Rewriting Techniques, 4: dependency pairs, argument filtering

05–12–2019

Exercise 1:

Definition 1 (Weakly monotone algebra). A weakly monotone \( \mathcal{F} \)-algebra \((A, >, \supseteq)\) consists of a non-empty \( \mathcal{F} \)-algebra \( A \) together with a proper order \( > \) and a preorder \( \supseteq \) on the carrier \( A \) of \( A \) such that \( > \cdot \supseteq > \) or \( \supseteq \cdot > \supseteq \), and every algebra operation is monotone with respect to \( \supseteq \) in all coordinates, i.e., if \( f \in \mathcal{F} \) has arity \( n \geq 1 \), for all \( a_1, \ldots, a_n, b \in A \) and \( i \in [1, n] \) with \( a_i \supseteq b \) then

\[
 f_A(a_1, \ldots, a_i, \ldots, a_n) \supseteq f_A(a_1, \ldots, b, \ldots, a_n)
\]

A monotone algebra \((A, >, \supseteq)\) is said well-founded if \( > \) is so.

Prove that if \((A, >, \supseteq)\) is a well-founded weakly monotone algebra then \((>_{\mathcal{A}}, \supseteq_{\mathcal{A}})\) is a reduction pair.

Solution:

We first prove that \( \supseteq_{\mathcal{A}} \) is a rewrite relation. Closure under context follows from monotonicity. Closure by substitution by definition of \( \supseteq_{\mathcal{A}}: t \supseteq_{\mathcal{A}} u \) if for any valuation \( \xi: \text{Var} \to A, t\xi \supseteq u\xi \) (where \( \xi \) is the substitution elevated to terms such that \( f(t_1, \ldots, t_n)\xi = f_A(t_1\xi, \ldots, t_n\xi) \)).

It is known from a previous theorem that \( >_{\mathcal{A}} \) is a reduction order for well-founded monotonic algebras \((A, >)\). In the proof of this theorem, monotonicity was used only for the closure under context of \( >_{\mathcal{A}} \), which is not needed here (\( > \) does not need to be a rewrite order). It can still be derived that \( >_{\mathcal{A}} \) is closed under substitution (and is a well founded order by assumption). The condition \( >_{\mathcal{A}} \cdot \supseteq_{\mathcal{A}} \subseteq >_{\mathcal{A}} \) or \( \supseteq_{\mathcal{A}} \cdot >_{\mathcal{A}} \subseteq >_{\mathcal{A}} \) follows from the definitions.

Exercise 2:

(a) Let \( R \) be a rewrite system and such that each defined symbol has positive arity. Prove that if every cycle \( C \) of the dependency graph of \( R \) has a simple projection \( \pi \) such that \( \pi(C) \subseteq \supseteq \) and \( \pi(C) \cap \supseteq \neq \emptyset \), where \( \pi(C) = \{(\pi(s), \pi(t)) \mid (s, t) \in C\} \) and \( \supseteq \) is the subterm relation, then \( R \) terminates.

Solution:

From the theorem of dependency pairs, suppose to the contrary that there exists a rewrite sequence

\[
t_1 \rightarrow_R^* u_1 \rightarrow_C t_2 \rightarrow_R^* u_2 \rightarrow_C t_3 \rightarrow_R^* \ldots
\]

where \( t_1 = f^*(s_1, \ldots, s_n) \) is headed by a marked symbol, \( f(s_1, \ldots, s_n) \) is a minimal non-terminating term and all rules of \( C \) are applied infinitely often. We apply the simple projection to this rewrite sequence:

- Consider \( u_i \rightarrow_C t_{i+1} \). There exists a dependency pair \( l \rightarrow r \in C \) and a substitution \( \sigma \) such that \( u_i = l\sigma \) and \( t_{i+1} = r\sigma \). We have \( \pi(u_i) = \pi(l)\sigma \) and \( \pi(t_{i+1}) = \pi(r)\sigma \). Since \( \pi(l) \rightarrow \pi(r) \in \pi(C) \), by hypothesis it holds \( \pi(l) \supseteq \pi(r) \). So \( \pi(l) = \pi(r) \) or \( \pi(l) \triangleright \pi(r) \).

