Arbres de syntaxe anonymes

Sémantique – devoir maison

Master 1 Informatique – Université Paris Cité

Version du 22 avril 2024

1 Introduction

Le cours vous a familiarisé avec la notion d'arbre de syntaxe abstraite avec lieurs, qui forme la base de la manipulation pratique et théorique des langages de programmation. Nous avons décrit une représentation concrète de ces arbres où les variables sont implémentées à l'aide d'un ensemble infini dénombrable de *noms*. Cette implémentation, appelée *représentation nommée*, est très intuitive puisque c'est celle que nous utilisons tous les jours lorsque nous écrivons des programmes ou des énoncés mathématiques. Cette simplicité a toutefois un prix : elle impose de considérer que deux arbres sont équivalents lorsqu'ils sont identiques à un renommage de leurs variables liées près. Nous avons pu constater que la relation d'équivalence correspondante, l'*équivalence* α, quoique conceptuellement simple, est d'un usage fort déplaisant, qu'il s'agisse de raisonner rigoureusement ou d'implémenter efficacement.

Les représentations dites *anonymes* constituent une alternative moins intuitive mais bien plus adaptée à l'implémentation ou au raisonnement sur machine. L'idée est, en quelques mots, de marquer chaque occurrence de chaque variable par un entier qui exprime la distance entre la dite occurrence et le lieur qui a introduit la variable correspondante. Les noms et l'équivalence α deviennent alors superflus. Pour cette raison, les représentations anonymes sont employées dans la plupart des logiciels qui doivent manipuler des termes avec lieurs, comme par exemple l'assistant de preuve Coq. L'objectif de ce devoir est d'étudier ces représentations anonymes, en particulier la conversion vers et depuis une représentation nommée.

Consignes. Vous avez reçu le présent document au sein d'une archive au format ZIP contenant des fichiers LaTeX. Vous devez compléter les fichiers `student-XXX.tex` contenus dans l'archive par vos réponses, puis compiler le document LaTeX manuellement ou bien à l'aide du `Makefile` fourni. Le fichier PDF généré doit être envoyé à l'adresse courriel guatto@irif.fr avant le **20 mai 2024** à 19h00. Votre rendu doit être le fruit d'un **travail individuel**, comme exprimé par la charte anti-plagiat de l'université [1]. Les preuves demandées doivent être détaillées convenablement. Vous veillerez à détailler les principes de raisonnement utilisés, en particulier la nature des inductions réalisées et l'usage des hypothèses d'induction le cas échéant.

1. `https://fr.u-paris.fr/charte-anti-plagiat`

Le document a été rédigé en utilisant la suite logicielle TeX Live 2024 [2]. Il peut sans doute être compilé avec d'autres versions de LaTeX, mais elles n'ont pas été testées.

2 Prolégomènes

2.1 Notations et généralités mathématiques

Les notations et concepts employés sont essentiellement identiques à ceux utilisés dans les notes de cours, mais on rappelle ici quelques notions utiles.

Entiers naturels. Les lettres n, m, p, i, j, k désignent toujours des entiers naturels. La soustraction est bien définie sur les entiers naturels en posant $n - m = 0$ lorsque $n \leq m$. Avec cette définition, la fonction qui à n associe $n - m$, notée $(-m)$, n'est plus l'inverse de la fonction qui a p associe de $p + m$, notée $(+m)$, mais son *adjoint à gauche*, ce qui signifie que $n - m \leq p$ si et seulement si $n \leq p + m$. Cette définition ne vérifie pas toutes les propriétés auxquelles nous ont habitué les entiers relatifs. Notamment, si on a toujours $(n + m) - m = n$, l'égalité $(n - m) + m = n$ n'est en revanche validée que lorsque m est inférieur ou égal à n ; par exemple, $(0 - 1) + 1 = 1 \neq 0$. De même, si on a toujours $n - (m + p) = n - m - p$, l'égalité $n - (p - m) = n - p + m$ n'est validée que lorsque m est inférieur ou égal au minimum de p et de $n - (p - m)$.

