Typage et programmation
Examen 2003-2004

François Pottier
Francois.Pottier@inria.fr

18 décembre 2003

La plupart des questions peuvent être traitées sans avoir nécessairement répondu à celles qui précèdent.

Un algorithme d'unification du premier ordre

On se donne un ensemble de *constructeurs* F, chacun étant muni d'une arité entière, positive ou nulle, fixe. On suppose que cet ensemble contient au moins un constructeur F_0 d'arité nulle.

Un *chemin* π est une suite finie d'entiers strictement positifs. On note ϵ le chemin de longueur nulle, i le chemin de longueur 1 constitué de la lettre i, si i est un entier strictement positif, et $\pi.\pi'$ la concaténation des chemins π et π'. Un *arbre* t est une fonction partielle des chemins vers les constructeurs dont le domaine est non vide, clos par préfixe, et cohérent avec l'arité des constructeurs – trois conditions que l'on peut écrire comme suit :

1. $t(\epsilon)$ est défini ;
2. si $t(\pi.\pi')$ est défini, alors $t(\pi)$ l'est également ;
3. $t(\pi.i)$ est défini si et seulement si le constructeur $t(\pi)$ a une arité supérieure ou égale à i.

Un arbre est dit *fini* si son domaine est un ensemble fini, *infini* dans le cas contraire. L'intérêt de cette définition des arbres est d'englober les arbres finis et infinis dans un cadre uniforme.

Question 1. *Supposons donnés deux constructeurs* `int` *et* $\rightarrow$*, d'arités respectives 0 et 2. Quels arbres peut-on naturellement associer aux notations usuelles* `int` $\rightarrow$ `int` *et* μ`X.int` $\rightarrow$ `X` *? (On en explicitera le domaine et on indiquera, parmi les chemins éléments du domaine, lesquels ont pour image* `int` *et lesquels ont pour image* $\rightarrow$*.) Ces arbres sont-ils finis ou infinis ?* ◇

Solution. L'arbre correspondant à `int` $\rightarrow$ `int` a pour domaine l'ensemble $\{\epsilon, 1, 2\}$. Il s'agit donc d'un arbre fini. L'image du chemin ϵ est $\rightarrow$, tandis que l'image des chemins 1 et 2 est `int`.

L'arbre correspondant à μ`X.int` $\rightarrow$ `X` a pour domaine l'ensemble des chemins de la forme 2^* ou 2^*1. Il s'agit donc d'un arbre infini. L'image des chemins de la première forme est $\rightarrow$, tandis que l'image des chemins de la seconde forme est `int`. □

Si $t(\pi)$ est défini, le *sous-arbre de t à la position π*, noté t/π, est l'arbre qui à π' associe $t(\pi.\pi')$. Un arbre est dit *régulier* s'il admet un nombre fini de sous-arbres distincts. Si F est un constructeur d'arité n et si $t_1, \ldots, t_n$ sont des arbres, on notera $F(t_1, \ldots, t_n)$ l'arbre t défini par $t(\epsilon) = F$ et $t(i.\pi) = t_i(\pi)$ pour $i \in \{1, \ldots, n\}$.

Question 2. *On définit souvent les arbres finis comme les termes engendrés par la définition inductive suivante :*

$$T ::= F\ T_1\ \ldots\ T_n \qquad \text{où } n \text{ est l'arité de } F$$

Indiquer comment tout terme T de cette forme peut être interprété comme un arbre au sens de la définition précédente, noté $[\![T]\!]$, et vérifier que cet arbre est fini. Réciproquement, vérifier que tout arbre fini est l'interprétation d'un tel terme. ◇

