technische universität dortmund

Thomas Schwentick
Gaetano Geck Martin Schuster

Lehrstuhl 1
Logik in der Informatik

WS 2016/17 Übungsblatt 6 16.1.2017

Aufgabe 6.1 [Prädikatenlogische Resolution] **6 Punkte**

Zeigen Sie durch prädikatenlogische Resolution, dass die Formel

$$\varphi = \exists w \forall x \exists y \forall z \Big[\big(\neg T(x) \vee S(x)\big) \wedge \big(\neg(A(x,z) \wedge T(z)) \vee T(x)\big) \wedge \big(F(x) \vee (A(y,x) \wedge \neg S(y))\big) \wedge (T(w) \wedge \neg F(w)) \Big]$$

unerfüllbar ist.

Lösungsvorschlag:

Die Formel φ befindet sich in geschlossener Pränexform. Nun bringen wir φ erst in Skolem- und dann in Matrixklauselform. Dazu wenden wir die Skolemregeln von links nach rechts an:

1. Die Variable w wird durch den ersten Existenzquantor quantifiziert; diesen entfernen wir und ersetzen w durch die Konstante a:

   $$\forall x \exists y \forall z \Big[\big(\neg T(x) \vee S(x)\big) \wedge \big(\neg(A(x,z) \wedge T(z)) \vee T(x)\big) \wedge \big(F(x) \vee (A(y,x) \wedge \neg S(y))\big) \wedge \big(T(a) \wedge \neg F(a)\big) \Big]$$

2. Der nächste Existenzquantor steht nun an 2. Stelle. Daher wird y durch die 1-stellige Funktion g ersetzt:

   $$\forall x \forall z \Big[\big(\neg T(x) \vee S(x)\big) \wedge \big(\neg(A(x,z) \wedge T(z)) \vee T(x)\big) \wedge \big(F(x) \vee (A(g(x),x) \wedge \neg S(g(x)))\big) \wedge \big(T(a) \wedge \neg F(a)\big) \Big]$$

Die resultierende Formel befindet sich in Skolemform und muss nun noch in Matrixklauselform gebracht werden. Dazu wird die quantorenfreie Teilformel

$$\big(\neg T(x) \vee S(x)\big) \wedge \big(\neg(A(x,z) \wedge T(z)) \vee T(x)\big) \wedge \big(F(x) \vee (A(g(x),x) \wedge \neg S(g(x)))\big) \wedge \big(T(a) \wedge \neg F(a)\big)$$

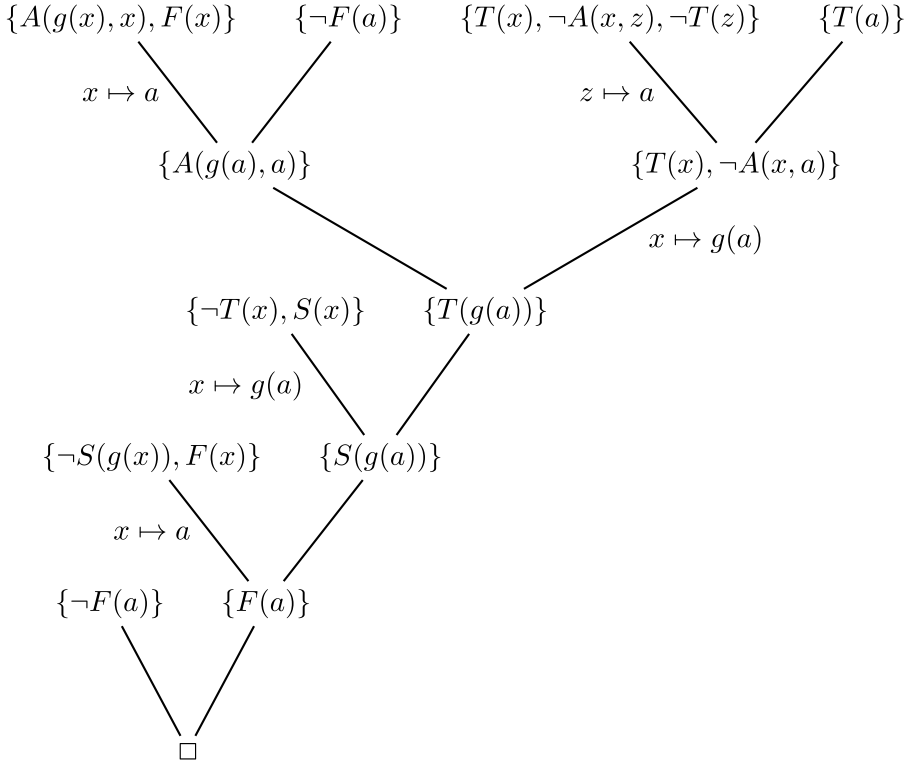
in KNF gebracht. Nach der Regel von DeMorgan und durch Distribution ergibt sich die Formel φ':