Fonctions et images. Soit f est une fonction de domaine A et codomaine B. On rappelle que si P est une partie de A, la notation $P; f$ désigne l'*image directe* de P par f, c'est-à-dire la partie $\{f(x) \mid x \in P\}$ de B. Si P est l'ensemble A tout entier, on parle simplement de l'*image* de f. Si a est un élément de A et b un élément de B, on note $f,\ a \mapsto b$ la fonction qui envoie a' dans b lorsque $a = a'$ et dans $f(a')$ lorsque $a' \neq a$.

Propriété 1. *Soient $f : B \to C$ et $g : A \to B$ des fonctions. Soit a, b et c des éléments des ensembles A, B et C respectivement. On a $(f,\ b \mapsto c) \circ (g,\ a \mapsto b) = ((f,\ b \mapsto c) \circ g),\ a \mapsto c$.*

Soit f une fonction de domaine A et codomaine B d'image finie. Alors $f,\ x \mapsto y$ est d'image finie pour tous x et y. Pour toutes fonctions $g : C \to A$ et $h : B \to D$ d'image pas nécessairement finie, la fonction $g; f; h$ est d'image finie.

Noms. On fixe un ensemble infini dénombrable $\mathbb{A}$ dont les éléments sont appelés *noms*. On désigne par Γ, Δ ou Θ des parties finies de $\mathbb{A}$. On suppose donnée une fonction ν qui associe à toute partie finie Γ de $\mathbb{A}$ un élément $\nu(\Gamma)$ de $\mathbb{A}$ qui n'appartient pas à Γ.

2.2 Signatures du second ordre

Toutes les représentations des termes du second ordre qui vont suivre utilisent la même notion de signature, identique à celle vue en cours. On rappelle qu'une telle signature Σ est la donnée d'un ensemble $|\Sigma|$, dont les éléments sont appelés *opérateurs* de la signature, ainsi que

2. `https://www.tug.org/texlive/` à installer avec le gestionnaire de paquets de votre système.

d'une fonction $\mathrm{ar}_\Sigma : |\Sigma| \to \mathsf{Sort}^*$ dite d'*arité* qui associe à chaque opérateur f de la signature une liste finie de *sortes* $\mathrm{ar}_\Sigma(f)$. L'ensemble des sortes est décrit par la grammaire suivante.

$$\mathsf{Sort} \ni \mathfrak{S}, \mathfrak{T} ::= \mathbb{T} \mid \mathbb{V} \to \mathfrak{S}$$

On écrit $\Sigma \ni f : \mathfrak{S}_1 \to \cdots \to \mathfrak{S}_n \to \mathbb{T}$ pour signifier que f est un opérateur de Σ dont l'arité est la liste $[\mathfrak{S}_1, \ldots, \mathfrak{S}_n]$.

Pour écrire des exemples, on utilisera à plusieurs reprises la signature Σ_λ du λ-calcul pur, sans doute la plus simple des signatures du second-ordre intéressantes. Celle-ci est intégralement spécifiée par les deux opérateurs suivants.

$$\Sigma_\lambda \ni \mathtt{app} : \mathbb{T} \to \mathbb{T} \to \mathbb{T} \qquad\qquad \Sigma_\lambda \ni \mathtt{fun} : (\mathbb{V} \to \mathbb{T}) \to \mathbb{T}$$

On appelle λ-*termes* les termes du second ordre sur la signature Σ_λ.

3 Représentation nommée

On commence par rappeler le fonctionnement de la représentation nommée et de ses termes, qu'on dira *nominaux*. Le cœur de cette représentation, déjà vu en cours, est résumé à la figure 1. La substitution se décompose en, d'une part, la recherche du terme $\sigma(x)$ associé à la variable x dans la substitution σ, et, d'autre part, l'application d'une substitution σ à un terme t pour obtenir le terme substitué $t[\sigma]$. La substitution est rendue hygiénique par l'application d'un renommage approprié des variables libres, comme vu en cours. L'équivalence α est une relation qui formalise l'idée d'égalité de deux termes à un renommage des variables liées près. On admettra tous les résultats démontrés en cours à son sujet. En particulier, c'est bien une relation d'équivalence.