Solution. On définit, par induction structurelle, une interprétation (c'est-à-dire une fonction) des termes vers les arbres, de la façon suivante :

$$[\![F\ T_1\ \ldots\ T_n]\!] = F([\![T_1]\!], \ldots, [\![T_n]\!])$$

On prouve ensuite, toujours par induction structurelle, que l'arbre $[\![T]\!]$ est fini. Notons que cette preuve ne comporte qu'un cas ; le « cas de base » correspond, si l'on veut, au cas où n est nul. Par hypothèse d'induction, $[\![T_1]\!], \ldots, [\![T_n]\!]$ sont des arbres finis. Leurs domaines $\Pi_1, \ldots, \Pi_n$ sont donc des ensembles finis. Le domaine de $[\![F\ T_1\ \ldots\ T_n]\!]$, qui par définition est $\{\epsilon\} \cup 1.\Pi_1 \cup \ldots \cup n.\Pi_n$, est donc une union finie d'ensembles finis, soit un ensemble fini également. L'arbre $[\![F\ T_1\ \ldots\ T_n]\!]$ est donc fini.

Réciproquement, nous devons prouver que tout arbre fini t appartient à l'image de $[\![\cdot]\!]$. La preuve se fait par induction sur le cardinal (du domaine) de t. On vérifie facilement que, si $t(\epsilon)$ est un constructeur F d'arité n, alors t peut s'écrire $F(t/1, \ldots, t/n)$. Comme plus haut, si le domaine de t/i est Π_i, alors le domaine de t contient $\{\epsilon\} \cup i.\Pi_i$, ce qui montre que si t est de cardinal fini, alors t/i est de cardinal fini et strictement inférieur. L'hypothèse d'induction s'applique alors à t/i, qui peut donc s'écrire $[\![T_i]\!]$ pour un certain terme T_i. Le terme t s'écrit ainsi $[\![F\ T_1\ \ldots\ T_n]\!]$. □

Dans la suite, on s'intéresse à la résolution de contraintes d'égalité entre termes avec variables. On se donne un ensemble infini de *variables* $\mathtt{X}$. La syntaxe des *termes* $\mathtt{T}$ est alors définie inductivement :

$$\mathtt{T} ::= \mathtt{X} \mid F\ \mathtt{T}_1\ \ldots\ \mathtt{T}_n \qquad \text{où } n \text{ est l'arité de } F$$

Une *multi-équation* e est un ensemble de termes. La syntaxe des contraintes est alors la suivante :

$$C ::= \mathtt{false} \mid \mathtt{T} =^? \mathtt{T} \mid e \mid C \wedge C \mid \exists \mathtt{X}.C$$

On a distingué les équations binaires $\mathtt{T}_1 =^? \mathtt{T}_2$ des multi-équations e, car ces deux formes satisferont des invariants différents et seront représentées de façon différente en mémoire. Néanmoins, d'un point de vue logique, il s'agit, dans les deux cas, de contraintes d'égalité.

On peut se donner (au moins) trois interprétations naturelles de ces contraintes.

Question 3. *Définir une interprétation des termes et des contraintes dans l'ensemble de tous les arbres (finis ou infinis). (On définira la notion de* valuation ϕ *et les prédicats de* satisfaction $\vdash$*, d'implication* $\Vdash$ *et d'*équivalence $\equiv$ *de contraintes.) Indiquer comment la modifier pour obtenir une interprétation dans l'ensemble des arbres réguliers, ou encore une interprétation dans l'ensemble des arbres finis.* ◇

Solution. Notons ϕ une *valuation*, c'est-à-dire une fonction totale des variables $\mathtt{X}$ vers les arbres (finis ou infinis). On dit que ϕ *satisfait* (i) la contrainte $\mathtt{false}$ jamais ; (ii) une équation $\mathtt{T}_1 =^? \mathtt{T}_2$ si et seulement si $\phi(\mathtt{T}_1)$ et $\phi(\mathtt{T}_2)$ coïncident ; (iii) une multi-équation e si et seulement si tous les éléments de e ont une même image à travers ϕ ; (iv) une conjonction $C_1 \wedge C_2$ si et seulement si ϕ satisfait C_1 et satisfait C_2 ; (v) une abstraction $\exists \mathtt{X}.C$ si et seulement si il existe une valuation ϕ' qui coïncide avec ϕ hors de $\{\mathtt{X}\}$ et qui satisfait C. Une contrainte C_1 *implique* une contrainte C_2 si et seulement si toute valuation qui satisfait C_1 satisfait C_2. Deux contraintes sont *équivalentes* si et seulement si elles s'impliquent mutuellement.