$$\big(\neg T(x) \vee S(x)\big) \wedge \big(\neg A(x,z) \vee \neg T(z) \vee T(x)\big) \wedge \big(F(x) \vee A(g(x),x)\big) \wedge \big(F(x) \vee \neg S(g(x))\big) \wedge T(a) \wedge \neg F(a)$$

Als Matrixklauselform $\mathcal{K}(\varphi')$ ergibt sich somit

$$\Big\{ \{\neg T(x), S(x)\}, \{\neg A(x,z), \neg T(z), T(x)\}, \{F(x), A(g(x),x)\}, \{F(x), \neg S(g(x))\}, \{T(a)\}, \{\neg F(a))\} \Big\}$$

Wir zeigen die Unerfüllbarkeit von $\mathcal{K}(\varphi')$ durch die prädikatenlogische Resolution:



Da die leere Klausel resolviert werden kann, ist φ' und, aufgrund der Erfüllbarkeitsäquivalenz auch Formel φ, unerfüllbar.

Aufgabe 6.2 [Resolutionsstrategien] **4 Punkte**

a) Geben Sie eine möglichst kleine unerfüllbare prädikatenlogische Klauselmenge $\mathcal{K}$ an, welche die Unvollständigkeit der Resolution mit Einheits-Restriktion beweist. Begründen Sie, warum die leere Klausel $\square$ durch Resolution mit Einheits-Restriktion nicht aus $\mathcal{K}$ abgeleitet werden kann. Beweisen Sie die Unerfüllbarkeit der Klauselmenge durch Resolution mit N-Restriktion. **(2 Punkte)**

b) Beweisen Sie die Unerfüllbarkeit der Formelmenge

$$\mathcal{K} = \big\{\{R(x), \neg T(x,y)\}, \{R(x), S(a,y), T(x,y)\}, \{\neg R(x), \neg S(a,y), \neg T(x,y)\},\\ \{\neg R(x), S(a,y)\}, \{\neg S(a,y), T(x,y)\}\big\}$$

durch Resolution mit linearer Restriktion. **(2 Punkte)**

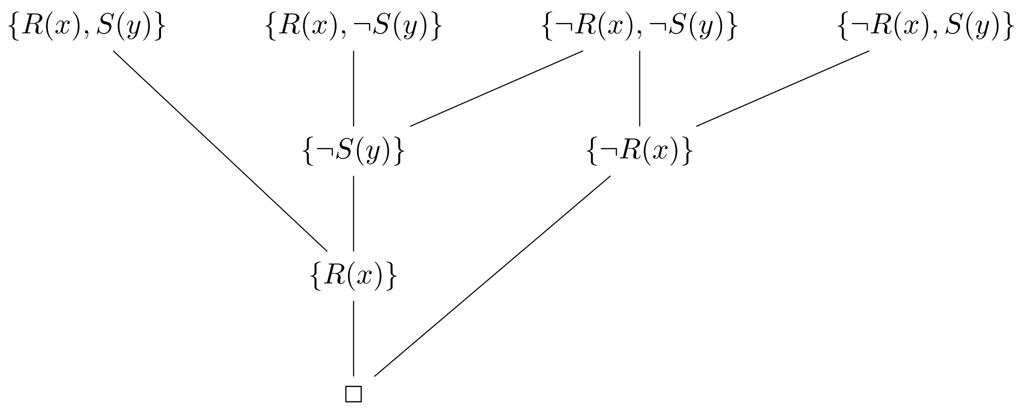
Lösungsvorschlag:

a) Eine mögliche Klauselmenge ist

$$\big\{\{R(x), S(y)\}, \{R(x), \neg S(y)\}, \{\neg R(x), \neg S(y)\}, \{\neg R(x), S(y)\}\big\},$$

