 = [z/f(u, v)] [w/g(a, y)] [u/a] [v/b]$$

$$L_2\sigma_3 = \neg P(f(f(a, v), g(a, y)), h(f(a, b)))$$

$$t_2 = b \quad \uparrow$$

5. $L_1\sigma_4 = \neg P(f(f(a, b), g(a, y)), h(f(a, b)))$

$$|\mathbf{L}\sigma| = 1$$

$$L_2\sigma_4 = \neg P(f(f(a, b), g(a, y)), h(f(a, b)))$$

→ Die Literalmenge \mathbf{L} wurde erfolgreich unifiziert.

Unifikationsalgorithmus (Termination und Korrektheit)

Termination

- Eingabe: Eine endliche Menge $L = \{L_1, \dots, L_n\}$ von Literalen. Die Gesamtzahl der auftretenden Variablen ist dann auch endlich.
- In jedem Schritt wird
 - entweder eine Variable gefunden, die durch eine Substitution ersetzt wird (somit wird die Anzahl der Variablen in $L\sigma_i$ kleiner),
 - oder es wird ein Paar nicht unifizierbarer Literale gefunden, und somit das Verfahren abgebrochen.
- Die Zahl der Schleifendurchläufe ist also durch die Gesamtzahl der in L auftretenden Variablen beschränkt.

Korrektheit

- Die while-Schleife wird nur dann erfolgreich verlassen, wenn $|L\sigma_i| = 1$, d.h. wenn ein Unifikator gefunden ist.
- Noch zu zeigen
 - Die gebildete Substitution σ ist ein allgemeinsten Unifikator.
 - Wenn L unifizierbar ist, dann wird die while-Schleife erfolgreich durchlaufen

Zum Selbststudium: Algorithmus liefert allgemeinsten Unifikator

Satz 11.11

Wenn L eine unifizierbare Menge von Literalen ist, dann ist der durch den Unifikationsalgorithmus konstruierte Unifikator σ ein allgemeinsten Unifikator für L .
D.h.: Ist σ' ein Unifikator von L , dann gibt es eine Substitution τ mit: $\sigma' = \sigma \tau$

Hilfssatz 11.12 (Verallgemeinerung von 11.11)

Ist σ' ein Unifikator von L , dann gilt für jede im Unifikationsalgorithmus gebildete Substitution σ_i : $\sigma' = \sigma_i \sigma'$.

Beweis

Induktionsverankerung: $\sigma' = [] \sigma' = \sigma_0 \sigma'$

Induktionsvoraussetzung: Es sei σ_i eine Substitution mit $\sigma' = \sigma_i \sigma'$

Induktionsschritt: $\sigma_{i+1} := \sigma_i [x/t]$, wobei x eine Variable und t ein Term ist und x und t in korrespondierenden Positionen in Literalen von $L\sigma_i$ auftauchen, also (nach Induktionsvoraussetzung) durch σ' unifizierbar sind. Das heißt: $x\sigma' = t\sigma'$.

Für x ist $x [x/t] \sigma' = t \sigma' = x \sigma'$.

Für jede Variable $y \neq x$ ist $y [x/t] \sigma' = y \sigma'$, da x in y nicht vorkommt.

Also ist $[x/t] \sigma' = \sigma'$ und damit ist $\sigma_{i+1} \sigma' = \sigma_i [x/t] \sigma' = \sigma_i \sigma' = \sigma'$

Hilfssatz 11.13

Ist σ' ein Unifikator von L , dann wird die Schleife erfolgreich durchlaufen bis $|L\sigma_i| = 1$.

Beweis

Da in jedem Durchlauf $L\sigma_i$ durch σ' unifizierbar ist, kann der im Algorithmus gefundene Unterschied zwischen den Literalen nur in den Termen liegen.