Si on modifie la définition ci-dessus en exigeant qu'une valuation soit une fonction totale des variables vers les arbres *réguliers*, on obtient une seconde interprétation. De même, on peut exiger qu'une valuation soit une fonction totale des variables vers les arbres *finis*, ce qui donne une troisième interprétation. □

On note $\bar{\mathtt{X}}$ une multi-équation dont tous les membres sont des variables. On note $\mathtt{T} = e$ ou $e = \mathtt{T}$ la multi-équation $\{\mathtt{T}\} \cup e$, c'est-à-dire la multi-équation dont les membres sont $\mathtt{T}$ et les membres de e. On pourra noter $\mathtt{X}$ la multi-équation $\{\mathtt{X}\}$, dont l'unique membre est $\mathtt{X}$. On note $\mathbb{T}$ un *contexte*, c'est-à-dire un terme doté d'un trou :

$$\mathbb{T} ::= [] \mid F\ \mathtt{T}_1\ \ldots\ \mathbb{T}\ \ldots\ \mathtt{T}_n$$

L'algorithme d'unification est alors défini par le système de réécriture suivant :

$$
\begin{array}{rcll}
\mathtt{T} =^? \mathbb{T}[F\,\mathtt{X}_1\,\ldots\,\mathtt{X}_n] & \to & \exists \mathtt{X}.(\mathtt{T} =^? \mathbb{T}[\mathtt{X}] \wedge \mathtt{X} = F\,\mathtt{X}_1\,\ldots\,\mathtt{X}_n) & \textsc{Découpage} \\
 & & \text{si } \mathtt{X} \notin \mathrm{ftv}(\mathtt{T}, \mathbb{T}, \mathtt{X}_1, \ldots, \mathtt{X}_n) & \\
(\exists \mathtt{X}.C_1) \wedge C_2 & \to & \exists \mathtt{X}.(C_1 \wedge C_2) & \textsc{Extrusion} \\
 & & \text{si } \mathtt{X} \notin \mathrm{ftv}(C_2) & \\
\mathtt{X}_1 =^? \mathtt{X}_2 \wedge \mathtt{X}_1 = \mathtt{X}_2 = e & \to & \mathtt{X}_1 = \mathtt{X}_2 = e & \textsc{Tautologie} \\
\mathtt{X}_1 =^? \mathtt{X}_2 \wedge \mathtt{X}_1 = \bar{\mathtt{X}}_1 \wedge \mathtt{X}_2 = e & \to & \mathtt{X}_1 = \bar{\mathtt{X}}_1 = \mathtt{X}_2 = e & \textsc{Fusion} \\
\mathtt{X}_1 =^? \mathtt{X}_2 \wedge \mathtt{X}_1 = \bar{\mathtt{X}}_1 = F\,\mathtt{X}_1^1\,\ldots\,\mathtt{X}_1^n & \to & \mathtt{X}_1 = \bar{\mathtt{X}}_1 = \mathtt{X}_2 = \bar{\mathtt{X}}_2 = F\,\mathtt{X}_1^1\,\ldots\,\mathtt{X}_1^n \wedge & \textsc{Propagation} \\
\wedge\, \mathtt{X}_2 = \bar{\mathtt{X}}_2 = F\,\mathtt{X}_2^1\,\ldots\,\mathtt{X}_2^n & & \wedge_{i=1}^n \mathtt{X}_1^i =^? \mathtt{X}_2^i & \\
\mathtt{X}_1 =^? \mathtt{X}_2 \wedge \mathtt{X}_1 = \bar{\mathtt{X}}_1 = F_1\,\mathtt{X}_1^1\,\ldots\,\mathtt{X}_1^{n_1} & \to & \mathtt{false} & \textsc{Échec} \\
\wedge\, \mathtt{X}_2 = \bar{\mathtt{X}}_2 = F_2\,\mathtt{X}_2^1\,\ldots\,\mathtt{X}_2^{n_2} & & & \\
 & & \text{si } F_1 \neq F_2 & \\
C \wedge \mathtt{false} & \to & \mathtt{false} & \textsc{Échec-}\wedge \\
\exists \mathtt{X}.\mathtt{false} & \to & \mathtt{false} & \textsc{Échec-}\exists
\end{array}
$$

