Lógica matemática y fundamentos (2013–14)

Tema 12: Resolución en lógica de primer orden

José A. Alonso Jiménez
María J. Hidalgo Doblado

Grupo de Lógica Computacional
Departamento de Ciencias de la Computación e I.A.
Universidad de Sevilla

Tema 12: Resolución en lógica de primer orden

Tema 12: Resolución en lógica de primer orden

Ejemplos de consecuencia mediante resolución

- Ejemplo 1:

  $$\{\forall x\ [P(x) \to Q(x)], \exists x\ P(x)\} \models \exists x\ Q(x)$$

  se reduce a

  $\{\{\neg P(x), Q(x)\}, \{P(a)\}, \{\neg Q(z)\}\}$ es inconsistente.

  Demostración:

1	$\{\neg P(x), Q(x)\}$	Hipótesis
2	$\{P(a)\}$	Hipótesis
3	$\{\neg Q(z)\}$	Hipótesis
4	$\{Q(a)\}$	Resolvente de 1 y 2 con $\sigma = [x/a]$
5	$\square$	Resolvente de 3 y 4 con $\sigma = [z/a]$

Ejemplos de consecuencia mediante resolución

- Ejemplo 2:

  $$\{\forall x\,[P(x) \to Q(x)], \forall x\,[Q(x) \to R(x)]\} \models \forall x\,[P(x) \to R(x)]$$

  se reduce a

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

  es inconsistente.

  Demostración:

1	$\{\neg P(x), Q(x)\}$	Hipótesis
2	$\{\neg Q(y), R(y)\}$	Hipótesis
3	$\{P(a)\}$	Hipótesis
4	$\{\neg R(a)\}$	Hipótesis
5	$\{Q(a)\}$	Resolvente de 1 y 3 con $\sigma = [x/a]$
6	$\{R(a)\}$	Resolvente de 2 y 5 con $\sigma = [y/a]$
5	$\square$	Resolvente de 4 y 6 con $\sigma = \epsilon$

Tema 12: Resolución en lógica de primer orden

Unificadores

- Def.: La sustitución σ es un unificador de los términos t_1 y t_2 si $t_1\sigma = t_2\sigma$.
- Def.: Los términos t_1 y t_2 son unificables si tienen algún unificador.
- Def.: t es una instancia común de t_1 y t_2 si existe una sustitución σ tal que $t = t_1\sigma = t_2\sigma$.
- Ejemplos:

t_1	t_2	Unificador	Instancia común
$f(x, g(z))$	$f(g(y), x)$	$[x/g(z), y/z]$	$f(g(z), g(z))$
$f(x, g(z))$	$f(g(y), x)$	$[x/g(y), z/y]$	$f(g(y), g(y))$
$f(x, g(z))$	$f(g(y), x)$	$[x/g(a), y/a]$	$f(g(a), g(a))$
$f(x, y)$	$f(y, x)$	$[x/a, y/a]$	$f(a, a)$
$f(x, y)$	$f(y, x)$	$[y/x]$	$f(x, x)$
$f(x, y)$	$g(a, b)$	No tiene	No tiene
$f(x, x)$	$f(a, b)$	No tiene	No tiene
$f(x)$	$f(g(x))$	No tiene	No tiene

- Nota: Las anteriores definiciones se extienden a conjuntos de términos y de literales.

Composición de sustituciones e identidad

- Composición de sustituciones:
  - Def.: La composición de las sustituciones σ_1 y σ_2 es la sustitución $\sigma_1\sigma_2$ definida por $x(\sigma_1\sigma_2) = (x\sigma_1)\sigma_2$, para toda variable x.
  - Ejemplo: Si $\sigma_1 = [x/f(z,a), y/w]$ y $\sigma_2 = [x/b, z/g(w)]$, entonces
    - $x\sigma_1\sigma_2 = (x\sigma_1)\sigma_2 = f(z,a)\sigma_2 = f(z\sigma_2, a\sigma_2) = f(g(w),a)$
    - $y\sigma_1\sigma_2 = (y\sigma_1)\sigma_2 = w\sigma_2 = w$
    - $z\sigma_1\sigma_2 = (z\sigma_1)\sigma_2 = z\sigma_2 = g(w)$
    - $w\sigma_1\sigma_2 = (w\sigma_1)\sigma_2 = w\sigma_2 = w$