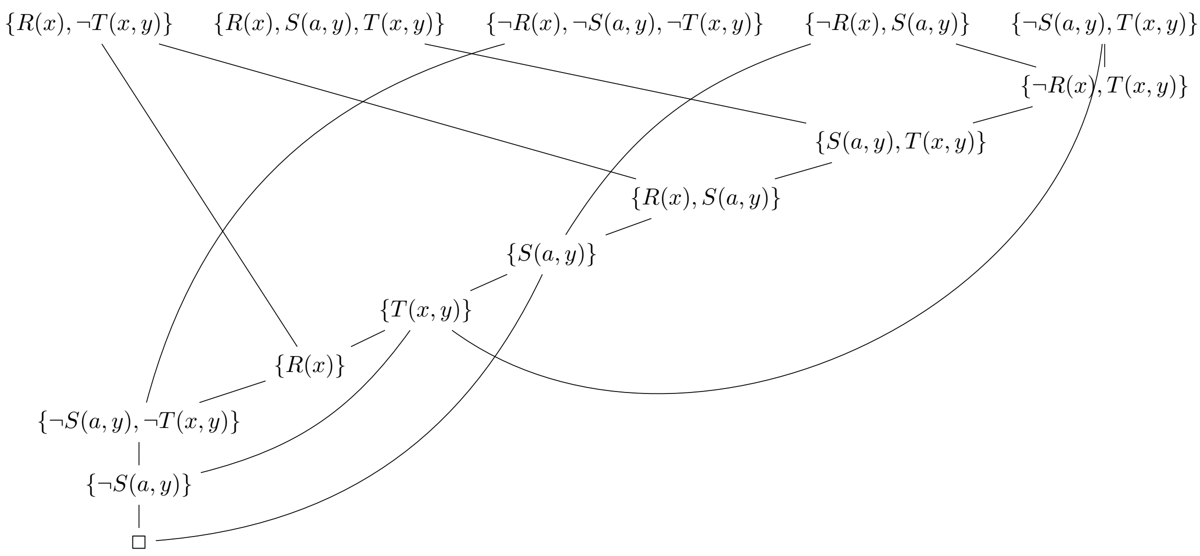
Offenbar steht in $\mathcal{K}$ keine Einheitsklausel für einen Resolutionsschritt zur Verfügung, sodass durch Resolution mit Einheits-Restriktion keine Resolventen gebildet werden können.

Durch Resolution mit N-Restriktion kann jedoch die leere Klausel abgeleitet werden:



Offensichtlich ist an jedem Resolutionsschritt eine negative Klausel beteiligt, sodass die N-Restriktion respektiert wird. Da die leere Klausel abgeleitet werden kann, folgt die Unerfüllbarkeit der Formelmenge.

b) Die folgende Resolution beachtet die lineare Restriktion:



Offenbar ist an jedem Resolutionsschritt nach dem ersten die zuletzt abgeleitete Resolvente beteiligt, sodass es sich um eine Resolution mit linearer Restriktion handelt. Da die leere Klausel abgeleitet werden kann, folgt die Unerfüllbarkeit der Formelmenge.

Aufgabe 6.3 [Tims Stammbaum] **10 Punkte**

Die vorgegebenen Programmabschnitte sind in der Datei `stammbaum.pl` zusammengefaasst. Mit dem SWI-Prolog-Interpreter (`http://www.swi-prolog.org`) können Sie die Anfragen beantworten lassen und Ihren eigenen Code testen.