La réécriture se fait modulo commutativité et associativité de la conjonction, modulo commutativité du symbole $=^?$, et modulo un contexte arbitraire.

On peut vérifier que la réécriture préserve l'invariant suivant : soit la contrainte courante contient `false`, soit (i) toute variable (qu'elle soit libre ou existentiellement quantifiée) est membre d'exactement une multi-équation ; et (ii) toute multi-équation est de la forme soit $\bar{\mathtt{X}}$ soit $\bar{\mathtt{X}} = F\,\mathtt{X}_1\,\ldots\,\mathtt{X}_n$. (On ne demande pas de démontrer cela.) Dans la suite, on ne s'intéressera qu'à des contraintes satisfaisant cet invariant. Les propriétés (i) et (ii) indiquent que les multi-équations constituent une *partition* des variables du problème, et qu'à chaque classe d'équivalence peut être associé un terme de la forme $F\,\mathtt{X}_1\,\ldots\,\mathtt{X}_n$.

Question 4. *À titre d'exemple, supposons qu'on dispose des constructeurs* `int` *et* `bool`*, d'arité* 0*, et des constructeurs* $\to$ *et* $\times$*, d'arité* 2*. On considère la contrainte suivante, laquelle possède les propriétés (i) et (ii) définies plus haut :*

$$\mathtt{X} \wedge \mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{X} =^? \mathtt{int} \to (\mathtt{Y} \times \mathtt{Z}) \wedge \exists \mathtt{Y}.(\mathtt{Y} \wedge \mathtt{W} = \mathtt{bool} \wedge \mathtt{X} =^? \mathtt{Y} \to (\mathtt{W} \times \mathtt{W}))$$

Quelles en sont les variables libres ? Donner une série de réductions permettant d'amener cette contrainte en forme normale, en indiquant le nom de la (des) règle(s) employée(s) à chaque étape. Indiquer, sans justification, si cette contrainte est satisfiable et quelles sont les valuations qui la satisfont. ◇

Solution. L'algorithme étant non déterministe, il existe plusieurs séquences de réduction ayant pour origine la contrainte ci-dessus ; en voici une.