    Por tanto, $\sigma_1\sigma_2 = [x/f(g(w),a), y/w, z/g(w)]$.
- Def.: La substitución identidad es la sustitución ϵ tal que, para todo x, $x\epsilon = x$.
- Propiedades:
  1. Asociativa: $\sigma_1(\sigma_2\sigma_3) = (\sigma_1\sigma_2)\sigma_3$
  2. Neutro: $\sigma\epsilon = \epsilon\sigma = \sigma$.

Comparación de sustituciones

- Def.: La sustitución σ_1 es más general que la σ_2 si existe una sustitución σ_3 tal que $\sigma_2 = \sigma_1\sigma_3$. Se representa por $\sigma_2 \leq \sigma_1$.
- Def.: Las sustituciones σ_1 y σ_2 son equivalentes si $\sigma_1 \leq \sigma_2$ y $\sigma_2 \leq \sigma_1$. Se representa por $\sigma_1 \equiv \sigma_2$.
- Ejemplos: Sean $\sigma_1 = [x/g(z), y/z], \sigma_2 = [x/g(y), z/y]$ y $\sigma_3 = [x/g(a), y/a]$. Entonces,
  1. $\sigma_1 = \sigma_2[y/z]$
  2. $\sigma_2 = \sigma_1[z/y]$
  3. $\sigma_3 = \sigma_1[z/a]$
  4. $\sigma_1 \equiv \sigma_2$
  5. $\sigma_3 \leq \sigma_1$
- Ejemplo: $[x/a, y/a] \leq [y/x]$, ya que $[x/a, y/a] = [y/x][x/a, y/a]$.

Unificador de máxima generalidad

- Def.: La sustitución σ es un unificador de máxima generalidad (UMG) de los términos t_1 y t_2 si
  - σ es un unificador de t_1 y t_2.
  - σ es más general que cualquier unificador de t_1 y t_2.
- Ejemplos:
  1. $[x/g(z), y/z]$ es un UMG de $f(x, g(z))$ y $f(g(y), x)$.
  2. $[x/g(y), z/y]$ es un UMG de $f(x, g(z))$ y $f(g(y), x)$.
  3. $[x/g(a), y/a]$ no es un UMG de $f(x, g(z))$ y $f(g(y), x)$.
- Nota: Las anterior definición se extienden a conjuntos de términos y de literales.

Unificación de listas de términos

- Notación de lista:
  - $(a_1, \dots, a_n)$ representa una lista cuyos elementos son $a_1, \dots, a_n$.
  - $(a|R)$ representa una lista cuyo primer elemento es a y resto es R.
  - $()$ representa la lista vacía.
- Unificadores de listas de términos:
  - Def.: σ es un unificador de $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ si $s_1\sigma = t_1\sigma, \dots, s_n\sigma = t_n\sigma$.
  - Def.: $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ son unificables si tienen algún unificador.
  - Def.: σ es un unificador de máxima generalidad (UMG) de $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ si σ es un unificador de $(s_1 \dots, s_n)$ y $(t_1 \dots, t_n)$ más general que cualquier otro.
- Aplicación de una sustitución a una lista de ecuaciones:
  - $(s_1 = t_1, \dots, s_n = t_n)\sigma = (s_1\sigma = t_1\sigma, \dots, s_n\sigma = t_n\sigma)$.

Algoritmo de unificación de listas de términos

- Entrada: Lista de ecuaciones $L = (s_1 = t_1, \dots, s_n = t_n)$ y sustitución σ.
- Salida:
  Un UMG de las listas $(s_1 \dots, s_n)\sigma$ y $(t_1 \dots, t_n)\sigma$, si son unificables;
  "No unificables", en caso contrario.

