Types, Logique et Coercions Implicites

Cody Roux

INRIA-Sophia Marelle

18 juin 2007

Motivation

Nous nous intéressons à la notion de *Sous Typage* dans la théorie des types : Si $A \leq B$ alors

$$\frac{\Gamma \vdash t : A \quad \Gamma \vdash A \leq B}{\Gamma \vdash t : B}$$

La motivation est double :

1. Mathématique : Formalisation plus naturelle de concepts mathématiques : 0 peut etre vu comme un élément de $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ etc.
2. Informatique : Possibilité de typer plus de termes donc de garantir la fiabilité de plus de programmes

coercions implicites

Une approche possible : étant donné $A \leq B$, donner une *coercion* $c\colon A \to B$ transparente à l'utilisateur. C'est le concept de *coercion implicite*.

Exemple : Si A est un anneau, $\mathtt{Mat} = [1..n] \to [1..m] \to A$. Nous voulons noter M à la place de $\phi(M)$ avec $\phi\colon \mathtt{Mat} \to (A^n \to A^m)$ où $A^k = [1..k] \to A$.

Problème : dans le système de types avec sous-typage $F_{\leq}$, le typage est indécidable !

Idées :

1. Se restreindre aux types simples
2. Faire marcher *l'isomorphisme de Curry-Howard* : trouver un terme $c\colon A \to B \iff$ prouver $A \vdash B$

La Logique des Coercions :

On se donne un ensemble $\mathcal{C}$ de types de la forme $U \to V$

$$\frac{}{\mathcal{C}; A \vdash A}Ax$$

$$\frac{\mathcal{C}, B \to C; A \vdash B}{\mathcal{C}, B \to C; A \vdash C}App$$

$$\frac{\mathcal{C}; C \vdash A \quad \mathcal{C}; B \vdash D}{\mathcal{C}; A \to B \vdash C \to D}Contr$$

$$\frac{\mathcal{C}; A \vdash B \quad \mathcal{C}; B \vdash C}{\mathcal{C}; A \vdash C}Cut$$

Décidabilité

Deux propriétés fondamentales :

1. La logique est décidable

- **idée :**normalisation forte + elimination des coupures
- Une preuve dans la logique des coercions $\longleftrightarrow$ un lambda terme sous forme *coercive*
- On rajoute *Appin* et *Contrin* pour “symétriser” la logique
- On prouve que tout terme est normalisable
- On montre que tout terme sous forme coercive normale est constructible sans *Cut*

2. Etant donné un terme du λ-calcul, il est décidable de savoir si il est typable avec les coercions.

- **idée** On se ramène à traiter un nombre fini de cas + logique décidable
- On sait typer les variables et les abstractions
- Pour typer une application : si $f : F$ et $a : A$, $(f\ a)$ est typable si et seulement si il existe X tel que $F \vdash A \rightarrow X$.
- Probleme : trouver tous les X possibles !
- On montre qu'il suffit de considérer un nombre fini de types.

Cohérence

Si c, c' sont des coercions de A vers B nous voulons

$$c \ =_{\beta\eta} \ c'$$

Un premier résultat : Si il n'y a qu'une unique coercion $c_0 : A \rightarrow B$ dans $\mathcal{C}$ et que A n'est pas un sous-terme de B qui n'est pas un sous-terme de A alors nous avons la cohérence.

Objectifs :

1. Généraliser ce résultat !
2. intégrer des équations entre coercions ($f \circ g = id \dots$)

Conclusions

- Coercions puissantes, mais...
- Types (trop) simples

Il reste à faire :

- Intégrer les types polymorphes, dépendants (sous une certaine forme)
- Intégrer les types inductifs
- Comprendre la situation des types implicites.