$$
\begin{array}{rcl}
 & & \mathtt{X} \wedge \mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{X} =^? \mathtt{int} \to (\mathtt{Y} \times \mathtt{Z}) \wedge \exists \mathtt{Y}.(\mathtt{Y} \wedge \mathtt{W} = \mathtt{bool} \wedge \mathtt{X} =^? \mathtt{Y} \to (\mathtt{W} \times \mathtt{W})) \\
(\textsc{Découpage}) & \to^* & \mathtt{X} \wedge \mathtt{Y} \wedge \mathtt{Z} \wedge \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3.(\mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X}_2 = \mathtt{Y} \times \mathtt{Z} \wedge \mathtt{X}_3 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \mathtt{X} =^? \mathtt{X}_3) \wedge \\
 & & \exists \mathtt{Y}.(\mathtt{Y} \wedge \mathtt{W} = \mathtt{bool} \wedge \exists \mathtt{X}_4\mathtt{X}_5.(\mathtt{X}_4 = \mathtt{W} \times \mathtt{W} \wedge \mathtt{X}_5 = \mathtt{Y} \to \mathtt{X}_4 \wedge \mathtt{X} =^? \mathtt{X}_5)) \\
(\textsc{Extrusion}) & \to^* & \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3\mathtt{Y}'\mathtt{X}_4\mathtt{X}_5.(\mathtt{X} \wedge \mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X}_2 = \mathtt{Y} \times \mathtt{Z} \wedge \mathtt{X}_3 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \mathtt{X} =^? \mathtt{X}_3 \wedge \\
 & & \mathtt{Y}' \wedge \mathtt{W} = \mathtt{bool} \wedge \mathtt{X}_4 = \mathtt{W} \times \mathtt{W} \wedge \mathtt{X}_5 = \mathtt{Y}' \to \mathtt{X}_4 \wedge \mathtt{X} =^? \mathtt{X}_5) \\
(\textsc{Fusion}) & \to & \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3\mathtt{Y}'\mathtt{X}_4\mathtt{X}_5.(\mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X}_2 = \mathtt{Y} \times \mathtt{Z} \wedge \mathtt{X} = \mathtt{X}_3 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \\
 & & \mathtt{Y}' \wedge \mathtt{W} = \mathtt{bool} \wedge \mathtt{X}_4 = \mathtt{W} \times \mathtt{W} \wedge \mathtt{X}_5 = \mathtt{Y}' \to \mathtt{X}_4 \wedge \mathtt{X} =^? \mathtt{X}_5) \\
(\textsc{Propagation}) & \to & \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3\mathtt{Y}'\mathtt{X}_4\mathtt{X}_5.(\mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X}_2 = \mathtt{Y} \times \mathtt{Z} \wedge \mathtt{X} = \mathtt{X}_3 = \mathtt{X}_5 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \\
 & & \mathtt{Y}' \wedge \mathtt{W} = \mathtt{bool} \wedge \mathtt{X}_4 = \mathtt{W} \times \mathtt{W} \wedge \mathtt{Y}' =^? \mathtt{X}_1 \wedge \mathtt{X}_4 =^? \mathtt{X}_2) \\
(\textsc{Fusion}) & \to & \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3\mathtt{Y}'\mathtt{X}_4\mathtt{X}_5.(\mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{Y}' = \mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X}_2 = \mathtt{Y} \times \mathtt{Z} \wedge \mathtt{X} = \mathtt{X}_3 = \mathtt{X}_5 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \\
 & & \mathtt{W} = \mathtt{bool} \wedge \mathtt{X}_4 = \mathtt{W} \times \mathtt{W} \wedge \mathtt{X}_4 =^? \mathtt{X}_2) \\
(\textsc{Propagation}) & \to & \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3\mathtt{Y}'\mathtt{X}_4\mathtt{X}_5.(\mathtt{Y} \wedge \mathtt{Z} \wedge \mathtt{Y}' = \mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X} = \mathtt{X}_3 = \mathtt{X}_5 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \\
 & & \mathtt{W} = \mathtt{bool} \wedge \mathtt{X}_2 = \mathtt{X}_4 = \mathtt{W} \times \mathtt{W} \wedge \mathtt{Y} =^? \mathtt{W} \wedge \mathtt{Z} =^? \mathtt{W}) \\
(\textsc{Fusion}) & \to^2 & \exists \mathtt{X}_1\mathtt{X}_2\mathtt{X}_3\mathtt{Y}'\mathtt{X}_4\mathtt{X}_5.(\mathtt{Y}' = \mathtt{X}_1 = \mathtt{int} \wedge \mathtt{X} = \mathtt{X}_3 = \mathtt{X}_5 = \mathtt{X}_1 \to \mathtt{X}_2 \wedge \\
 & & \mathtt{Y} = \mathtt{Z} = \mathtt{W} = \mathtt{bool} \wedge \mathtt{X}_2 = \mathtt{X}_4 = \mathtt{W} \times \mathtt{W})
\end{array}
$$