Le jugement $\Sigma \mid \Gamma \vdash t : \mathfrak{S}$ exprime que le terme du second-ordre t a la sorte $\mathfrak{S}$ sous l'hypothèse que la signature soit Σ et que l'ensemble de ses variables libres soit inclus dans l'ensemble fini Γ. On l'étend à un jugement $\Sigma \mid \Delta \vdash \sigma : \Gamma$ qui exprime que la substitution σ définit toutes les variables dans Γ par des termes dont les variables libres sont comprises dans Δ. Ce nouveau jugement permet de décrire la relation entre les variables libres de t et de σ d'une manière plus simple que celle vue en cours.

Propriété 2. *Si* $\Sigma \mid \Delta \vdash \sigma : \Gamma$ *et* $x \in \Gamma$ *alors* $\Sigma \mid \Delta \vdash \sigma(x) : \mathbb{T}$.

Démonstration. Il s'agit d'une induction de routine qu'on va détailler pour donner une idée du style de rédaction attendu.

On peut démontrer le résultat en raisonnant par induction, soit sur la substitution σ, soit directement sur la dérivation du jugement $\Sigma \mid \Delta \vdash \sigma : \Gamma$. On choisit la première option.

- Cas $\sigma = \mathtt{id}$: par inversion du jugement $\Sigma \mid \Delta \vdash \mathtt{id} : \Gamma$, on obtient que Γ est inclus dans Δ, et donc x appartient à Δ. On a $\mathtt{id}(x) = \mathtt{var}(x)$ et on conclut immédiatement $\Sigma \mid \Delta \vdash \mathtt{var}(x) : \mathbb{T}$.
- Cas $\sigma = \sigma', y \backslash t$: par inversion du jugement $\Sigma \mid \Delta \vdash \Gamma, y \backslash t : \Gamma$, on sait que $\Gamma = \Gamma', y$ avec $\Sigma \mid \Delta \vdash \sigma' : \Gamma'$ et $\Sigma \mid \Delta \vdash t : \mathbb{T}$. On raisonne par cas sur $x = y$.
  - Cas $x = y$: on a $\sigma(x) = t$ et on conclut immédiatement.

$$t, s ::= \mathtt{var}(x) \mid \langle f \rangle(t_1, \ldots, t_k) \mid x.t \qquad\qquad \sigma, \varphi ::= \mathtt{id} \mid \sigma, x \backslash t$$

(a) grammaire des termes du second ordre

$$\mathtt{id}(x) = \mathtt{var}(x) \qquad (\sigma, x \backslash t)(x) = t \qquad (\sigma, y \backslash t)(x) = \sigma(x) \quad (\text{avec } x \neq y)$$

(b) recherche du terme associé à un nom dans une substitution

$$\begin{aligned} \mathtt{var}(x)[\sigma] &= \sigma(x) \\ \langle f \rangle(t_1, \ldots, t_k)[\sigma] &= \langle f \rangle(t_1[\sigma], \ldots, t_k[\sigma]) \\ (x.t)[\sigma] &= y.t[\sigma, x \backslash y] \text{ où } y = \nu(t, \sigma) \end{aligned}$$

(c) substitution hygiénique

$$\frac{}{x \equiv_\alpha x} \qquad \frac{t_1 \equiv_\alpha s_1 \qquad \ldots \qquad t_k \equiv_\alpha s_k}{\langle f \rangle(t_1, \ldots, t_k) \equiv_\alpha \langle f \rangle(s_1, \ldots, s_k)} \qquad \frac{\exists F \subseteq_{\mathsf{cof}} \mathbb{A}, \forall z \in F, t[x \backslash z] \equiv_\alpha s[y \backslash z]}{x.t \equiv_\alpha y.s}$$