Weiterhin muß mindestens einer der Terme eine Variable x sein (sonst haben wir abweichende Funktionssymbole / Konstanten und die Terme sind nicht unifizierbar oder wir haben den Unterschied noch nicht genau lokalisiert), den anderen Term nennen wir t . Weiterhin muss gelten: $x\sigma' = t\sigma'$.

Wäre nun x ein echter Teilterm von t , dann wäre $x\sigma'$ auch ein echter Teilterm von $t\sigma'$, die beiden also nicht identisch. Also kommt x nicht in t vor und der Algorithmus bricht auch bei der dritten Möglichkeit nicht ab.

Prädikatenlogische Resolution: Resolventenbildung

Definition 11.14

Seien K_1, K_2 und R prädikatenlogische Klauseln in Mengendarstellung. R heißt **prädikatenlogische Resolvente** von K_1 und K_2 ($\{K_1, K_2\} \vdash_{\text{res}} R$) falls gilt:

1. Es gibt Variablenumbenennungen (Substitutionen) σ_1 und σ_2 , so dass $K_1\sigma_1$ und $K_2\sigma_2$ keine gemeinsamen Variablen aufweisen.
2. Es gibt eine nicht leere Menge von Literalen $\{L_1, \dots, L_m\}$ aus $K_1\sigma_1$ und eine nicht leere Menge von Literalen $\{L'_1, \dots, L'_n\}$ aus $K_2\sigma_2$, so dass $\{\overline{L_1}, \dots, \overline{L_m}, L'_1, \dots, L'_n\}$ mit dem allgemeinsten Unifikator σ unifizierbar ist.
3. $R = (K_1\sigma_1 - \{L_1, \dots, L_m\}) \cup (K_2\sigma_2 - \{L'_1, \dots, L'_n\}) \sigma$

-
- Nach Konstruktion gilt auch: $R = (K_1\sigma_1\sigma \setminus \{L_1\}\sigma) \cup (K_2\sigma_2\sigma \setminus \{\overline{L_1}\}\sigma)$
 - Die aussagenlogische Resolventenbildung ist ein Spezialfall der prädikatenlogischen Resolventenbildung (da keine Variablen auftreten):

→ $\sigma_1 = \sigma_2 = \sigma = []$

→ $m = n = 1$

Prädikatenlogische Resolution – Beispiel zu 11.14 (1)

$$K_1 = \{P(f(x)), \neg Q(z), P(z)\}$$

$$K_2 = \{\neg P(x), R(g(x), a)\}$$

Auswahl von Literalen für die Resolution

$$\{P(f(x)), \neg Q(z), P(z)\}$$

$$\{\neg P(x), R(g(x), a)\}$$

$$\sigma_1 = []$$

$$\sigma_2 = [x/u]$$

$$\{P(f(x)), \neg Q(z), P(z)\}$$

$$\{\neg P(u), R(g(u), a)\}$$

$$\sigma = [z/f(x)][u/f(x)]$$

(Unifikation)

$$\{P(f(x)), \neg Q(f(x))\}$$

$$\{\neg P(f(x)), R(g(f(x)), a)\}$$

(Resolution)

$$\{\neg Q(f(x)), R(g(f(x)), a)\}$$

Prädikatenlogische Resolution – Beispiel zu 11.14 (2)

$$K_1 = \{P(f(x)), \neg Q(z), P(z)\}$$

$$K_2 = \{\neg P(x), R(g(x), a)\}$$

Auswahl von Literalen für die Resolution

$$\{P(f(x)), \neg Q(z), P(z)\}$$

$$\{\neg P(x), R(g(x), a)\}$$

$$\sigma_1 = []$$

$$\sigma_2 = [x/u]$$

$$\{P(f(x)), \neg Q(z), P(z)\}$$

$$\{\neg P(u), R(g(u), a)\}$$

$$\sigma = [u/z]$$

(Unifikation)

$$\{P(f(x)), \neg Q(z), P(z)\}$$

$$\{\neg P(z), R(g(z), a)\}$$

(Resolution)