La contrainte finale est bien en forme normale : elle ne contient aucune équation à résoudre, ne contient pas `false`, et présente tous ses quantificateurs existentiels en position prénexe. Aucune règle de réduction ne lui

est donc applicable. On constate aisément, d'après sa forme, que les valuations qui la satisfont sont celles qui à $\mathtt{X}$ associent l'arbre $\mathtt{int} \to (\mathtt{bool} \times \mathtt{bool})$ et à $\mathtt{Y}, \mathtt{Z}, \mathtt{W}$ associent l'arbre $\mathtt{bool}$. La contrainte initiale, qui lui est équivalente, est satisfaite par les mêmes valuations. L'ensemble de ces valuations étant non vide, elle est satisfiable. □

Question 5. *Démontrer que toute forme normale (c'est-à-dire toute contrainte irréductible) vis-à-vis de* $\to$ *est soit* `false`*, soit de la forme* $\exists \bar{\mathtt{X}}.C$*, où* C *est une conjonction de multi-équations satisfaisant les propriétés (i) et (ii) définies plus haut.* ◇

Solution. Considérons une forme normale, c'est-à-dire une contrainte irréductible. Si elle contient `false`, alors, parce qu'elle n'est pas réductible par ÉCHEC-$\wedge$ ou ÉCHEC-$\exists$, elle ne peut être que `false` lui-même. Si elle ne contient pas `false`, alors, puisqu'elle elle satisfait par hypothèse l'invariant donné plus haut, elle doit satisfaire les propriétés (i) et (ii). De plus, parce qu'elle n'est pas réductible par EXTRUSION, elle doit être de la forme $\exists \bar{\mathtt{X}}.C$, où C satisfait (i) et (ii). Il reste à montrer que C est nécessairement une conjonction de multi-équations, c'est-à-dire que C ne peut contenir aucune équation à résoudre $\mathtt{T}_1 =^? \mathtt{T}_2$. On remarque d'abord que si $\mathtt{T}_2$ n'est pas une variable, alors une telle équation serait réductible par DÉCOUPAGE, et de même si $\mathtt{T}_1$ n'est pas une variable, puisque la réduction se fait modulo commutativité du symbole $=^?$. Intéressons-nous donc à une équation $\mathtt{X}_1 =^? \mathtt{X}_2$. D'après la propriété (i), $\mathtt{X}_1$ et $\mathtt{X}_2$ sont chacune membre d'exactement une multi-équation dans C. S'il s'agit de la même multi-équation, alors $\mathtt{X}_1 =^? \mathtt{X}_2$ est réductible par TAUTOLOGIE, ce qui est impossible. Il s'agit donc de deux multi-équations distinctes. D'après la propriété (ii), chacune est de la forme soit $\bar{\mathtt{X}}$ soit $\bar{\mathtt{X}} = F\,\mathtt{X}_1 \ldots \mathtt{X}_n$. Si l'une est de la première forme, alors $\mathtt{X}_1 =^? \mathtt{X}_2$ est réductible par FUSION. Si les deux sont de la seconde forme, alors $\mathtt{X}_1 =^? \mathtt{X}_2$ est réductible par PROPAGATIONou ÉCHEC, selon qu'elles font apparaître ou non le même constructeur. Dans tous les cas, nous obtenons donc une contradiction, ce qui montre bien que C ne peut contenir aucune équation à résoudre $\mathtt{T}_1 =^? \mathtt{T}_2$. □

Question 6. *Démontrer que, si termes et contraintes sont interprétés dans l'ensemble de tous les arbres (finis ou infinis), alors toute forme normale autre que* `false` *est satisfiable. Pour cela, on s'intéressera à une conjonction* C *de multi-équations satisfaisant les propriétés (i) et (ii) définies plus haut, et on exhibera une valuation* ϕ *qui la satisfait.* ◇

Solution. Considérons une contrainte C de la forme ci-dessus. On dira qu'une variable $\mathtt{X}$ est *contrainte à* $F\,\mathtt{X}_1 \ldots \mathtt{X}_n$ si elle apparaît au sein d'une multi-équation de la forme $\bar{\mathtt{X}} = F\,\mathtt{X}_1 \ldots \mathtt{X}_n$, et *non contrainte* sinon. D'après les propriétés (i) et (ii), toute variable est soit contrainte à un terme unique, soit non contrainte.