In the former case, trivially \( \pi(u_i) = \pi(t_{i+1}) \). In the latter case, the closure under substitution of \( \triangleright \) yields \( \pi(u_i) \triangleright \pi(t_{i+1}) \). Because of the assumption \( \pi(C) \cap \triangleright \neq \emptyset \), and all rules of \( C \) are applied infinitely often, \( \pi(u_i) \triangleright \pi(t_{i+1}) \) will hold for infinitely many \( i \).
• Consider now \( t_i \rightarrow_R^* u_i \). All steps in this sequence take place below the (marked) root symbol, which is therefore the same for \( t_i \) and \( u_i \). Therefore \( \pi(t_i) \rightarrow_R^* \pi(u_i) \) holds.

By applying our simple projection \( \pi \) to the rewrite sequence, we transform it into an infinite \( \rightarrow_R \cup \triangleright \) sequence containing infinitely many \( \triangleright \) steps, starting from \( \pi(t_1) \). Since \( \triangleright \) is well-founded, the sequence must also contain infinitely many \( \rightarrow_R \) steps. By making repeated use of the commutation \( (\rightarrow_R \cup \triangleright) \subseteq (\rightarrow_R \triangleright) \) we obtain an infinite sequence of \( \rightarrow_R \) starting from \( \pi(t_1) \). Therefore \( \pi(t_1) \) is not terminating w.r.t. \( R \). But \( f(s_1, \ldots, s_n) \triangleright \pi(t_1) \) and \( f(s_1, \ldots, s_n) \) is a minimal non-terminating term: contradiction.

Consider the following rewriting system:

\[
\begin{align*}
m(1) & \rightarrow 1 \\
m(a(x, y)) & \rightarrow a(s(x), m(y)) \\
q(0, 0) & \rightarrow a(0, 1) \\
q(s(x), 0) & \rightarrow 1 \\
q(s(x), s(y)) & \rightarrow m(q(x, y)) \\
q(0, s(y)) & \rightarrow a(0, q(s(0), s(y)))
\end{align*}
\]

(b) Compute the marked dependency pairs and the dependency graph approximation.

<table>
<thead>
<tr>
<th>Solution:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The defined symbols are {m, q}. The marked dependency pairs are:</td>
</tr>
<tr>
<td>1 : ((m^#(a(x, y)), m^#(y)))</td>
</tr>
<tr>
<td>2 : ((q^#(s(x), s(y)), m^#(q(x, y))))</td>
</tr>
<tr>
<td>3 : ((q^#(s(x), s(y)), q^#(x, y)))</td>
</tr>
<tr>
<td>4 : ((q^#(0, s(y)), q^#(s(0), s(y))))</td>
</tr>
<tr>
<td>whereas the dependency graph approximation is</td>
</tr>
</tbody>
</table>
| \[
\begin{array}{cccc}
& 1 & 2 & \\
\triangleright & & & \\
3 & & & \\
\triangleright & & & \\
& 3 & 4 & \\
\end{array}
\] |

(c) Prove the termination of the rewrite system by finding a suitable simple projection that satisfied the constraints in question 1.

<table>
<thead>
<tr>
<th>Solution:</th>
</tr>
</thead>
<tbody>
<tr>
<td>There are 3 loops: {1}, {3}, {3, 4}. We define ( \pi(f^#) = 1 ) and ( \pi(g^#) = 2 ). For the first loop it holds ( a(x, y) \triangleright y ); for the second loop ( s(y) \triangleright y ) and for the last loop ( s(y) \triangleright s(y) ). The conditions to apply the result in question 1 are therefore satisfied and the TRS terminates.</td>
</tr>
</tbody>
</table>

An argument filter \( AF \) for short) for a signature \( \mathcal{F} \) is a mapping \( \pi \) that associates with every \( n \)-ary function symbol an argument position \( i \in [1, n] \) or a (possibly empty) list \([i_1, \ldots, i_m]\) of argument positions with \( 1 \leq i_1 < \cdots < i_m \leq n \).

