Devoir de λ-calcul

Isomorphismes de types

David Baelde

À rendre le 29 avril

Informellement, deux types T et T' sont isomorphes s'il existe $f : T \Rightarrow T'$ et $g : T' \Rightarrow T$ tel que $f \circ g$ et $g \circ f$ sont respectivement l'identité sur T et T'. La notion d'isomorphisme de types est d'abord venue de considérations sémantiques — on se demande quand est-ce que les interprétations de deux types sont isomorphes au sens mathématique usuel — mais a aussi trouvé des applications très concrètes dans le contexte des langages de programmation.
Considérons l'implémentation d'une structure de données représentant des ensembles. Elle comporterait par exemple un objet `empty` : α `set`, une fonction `map` : α `set` $\Rightarrow (\alpha \Rightarrow \beta) \Rightarrow \beta$ `set`, etc. L'utilisateur à la recherche de l'opération permettant d'appliquer une fonction à tous les éléments d'un ensemble pourrait effectuer une recherche, indiquant qu'il cherche une fonction de type $(\alpha \Rightarrow \beta) \Rightarrow \alpha$ `set` $\Rightarrow \beta$ `set`. Il serait alors légitime de lui proposer en réponse la fonction `map` même si l'ordre de ses deux arguments est inversé par rapport à la requête. On pourrait même imaginer que le système génère automatiquement le code effectuant la conversion entre l'opération désirée et celle fournie par la bibliothèque. Par contre, si l'utilisateur recherche une fonction de type α `set` $\Rightarrow \alpha \Rightarrow \alpha$ `set`, i.e., le type de l'ajout d'un élément, il serait malvenu de lui proposer la fonction constante égale à `empty`, bien qu'une conversion existe. Dans le premier exemple, on a un isomorphisme de type, mais pas dans le second. L'isomorphisme de type permet ici d'obtenir des fonctions de conversion, avec l'assurance que la conversion ne perd (et n'ajoute) pas d'information au passage. Une autre utilisation similaire est l'adaptation de composants logiciels : si l'on veut connecter deux composants, au lieu de demander que leurs interfaces coincident, on pourrait se contenter de demander qu'il y ait un isomorphisme entre les deux, et utiliser encore une fois les fonctions réalisant l'isomorphisme pour connecter les deux composants.
Dans ce devoir, nous allons voir que l'isomorphisme de types est décidable et même finiment axiomatisable dans le cas du λ-calcul simplement typé enrichi avec des paires. Si les choses se passent bien dans ce "petit" calcul, elles peuvent se corser assez vite. Par exemple, il n'est plus possible d'obtenir une axiomatisation finie dès qu'on ajoute des types sommes et leur unité, correspondant à la disjonction et aux faux. Cela n'a pas empêché le développement d'applications, qui s'appuient alors sur des approximations correctes mais incomplètes de la notion d'isomorphisme.

Remarque Dans ce sujet, on ne considère que des termes bien typés, et donc fortement normalisants. On supposera qu'un terme est toujours implicitement donné avec un type et un environnement de typage. On pourrait expliciter ces informations en annotant les variables par leur type (un terme ainsi annoté admet un unique type, s'il est bien typé) mais on ne l'exige pas ici car cela alourdirait les notations.

1 Isomorphismes de types dans $\Lambda^{\Rightarrow}$

Dans cette partie on considère le λ-calcul pur, simplement typé : les termes (notés M, N, etc. ou u, v, etc.) sont construits à partir des variables par application et abstraction, et les types (notés T, T', etc.) sont construits à partir des types *atomiques* (notés A, B, etc.) en utilisant la flèche $\Rightarrow$.