(d) équivalence α

$$\frac{\Gamma \ni x}{\Sigma \mid \Gamma \vdash \mathtt{var}(x) : \mathbb{T}} \qquad \frac{\Sigma \mid \Gamma, x \vdash t : \mathfrak{S}}{\Sigma \mid \Gamma \vdash x.t : \mathbb{V} \to \mathfrak{S}} \qquad \frac{(\Sigma \mid \Gamma \vdash t_i : \mathfrak{S}_\mathrm{i})_{1 \leq i \leq k} \quad \Sigma \ni f : \mathfrak{S}_1 \to \ldots \to \mathfrak{S}_k \to \mathbb{T}}{\Sigma \mid \Gamma \vdash \langle f \rangle(t_1, \ldots, t_k) : \mathbb{T}}$$

$$\frac{\Delta \supseteq \Gamma}{\Sigma \mid \Delta \vdash \mathtt{id} : \Gamma} \qquad \frac{\Sigma \mid \Delta \vdash \sigma : \Gamma \qquad \Sigma \mid \Delta \vdash t : \mathbb{T}}{\Sigma \mid \Delta \vdash \sigma, x \backslash t : \Gamma, x}$$

(e) jugements $\boxed{\Sigma \mid \Gamma \vdash t : \mathfrak{S}}$ et $\boxed{\Sigma \mid \Delta \vdash \sigma : \Gamma}$ de bonne formation des termes et substitutions

Figure 1 – Représentation nommée des termes du second ordre

– Cas $x \neq y$: on a $\sigma(x) = \sigma'(x)$ et on conclut par hypothèse d'induction. $\square$

Lemme 1. *Si* $\Sigma \mid \Gamma \vdash t : \mathfrak{S}$ *et* $\Sigma \mid \Delta \vdash \sigma : \Gamma$ *alors* $\Sigma \mid \Delta \vdash t[\sigma] : \mathfrak{S}$.

Démonstration. À compléter.

$\square$

4 Syntaxe anonyme avec indices de De Bruijn

4.1 Généralités

La forme la plus classique de termes anonymes utilise les *indices de De Bruijn*, du nom de leur inventeur, le mathématicien néerlandais Nicolaas Govert de Bruijn [3]. Elle est présentée à la figure 2a. Comme dans toutes les représentations anonymes, les lieurs ne spécifient plus le nom à lier, et chaque occurrence de variable liée est marquée par un entier qui indique quel est le lieur qui a introduit cette variable. Avec les indices de De Bruijn, l'entier 0 désigne la dernière variable introduite, l'entier 1 désigne l'avant-dernière variable introduite, et ainsi de suite. Donc, plus l'entier qui marque une occurrence est grand, plus le lieur est éloigné de l'occurrence de cette occurrence. On peut y penser comme à une discipline *dernier arrivé, premier servi* rappelant celle des piles. Le jugement $\Sigma \mid m \vdash t : \mathfrak{S}$ présenté à la figure 2e indique que le terme t est de sorte $\mathfrak{S}$ et comprend au plus m variables libres. On reviendra au traitement des variables libres plus loin, mais on peut noter que l'entier m joue le rôle du contexte Γ de la syntaxe avec noms.

Les équations ci-dessous présentent des exemples de λ-termes clos (sans variables libres) t en représentation nommée, et le λ-terme anonyme $\mathsf{index}(t)$ correspondant. La fonction index, qui convertit un terme nominal en terme anonyme, sera définie ultérieurement. On omet $\mathtt{var}(-)$ et $\mathtt{idx}(-)$ par souci de lisibilité.

$$\mathsf{index}(\langle\mathtt{fun}\rangle(x.x)) = \langle\mathtt{fun}\rangle(\star.0) \tag{1}$$
$$\mathsf{index}(\langle\mathtt{fun}\rangle(y.y)) = \langle\mathtt{fun}\rangle(\star.0) \tag{2}$$
$$\mathsf{index}(\langle\mathtt{fun}\rangle(x.\langle\mathtt{fun}\rangle(y.y))) = \langle\mathtt{fun}\rangle(\star.\langle\mathtt{fun}\rangle(\star.0)) \tag{3}$$
$$\mathsf{index}(\langle\mathtt{fun}\rangle(x.\langle\mathtt{fun}\rangle(y.x))) = \langle\mathtt{fun}\rangle(\star.\langle\mathtt{fun}\rangle(\star.1)) \tag{4}$$
$$\mathsf{index}(\langle\mathtt{fun}\rangle(x.\langle\mathtt{app}\rangle(x, \langle\mathtt{fun}\rangle(y.y)))) = \langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(0, \langle\mathtt{fun}\rangle(\star.0))) \tag{5}$$
$$\mathsf{index}(\langle\mathtt{fun}\rangle(x.\langle\mathtt{app}\rangle(x, \langle\mathtt{fun}\rangle(y.x)))) = \langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(0, \langle\mathtt{fun}\rangle(\star.1))) \tag{6}$$