The signature \( \mathcal{F}_\pi \) consists of all function symbols \( f \) such that \( \pi(f) \) is some list \([i_1, \ldots, i_m]\), where in \( \mathcal{F}_\pi \) the arity of \( f \) is \( m \). Every argument filter \( \pi \) induces a mapping from \( T(\mathcal{F}, \text{Var}) \) to \( T(\mathcal{F}_\pi, \text{Var}) \), also denoted by \( \pi \):

\[
\pi(t) = \begin{cases} 
  t & \text{if } t \text{ is a variable} \\
  \pi(t_i) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } \pi(f) = i \\
  f(\pi(t_{i_1}), \ldots, \pi(t_{i_m})) & \text{if } t = f(t_1, \ldots, t_n) \text{ and } \pi(f) = [i_1, \ldots, i_m]
\end{cases}
\]

Exercise 3:
Let $R$ be the following TRS,

$$
\begin{align*}
0 \cdot y &\rightarrow 0 \\
0 \cdot 0 &\rightarrow x \\
x \cdot s \cdot y &\rightarrow x \cdot y \\
0 \cdot s \cdot y &\rightarrow 0 \\
x \cdot s \cdot s \cdot y &\rightarrow s \cdot (x-y) \div s \cdot y \\
\end{align*}
$$

(a) Give dependency pairs of $R$

**Solution:**

$$
\begin{align*}
s \cdot x \cdot s \cdot y &\rightarrow x \cdot s \cdot y \\
s \cdot x \cdot s \cdot y &\rightarrow (x-y) \div s \cdot y \\
s \cdot x \cdot s \cdot y &\rightarrow x \div s \cdot y \\
\end{align*}
$$

(b) Find a weakly monotone polynomial interpretation on $\mathbb{N}$ to prove termination of $R$.

**Solution:**

$$P_-(X,Y) = P_+(X,Y) = P_{-1}(X,Y) = P_{+1}(X,Y) = X, \quad P_{s}(X) = X + 1 \quad \text{and} \quad P_0 = 0.$$  

(c) Find an argument filter to prove termination using LPO with empty precedence.

**Solution:**

$$\pi(-) = \pi(\div) = \pi(\cdot) = \pi(s) = 1, \quad \pi(0) = [\ ] \quad \text{and} \quad \pi(s) = [1]$$

which simplifies the system to

$$
\begin{align*}
0 \geq 0 &\quad 0 \geq 0 &\quad s \cdot x > x \\
x \geq x &\quad s \cdot x \geq s \cdot x &\quad s \cdot x > x \\
s \cdot x \geq x &\quad s \cdot x > x \\
\end{align*}
$$

**Exercise 4:**

Consider the rewriting system $R$:

$$
\begin{align*}
0 \leq y &\rightarrow \text{true} & x - 0 &\rightarrow x & \text{gcd} 0 \cdot y &\rightarrow y \\
(s \cdot x) \leq 0 &\rightarrow \text{false} & x - \text{(s \cdot y)} &\rightarrow \text{p} \cdot (x-y) & \text{gcd} (s \cdot x) \geq 0 &\rightarrow s \cdot x \\
s \cdot x \leq s \cdot y &\rightarrow x \leq y & \text{p} \cdot (s \cdot x) &\rightarrow x & \text{gcd} (s \cdot x) \cdot (s \cdot y) &\rightarrow \text{if} \ (y \leq x) \cdot (s \cdot x) \cdot (s \cdot y) \\
\text{if true} (s \cdot x) \cdot (s \cdot y) &\rightarrow \text{gcd} (x-y) \cdot (s \cdot y) & \text{if false} (s \cdot x) \cdot (s \cdot y) &\rightarrow \text{gcd} (y-x) \cdot (s \cdot x) \\
\end{align*}
$$

(a) Compute the dependency pairs of $R$.

**Solution:**

$$
\begin{align*}
s \cdot x \cdot s \cdot y &\rightarrow x \leq s \cdot y \\
x - \#(s \cdot y) &\rightarrow \text{p} \cdot (x-y) \\
x - \#(s \cdot y) &\rightarrow x - \#y \\
\text{if}^\dagger \text{true} (s \cdot x) \cdot (s \cdot y) &\rightarrow \text{gcd}^\dagger (x-y) \cdot (s \cdot y) \\
\text{if}^\dagger \text{true} (s \cdot x) \cdot (s \cdot y) &\rightarrow x - \#y \\
\text{gcd}^\dagger (s \cdot x) \cdot (s \cdot y) &\rightarrow \text{if}^\dagger \ (y \leq x) \cdot (s \cdot x) \cdot (s \cdot y) \\
\text{gcd}^\dagger (s \cdot x) \cdot (s \cdot y) &\rightarrow y \leq x \\
\text{if}^\dagger \text{false} (s \cdot x) \cdot (s \cdot y) &\rightarrow \text{gcd}^\dagger (y-x) \cdot (s \cdot x) \\
\text{if}^\dagger \text{false} (s \cdot x) \cdot (y) &\rightarrow y - \#x \\
\end{align*}
$$
(b) How many different argument filters does \( R \cup DP(R) \) admit?