On exhibe alors une valuation ϕ en définissant la famille des $\phi(\mathtt{X})(\pi)$, où $\mathtt{X}$ parcourt $\mathrm{ftv}(C)$, par induction sur la longueur du chemin π :

$$\begin{array}{rcll} \phi(\mathtt{X})(\epsilon) & = & F_0 & \text{si } \mathtt{X} \text{ est non contrainte} \\ \phi(\mathtt{X})(\epsilon) & = & F & \text{si } \mathtt{X} \text{ est contrainte à } F\,\mathtt{X}_1 \ldots \mathtt{X}_n \\ \phi(\mathtt{X})(i.\pi) & = & \phi(\mathtt{X}_i)(\pi) & \text{si } \mathtt{X} \text{ est contrainte à } F\,\mathtt{X}_1 \ldots \mathtt{X}_n \text{ et } i \in \{1, \ldots, n\} \end{array}$$

On vérifie alors que, pour tout $\mathtt{X}$, la fonction $\phi(\mathtt{X})$ est bien un arbre. En effet, $\phi(\mathtt{X})(\epsilon)$ est toujours défini, ce qui donne (i). Par ailleurs, on démontre par induction sur π que si $\phi(\mathtt{X})(\pi.j)$ est défini, alors $\phi(\mathtt{X})(\pi)$ l'est également. Si π est ϵ, c'est immédiat. Sinon, π est de la forme $i.\pi'$. Par hypothèse, $\phi(\mathtt{X})(i.\pi'.j)$ est défini ; on peut l'écrire $\phi(\mathtt{X}_i)(\pi'.j)$, pour un certain $\mathtt{X}_i$. Par hypothèse d'induction, $\phi(\mathtt{X}_i)(\pi')$ est donc également défini ; or on peut l'écrire $\phi(\mathtt{X})(i.\pi')$, c'est-à-dire $\phi(\mathtt{X})(\pi)$. Ceci donne donc (ii). La propriété (iii) peut également se démontrer par induction ; nous ne le ferons pas explicitement ici.

Il reste à vérifier que ϕ est bien une solution de C. Pour cela, on vérifie qu'elle satisfait bien chaque multi-équation. Le résultat est immédiat, car si deux variables $\mathtt{X}$ et $\mathtt{Y}$ appartiennent à une même multi-équation, alors $\phi(\mathtt{X})$ et $\phi(\mathtt{Y})$ coïncident par construction, et si cette multi-équation contient un terme de la forme $F\,\mathtt{X}_1 \ldots \mathtt{X}_n$, alors, toujours par construction, ils coïncident également avec $F(\phi(\mathtt{X}_1), \ldots, \phi(\mathtt{X}_n))$.

Pour conclure que toute forme normale autre que `false` est satisfiable, il reste à faire appel au résultat de la question précédente et à remarquer que $\exists \bar{\mathtt{X}}.C$ est satisfiable si et seulement si C est satisfiable. □

Question 7. *D'après la question précédente, toute forme normale pour* $\to$ *est soit* `false`*, soit satisfiable. Quelles autres propriétés de* $\to$ *faut-il démontrer pour pouvoir affirmer que le problème de la satisfiabilité d'une contrainte (dans le modèle constitué de tous les arbres, finis ou infinis) est décidable ?* ◇

Solution. On peut démontrer d'une part que la relation $\rightarrow$ est fortement normalisante, c'est-à-dire qu'elle n'admet aucune séquence de réduction infinie, et d'autre part que la réduction préserve l'interprétation des contraintes, c'est-à-dire que $\rightarrow$ est incluse dans $\equiv$. La combinaison de ces deux propriétés indique que toute contrainte C admet une forme normale C', qui lui est logiquement équivalente, et que l'on peut calculer de façon effective. La question précédente montre que C' est satisfiable si et seulement si elle n'est pas `false`. Donc, C est satisfiable si et seulement si C' n'est pas `false`. Nous avons décrit un algorithme permettant de décider si C est satisfiable. □