Dans ce calcul, la η-expansion, notée $\to_\eta$, est restreinte aux cas bien typés : on a $M \to_\eta (\lambda x.\ M\ x)$ uniquement quand $M : T \Rightarrow T'$, avec la condition usuelle $x \notin \mathrm{FV}(M)$. On définit aussi la η_0-expansion comme la restriction de la η-expansion au cas où le terme à expanser n'est pas déja une abstraction, et qu'il n'est pas le sous-terme gauche d'une application. Ceci évite que l'expansion crée immédiatemment un β-redex. On voit aisément que (contrairement à la η-expansion) la η_0-expansion est fortement normalisante. Elle est aussi confluente.
On définit inductivement les formes *β-normales η-longues* comme s'écrivant $\lambda x_1 \dots \lambda x_p.\ y\ M_1 \dots M_n$ où y est une variable (libre ou liée), les M_i sont des formes β-normales η-longues, et $y\ M_1 \dots M_n$ est de type atomique. On admettra qu'un terme donné ne peut se réduire que vers une unique[1] forme β-normale η-longue, et que celle-ci peut toujours être atteinte en prenant la forme η_0-normale de la forme β-normale du terme. On remarquera que si un terme est en forme normale de tête, il aura la même variable de tête que sa forme β-normale η-longue. On peut alors définir la variable de tête d'un terme quelconque comme la variable de tête de sa forme β-normale η-longue. On notera enfin que le nombre d'abstractions en tête d'une forme β-normale η-longue est égal à l'*arité* de son type, définie par $\mathrm{ar}(A) = 0$ et $\mathrm{ar}(T \Rightarrow T') = 1 + \mathrm{ar}(T')$.
On note $\to_{\beta\eta}$ l'union des relations $\to_\beta$ et $\to_\eta$. La $\beta\eta$-convertibilité, notée $=_{\beta\eta}$, est la plus petite relation d'équivalence qui passe au contexte[2] et contienne $\to_{\beta\eta}$. On admettra que $M =_{\beta\eta} N$ ssi M et N ont la même forme β-normale η-longue.
On dit que les types T et T' sont *isomorphes*, noté $T \simeq T'$, s'il existe deux termes clos $M : T \Rightarrow T'$ et $N : T' \Rightarrow T$ qui soient inverses l'un de l'autre, c'est à dire qu'on a $(\lambda x'.\ M\ (N\ x')) =_{\beta\eta} (\lambda x'.\ x')$ (identité sur T') et $(\lambda x.\ N\ (M\ x)) =_{\beta\eta} (\lambda x.\ x)$ (identité sur T).

Question 1
Montrer que $(A \Rightarrow B \Rightarrow C) \Rightarrow D \Rightarrow E$ et $D \Rightarrow (B \Rightarrow A \Rightarrow C) \Rightarrow E$ sont isomorphes.

Question 2
Montrer que si un terme M est inversible, alors sa variable de tête est son premier argument et aucune de ses abstractions de tête n'est vide, c'est à dire qu'on a $M =_{\beta\eta} \lambda x \lambda y_1 \dots \lambda y_k.\ x\ M_1 \dots M_n$ avec pour tout $1 \le i \le k$, y_i apparaissant dans au moins un M_j.

Question 3
On considère deux formes β-normales η-longues inverses l'une de l'autre :

$$M = \lambda x \lambda y_1 \dots \lambda y_n.\ x\ M_1 \dots M_p \qquad N = \lambda x' \lambda y'_1 \dots \lambda y'_m.\ x'\ N_1 \dots N_q$$

On notera $\delta(M) = n - p$; de même $\delta(N) = m - q$. Si $M : T \Rightarrow T'$, exprimer $\delta(M)$ en fonction de $\mathrm{ar}(T)$ et $\mathrm{ar}(T')$. En déduire une relation entre $\delta(M)$ et $\delta(N)$.

Question 4
Soit M et N formes β-normales η-longues inverses l'une de l'autre, s'écrivant

$$M = \lambda x \lambda y_1 \dots \lambda y_n.\ x\ M_1 \dots M_p \qquad \text{et} \qquad N = \lambda x' \lambda y'_1 \dots \lambda y'_p.\ x'\ N_1 \dots N_n$$

Montrer qu'il existe une bijection σ telle que, pour tout $1 \le i \le p$ et $1 \le j \le n$, M_i a pour variable de tête $y_{\sigma(i)}$ et N_j a pour variable de tête $y'_{\sigma^{-1}(j)}$. De plus, on concluera bien sûr qu'on a $p = n$.