**Solution:**
Define \( N(f) = a + 2^a \) where \( a \) is the arity of \( f \in \Sigma \). Then \( N \) is the number of argument filters for function \( f \). The number of argument filters for the system \( R \) is then \( N(\Sigma) = \prod_{f \in \Sigma} N(f) \). For \( R \), we have \( N(R) = 21384 \).

(c) Prove the termination of \( R \).

**Solution:**
Argument filter \( \pi(\leq) = 2, \pi(-) = [ ], \pi(-^\perp) = 2 \) yields the system
\[
\begin{align*}
\text{if}^2 \text{ true } (s x)(s y) & \rightarrow \gcd^2 - (s y) \\
\text{if}^2 \text{ true } (s x)(s y) & \rightarrow y \\
\gcd^2 (s x)(s y) & \rightarrow \text{if}^2 x (s x)(s y) \\
\text{if}^2 \text{ false } (s x)(s y) & \rightarrow \gcd^2 - (s x) \\
\text{if}^2 \text{ false } (s x)y & \rightarrow x
\end{align*}
\]
and a weakly monotone polynomial interpretation
\[
\begin{align*}
P_{\gcd^2}(X,Y) &= 2X + Y \\
P_{\text{if}^2}(X,Y,Z) &= X + Y + Z \\
P_{\text{true}} &= P_{\text{false}} = 0 \\
P_{\leq}(X,Y) &= Y
\end{align*}
\]
these interpretations yield in particular
- for 20, \( X + Y + 2 > Y + 1 \);
- for 21, \( X + Y + 2 > Y \);
- for 22, \( 2X + Y + 3 > 2X + Y + 2 \);
- for 23, \( X + Y + 2 > X + 1 \);
- for 24, \( X + Y + 2 > X \).

The other rules follow easily enough.

---

For next week: critical pairs, KB completion

**Exercise 5:**
Compute the critical pairs of the following rewrite systems. Which one are locally confluent?

(a) \( s(p(s(y))) \rightarrow y, s(p(x)) \rightarrow p(s(x)) \)
(b) \( 0 + y \rightarrow y, x + 0 \rightarrow x, s(w) + z \rightarrow s(w + z), v + s(k) \rightarrow s(v + k) \)
(c) \( a(x,x) \rightarrow 0, a(y,p(y)) \rightarrow 1 \)
(d) \( a(a(x,y),z) \rightarrow a(x,a(y,z)), a(w,1) \rightarrow w \)

**Solution:**
(a) The critical pairs are determined from the substitutions \( \sigma_1 = [x/s(y)] \) and \( \sigma_2 = [y/p(x)] \) and \( \sigma_3 = [y/p(s(z))] \), where \( \sigma_3 \) is obtained considering \( s(p(s(y))) \) with its renaming \( s(p(s(z))) \).
From these three substitutions we get respectively the following three diagrams:
where the critical pairs are \((s(p(s(y))), y), (p(x), s(p(s(x)))), (p(x), p(s(s(x)))))\) and \((s(p(z)), p(s(z)))\). From the diagrams it follows that the TRS is not locally confluent.

(b) The critical pairs are determined from the substitutions \(\sigma_{1,2} = [x/0, y/0], \sigma_{1,4} = [v/0, y/s(k)], \sigma_{2,3} = [x/s(w), z/0], \sigma_{3,4} = [v/s(w), z/s(k)]\), where \(\sigma_{i,j}\) is defined from the rules \(i\) and \(j\). From these substitutions we get the following diagrams:

\[
\begin{align*}
0 + 0 & \quad 0 + s(k) & \quad s(w) + 0 & \quad s(w) + s(k) \\
0 & \quad s(0 + k) & \quad s(w + o) & \quad s(w + s(k)) & \quad s(s(w + k))
\end{align*}
\]

where the critical pairs are \((0, 0), (s(0 + k), s(k)), (s(w + o), s(w)), (s(w + s(k)), s(s(w + k)))\). From the diagrams it follows that the TRS is locally confluent (actually stronger than that: the diamond property is satisfied).