Le dernier exemple mérite une attention particulière. Les deux occurrences de la variable nommée x dans $\langle\mathtt{fun}\rangle(x.\langle\mathtt{app}\rangle(x, \langle\mathtt{fun}\rangle(y.x)))$ se trouvent associées à deux indices différents dans sa représentation anonyme $\langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(0, \langle\mathtt{fun}\rangle(\star.1)))$. En effet, la seconde occurrence se trouve placée sous un lieur supplémentaire, et l'indice doit donc être incrementé. Il s'agit d'un phénomène typique des représentations anonymes, qui va induire une certaine subtilité dans l'implémentation de la substitution.

3. On trouve les linéaments de cette représentation dans les travaux antérieurs de Nicolas Bourbaki.

$$t, s ::= \texttt{idx}(n) \mid \langle f \rangle(t_1, \ldots, t_k) \mid \star.t \qquad\qquad \sigma, \varphi ::= \texttt{id} \mid \sigma.t \mid \uparrow\sigma$$

(a) grammaire des termes du second ordre

$$\uparrow_m \texttt{idx}(i) = \begin{cases} \texttt{idx}(i+1) & \text{si } i \geq m \\ \texttt{idx}(i) & \text{si } i < m \end{cases}$$
$$\uparrow_m 1\langle f \rangle(t_1, \ldots, t_k) = \langle f \rangle(\uparrow_m t_1, \ldots, \uparrow_m t_k)$$
$$\uparrow_m \star.t = \star.\uparrow_{m+1} t$$

(b) décalage d'un terme du second ordre

$$\texttt{id}(n) = \texttt{idx}(n) \qquad (\sigma.t)(0) = t \qquad (\sigma.t)(n+1) = \sigma(n) \qquad \uparrow\sigma(n) = \uparrow_0 \sigma(n)$$

(c) recherche du terme associé à un indice dans une substitution

$$\texttt{idx}(i)[\sigma] = \sigma(i)$$
$$\langle f \rangle(t_1, \ldots, t_k)[\sigma] = \langle f \rangle(t_1[\sigma], \ldots, t_k[\sigma])$$
$$(\star.t)[\sigma] = \star.t[(\uparrow\sigma).\texttt{idx}(0)]$$

(d) substitution

$$\frac{i < n}{\Sigma \mid n \vdash \texttt{idx}(i) : \mathbb{T}} \qquad \frac{\Sigma \mid n+1 \vdash t : \mathfrak{S}}{\Sigma \mid n \vdash \star.t : \mathbb{V} \to \mathfrak{S}} \qquad \frac{(\Sigma \mid n \vdash t_j : \mathfrak{S}_{\mathrm{i}})_{1 \leq i \leq k} \quad \Sigma \ni f : \mathfrak{S}_1 \to \ldots \to \mathfrak{S}_k \to \mathbb{T}}{\Sigma \mid n \vdash \langle f \rangle(t_1, \ldots, t_k) : \mathbb{T}}$$

$$\frac{}{\Sigma \mid m \vdash \texttt{id} : m} \qquad \frac{\Sigma \mid m \vdash \sigma : n \qquad \Sigma \mid m \vdash t : \mathbb{T}}{\Sigma \mid m \vdash \sigma.t : n+1} \qquad \frac{\Sigma \mid m \vdash \sigma : n}{\Sigma \mid m+1 \vdash \uparrow\sigma : n}$$

