TD9 27 mars 2014

Logique

David Baelde
<baelde@lsv.ens-cachan.fr>

Exercice 1 (Arbre-forme résolue)
Un problème d'unification est en arbre-forme résolue s'il s'écrit $\perp$, $\top$ ou bien

$$x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$$

où les variables x_i n'apparaissent qu'une seule fois dans le problème. (On ne se préoccupe pas de l'orientation des équations : $t \stackrel{?}{=} x$ est autorisé au même titre que $x \stackrel{?}{=} t$.)

1. Donner un sous-ensemble minimal des règles de simplification du cours qui soit suffisant pour réduire tout problème en un problème en arbre-forme résolue.

Exercice 2 (DAG-forme résolue)
Un problème d'unification est en DAG-forme résolue s'il s'écrit $\perp$, $\top$ ou bien

$$x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$$

avec les variables x_i distinctes et pour tout $i \leq j$, $x_j \notin \mathrm{Var}(t_i)$.

1. Justifier qu'il s'agit bien d'une forme résolue, et montrer que le mgu d'une DAG-forme résolue peut être exponentiellement plus grand que le problème lui-même. (On prendra comme notions tailles le nombre de constantes et symboles de fonctions.)
2. Proposer une stratégie d'application des règles d'unification qui permette de réduire tout problème en DAG-forme résolue, et exploite l'intérêt de cette forme.

Exercice 3 (RT-forme résolue)
Un problème d'unification est en RT-forme résolue s'il s'écrit $\perp$, $\top$ ou bien

$$x_1 \stackrel{?}{=} t_1 \wedge \ldots \wedge x_n \stackrel{?}{=} t_n$$

où les x_i sont distinctes et où il n'y a pas de cycles de variables $x_{i_1} \stackrel{?}{=} x_{i_2} \wedge x_{i_2} \stackrel{?}{=} x_{i_3} \wedge \ldots \wedge x_{i_n} \stackrel{?}{=} x_{i_1}$.

1. Donner un sous-ensemble minimal des règles permettant de toujours atteindre ces formes résolues.
2. Montrer que ces règles sont correctes si on interprète les termes comme des arbres finis ou infinis.
3. Montrer que toute RT-forme résolue admet une solution pour ces arbres généralisés.
4. Montrer qu'on a défini un *mgu* pour les termes finis ou infinis. Le calculer pour $x \stackrel{?}{=} f(f(x)) \wedge f(x) \stackrel{?}{=} f(f(f(f(x))))$.

Exercice 4 (Résolution au premier ordre)

1. Donner l'ensemble des clauses que l'on peut obtenir par résolution à partir de la formule $\forall x. \neg p(x) \vee p(s(x))$.
2. Donner un modèle ou une preuve d'insatisfaisabilité par résolution pour la théorie suivante :

$$\begin{array}{l}\{\ \forall x. p(x, f(f(x))),\ \forall y.\ p(f(f(f(y))), y), \\ \quad \forall x \forall y \forall z.\ p(x,y) \rightarrow p(y,z) \rightarrow p(x,z),\ \forall x \forall y.\ \neg p(x,y) \vee \neg p(y,x)\ \}\end{array}$$

Exercice 5

On se propose d'abandonner la règle de factorisation, et de traiter les clauses du premier ordre comme des ensembles de littéraux, en identifiant notamment $L \vee L$ et L à $\{L\}$, et $\perp$ à $\emptyset$.

1. Dériver $\perp$ à partir de

$$\forall x \forall y.\ \big(p(x) \vee p(y) \vee q(x,y)\big) \wedge \big(\neg p(x) \vee \neg p(y) \vee q(x,y)\big) \wedge \big(\neg q(x,x)\big)$$

2. On considère maintenant la théorie

$$\{\ \forall x \forall y. (p(x,y) \vee p(y,x)),\ \forall x \forall y. (\neg p(x,y) \vee \neg p(y,x))\ \}$$

Quelles clauses peuvent être dérivées à partir de cette théorie dans notre système modifié ? Que peut on en conclure ?