(c) There are no non-trivial critical pairs since the unification problem \(y = ^7 p(y)\) does not admit any solution. Therefore, the TRS is locally confluent.

(d) The critical pairs are determined from the substitutions \(\sigma_1 = [w/a(x, y), z/1], \sigma_2 = [x/w, y/1]\) and \(\sigma_3 = [x/a(x', y'), y'/z'],\) where \(\sigma_3\) is obtained considering \(a(a(x, y), z)\) with its renaming \(a(x', y'), z').\) From these three substitutions we get respectively the following diagrams:

\[
\begin{align*}
a(a(x, y), 1) & \quad a(a(w, 1), z) \\
a(x, a(y, 1)) & \quad a(x, y) & \quad a(w, z) & \quad a(w, a(1, z))
\end{align*}
\]

\[
\begin{align*}
a(a(x', y'), z'), z & \\
a(a(x', y'), z'), z & \quad a(a(x', y'), a(z', z))
\end{align*}
\]

\[
\begin{align*}
a(x', a(y', z'), z) & \quad a(x', a(y', z'), z)
\end{align*}
\]

where the critical pairs are \((a(x, a(y, 1)), a(x, y))\) and \((a(w, z), a(w, a(1, z)))\). From the diagrams it follows that the TRS is not locally confluent.

Exercise 6:
Is the TRS consisting of the rewriting rules

\[
\begin{align*}
0 + x & \rightarrow x & \text{gcd} \ x \ 0 & \rightarrow x & \text{gcd} \ (x + y) \ x & \rightarrow \text{gcd} \ x \ y \\
s \ x + y & \rightarrow s \ (x + y) & \text{gcd} \ 0 \ x & \rightarrow x & \text{gcd} \ x \ (x + y) & \rightarrow \text{gcd} \ x \ y
\end{align*}
\]

confluent?

Solution:
There are two non trivial overlaps,
\[
\text{gcd} (s \cdot x + y) (s \cdot x) \quad \text{gcd} (s \cdot x) ((s \cdot x) + y)
\]
\[
gcd (s \cdot x) y \quad \text{gcd} (s \cdot (x + y)) (s \cdot x) \quad \text{gcd} (s \cdot x) (s \cdot (x + y))
\]

\[(26)\]

**Exercise 7:**
Complete the ES consisting of the equation \((x \cdot y) \cdot (y \cdot z) \approx y\) (of central groupoids).

**Solution:**
Let \(>\) be an arbitrary simplification order. Because of the subterm property, we have \((x \cdot y) \cdot (y \cdot z) > y\), and thus, \(R_0 = \{(x \cdot y) \cdot (y \cdot z) \rightarrow y\}\).

Overlapping the left-hand side with its renamed copy \((x' \cdot y') \cdot (y' \cdot z') \rightarrow y'\) yields two critical pairs

- \(x \cdot y\) unifies with \((x' \cdot y') \cdot (y' \cdot z')\) with mgu \(\{x \mapsto x', y \mapsto y' \cdot z'\}\):

\[
((x' \cdot y') \cdot (y' \cdot z')) \cdot ((y' \cdot z') \cdot z)
\]
\[
y' \cdot z' \quad (y' \cdot z') \cdot z
\]

- \(y \cdot z\) unifies with \((x' \cdot y') \cdot (y' \cdot z')\) with mgu \(\{y \mapsto x' \cdot y', z \mapsto y' \cdot z'\}\),

\[
(x \cdot (x' \cdot y')) \cdot ((x' \cdot y') \cdot (y' \cdot z'))
\]
\[
x' \cdot y' \quad (x \cdot (x' \cdot y')) \cdot y'
\]

Since the terms in these critical pairs are \(R_0\)-irreducible and can be ordered by the subterm property, we obtain the new rewrite system \(R_1\)

\[
\{(x \cdot y) \cdot (y \cdot z) \rightarrow y, x \cdot ((x \cdot y) \cdot z) \rightarrow x \cdot y, (x \cdot (y \cdot z)) \cdot z \rightarrow y \cdot z\}
\]

Next iteration yields only joinable critical pairs.