Algoritmo de unificación de listas de términos

- Procedimiento $\text{unif}(L, \sigma)$:
  1. Si $L = ()$, entonces $\text{unif}(L, \sigma) = \sigma$.
  2. Si $L = (t = t|L')$, entonces $\text{unif}(L, \sigma) = \text{unif}(L', \sigma)$.
  3. Si $L = (f(t_1, \dots, t_m) = f(t'_1 \dots, t'_m)|L')$, entonces $\text{unif}(L, \sigma) = \text{unif}((t_1 = t'_1, \dots, t_m = t'_m|L'), \sigma)$.
  4. Si $L = (x = t|L')$ (ó $L = (t = x|L')$) y x no aparece en t, entonces $\text{unif}(L, \sigma) = \text{unif}(L'[x/t], \sigma[x/t])$.
  5. Si $L = (x = t|L')$ (ó $L = (t = x|L')$) y x aparece en t, entonces $\text{unif}(L, \sigma) =$ "No unificables".
  6. Si $L = (f(t_1, \dots, t_m) = g(t'_1 \dots, t'_m)|L')$, entonces $\text{unif}(L, \sigma) =$ "No unificables".
  7. Si $L = (f(t_1, \dots, t_m) = f(t'_1 \dots, t'_p)|L')$ y $m \neq p$, entonces $\text{unif}(L, \sigma) =$ "No unificables".

Algoritmo de unificación de dos términos

- Entrada: Dos términos t_1 y t_2.
- Salida: Un UMG de t_1 y t_2, si son unificables;
  "No unificables", en caso contrario.
- Procedimiento: $\text{unif}((t_1 = t_2), \epsilon)$.
- Ejemplo 1: Unificar $f(x, g(z))$ y $f(g(y), x)$:

$$
\begin{array}{cll}
 & \text{unif}((f(x, g(z)) = f(g(y), x)), \epsilon) & \\
= & \text{unif}((x = g(y), g(z) = x), \epsilon) & \text{por } 3 \\
= & \text{unif}((g(z) = x)[x/g(y)], \epsilon[x/g(y)]) & \text{por } 4 \\
= & \text{unif}((g(z) = g(y)), [x/g(y)]) & \\
= & \text{unif}((z = y), [x/g(y)]) & \text{por } 3 \\
= & \text{unif}((), [x/g(y)][z/y]) & \text{por } 4 \\
= & \text{unif}((), [x/g(y), z/y]) & \\
= & [x/g(y), z/y] & \text{por } 1
\end{array}
$$

Ejemplos de unificación

- Ejemplo 2: Unificar $f(x,b)$ y $f(a,y)$:

$$\begin{array}{lll} & \text{unif}((f(x,b)=f(a,y),\epsilon) & \\ = & \text{unif}((x=a,b=y),\epsilon) & \text{por 3} \\ = & \text{unif}((b=y)[x/a],\epsilon[x/a]) & \text{por 4} \\ = & \text{unif}((b=y),[x/a]) & \\ = & \text{unif}((),[x/a][y/b]) & \text{por 4} \\ = & [x/a,y/b]) & \text{por 1} \end{array}$$

- Ejemplo 3: Unificar $f(x,x)$ y $f(a,b)$:

$$\begin{array}{lll} & \text{unif}((f(x,x)=f(a,b)),\epsilon) & \\ = & \text{unif}((x=a,x=b),\epsilon) & \text{por 3} \\ = & \text{unif}((x=b)[x/a],\epsilon[x/a]) & \text{por 4} \\ = & \text{unif}((a=b),[x/a]) & \\ = & \text{“No unificable”} & \text{por 6} \end{array}$$

Ejemplos de unificación

- Ejemplo 4: Unificar $f(x, g(y))$ y $f(y, x)$:

$$
\begin{array}{lll}
 & \mathrm{unif}((f(x,g(y)) = f(y,x)), \epsilon) & \\
= & \mathrm{unif}((x = y, g(y) = x), \epsilon) & \text{por 3} \\
= & \mathrm{unif}((g(y) = x)[x/y], \epsilon[x/y]) & \text{por 4} \\
= & \mathrm{unif}((g(y) = y), [x/y]) & \\
= & \text{``No unificable''} & \text{por 5}
\end{array}
$$