Der kleine Tim hat seiner Oma beim Weihnachtsessen versprochen ihr dabei zu helfen, den Stammbaum der Familie auf den neuesten Stand zu bringen. Nachdem Tim uns das ganze Semester über begleitet hat, weiß er, dass eine Modellierung mit Hilfe von Logik viele Vorteile hat. Diese Vorteile möchte er nutzen, und schreibt deshalb sein Wissen über den Stammbaum in einem Prolog-Programm $\mathcal{K}$ auf. Das Programm $\mathcal{K}$ enthält die Fakten:

```
kind_von(tim, anna).                                   w(anna).
kind_von(anna, fanny).                                 w(fanny).
kind_von(daniel, fanny).                               w(celine).
kind_von(celine, gertrude).                            w(gertrude).
kind_von(tim, bernd).                                  m(bernd).
kind_von(anna, ephraim).                               m(daniel).
kind_von(daniel, ephraim).                             m(ephraim).
kind_von(celine, daniel).                              m(tim).
```

Dabei soll ein Fakt `kind_von(X, Y)` bedeuten, dass `X` ein Kind von `Y` ist und Fakten `w(X)` und `m(X)` sollen bedeuten, dass `X` weiblich bzw. männlich ist.
Zusätzlich enthält $\mathcal{K}$ die folgenden Regeln, mit denen sich neue Beziehungen und Eigenschaften ableiten lassen:

```
vater_von(X,Y)     :- m(X), kind_von(Y,X).
oma(X)             :- w(X), kind_von(Z,Y), kind_von(Y,X).
eltern_von(X,Y,Z) :- kind_von(Z,X), kind_von(Z,Y), w(X), m(Y).
```

a) Geben Sie zum Prolog-Programm $\mathcal{K}$ korrespondierende prädikatenlogische Formeln an; wählen Sie dazu insbesondere eine Signatur. Beschränken Sie sich bei der Angabe der Formeln für die `w`-, `m`- und `kind_von`-Fakten auf jeweils 2 Beispiele. **(3 Punkte)**

b) Geben Sie eine erfolgreiche Berechnung für $\mathcal{K}$ und die Anfrage `eltern_von(X, Y, tim)` an. Geben Sie dazu zunächst die dem Progamm entsprechende Klauselmenge an. Geben Sie die Klauseln zu allen Regeln an, beschränken Sie sich bei den Klauseln zu den `w`-, `m`- und `kind_von`-Fakten aber auf jene, die für die Berechnung benötigt werden. **(3 Punkte)**

c) Ergänzen Sie das Prolog-Programm $\mathcal{K}$ um folgende Regeln:

- `geschwister(X,Y)`: Gilt wenn verschiedene Kinder `X` und `Y` sowohl dieselbe Mutter als auch denselben Vater haben.
- `schwester_von(X,Y)`: Gilt wenn `X` Schwester von `Y` ist.
- `tante_von(X,Y)`: Gilt wenn `X` Tante von `Y` ist.

Hinweis: Um sicherzustellen, dass zwei Variablen unterschiedliche Personen enthalten, dürfen Sie das Prolog-Prädikat `\=(X,Y)` verwenden. Dieses ist genau dann wahr, wenn `X` und `Y` zum Auswertungszeitpunkt nicht unifizierbar sind.

(2 Punkte)

d) Tim hat in den Folien der Logik-Vorlesung geschmökert und gelernt, wie ein Prolog-Programm aussehen kann, das testet, ob `X` ein Nachfahre von `Y` ist:

```
nachfahre_von(X,Y) :- kind(X,Y).
nachfahre_von(X,Y) :- kind(X,Z), nachfahre_von(Z,Y).
```

Der kleine Tim weiß, dass das für seine Oma nicht gut genug ist. Er möchte ein Prolog-Programm `nachfahre_von(X,Y,V)` schreiben, mit der intuitiven Bedeutung, dass `X` ein Nachfahre von `Y` ist und `V` das Verwandschaftsverhältnis von `X` und `Y` beschreibt. Zur Repräsentation des Verwandschaftsverhältnisses sollen die zusätzlichen Funktionssymbol `k`, `e` und `u` mit folgender Bedeutung verwendet werden:

- falls `X` Kind von `Y` ist, gilt `nachfahre_von(X,Y,k(Y))`;
- falls `X` Enkel von `Y` ist, gilt `nachfahre_von(X,Y,e(Y))`;
- falls `X` Urenkel von `Y` ist, gilt `nachfahre_von(X,Y,u(e(Y)))`;
- falls `X` Ur-...-urenkel von `Y` ist, gilt `nachfahre_von(X,Y,u(...u(e(Y))...))`.

Helfen Sie dem kleinen Tim, indem Sie ein Prolog-Programm für `nachfahre_von(X,Y,Z)` angeben.

