Computing Science
Thierry Coquand
Professor

Rapport de la thèse de Rodolphe Lepigre

Le but de cette thèse est la mise au point un système de preuves pour la correction de programmes fonctionnels. Si l'approche se place dans le cadre traditionnel de la réalisabilité (qui remonte sous cette forme aux travaux de Jean-Louis Krivine) où l'on manipule des assertions sur un langage de programmation non typé, une originalité de ce travail est d'obtenir un système qui combine la logique classique et les types dépendants.

La thèse commence par la présentation d'une machine abstraite pour un calcul non typé. C'est une version en appel par valeur de la machine de Krivine, étendue avec constructeurs et "record", et une distinction entre "programme" et "pile" (ou contexte d'évaluation). On peut alors définir (chapitre 3) une première notion d'équivalence observationnelle des programmes, qui a la propriété fondamentale que si l'on substitue dans un programme un sous-programme par un sous-programme équivalent, on obtient un programme équivalent (Théorèmes 3.1.5 et 3.1.7). Cette équivalence observationnelle constitue un exemple d'une "relation d'équivalence compatible" (Définition 3.2.10 et Théorème 3.2.11). En partant d'une relation d'équivalence compatible abstraite, le chapitre 4 explique alors comment interpréter une notion assez riche et naturelle de types (Définition 4.5.43), et le théorème 4.5.45 montre la correction du système de types pour cette interprétation.

Le chapitre 5 est pour moi le chapitre central de ce travail. À ce point, on peut déjà définir une notion de types dépendants, mais son utilisation est limitée par le fait que la loi $t\ u : B(u)$ si $t : \Pi(x : A)B$ ne peut être utilisée que si u est une *valeur*. C'est le problème fondamental de la combinaison des types dépendants et des effets de bord. Ce chapitre contient alors des résultats mathématiques non triviaux motivés par cette question. L'idée est d'avoir une notion d'une relation d'équivalence compatible telle que l'on pourra conclure $t\ u : B(u)$ dès que u est *équivalent* à une valeur. Sémantiquement, cela correspond au fait que toute valeur dans $[\![A]\!]^{\perp\perp}$ doit aussi être dans $[\![A]\!]$. Un contre-exemple (Théorème 5.4.9) montre que ce n'est pas vérifié pour la relation d'équivalence observationnelle. L'auteur montre alors qu'il est possible de définir une autre relation d'équivalence qui elle vérifie cette propriété. Ceci est obtenu par l'introduction d'un opérateur qui internalise (de manière non effective) la relation d'équivalence elle-même et un procédé de "feed-back" entre cette relation d'équivalence et cet opérateur. Ceci donne une justification sémantique pour obtenir un système satisfaisant qui combine types dépendants et effet de bord. Le chapitre suivant est une extension avec la notion de sous-types, qui contient une utilisation astucieuse de l'opérateur ϵ de Hilbert. Le dernier chapitre montre que toutes ces idées marchent en pratique en donnant des exemples de preuves de programme qui ont été effectivement vérifiées dans un prototype.

Le tout constitue une excellente thèse d'informatique théorique. Les questions de sémantique sont directement motivées par des exemples concrets suggérés par le prototype et suggèrent des développements mathématiquement non triviaux et intéressants. Comme le signale l'auteur, il y a des points communs avec les travaux de Doug Howe et la relation $\sim$ du système NuPrl (mais ces travaux se situaient dans un cadre intuitioniste). Le fait que cette approche suscite des résultats sémantiques pertinent la justifie a posteriori. a présentation est claire (et j'ai été aidé par l'article correspondant au chapitre 5, qui résume aussi très

CHALMERS UNIVERSITY OF TECHNOLOGY
GÖTEBORG UNIVERSITY
Computing Science
SE-412 96 Göteborg, SWEDEN
Visiting address: Eklandagatan 86
Telephone: +46 - (0)31 772 1035 Fax: +46 - (0)31 16 56 55
Chalmers tekniska högskola AB, Reg. No: 556479-5598, VAT No. SE556479559801
www-address: http://www.md.chalmers.se/~coquand E-mail: coquand@cs.chalmers.se

bien ce qui se passe). En conclusion, ce travail mérite amplement l'obtention d'une thèse.

Göteborg,

Thierry Coquand
Professor of Computing Science