

# Étude des automorphismes de types et méthode de Monte-Carlo

Sergei SOLOVIEV  
ACADIE, IRIT, UPS  
[sergei.soloviev@irit.fr](mailto:sergei.soloviev@irit.fr)

Érik MARTIN-DOREL  
ACADIE, IRIT, UPS  
[erik.martin-dorel@irit.fr](mailto:erik.martin-dorel@irit.fr)

22 novembre 2016

## 1 Informations pratiques

Le stage se déroulera au sein de l'Institut de Recherche en Informatique de Toulouse ([IRIT](#)) sur le campus de l'Université Paul Sabatier ([UPS](#)), sous la direction de Sergei SOLOVIEV et Erik MARTIN-DOREL.

Le stage s'effectuera dans le cadre du projet *FAGames (Formal Analysis of Games using Interactive Theorem Proving)* financé par le LabEx [CIMI](#).

## 2 Contexte

Ce stage s'articule autour de la notion d'isomorphisme de types [1]. En théorie des types, la notion de type est très proche de la notion de formule logique. La correspondance entre les deux est appelée *correspondance de Curry-Howard* et peut être résumée de la façon suivante :

- un habitant  $p$  d'un type  $P$  peut être vu comme une preuve  $p$  d'une formule  $P$
- une fonction du type  $A \rightarrow B$  peut être vue comme une preuve de l'implication  $A \Rightarrow B$
- ...

Ensuite, vient la notion d'isomorphisme entre deux types donnés. Par exemple, les deux types suivants (qui sont des types de fonctions à deux arguments) sont isomorphes :  $A \rightarrow (B \rightarrow C) \sim B \rightarrow (A \rightarrow C)$ . En effet, ils représentent des formules équivalentes et de plus, les preuves des deux formules peuvent être mises en bijection.

L'automorphisme est le cas particulier d'isomorphisme, quand une transformation isomorphe ne change pas le type. Par exemple, on obtient un automorphisme  $A \rightarrow (A \rightarrow C) \sim A \rightarrow (A \rightarrow C)$  en permutant  $A$  et  $A$ , qui n'est cependant pas un automorphisme trivialement égal à l'identité : on

remarque que  $f(x, y)$  est différent de  $f(y, x)$ , même si  $x$  et  $y$  ont le même type  $A$ .

Pour les types complexes le nombre total des isomorphismes peut être très grand (comme pour l'isomorphisme des graphes) et l'estimation du nombre des automorphismes parmi les isomorphismes peut être également très complexe. Cependant, on peut engendrer les isomorphismes de manière aléatoire et vérifier assez facilement si l'isomorphisme est un automorphisme, ce qui permet d'estimer expérimentalement la proportion d'automorphismes.

### 3 Objectifs du stage

Ce stage portera sur l'étude quantitative des automorphismes de types, en utilisant notamment la méthode de Monte-Carlo. L'étude s'appuiera donc sur une implémentation et le choix du langage de programmation reste à discuter avec le candidat (par exemple : OCaml, Java ou Python).

### 4 Prérequis

Le candidat devra avoir des connaissances en logique des prédicats, en théorie des graphes et des probabilités, ainsi que des compétences en programmation et génie logiciel. Des connaissances de base en théorie des types sont les bienvenues.

### Références

- [1] Roberto Di Cosmo. *Isomorphisms of Types : from  $\lambda$ -calculus to information retrieval and language design*. Progress in Theoretical Computer Science. Birkhäuser, 1995.