Automated Reasoning and Learning
-
CP1 SLD Resolution Trees (Examples)

Ramon Béjar Torres[†]

March 21, 2024

[†] ramon.bejar@udl.cat

Table of Contents

Solving CP1 Problems with with SLD Resolution

For CP1, the kind of logical consequence questions we want to answer is of this type

$$P \models \exists x_1 \exists x_2 \ldots \exists x_n Q(x_1, x_2, \ldots, x_n)$$

This query is always transformed to a single logical formula:

$$P \cup \{\neg \exists x_1 \exists x_2 \ldots \exists x_n Q(x_1, x_2, \ldots, x_n)\} \equiv$$
$$P \cup \{\forall x_1 \forall x_2 \ldots \forall x_n \neg Q(x_1, x_2, \ldots, x_n)\}$$

But in CP1, a linear resolution proof will use, for every pair of resolved clauses, a most general unifier that shows the specific cases where the two resolved literals are contradictory for any interpretation (so they can be resolved).

CP1 SLD Resolution Trees - Example 1

Given the following CP1 logic program P (remember, all the variables are assumed to be universally quantified):

$$P = \begin{cases} 1.\ q(a) \\ 2.\ p(X) \leftarrow q(X) \end{cases}$$

Then, for the question:

$$P \models p(a) \ \equiv \ P \cup \{\neg p(a)\} \text{ is } \ \mathrm{UNSAT}$$

a SLD resolution proof is as follows:

CP1 SLD Resolution Trees - Example 1

$\leftarrow p(a)$

$\quad | \; 2.\{a/X_1\}$

$\leftarrow q(a)$

$\quad | \; 1.\{\}$

$\square$

Steps in the SLD resolution proof

1. From the top goal $\{\neg p(a)\}$ and the clause $\{p(X_1), \neg q(X_1)\}$ we obtain the resolvent (new goal) $\{\neg q(a)\}$ using the mgu $\theta = \{a/X_1\}$.
2. Then, from the new goal $\{\neg q(a)\}$ and the clause $\{q(a)\}$ we obtain the resolvent $\{\}$ (empty clause) using the mgu $\theta = \{\}$.

CP1 SLD Resolution Trees - Example 2

Given the following CP1 logic program P:

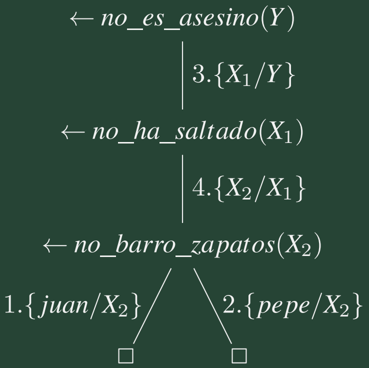
$$P = \left\{ \begin{array}{l} 1.\ no\_barro\_zapatos(juan). \\ 2.\ no\_barro\_zapatos(pepe). \\ \\ 3.\ no\_es\_asesino(X) \leftarrow no\_ha\_saltado(X). \\ 4.\ no\_ha\_saltado(X) \leftarrow no\_barro\_zapatos(X). \end{array} \right.$$

Then, the answer to the question:

$$P \models \exists Y\, no\_es\_asesino(Y) \quad \equiv \quad P \cup \{\forall Y \neg no\_es\_asesino(Y)\} \text{ is UNSAT}$$

is obtained with SLD resolution as follows

CP1 SLD Resolution Trees - Example 2



Steps:

1. top goal and clause 3 are resolved with mgu = $\{X_1/Y\}$
2. goal $\{\neg no\_ha\_saltado(X_1)\}$ and clause 4 are resolved with mgu = $\{X_2/X_1\}$
3. goal $\{\neg no\_barro\_zapatos(X_2)\}$ is resolved with clause 1 using mgu=$\{juan/X_2\}$ and with clause 2 using mgu=$\{pepe/X_2\}$.

So, we obtain two different answers:

- $\{X_1/Y\} \cdot \{X_2/X_1\} \cdot \{juan/X_2\} = \{juan/Y\}$
- $\{X_1/Y\} \cdot \{X_2/X_1\} \cdot \{pepe/X_2\} = \{pepe/Y\}$

CP1 SLD Resolution Trees - Example 3

Given the following CP1 logic program P:

$$P = \begin{cases} 1.\ append(l(X,e),L2,l(X,L2)). \\ 2.\ append(l(X_1,T1),L2',l(X_1,T3)) \leftarrow append(T1,L2',T3). \end{cases}$$

Then, the answer to the question:

$$P \models \exists L3\ append(l(a,l(b,e)),l(c,e),L3)$$

is obtained with SLD resolution as follows

CP1 SLD Resolution Trees - Example 3

$$\leftarrow append(l(a,l(b,e)),l(c,e),L3)$$

$$\Big|\; 2.\{a/X_1, l(b,e)/T_1, l(c,e)/L2', l(a,T3)/L3\}$$

$$\leftarrow append(l(b,e),l(c,e),T3)$$

$$\Big|\; 1.\{b/X, l(c,e)/L2, l(b,l(c,e))/T3\}$$

$$\square$$

Steps in the SLD resolution proof

1. From $\{\neg append(l(a,l(b,e)),l(c,e),L3)\}$ and $\{append(l(X_1,T1),L2',l(X_1,T3)) \vee \neg append(T1,L2',T3)\}$ we obtain the resolvent $\{\neg append(l(b,e),l(c,e),T3)\}$ with unifier $\theta_1 = \{a/X_1,\ l(b,e)/T_1,\ l(c,e)/L2',\ l(a,T3)/L3\}$.
2. From $\{\neg append(l(b,e),l(c,e),T3)\}$ and $\{append(l(X,e),L2,l(X,L2))\}$ we obtain the resolvent $\{\}$ using the unifier $\theta_2 = \{b/X,\ l(c,e)/L2,\ l(b,l(c,e))/T3\}$.

CP1 SLD Resolution Trees - Example 3

So, the value of $L3$ in the input question is obtained trough the composition $\theta_1 \cdot \theta_2$:

$$\{a/X_1,\ l(b,e)/T_1,\ l(c,e)/L2',\ l(a,T3)/L3\} \cdot$$
$$\{b/X, l(c,e)/L2,\ l(b,l(c,e))/T3\}$$

where we are only interested in tha value for $L3$:

$$\{l(a,T3)/L3\} \cdot \{l(b,l(c,e))/T3\} = \{l(a,l(b,l(c,e)))/L3\}$$

Observe that we have used this append predicate to simulate the appending of one list with another one. In particular our query was equivalent to:

$$[a,b]+[c] = [a,b,c]$$

We have used the constant e to encode an empty list.