- Ejemplo 5: Unificar $j(w, a, h(w))$ y $j(f(x, y), x, z)$

$$
\begin{array}{lll}
 & \mathrm{unif}((j(w,a,h(w)) = j(f(x,y),x,z))\epsilon) & \\
= & \mathrm{unif}((w = f(x,y), a = x, h(w) = z), \epsilon) & \text{por 3} \\
= & \mathrm{unif}((a = x, h(w) = z)[w/f(x,y)], \epsilon[w/f(x,y)]) & \text{por 4} \\
= & \mathrm{unif}((a = x, h(f(x,y)) = z), [w/f(x,y)]) & \\
= & \mathrm{unif}((h(f(x,y)) = z)[x/a], [w/f(x,y)][x/a]) & \text{por 4} \\
= & \mathrm{unif}((h(f(a,y)) = z), [w/f(a,y), x/a]) & \\
= & \mathrm{unif}((), [w/f(a,y), x/a][z/h(f(a,y))]) & \text{por 4} \\
= & [w/f(a,y), x/a, z/h(f(a,y))] & \text{por 1}
\end{array}
$$

Ejemplos de unificación

- Ejemplo 6: Unificar $j(w, a, h(w))$ y $j(f(x, y), x, y)$

$$
\begin{array}{lll}
 & \mathrm{unif}((j(w, a, h(w)) = j(f(x, y), x, y))\epsilon) & \\
= & \mathrm{unif}((w = f(x, y), a = x, h(w) = y), \epsilon) & \text{por 3} \\
= & \mathrm{unif}((a = x, h(w) = y)[w/f(x, y)], \epsilon[w/f(x, y)]) & \text{por 4} \\
= & \mathrm{unif}((a = x, h(f(x, y)) = y), [w/f(x, y)]) & \\
= & \mathrm{unif}((h(f(x, y)) = y)[x/a], [w/f(x, y)][x/a]) & \text{por 4} \\
= & \mathrm{unif}((h(f(a, y)) = y), [w/f(a, y), x/a]) & \\
= & \text{“No unificable”} & \text{por 5}
\end{array}
$$

- Ejemplo 7: Unificar $f(a, y)$ y $f(a, b)$:

$$
\begin{array}{lll}
 & \mathrm{unif}((f(a, y) = f(a, b), \epsilon)) & \\
= & \mathrm{unif}((a = a, y = b), \epsilon) & \text{por 3} \\
= & \mathrm{unif}((y = b), \epsilon) & \text{por 2} \\
= & \mathrm{unif}((), [y/b]) & \text{por 4} \\
= & [y/b] & \text{por 1}
\end{array}
$$

Tema 12: Resolución en lógica de primer orden

Separación de variables

- Def.: La sustitución $[x_1/t_1, \dots, x_n/t_n]$ es un renombramiento si todos los t_i son variables.
- Prop.: Si θ es un renombramiento, entonces $C \equiv C\theta$.
- Def.: Las cláusulas C_1 y C_2 están separadas si no tienen ninguna variable común.
- Def.: Una separación de las variables de C_1 y C_2 es un par de renombramientos (θ_1, θ_2) tales que $C_1\theta_1$ y $C_2\theta_2$ están separadas.
- Ejemplo: Una separación de variables de
  $$C_1 = \{P(x), Q(x,y)\} \text{ y } C_2 = \{R(f(x,y))\}$$
  es
  $$(\theta_1 = [x/x_1, y/y_1], \theta_2 = [x/x_2, y/y_2]).$$

Resolvente binaria

- Def.: La cláusula C es una resolvente binaria de las cláusulas C_1 y C_2 si existen una separación de variables (θ_1, θ_2) de C_1 y C_2, un literal $L_1 \in C_1$, un literal $L_2 \in C_2$ y un UMG σ de $L_1\theta_1$ y $L_2^c\theta_2$ tales que
$$C = (C_1\theta_1\sigma \setminus \{L_1\theta_1\sigma_1\}) \cup (C_2\theta_2\sigma \setminus \{L_2\theta_2\sigma\}).$$
- Ejemplo: Sean
$$\begin{array}{llll} C_1 & = \{\neg P(x), Q(f(x))\}, & C_2 & = \{\neg Q(x), R(g(x))\}, \\ L_1 & = Q(f(x)), & L_2 & = \neg Q(x), \\ \theta_1 & = [x/x_1], & \theta_2 & = [x/x_2], \\ L_1\theta_1 & = Q(f(x_1)), & L_2^c\theta_2 & = Q(x_2), \\ \sigma & = [x_2/f(x_1)] & & \end{array}$$
Entonces, $C = \{\neg P(x_1), R(g(f(x_1)))\}$ es una resolvente binaria de C_1 y C_2.

