TD 2012 : Réécriture et bonne année

Ioana Pasca, Marc Lasson

Exercice 1. *Confluence des systèmes de réécriture de termes*

1. Proposez r_1 et r_2 tels que le système $\{f(g(x)) \to r_1, g(h(x)) \to r_2\}$ soit confluent.

2. Est-ce que le système de réécriture de termes $R = \{f(g(f(x))) \to g(x)\}$ est confluent ?
Trouver un système R' convergent (i.e., terminant et confluent) tel que $\approx_R = \approx_{R'}$.

Exercice 2. *Retour sur le cours*

On rappelle qu'une paire critique pour un système de réécriture de termes est la donnée
– de deux règles $l_1 \to r_1$, $l_2 \to r_2$ telles que $Vars(l_1) \cap Vars(l_2) = \emptyset$ (ce que l'on peut toujours supposer, modulo renommage des règles),
– d'une position p de l_1 telle que $l_{1|p}$ n'est pas une variable,
– et d'un mgu θ pour le problème d'unification $\{l_{1|p} \stackrel{?}{=} l_2\}$.
La paire critique correspondante est alors $(\,\theta(r_1), (\theta(l_1)[\theta(r_2)]_p\,)$ — il s'agit des deux termes vers lesquels peut se réécrire l_1.

1. En quoi la situation où $l_{1|p}$ est une variable est-elle particulière ?

2. Il est facile de voir que si un système de réécriture de termes est localement confluent, alors toutes ses paires critiques sont joignables.
Démontrer la réciproque.

Exercice 3.

Une règle $l \to r$ est appelée **linéaire à gauche** (resp. **à droite**) si toutes les variables apparaisent au plus une fois dans l (resp. r). La règle est appelée **linéaire** si elle est linéaire à gauche et à droite. Un TRS est appelé linéaire à gauche (resp. linéaire à droite, resp. linéaire) si toutes ses règles sont linéaires à gauche (resp. linéaires à droite, resp. linéaires).
Deux termes s_1 et s_2 sont **fortement joignables** par rapport à $\to$ s'il existe des termes t_1 et t_2 tels que

$$s_1 \to^{=} t_1 \;{}^{*}\!\!\leftarrow s_2 \quad \text{et} \quad s_1 \to^{*} t_2 \;{}^{=}\!\!\leftarrow s_2$$

$(\to^{=} \,=\, \to \cup \{(t,t)\})$. Montrer que si R est linéaire et toute paire critique de R est fortement joignable, alors R est fortement confluent, i.e. $y_1 \leftarrow x \to y_2$ implique l'existence de z tel que $y_1 \to^{*} z \;{}^{=}\!\!\leftarrow y_2$ (du fait de la symétrie, on doit aussi avoir un z' tel que $y_1 \to^{=} z' \;{}^{*}\!\!\leftarrow y_2$).

Exercice 4.

On dit que R est **réduit à gauche** si pour tout $(l \to r) \in R$, l est en forme normale par rapport à $R \setminus \{l \to r\}$.

1. Montrer que si un TRS est réduit à gauche, terminant et clos (en anglais, *ground* – pas de variables dans les règles de réécriture) alors il est confluent.

2. Soit E un ensemble fini d'identités closes sur Σ et soit $>$ un ordre de réduction qui est total sur les termes clos sur Σ. Décrire un algorithme qui transforme E en un TRS R fini, réduit à gauche tel que $\approx_E = \approx_R$ et $R \subseteq >$ (où $\approx_E$ est l'égalité engendrée par l'ensemble des identités E).

3. En déduire que le problème du mot est décidable pour un ensemble fini d'identités closes.

Exercice 5. *Interprétations Polynômiales – Retour de la terminaison*

1. Pour chacun des systèmes suivants, déterminer s'il termine ou non en utilisant la méthode d'interpretation polynômiale.

a) $$\begin{cases} x + 0 \to x \\ x + S(y) \to S(x+y) \\ x \times 0 \to 0 \\ x \times S(y) \to (x \times y) + x \end{cases}$$

b) $$\begin{cases} \neg\neg x \to x \\ \neg(x \wedge y) \to (\neg x) \vee (\neg y) \\ \neg(x \vee y) \to (\neg x) \wedge (\neg y) \end{cases}$$

c) $$\begin{cases} \neg\neg x \to x \\ \neg(x \vee y) \to (\neg x) \vee (\neg y) \\ \neg(x \wedge y) \to (\neg x) \wedge (\neg y) \\ x \wedge (y \vee z) \to (x \wedge y) \vee (x \wedge z) \\ (x \vee y) \wedge z \to (x \wedge z) \vee (y \wedge z) \end{cases}$$

d) $$\begin{cases} f(f(x,y),z) \to f(x, f(y,z)) \\ f(x, f(y,z)) \to f(y,y) \end{cases}$$

e) $$\begin{cases} or(x,y) \to x \\ or(x,y) \to y \\ f(a,b,x) \to f(x,x,x) \end{cases}$$

f) $$\begin{cases} a(0,x) \to s(x) \\ a(s(x),0) \to a(x,s(0)) \\ a(s(x),s(y)) \to a(x, a(s(x),y)) \end{cases}$$

2. Prouver que si un système de réécriture R peut être prouvé terminant grâce à cette méthode, alors on peut trouver une constante $c > 0$ telle que pour tout terme t, on peut borner le nombre de réductions à partir de t par $2^{2^{c|t|}}$.

Indice : Soit $a \in \mathcal{A}$, prendre $c \geqslant km + \log d$ avec k, m et d tels que

$$a \leqslant d \text{ et } \forall f \in \Sigma_h : f^{\mathcal{A}}(a_1, ..., a_h) \leqslant d \cdot \prod_{i=1}^{h} {a_i}^k \text{ et } h \leqslant m$$

et essayez de borner $\pi_a(t)$ où π_a est le morphisme qui envoie toutes les variables sur a.

3. En déduire que vous ne pouviez pas traiter le dernier exemple de la question 1 avec cette méthode.