Hinweis: Möglicherweise hilft die Methode aus dem Affe-Stuhl-Banane-Beispiel der Vorlesung. **(2 Punkte)**

Lösungsvorschlag:

a) Wir verwenden die Relationssymbole

- M für `m`,
- W für `w`,
- K für `kind_von`,
- V für `vater_von`,
- O für `oma`, und
- E für `eltern_von`

sowie die Konstantensymbole a für `anna`, b für `bernd`, c für `celine`, d für `daniel`, e für `ephraim`, f für `fanny`, g für `gertrude` und t für `tim`.

Dann ergibt sich für die einzelnen Fakten:

- `w(anna).`, `w(fanny).`, `w(celine).` und `w(gertrude).` korrespondieren zu $W(a)$, $W(f)$, $W(c)$ und $W(g)$
- `m(bernd).`, `m(daniel).`, `m(ephraim).` und `m(tim).` korrespondieren zu $M(b)$, $M(d)$,$M(e)$ und $M(t)$,
- `kind_von(tim, anna).` korrespondiert zu $K(t, a)$
- `kind_von(anna, fanny).` korrespondiert zu $K(a, f)$
- `kind_von(daniel, fanny).` korrespondiert zu $K(d, f)$
- `kind_von(celine, gertrude).` korrespondiert zu $K(c, g)$
- `kind_von(tim, bernd).` korrespondiert zu $K(t, b)$

- `kind_von(anna, ephraim).` korrespondiert zu $K(a, e)$
- `kind_von(daniel, ephraim).` korrespondiert zu $K(d, e)$
- `kind_von(celine, daniel).` korrespondiert zu $K(c, d)$

Die Regeln entsprechen den folgenden prädikatenlogischen Formeln:

- `vater_von(X,Y) :- m(X), kind_von(Y,X).` korrespondiert zu
$$\forall x \forall y \big((M(x) \wedge K(y, x)) \rightarrow V(x, y)\big).$$

- `oma(X) :- w(X), kind_von(Z,Y), kind_von(Y,X).` korrespondiert zu
$$\forall x \Big(\big(\exists y \exists z (W(x) \wedge K(z, y) \wedge K(y, x)) \rightarrow O(x)\big)\Big).$$

- `eltern_von(X,Y,Z) :- kind_von(Z,X), kind_von(Z,Y), w(X), m(Y).` korrespondiert zu
$$\forall x \forall y \forall z ((K(z, x) \wedge K(z, y) \wedge W(x) \wedge M(y)) \rightarrow E(x, y, z)).$$

Somit ergibt sich insgesamt als zu $\mathcal{K}$ korrespondierende prädikatenlogischen Formel $\varphi_{\mathcal{K}}$:

$$\begin{array}{rcl}
\varphi_{\mathcal{K}} & = & W(a) \wedge \cdots \wedge W(g) \wedge M(b) \wedge \cdots \wedge M(t) \\
& \wedge & K(t, a) \wedge \cdots \wedge K(c, d) \\
& \wedge & \forall x \forall y \big((M(x) \wedge K(y, x)) \rightarrow V(x, y)\big) \\
& \wedge & \forall x \Big(\big(\exists y \exists z (W(x) \wedge K(z, y) \wedge K(y, x)) \rightarrow O(x)\big)\Big) \\
& \wedge & \forall x \forall y \forall z ((K(z, x) \wedge K(z, y) \wedge W(x) \wedge M(y)) \rightarrow E(x, y, z))
\end{array}$$

b) Bevor wir die Anfrage `eltern_von(X, Y, tim)` auswerten, formen wir die Formeln

$$\begin{array}{rcl}
\varphi_{\mathcal{K}} & = & W(a) \wedge \cdots \wedge W(g) \wedge M(b) \wedge \cdots \wedge M(t) \\
& \wedge & K(t, a) \wedge \cdots \wedge K(c, d) \\
& \wedge & \forall x \forall y \big((M(x) \wedge K(y, x)) \rightarrow V(x, y)\big) \\
& \wedge & \forall x \Big(\big(\exists y \exists z (W(x) \wedge K(z, y) \wedge K(y, x)) \rightarrow O(x)\big)\Big) \\
& \wedge & \forall x \forall y \forall z ((K(z, x) \wedge K(z, y) \wedge W(x) \wedge M(y)) \rightarrow E(x, y, z))
\end{array}$$