(e) jugements $\boxed{\Sigma \mid m \vdash t : \mathfrak{S}}$ et $\boxed{\Sigma \mid p \vdash t : m}$ de bonne formation des termes et substitutions

FIGURE 2 – Représentation anonyme des termes du second ordre, indices de Bruijn

Les occurrences d'un indice qui désignent un lieur absent du terme, comme par exemple l'indice 1 dans $\star.1$, désignent les variables libres. Celles-ci n'ont donc pas plus d'identité que les variables liées ! On peut simplement parler de la première variable libre, de la seconde, de la troisième, etc. Par conséquent, pour convertir un terme nommé t en terme utilisant les indices de De Bruijn, il faut avoir choisi pour chaque nom x apparaîssant libre dans t un indice $\theta(x)$ à la racine du terme. On appelle θ un *indiçage* pour t ; il s'agit d'une simple fonction des noms libres de t dans les entiers. Dans les équations ci-dessous, on utilise la fonction θ qui envoie x sur 0 et y sur 1.

$$\mathsf{index}_\theta(x) = 0 \tag{7}$$
$$\mathsf{index}_\theta(y) = 1 \tag{8}$$
$$\mathsf{index}_\theta(\langle\texttt{fun}\rangle(z.x)) = \langle\texttt{fun}\rangle(\star.1) \tag{9}$$
$$\mathsf{index}_\theta(\langle\texttt{fun}\rangle(z.y)) = \langle\texttt{fun}\rangle(\star.2) \tag{10}$$
$$\mathsf{index}_\theta(\langle\texttt{fun}\rangle(k.\langle\texttt{fun}\rangle(z.x))) = \langle\texttt{fun}\rangle(\star.\langle\texttt{fun}\rangle(\star.2)) \tag{11}$$
$$\mathsf{index}_\theta(\langle\texttt{app}\rangle(x, \langle\texttt{fun}\rangle(z.z))) = \langle\texttt{app}\rangle(0, \langle\texttt{fun}\rangle(\star.0)) \tag{12}$$
$$\mathsf{index}_\theta(\langle\texttt{app}\rangle(y, \langle\texttt{fun}\rangle(z.\langle\texttt{app}\rangle(z, x)))) = \langle\texttt{app}\rangle(1, \langle\texttt{fun}\rangle(z.\langle\texttt{app}\rangle(0, 1))) \tag{13}$$

4.2 Substitution

La syntaxe des substitutions utilisées pour la syntaxe anonyme est présentée à la figure 2. Celle-ci diffère de celle pour les variables nommées en deux points. La première différence est attendue : les noms ont été éliminés, c'est-à-dire qu'on a remplacé la forme $\sigma, x \backslash t$ de la syntaxe avec noms par la forme anonyme $\sigma.t$, qui indique que l'indice 0 doit être envoyé sur t et l'indice $n+1$ sur le terme t sur lequel σ envoie n. La deuxième différence est plus délicate, et mérite quelques explications.

Dans notre formulation de la substitution à la figure 1c, nous avons utilisé implicitement une propriété qu'on peut appeler *stabilité des termes nominaux par extension du contexte*. Il s'agit simplement du fait que tout terme nonimal bien formé dans un contexte Γ est également bien formé dans un contexte Δ plus grand que Γ, et ce *sans modification*. Formellement, si $\Sigma \mid \Gamma \vdash t : \mathbb{T}$ et que Γ est inclus dans Δ, alors $\Sigma \mid \Delta \vdash t : \mathbb{T}$. La situation est plus complexe pour les termes anonymes. C'est la conséquence du fait que les variables libres sont représentées par des indices, et sont donc ordonnées par nature. Dans le jugement $\Gamma \mid n \vdash \texttt{idx}(i) : \mathfrak{S}$, il faut penser à l'entier i comme désignant une position dans toute substitution définissant n termes, ou, autrement dit, comme à un indice dans un tableau de taille n. Les indices de de Bruijn indicent ce tableau en partant de la droite : l'indice 0 désigne le dernier terme (le plus à droite), l'indice 1 l'avant-dernier, etc. Mais les substitutions sont aussi étendues en ajoutant des termes vers la droite, via la construction $\sigma.t$. La position d'indice i dans σ ne correspond donc pas à la position d'indice i dans $\sigma.t$, mais à la position $i+1$. Plus généralement, le terme $\Sigma \mid m \vdash \texttt{idx}(i) : \mathbb{T}$ désigne le même indice que le terme $\Sigma \mid m+k \vdash \texttt{idx}(i+k) : \mathbb{T}$ dans un contexte étendu. On a besoin de généraliser cette opération à tous les termes. Pour cela, on introduit l'opération de *décalage*, notée $\uparrow_m t$, définie comme à la figure 2b. Cette opération incrémente incrémente de 1 toutes les variables libres présentes dans t qui sont d'indice supérieur ou égal à p. La présence du paramètre p est nécessaire pour traiter les lieurs, comme le montre l'exemple suivant.