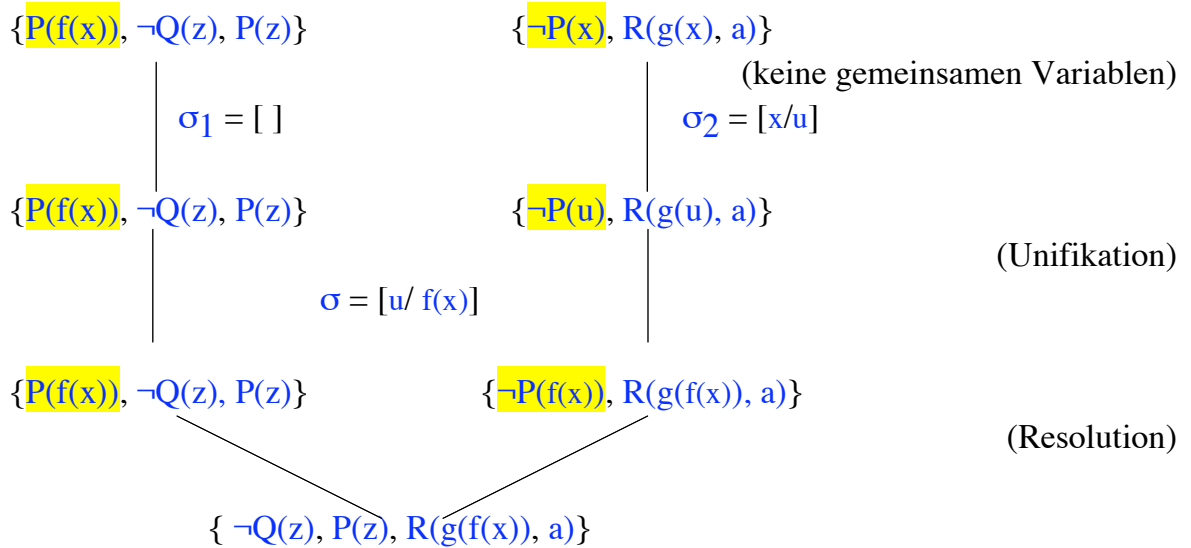
$$\{P(f(x)), \neg Q(z), R(g(z), a)\}$$

Prädikatenlogische Resolution – Beispiel zu 11.14 (3)

$$K_1 = \{P(f(x)), \neg Q(z), P(z)\}$$

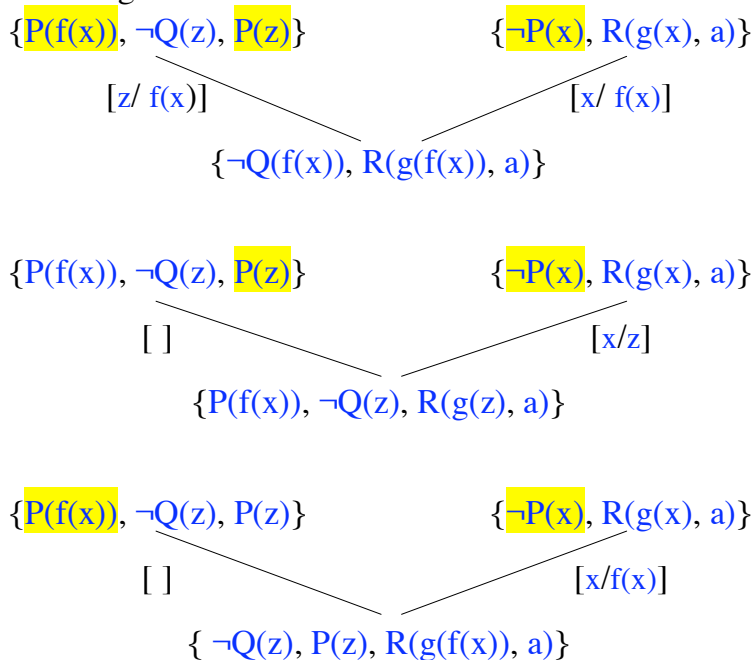
$$K_2 = \{\neg P(x), R(g(x), a)\}$$

Auswahl von Literalen für die Resolution



Kurzdarstellung der Beispiele zu 11.14

Anstelle der ausführlichen Resolutionsgraphen oben verwenden wir im folgenden die Kurzdarstellung:



Beobachtung zu Beispiel zu 11.14

Abgesehen von Varianten in der Variablenbenennung haben $K_1 = \{P(f(x)), \neg Q(z), P(z)\}$ und $K_2 = \{\neg P(x), R(g(x), a)\}$ drei prädikatenlogische Resolventen

- $\{\neg Q(f(x)), R(g(f(x)), a)\}$
- $\{\neg Q(z), P(z), R(g(f(x)), a)\}$
- $\{P(f(x)), \neg Q(z), R(g(z), a)\}$

(Welche Resolvente bei der Ableitung der leeren Klausel gebildet werden muss, hängt von den anderen beteiligten Klauseln ab.)

Satz 11.15

Seien K_1 und K_2 prädikatenlogische Klauseln in Mengendarstellung. Abgesehen von Variationen in der Variablenbenennung gibt es für K_1 und K_2 nur endlich viele prädikatenlogische Resolventen.

Beweisidee

Die Resolvente wird eindeutig durch die Klauseln und die Wahl der Literalmenge in Def. 11.14.2 bestimmt. Alternativen bei der Festlegung der beteiligten Substitutionen führen nur zu Variationen bei den Variablenbenennungen.

Zum Selbststudium: Prädikatenlogische Resolutionsableitung

Definition 11.16 (Resolventenmengen; vgl. 8.4)

Sei F eine Formel in Klauselnormalform in Mengendarstellung.

$$\text{Res}(F) := F \cup \{ R \mid R \text{ ist prädikatenlogische Resolvente zweier Klauseln in } F \}$$

Dies wird induktiv fortgesetzt durch:

$$\text{Res}^0(F) := F$$

$$\text{Res}^{n+1}(F) := \text{Res}(\text{Res}^n(F)) \quad n \geq 0$$

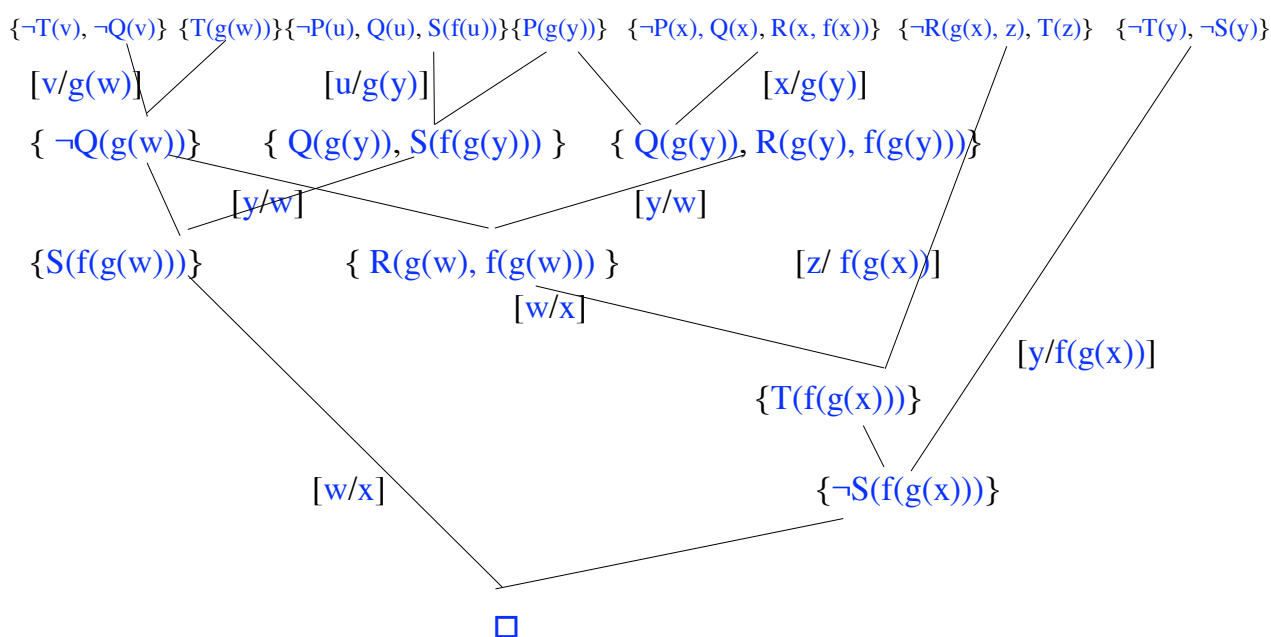
$$\text{Res}^*(F) := \bigcup_{n \geq 0} \text{Res}^n(F)$$