in Matrixklauselform um. Dazu formen wir $\varphi_{\mathcal{K}}$ zunächst in eine erfüllbarkeitsäquivalente Skolemform um:

$$\begin{array}{rcl}
\varphi_{\mathcal{K}} & \equiv & W(a) \wedge \cdots \wedge W(g) \wedge M(b) \wedge \cdots \wedge M(t) \\
& \wedge & K(t, a) \wedge \cdots \wedge K(c, d) \\
& \wedge & \forall x \forall y \big(\neg W(x) \vee \neg K(x, y) \vee T(x, y)\big) \\
& \wedge & \forall x \forall y \forall z \big(\neg K(x, z) \vee \neg K(z, y) \vee N(x, y)\big) \\
& \wedge & \forall x \forall y \forall z \big(\neg K(z, x) \vee \neg K(z, y) \vee \neg W(x) \vee \neg M(y) \vee E(x, y, z)\big) \\
& \equiv & \forall x \forall y \forall z \Big(W(a) \wedge \cdots \wedge W(g) \wedge M(b) \wedge \cdots \wedge M(t) \\
& \wedge & K(t, a) \wedge \cdots \wedge K(c, d) \\
& \wedge & \big(\neg M(x) \vee \neg K(y, x) \vee V(x, y)\big) \\
& \wedge & \big(\neg W(x) \vee \neg K(z, y) \vee \neg K(y, x) \vee O(x)\big) \\
& \wedge & \big(\neg K(z, x) \vee \neg K(z, y) \vee \neg W(x) \vee \neg M(y) \vee E(x, y, z)\big)\Big)
\end{array}$$

Als Matrixklauselform ergibt sich:

$$\begin{aligned}\{\quad & \{W(a)\}, \{W(f)\}, \{W(c)\}\{W(g)\}, \{M(b)\}, \{M(d)\}, \{M(e)\}, \{M(t)\}, \\ & \{K(t,a)\}, \{K(a,f)\}, \{K(d,f)\}, \{K(c,g)\}, \{K(t,b)\}, \{K(a,e)\}, \{K(d,e)\}, \{K(c,d)\}, \\ & \{\neg M(x), \neg K(y,x), V(x,y)\} \\ & \{\neg W(x), \neg K(z,y), \neg K(y,x), O(x)\} \\ & \{\neg K(z,x), \neg K(z,y), \neg W(x), \neg M(y), E(x,y,z)\} \quad \}\end{aligned}$$

Aus der Anfrage `eltern_von(X, Y, tim)` ergibt sich die Zielklausel $\{\neg E(x,y,t)\}$. Eine erfolgreiche Berechnung ist:

$$\begin{aligned} & (\{\neg E(x,y,t)\}, []) \\ \vdash_{\mathcal{K}} \quad & (\{\neg K(t,x), \neg K(t,y), \neg W(x), \neg M(y)\}, []) \\ \vdash_{\mathcal{K}} \quad & (\{\neg K(t,y), \neg W(a), \neg M(y)\}, [x \mapsto a]) \\ \vdash_{\mathcal{K}} \quad & (\{\neg W(a), \neg M(b)\}, [x \mapsto a, y \mapsto b]) \\ \vdash_{\mathcal{K}} \quad & (\{\neg M(b)\}, [x \mapsto a, y \mapsto b]) \\ \vdash_{\mathcal{K}} \quad & (\{\}, [x \mapsto a, y \mapsto b])\end{aligned}$$

Somit sind Anna und Bernd die Eltern von Tim.

c) Folgende Regeln entsprechen den Anforderungen:

- `X` und `Y` haben dieselben Eltern und sind voneinader verschieden:
  `geschwister(X,Y) :- eltern_von(M,V,X), eltern_von(M,V,Y), \=(X,Y).`

- Eine Schwester ist ein weibliches Geschisterkind:
  `schwester_von(X,Y) :- w(X), geschwister(X,Y).`

- `X` ist Tante von `Y` wenn sie die Schwester von `Y`s Mutter oder von `Y`s Vater ist:
  `tante_von(X,Y) :- kind_von(Y,Z), schwester_von(X,Z).`