$$\begin{aligned}
\uparrow_0\langle\mathtt{app}\rangle(0,\langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(0,1))) &= \langle\mathtt{app}\rangle(\uparrow_0 0, \uparrow_0\langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(0,1)))\\
&= \langle\mathtt{app}\rangle(\uparrow_0 0, \langle\mathtt{fun}\rangle(\star.\uparrow_1\langle\mathtt{app}\rangle(0,1)))\\
&= \langle\mathtt{app}\rangle(1, \langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(0,2)))\\
&\neq \langle\mathtt{app}\rangle(1, \langle\mathtt{fun}\rangle(\star.\langle\mathtt{app}\rangle(1,2)))\\
&= \langle\mathtt{app}\rangle(\uparrow_0 0, \langle\mathtt{fun}\rangle(\star.\uparrow_0\langle\mathtt{app}\rangle(0,1)))
\end{aligned}$$

La deuxième différence entre substitutions nominales et substitutions anonymes est donc la présence de cette opération de décalage dans les substitutions anonymes. La substitution $\uparrow\sigma$ envoie l'indice i sur le terme $\uparrow_0 \sigma(i)$. On dit qu'elle *affaiblit* σ pour la faire travailler sur un contexte étendu avec une variable libre, comme l'exprime sa règle de bonne formation donnée à la figure 2e.

Propriété 3. *Si* $\Sigma \mid n \vdash t : \mathfrak{S}$ *alors* $\Sigma \mid n+1 \vdash \uparrow_m t : \mathfrak{S}$.

Démonstration. À compléter. □

Propriété 4. *Si* $\Sigma \mid p \vdash \sigma : m$ *et* $n < m$ *alors* $\Sigma \mid p \vdash \sigma(n) : \mathbb{T}$.

Démonstration. À compléter. □

Lemme 2. *Si* $\Sigma \mid p \vdash \sigma : m$ *et* $\Sigma \mid m \vdash t : \mathfrak{S}$ *alors* $\Sigma \mid p \vdash t[\sigma] : \mathfrak{S}$.

Démonstration. À compléter. □

Pour démontrer les théorèmes qui suit, on définit la substitution $\mathrm{up}(n)$ pour tout entier n par récurrence en posant $\mathrm{up}(0) = \mathtt{id}$ et $\mathrm{up}(n+1) = \uparrow\mathrm{up}(n).0$.

Propriété 5. *Pour tous* n *et* m*, on a* $\mathrm{up}(n)(m) = \mathtt{idx}(m)$.

Démonstration. À compléter. □

Théorème 1. *Soit* t *un terme anonyme. On a* $t[\mathtt{id}] = t$.

$$\begin{aligned}
\mathsf{index}_\theta(\texttt{var}(x)) &= \texttt{idx}(\theta(x)) \\
\mathsf{index}_\theta(\langle f\rangle(t_1,\dots,t_k)) &= \langle f\rangle(\mathsf{index}_\theta(t_1),\dots,\mathsf{index}_\theta(t_n)) \\
\mathsf{index}_\theta(\star.t) &= \star.\mathsf{index}_{(\theta;(+1)),\,x\mapsto 0}(t)
\end{aligned}$$