Question 8. *Vérifier que la valuation ϕ exhibée par la question 6 est en fait régulière, c'est-à-dire que, pour tout* `X`*, ϕ(*`X`*) est un arbre régulier. En déduire qu'une contrainte est satisfiable dans le modèle constitué de tous les arbres (finis ou infinis) si et seulement si elle est satisfiable dans le modèle constitué des arbres réguliers uniquement.* ◇

Solution. On vérifie que si `X` est contrainte à $F\,\mathtt{X}_1 \ldots \mathtt{X}_n$ et $i \in \{1, \ldots, n\}$, alors le sous-arbre $\phi(\mathtt{X})/i$ n'est autre que $\phi(\mathtt{X}_i)$. Par ailleurs, si `X` est non contrainte, alors ϕ(`X`) n'admet pas de sous-arbres. On peut alors démontrer, par induction sur π, que pour tout `X` et pour tout π, le sous-arbre $\phi(\mathtt{X})/\pi$, s'il est défini, est élément de l'ensemble des $\phi(\mathtt{Y})$, où `Y` varie parmi les variables de C. Cet ensemble est fini, ce qui montre que, pour tout `X`, l'arbre ϕ(`X`) est régulier.

Il en découle que, quel que soit le modèle choisi (arbres quelconques ou arbres réguliers), les énoncés démontrés jusqu'ici sont inchangés, et l'algorithme de décision du problème de satisfiabilité est identique. Par conséquent, les contraintes satisfiables sont les mêmes dans les deux modèles. □

Question 9. *De quelle façon l'algorithme doit-il être modifié pour décider le problème de la satisfiabilité dans le modèle des arbres finis ? De quelle façon le développement précédent, et en particulier la preuve qui répond à la question 6, est-il alors modifié ?* ◇

Solution. Considérons une conjonction C de multi-équations satisfaisant les propriétés (i) et (ii) définies plus haut. Si `X` est contrainte à $F\,\mathtt{X}_1 \ldots \mathtt{X}_n$, alors on dira que `X` *domine* $\mathtt{X}_i$, pour $\mathtt{X} \in \{1, \ldots, n\}$. Si le graphe de la relation de domination contient un cycle, alors on dira que C est *cyclique*. On ajoute alors à l'algorithme un test d'acyclicité, sous forme d'une nouvelle règle :

$$C \rightarrow \mathtt{false} \qquad \text{si } C \text{ est cyclique}$$

La preuve qui répond à la question 6 est alors modifiée de la façon suivante. La construction de ϕ est inchangée : il faut simplement vérifier que, pour tout `X`, ϕ(`X`) est un arbre fini. Pour cela, on démontre, par induction sur π, que si $\phi(\mathtt{X})(\pi)$ est défini, alors le graphe de domination associé à C admet un chemin de longueur $\mid \pi \mid$. Si C est irréductible vis-à-vis de notre nouvelle règle, alors C est acyclique, donc les chemins de ce graphe sont de longueur bornée. Par conséquent, les chemins π tels que $\phi(\mathtt{X})(\pi)$ est défini sont de longueur bornée. Or, ces chemins sont des mots sur un alphabet fini : chaque lettre i est comprise entre 1 et l'arité maximale des constructeurs présents dans C. Par conséquent, le domaine de ϕ(`X`) est fini.

Le reste du développement est modifié comme suit. D'abord, on vérifie que la nouvelle règle préserve l'interprétation des contraintes, c'est-à-dire qu'une contrainte cyclique ne peut admettre de solution. Cela revient à supposer l'existence d'une valuation ϕ satisfaisant une contrainte cyclique C, et à en tirer une contradiction. On peut obtenir celle-ci en démontrant que ϕ(`X`) est nécessairement un arbre infini, si `X` est un point du cycle. Ensuite, on vérifie que le système reste fortement normalisant, ce qui est immédiat, car le membre droit de notre nouvelle règle est une forme normale. □