-
- $\square \in \text{Res}^*(F)$ GDW. Es gibt eine Folge von Klauseln K_1, K_2, \dots, K_n derart, dass
 - $K_n = \square$
 - Für $i = 1, \dots, n$ ist K_i entweder Element von F oder K_i ist Resolvente von K_{i1}, K_{i2} mit $i1, i2 < i$.

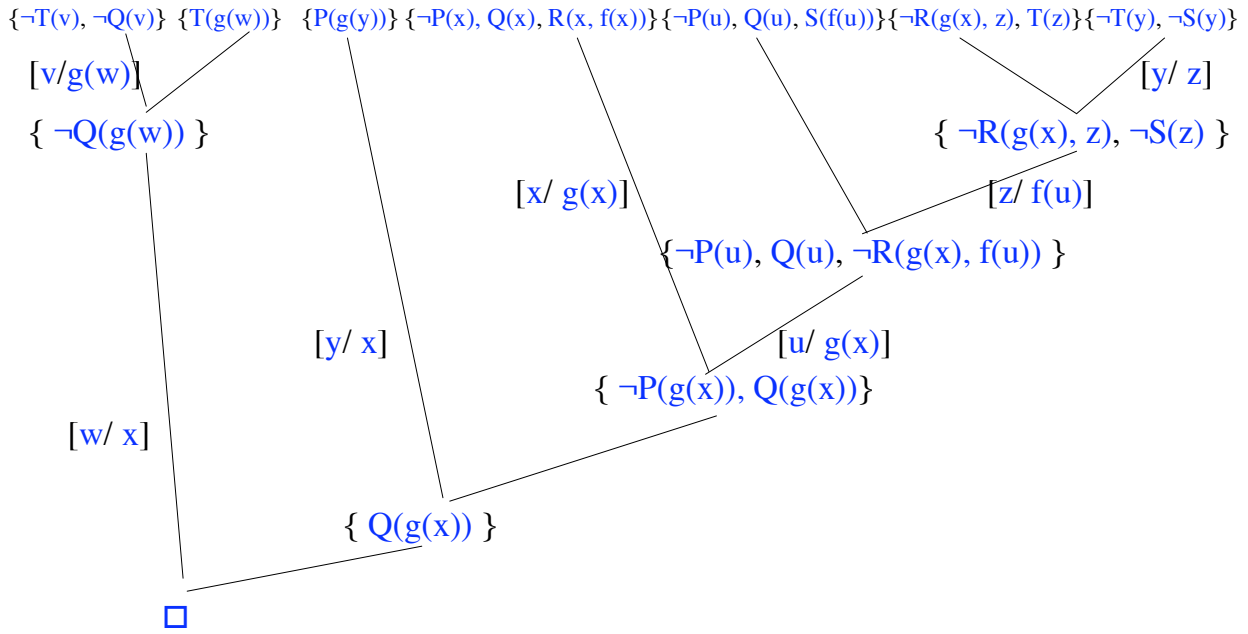
Zum Selbststudium: Prädikatenlogische Resolutionsableitung – Beispiel