(a) Conversion de la syntaxe nommée vers la syntaxe anonyme

$$\begin{aligned}
\mathsf{name}_\psi(\texttt{idx}(i)) &= \texttt{var}(\psi(i)) \\
\mathsf{name}_\psi(\langle f\rangle(t_1,\dots,t_k)) &= \langle f\rangle(\mathsf{name}_\psi(t_1),\dots,\mathsf{name}_\psi(t_n)) \\
\mathsf{name}_\psi(\star.t) &= \star.\mathsf{name}_{((-1);\psi),\,0\mapsto\nu(\psi)}(t)
\end{aligned}$$

(b) Conversion de la syntaxe anonyme vers la syntaxe nommée

FIGURE 3 – Conversion depuis et vers la syntaxe nominale

Démonstration. À compléter.

□

(*La question suivante est facultative. Il est très fortement conseillé de ne la traiter qu'une fois les autres démonstrations complétées.*)

Définir une fonction (totale) de composition des substitutions anonymes $(-)\circ(=)$ de sorte que cette opération vérifie le théorème attendu. Énoncer et démontrer ce théorème.

4.3 Conversion vers et depuis une syntaxe nommée

Un *indiçage* est une fonction d'image finie des noms dans les entiers naturels. Un *nommage* est une fonction d'image finie des entiers naturels dans les noms. Si ψ est un nommage, on note $\nu(\psi)$ le nom $\nu(\mathbb{N}\,;\psi)$. La figure 3a présente la fonction index, qui convertit un terme nominal t en terme anonyme $\mathsf{index}_\theta(t)$ en utilisant l'indiçage θ pour traiter les variables libres de t. La figure 3b présente la fonction name, qui convertit un terme anonyme t en terme nominal $\mathsf{name}_\psi(t)$ en utilisant le nommage ψ pour traiter les variables libres de t.

Théorème 2. *Soit t un terme anonyme. Si $\psi;\theta = id$, on a $\mathsf{index}_\theta(\mathsf{name}_\psi\, t) = t$.*

Démonstration. À compléter.

□

Théorème 3. *Soit t un terme avec noms. Si $(\theta;\psi)(x) = x$ pour tout nom x apparaissant libre dans t, alors* $\mathsf{name}_\psi(\mathsf{index}_\theta\, t) \equiv_\alpha t$.

Démonstration. À compléter.

□

Lemme 3. *Soient t un terme et θ, θ' des indiçages. Si $\theta(x) = \theta'(x)$ pour tout nom x apparaissant libre dans t, alors* $\mathsf{index}_\theta(t) = \mathsf{index}_{\theta'}(t)$.

Démonstration. À compléter.

□

Lemme 4. *On a* $\mathsf{index}_{\rho;\theta}(t) = \mathsf{index}_\theta(t[\rho])$.

Démonstration. À compléter.

□

Théorème 4. *Si $t \equiv_\alpha u$ alors* $\mathsf{index}_\theta(t) = \mathsf{index}_\theta(u)$.

Démonstration. À compléter.

□

5 Normalisation α

L'objectif de cette courte section finale est de montrer que les indices de De Bruijn peuvent être utilisés pour décider l'α-équivalence. Pour ce faire, on suppose donnée une fonction injective n des noms dans les entiers. Une telle fonction existe toujours puisque $\mathbb{A}$ est dénombrable. Si t est un terme, l'indiçage noté $\mathsf{can}(t)$ envoie un nom x soit sur $n(x)$ lorsque x apparaît libre dans t, soit sur 0 sinon.

La *forme α-normale* d'un terme nommé t, notée $\mathsf{nf}(t)$, est le terme anonyme $\mathsf{index}_{\mathsf{can}(t)}(t)$.

Théorème 5. *Deux termes nommés t et u sont α-équivalents si et seulement s'ils ont la même forme α-normale.*

Démonstration. À compléter.

□

Corollaire 1. *L'équivalence α entre termes nominaux est décidable.*

Historique des versions du document

22/04/2024 Version initiale.

22/04/2024 après-midi Bogues et typos.