Factorización

- Def.: La cláusula C es un factor de la cláusula D si existen dos literales L_1 y L_2 en D que son unificables y $C = D\sigma \setminus \{L_2\sigma\}$ donde σ es un UMG de L_1 y L_2.
- Ejemplo: Sean

$$\begin{array}{ll} D & = \{P(x,y), P(y,x), Q(a)\} \\ L_1 & = P(x,y) \\ L_2 & = P(y,x) \\ \sigma & = [y/x] \end{array}$$

Entonces,

$C = \{P(x,x), Q(a)\}$ es un factor de D.

Ejemplos de refutación por resolución

- Refutación de
  $S = \{\{\neg P(x, f(x, y))\}, \{P(a, z), \neg Q(z, v)\}, \{Q(u, a)\}\}$

1	$\{\neg P(x, f(x, y))\}$	Hipótesis
2	$\{P(a, z), \neg Q(z, v)\}$	Hipótesis
3	$\{Q(u, a)\}$	Hipótesis
4	$\{\neg Q(f(a, y), v)\}$	Resolvente de 1 y 2 con $\sigma = [x/a, z/f(a, y)]$
5	$\square$	Resolvente de 3 y 4 con $\sigma = [u/f(a, y), v/a]$

- Refutación de $S = \{\{P(x)\}, \{\neg P(f(x))\}\}$

1	$\{P(x)\}$	Hipótesis
2	$\{\neg P(f(x))\}$	Hipótesis
3	$\square$	Resolvente de 1 y 2 con con $\theta_1 = \epsilon, \theta_2 = [x/x'], \sigma = [x/f(x')]$

Ejemplos de refutación por resolución

- Refutación de $S = \{\{P(x,y), P(y,x)\}, \{\neg P(u,v), \neg P(v,u)\}\}$

1	$\{P(x,y), P(y,x)\}$	Hipótesis
2	$\{\neg P(u,v), \neg P(v,u)\}$	Hipótesis
3	$\{P(x,x)\}$	Factor de 1 con $[y/x]$
4	$\{\neg P(u,u)\}$	Factor de 2 con $[v/u]$
5	$\square$	Resolvente de 3 y 4 con $[x/u]$

Demostraciones de cláusulas por resolución

- Sea S un conjunto de cláusulas.
- La sucesión $(C_1, \dots, C_n)$ es una demostración por resolución de la cláusula C a partir de S si $C = C_n$ y para todo $i \in \{1, ..., n\}$ se verifica una de las siguientes condiciones:
  - $C_i \in S$;
  - existen $j, k < i$ tales que C_i es una resolvente de C_j y C_k
  - existe $j < i$ tal que C_i es un factor de C_j
- La cláusula C es demostrable por resolución a partir de S si existe una demostración por resolución de C a partir de S.
- Una refutación por resolución de S es una demostración por resolución de la cláusula vacía a partir de S.
- Se dice que S es refutable por resolución si existe una refutación por resolución a partir de S.