$\{\neg P(x), Q(x), R(x, f(x))\}$	$\{\neg P(u), Q(u), S(f(u))\}$	$\{T(g(w))\}$	$\{P(g(y))\}$
$\{\neg R(g(x), z), T(z)\}$	$\{\neg T(v), \neg Q(v)\}$	$\{\neg T(y), \neg S(y)\}$	
(1) $\{T(g(w))\}$	in F		
(2) $\{\neg T(v), \neg Q(v)\}$	in F		
(3) $\{\neg Q(g(w))\}$	Resolvente	(1)	(2) $[v/g(w)]$
(4) $\{\neg P(u), Q(u), S(f(u))\}$	in F		
(5) $\{P(g(y))\}$	in F		
(6) $\{Q(g(y)), S(f(g(y)))\}$	Resolvente	(4) $[u/g(y)]$	(5)
(7) $\{S(f(g(w)))\}$	Resolvente	(3)	(6) $[y/w]$
(8) $\{\neg P(x), Q(x), R(x, f(x))\}$	in F		
(9) $\{Q(g(y)), R(g(y), f(g(y)))\}$	Resolvente	(5)	(8) $[x/g(y)]$
(10) $\{R(g(w), f(g(w)))\}$	Resolvente	(3)	(9) $[y/w]$
(11) $\{\neg R(g(x), z), T(z)\}$	in F		
(12) $\{T(f(g(x)))\}$	Resolvente	(10) $[w/x]$	(11) $[z/f(g(x))]$
(13) $\{\neg T(y), \neg S(y)\}$	in F		
(14) $\{\neg S(f(g(x)))\}$	Resolvente	(12)	(13) $[y/f(g(x))]$
(15) \square	Resolvente	(7) $[w/x]$	(14)

Zum Selbststudium: Resolution in der Prädikatenlogik



Resolution in der Prädikatenlogik – Resolutionsgraph

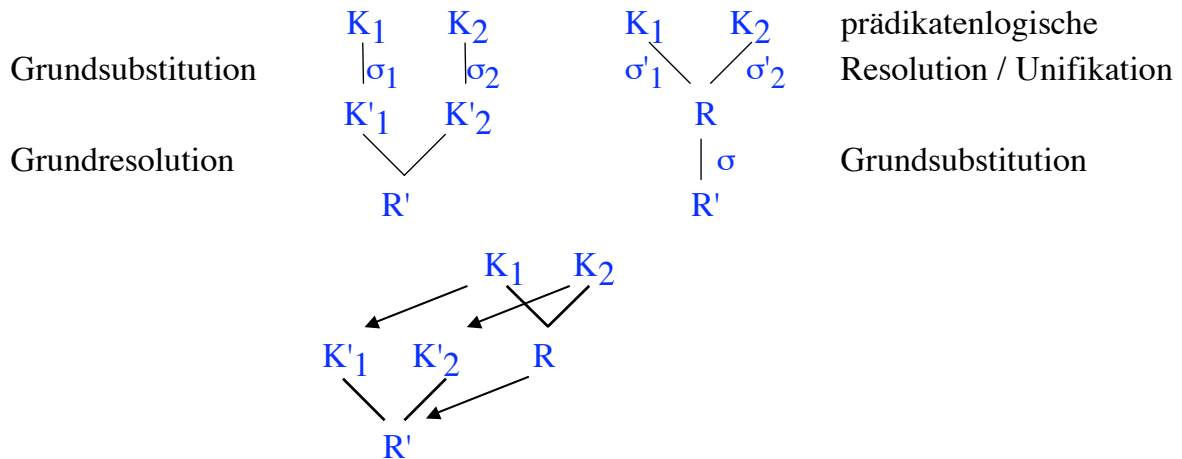


Zum Selbststudium: Lifting-Lemma

Satz 11.17 (Lifting Lemma)

Seien K_1 und K_2 zwei prädikatenlogische Klauseln und K'_1 und K'_2 zwei Grundinstanzen dieser Klauseln, die zu R' resolvierbar sind.