Question 5
Soient M, N, θ, θ' et σ comme dans la question précédente, avec $p = n$. En utilisant le même argument

1. Unique modulo α-conversion, comme toujours. On ne le répètera pas.
2. Cette condition signifie que $M =_{\beta\eta} N$ entraîne $C[M] =_{\beta\eta} C[N]$. Elle est inutile ici car $\to_{\beta\eta}$ passe déja au contexte.

qu'en question 2, on observe aisément que les sous-termes M_i et N_j ne peuvent comporter aucune abstraction vide en tête. On admettra en fait plus généralement que ces termes ne peuvent contenir d'abstraction vide, y compris en profondeur. Montrer que $(\lambda y_{\sigma(i)}.\ M_i)$ et $(\lambda y'_i.\ N_{\sigma(i)})$ sont clos, puis qu'ils sont inverses l'un de l'autre.

Question 6
Déduire des questions précédentes une caractérisation des termes inversibles en forme β-normale. Dans la litérature, on appelle ces termes *permutations héréditaires*.

Question 7
En déduire une axiomatisation finie, correcte et complète des isomorphismes de types pour $\Lambda^{\Rightarrow}$. Autrement dit, on cherche à faire coïncider $\simeq$ avec $\simeq_{\Rightarrow}$, définie comme la plus petite relation d'équivalence qui passe au contexte et satisfasse certaines équations, en nombre fini.

2 Extension à $\Lambda^{\Rightarrow\times\mathbf{1}}$

On étend le language des termes avec les constructions $\langle\rangle$, $\langle M, N\rangle$, $\pi_1(M)$ et $\pi_2(M)$. Du côté des types, on ajoute la constante $\mathbf{1}$ et le constructeur $T \times T'$. Les règles de typage associées sont :

$$\frac{}{\Gamma \vdash \langle\rangle : \mathbf{1}} \qquad \frac{\Gamma \vdash M : T \quad \Gamma \vdash N : T'}{\Gamma \vdash \langle M, N\rangle : T \times T'} \qquad \frac{\Gamma \vdash M : T \times T'}{\Gamma \vdash \pi_1(M) : T} \qquad \frac{\Gamma \vdash M : T \times T'}{\Gamma \vdash \pi_2(M) : T'}$$

Dans ce calcul étendu $\Lambda^{\Rightarrow\times\mathbf{1}}$, la relation $\to_\beta$ est enrichie des réductions $\pi_i(\langle M_1, M_2\rangle) \to_\beta M_i$ pour $i \in \{1, 2\}$, et $\to_\eta$ est enrichie de $M \to_\eta \langle \pi_1(M), \pi_2(M)\rangle$ si $M : T \times T'$, ainsi que $M \to_\eta \langle\rangle$ si $M : \mathbf{1}$. On restreint ensuite la η-expansion comme avant, en interdisant la création immédiate d'un redex : la η_0-expansion ne s'applique pas sur la constante $\langle\rangle$ et le constructeur $\langle \_, \_\rangle$ ni immédiatemment sous les destructeurs π_i. On définit alors les formes β-normales η-longues comme les formes η_0-normales des formes β-normales. On admettra que la $\beta\eta$-convertibilité coïncide, comme avant, avec l'égalité des formes β-normales η-longues. Enfin, on pourra s'appuyer sur la caractérisation des formes β-normales η-longues par les deux conditions suivantes :

- Une forme β-normale η-longue de type non atomique est construite avec le constructeur associé à ce type, et ses sous-termes sont des formes β-normales η-longues. Par exemple, $M : T_1 \times T_2$ est en forme β-normale η-longue si $M = \langle M_1, M_2\rangle$ avec $M_1 : T_1$ et $M_2 : T_2$ en formes β-normales η-longues.
- Une forme β-normale η-longue dont le type est atomique est une succession de projections et d'applications à des formes β-normales η-longues, terminée par une variable. Par exemple, si $x : A \times (B \Rightarrow (C \times D))$ et $M : B$ est une forme β-normale η-longue, alors $\pi_1(\pi_2(x)\ M)$ est une forme β-normale η-longue de type C.