Demostraciones de fórmulas por resolución

- Def.: Sean $S_1, \dots, S_n$ formas clausales de las fórmulas $F_1, \dots, F_n$ y S una forma clausal de $\neg F$. Una demostración por resolución de F a partir de $\{F_1, \dots, F_n\}$ es una refutación por resolución de $S_1 \cup \dots \cup S_n \cup S$.
- Def.: La fórmula F es demostrable por resolución a partir de $\{F_1, \dots, F_n\}$ si existe una demostración por resolución de F a partir de $\{F_1, \dots, F_n\}$.
  Se representa por $\{F_1, \dots, F_n\} \vdash_{Res} F$.
- Ejemplo: (tema 8 p. 21)
  $\{\forall x\ [P(x) \to Q(x)], \exists x\ P(x)\} \vdash_{Res} \exists x\ Q(x)$

1	$\{\neg P(x), Q(x)\}$	Hipótesis
2	$\{P(a)\}$	Hipótesis
3	$\{\neg Q(z)\}$	Hipótesis
4	$\{Q(a)\}$	Resolvente de 1 y 2 con $[x/a]$
5	$\square$	Resolvente de 3 y 4 con $[z/a]$

Ejemplos de demostraciones por resolución

- Ejemplo: (tema 8 p. 21)
  $\{\forall x\ [P(x) \to Q(x)], \forall x\ [Q(x) \to R(x)] \vdash_{Res} \forall x\ [P(x) \to R(x)]\}$

1	$\{\neg P(x), Q(x)\}$	Hipótesis
2	$\{\neg Q(y), R(y)\}$	Hipótesis
3	$\{P(a)\}$	Hipótesis
4	$\{\neg R(a)\}$	Hipótesis
5	$\{Q(a)\}$	Resolvente de 1 y 3 con $[x/a]$
6	$\{R(a)\}$	Resolvente de 5 y 2 con $[y/a]$
5	$\square$	Resolvente de 6 y 4

- Ejemplo: (tema 6 p. 55) $\vdash_{Res} \exists x\ [P(x) \to \forall y\ P(y)]$

1	$\{P(x)\}$	Hipótesis
2	$\{\neg P(f(x))\}$	Hipótesis
3	$\square$	Resolvente de 1 y 2 con $\theta_2 = [x/x'], \sigma = [x/f(x')]$

Ejemplos de demostraciones por resolución

- Ejemplo: $\vdash_{Res} \forall x\ \exists y\ \neg(P(y,x) \leftrightarrow \neg P(y,y))$
  - Forma clausal:

$$
\begin{array}{ll}
 & \neg\forall x\ \exists y\ \neg(P(y,x) \leftrightarrow \neg P(y,y)) \\
\equiv & \neg\forall x\ \exists y\ \neg((P(y,x) \rightarrow \neg P(y,y)) \wedge (\neg P(y,y) \rightarrow P(y,x))) \\
\equiv & \neg\forall x\ \exists y\ \neg((\neg P(y,x) \vee \neg P(y,y)) \wedge (\neg\neg P(y,y) \vee P(y,x))) \\
\equiv & \neg\forall x\ \exists y\ \neg((\neg P(y,x) \vee \neg P(y,y)) \wedge (P(y,y) \vee P(y,x))) \\
\equiv & \exists x\ \forall y\ \neg\neg((\neg P(y,x) \vee \neg P(y,y)) \wedge (P(y,y) \vee P(y,x))) \\
\equiv & \exists x\ \forall y\ ((\neg P(y,x) \vee \neg P(y,y)) \wedge (P(y,y) \vee P(y,x))) \\
\approx & \forall y\ ((\neg P(y,a) \vee \neg P(y,y)) \wedge (P(y,y) \vee P(y,a))) \\
\equiv & \{\{\neg P(y,a), \neg P(y,y)\}, \{P(y,y), P(y,a)\}\}
\end{array}
$$

  - Refutación:

1	$\{\neg P(y,a), \neg P(y,y)\}$	Hipótesis
2	$\{P(y,y), P(y,a)\}$	Hipótesis
3	$\{\neg P(a,a)\}$	Factor de 1 con $[y/a]$
4	$\{P(a,a)\}$	Factor de 2 con $[y/a]$
5	$\square$	Resolvente de 3 y 4

Paradoja del barbero de Russell

En una isla pequeña hay sólo un barbero. El gobernador de la isla ha publicado la siguiente norma: *"El barbero afeita a todas las personas que no se afeitan a sí misma y sólo a dichas personas"*. Demostrar que la norma es inconsistente.