Dann gibt es eine prädikatenlogische Resolvente R von K_1 und K_2 , so dass R' Grundinstanz von R ist.



Beweis: siehe Schönig (Kap. 2.5)

Resolutionssatz der Prädikatenlogik

Satz 11.18 (Resolutionssatz)

Sei F eine Aussage in Klauselnormalform und \mathbf{F} die Mengendarstellung von F .
Dann gilt: F ist genau dann unerfüllbar, wenn $\square \in \text{Res}^*(\mathbf{F})$.

Beweisskizze (Details siehe Schöning)

Korrektheit

Wenn $\mathbf{F} \vdash_{\text{res}} \mathbf{R}$, dann $\mathbf{F} \models \{ \mathbf{R} \}$, der Beweis stützt sich wesentlich auf:

Wenn $\{ \mathbf{K}_1, \mathbf{K}_2 \} \vdash_{\text{res}} \mathbf{R}$, dann $\{ \mathbf{K}_1, \mathbf{K}_2 \} \models \{ \mathbf{R} \}$

- Hier ist zu berücksichtigen:
 - die implizite Allquantifizierung von Variablen in Klauseln
 - die Unifikation (der Terme bzw. Literale)

Der Rest ergibt sich dann durch einen einfachen Schluss:

➔ Also: Wenn $\square \in \text{Res}^*(\mathbf{F})$, dann $\mathbf{F} \models \perp$, also ist dann F eine Kontradiktion.

Resolutionssatz der Prädikatenlogik [Fortsetzung]

Vollständigkeit

Sei F unerfüllbar

- Der Grundresolutionssatz sichert die Existenz einer Folge von Klauseln K'_1, K'_2, \dots, K'_n , so dass
 - K'_1, K'_2, \dots, K'_n ist eine aussagenlogische Resolutionsableitung
 - $K'_n = \square$
- Zu dieser Folge von Grundinstanzklauseln existiert eine korrespondierende Folge von prädikatenlogischen Klauseln, die wir von $i = 1$ ausgehend konstruieren:
 - Falls K'_i eine Grundinstanz zu einer Klausel $K \in F^*$ ist, wähle $K_i = K$.
 - Falls K'_i die Resolvente von K'_{i1} und K'_{i2} ($i1, i2 < i$) ist, so sind für K'_{i1} und K'_{i2} schon korrespondierende prädikatenlogische Klauseln, K_{i1} und K_{i2} festgelegt. Aufgrund des Lifting-Lemmas gibt es eine prädikatenlogische Resolvente R , die K'_i als Grundinstanz besitzt. Wir legen fest: $K_i = R$.

➔ K_1, K_2, \dots, K_n ist eine prädikatenlogische Resolutionsableitung, mit $K_n = \square$.

Übersicht: Resolution

- Aussagenlogische Resolution
 - Prädikatenlogische Resolution
 - ➔ Grundresolution: Bildung von Grundinstanzen + aussagenlogische Resolution
 - ➔ Prädikatenlogische Resolution durch Unifikation
 - Vollständigkeit der Grundresolution und der prädikatenlogischen Resolution
-
- Verfeinerung der Resolution
 - Restriktionen
 - Strategien
 - können vom aussagenlogischen Fall in den prädikatenlogischen Fall übertragen werden
 - ➔ Vollständigkeit muss bewiesen werden (Modifikation des Resolutionsatzes)