On définit la relation $\rightsquigarrow$ sur les types comme la plus petite relation qui passe au contexte et satisfasse les conditions suivantes :

$$\begin{aligned}
A \times \mathbf{1} &\rightsquigarrow A && (1)\\
\mathbf{1} \times A &\rightsquigarrow A && (2)\\
A \times (B \times C) &\rightsquigarrow (A \times B) \times C && (3)\\
(A \times B) \Rightarrow C &\rightsquigarrow A \Rightarrow (B \Rightarrow C) && (4)\\
A \Rightarrow (B \times C) &\rightsquigarrow (A \Rightarrow B) \times (A \Rightarrow C) && (5)\\
A \Rightarrow \mathbf{1} &\rightsquigarrow \mathbf{1} && (6)\\
\mathbf{1} \Rightarrow A &\rightsquigarrow A && (7)
\end{aligned}$$

Question 8
Montrer que $T \rightsquigarrow T'$ implique $T \simeq T'$.

Question 9
Montrer que $\rightsquigarrow$ termine et conflue. Pour le second point, on pourra ne pas énumérer tous les cas, mais l'on veillera à bien expliquer ce qu'il faudrait énumérer.

Question 10
On considère l'axiomatisation $\simeq_\times$ obtenue à partir des équations induites par $\rightsquigarrow$, et la commutativité $A \times B \simeq_\times B \times A$. Montrer que $T \simeq_\Rightarrow T'$ implique $T \simeq_\times T'$.

On se donne quelques notations. Pour $n > 1$, on écrit $\langle M_1, M_2, \ldots, M_n\rangle$ pour $\langle\langle\langle M_1, M_2\rangle, \ldots\rangle, M_n\rangle$. De même on pose $T_1 \times T_2 \times \ldots T_n$ pour $((T_1 \times T_2) \times \ldots) \times T_n$. Enfin, on définit la notation $\pi_k^n(M)$ pour $1 \leq k \leq n$: on pose $\pi_n^n(M) := \pi_2(M)$ pour tout $n \geq 2$, $\pi_1^2(M) = \pi_1(M)$ et $\pi_k^n(M) := \pi_{k-1}^{n-1}(\pi_1(M))$ pour $n > 2$. On a ainsi $\pi_k^n(\langle M_1, \ldots, M_n\rangle) \to_\beta M_k$ pour tout $1 \leq k \leq n$.

Question 11
On suppose $p : T_1 \times \ldots \times T_n, \Gamma \vdash M : T'$ avec M en forme β-normale η-longue, $\Gamma = z_1 : A_1, \ldots, z_n : A_n$, et les A_j, T_i et T' sont des types de $\Lambda^\Rightarrow$, c'est à dire sans occurrence des constructeurs $\times$ et $\mathbf{1}$. Montrer que M peut s'écrire $M'[x_i := \pi_i^n(p)]_{1\leq i\leq n}$ où les variables x_i n'apparaissent pas dans M, et M' est un terme de $\Lambda^\Rightarrow$.

Question 12
Soit $m, n > 1$, $M = \lambda p.\ \langle M_1, \ldots, M_m\rangle\theta$ et $N = \lambda q.\ \langle N_1, \ldots, N_n\rangle\theta'$ des termes en formes β-normales η-longues, avec les M_i et N_j termes de $\Lambda^\Rightarrow$, et $\theta = [y_i := \pi_i^n(p)]_{1\leq i\leq n}$ et $\theta' = [y'_i := \pi_i^m(q)]_{1\leq i\leq m}$. Montrer que si M et N sont inverses l'un de l'autre, alors $m = n$ et il existe une bijection σ tel que $(\lambda y_{\sigma(i)}.\ M_i)$ et $(\lambda y'_i.\ N_{\sigma(i)})$ soient (clos et) inverses l'un de l'autre. On admettra, comme dans la première partie, que M et N ne comportent aucune abstraction vide.

Question 13
Montrer que $T \simeq_\times T'$ si et seulement si $T \simeq T'$ dans $\Lambda^{\Rightarrow\times\mathbf{1}}$.