- Representación: $\forall x\ [\text{afeita}(b,x) \leftrightarrow \neg\text{afeita}(x,x)]$
- Forma clausal:

$$\begin{array}{ll}
 & \forall x\ [\text{afeita}(b,x) \leftrightarrow \neg\text{afeita}(x,x)] \\
\equiv & \forall x\ [(\text{afeita}(b,x) \to \neg\text{afeita}(x,x)) \land (\neg\text{afeita}(x,x) \to \text{afeita}(b,x))] \\
\equiv & \forall x\ [(\neg\text{afeita}(b,x) \lor \neg\text{afeita}(x,x)) \land (\neg\neg\text{afeita}(x,x) \lor \text{afeita}(b,x))] \\
\equiv & \forall x\ [(\neg\text{afeita}(b,x) \lor \neg\text{afeita}(x,x)) \land (\text{afeita}(x,x) \lor \text{afeita}(b,x))] \\
\equiv & \{\{\neg\text{afeita}(b,x), \neg\text{afeita}(x,x)\}, \{\text{afeita}(x,x), \text{afeita}(b,x)\}\}
\end{array}$$

- Refutación:

1	$\{\neg\text{afeita}(b,x), \neg\text{afeita}(x,x)\}$	Hipótesis
2	$\{\text{afeita}(x,x), \text{afeita}(b,x)\}$	Hipótesis
3	$\{\neg\text{afeita}(b,b)\}$	Factor de 1 con $[x/b]$
4	$\{\text{afeita}(b,b)\}$	Factor de 2 con $[x/b]$
5	$\square$	Resolvente de 3 y 4

Adecuación y completitud de la resolución

- Propiedades:
  - Si C es una resolvente de C_1 y C_2, entonces $\{C_1, C_2\} \models C$.
  - Si D es un factor de C entonces $C \models D$.
  - Si $\square \in S$, entonces S es inconsistente.
  - Si el conjunto de cláusulas S es refutable por resolución, entonces S es inconsistente.
- Teor.: El cálculo de resolución (para la lógica de primer orden sin igualdad) es adecuado y completo; es decir,

  Adecuado: $S \vdash_{Res} F \quad\Longrightarrow\quad S \models F$

  Completo: $S \models F \quad\Longrightarrow\quad S \vdash_{Res} F$

Decisión de no–consecuencia por resolución

- Enunciado: Comprobar, por resolución, que
  $\forall x\ [P(x) \lor Q(x)] \not\models \forall x\ P(x) \lor \forall x\ Q(x)$.
- Reducción 1: Comprobar que es consistente
  $\{\forall x\ [P(x) \lor Q(x)],\ \neg(\forall x\ P(x) \lor \forall x\ Q(x))\}$
- Reducción 2: Comprobar que es consistente
  $\{\{P(x), Q(x)\}, \{\neg P(a)\}, \{\neg Q(b)\}\}$
- Resolución:

1	$\{P(x), Q(x)\}$	Hipótesis
2	$\{\neg P(a)\}$	Hipótesis
3	$\{\neg Q(b)\}$	Hipótesis
4	$\{Q(a)\}$	Resolvente de 1 y 2
5	$\{P(b)\}$	Resolvente de 1 y 3

- Modelo: $U = \{a, b\}, I(P) = \{b\}, I(Q) = \{a\}$.

Bibliografía

1. Fitting, M. *First-Order Logic and Automated Theorem Proving (2nd ed.)* (Springer, 1996) pp. 137–141.
2. M.L. Bonet *Apuntes de LPO.* (Univ. Politécnica de Cataluña, 2003) pp. 34–40.
3. C.L. Chang y R.C.T. Lee *Symbolic logic and mechanical theorem proving* (Academic Press, 1973) pp. 70–99.
4. M. Genesereth *Computational Logic (Chapter 9: Relational Resolution)* (Stanford University, 2003)
5. S. Hölldobler *Computational logic.* (U. de Dresden, 2004) pp. 71–74.
6. M. Ojeda e I. Pérez *Lógica para la computación (Vol. 2: Lógica de Primer Orden)* (Ágora, 1997) pp. 138–164.
7. L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 50–61.
8. U. Schöning *Logic for computer scientists* (Birkäuser, 1989) pp. 79–96.