Strong normalisation for simply typed λ-calculus: a quest for new combinatorial proofs

Internship proposal

December 2015

Keywords
Lambda-calculus, Curry-Howard correspondence, linear logic, proof-nets, denotational semantics.

Thematics
Analysis of strong normalisation proof techniques based on denotational semantics and game theory.

Audience
M2/M1 students in logic or computer science.

Hosting institution
LIPN, Laboratoire d'Informatique de Paris Nord (UMR 7030 CNRS). Institut Galilée, Université Paris 13.
99, avenue Jean-Baptiste Clément. 93430 Villetaneuse.
`http://www.lipn.univ-paris13.fr`

Supervisors
Stefano Guerrini (email: `stefano.guerrini@univ-paris13.fr`)
Giulio Manzonetto (email: `giulio.manzonetto@univ-paris13.fr`)

PhD Thesis
Possible.

Duration
From 3 to 6 months.

Period
From February to July.

Required competencies
The candidate is supposed to have some basic knowledge of λ-calculus and first order logic.

Context

The simply typed λ-calculus $\Lambda^{\rightarrow}$ is a prototypical programming language endowed with a type system having $\rightarrow$ as the sole type constructor. It is the canonical and simplest example of a typed programming language, and it enjoys many desirable and interesting properties like *normal form uniqueness* and *strong normalisation*. This means that all simply typed λ-terms have a unique normal form, and that one can get it independently from the reduction strategy chosen to reduce the term.

The simply typed λ-calculus was defined by Alonso Church [Chu40] as a way to to avoid the fact that in the logical theory based on untyped λ-calculus [Chu32] one could reproduce Russell's paradox. Although the fact that it has introduced such a longtime ago, the research on simply typed λ-calculus is still ongoing [BS11, SMGB12, Sal09]. First, because the recently published book "Lambda Calculus with Types" by Henk Barendregt [BDS13] is giving a second youth to this subject. Second, because there are a number of fundamental problems — rather central in theoretical computer science also — that can be easily formulated in the simply typed λ-calculus, which are still open.

During this *stage*, we will mainly focus our attention on the following problem [Bar14]:

Problem. Construct an 'easy' assignment of (possibly transfinite) ordinals to terms of the simply typed λ-calculus, i.e. a map

$$\# : \Lambda \to \{\alpha \mid \alpha \text{ is an ordinal }\},$$

that strictly decreases along β-reduction. This means that, whenever $M \to_\beta N$ we also have $\#N < \#M$. Since ordinals are well-ordered, this immediately shows that β-reduction is strongly normalising on $\Lambda^{\to}$.

Actually, this is not a formal problem but rather a (mathematical) 'koan'. A problem, according to [Pol04], should have the property that a candidate solution should be clearly recognisable as such. A koan is instead a non-precise question for which the space of solutions is *a priori* not clear, but with the property that once a right answer would be proposed, it should be immediately recognised as "the solution". The keyword to understand the above problem is the adjective *easy* — what does it mean for an assignment to be easy? From our point of view, un assignment is easy if it shows what is actually decreasing during the reduction of a term (obviously not the size of the term, since in the simply typed λ-calculus we know that it may increase). Several existing proofs of strong normalisation are based on assignments of this kind [Loa98, dV87, How70, WW12], but none can be qualified as easy in this sense.

Objectives

During the stage, the student is supposed to write a survey on the state of the art of the research on this subject, developing one (or more) of the aspects stated below according to his/her main interests. In general, we expect the outcome of his/her work to be a clearer road map of the existing proofs of strong normalisation and a study of applicability of some new methods that we have in mind. The survey should be a good starting point for a possible PhD thesis.

Logical Approach. Linear logic [Gir87] plays a key role in the fine analysis of computational properties of λ-calculus. Its decomposition of logical connectives into linear connectives and exponential modalities allows to have a fine use of resources, and in particular on how and when to use values or to evaluate functions. Moreover, the graph representation of proofs introduced by Girard, the so-called proof nets, give a particularly useful framework for studying operational properties of λ-terms (for a simple introduction on the correspondence between proof nets and λ-calculus, see [Gue04]).

Game theoretic approach. Game theory has come to play an increasingly important role in logic and in computer science. There are proofs of strong normalisation [DM07] in term rewriting systems which are obtained by transforming terms into trees (the arenas of the game) and rewriting rules into tree transformations (the plays). The reduction of a term becomes —under this analogy — an interactive play between the player and the opponent. Once the game is well-defined it is enough to prove that the player has always a winning strategy to get as a conclusion that all terms are normalisable. We claim that a game theoretic proof of strong normalisation for the simply typed λ-calculus based on trees could be easily transformed in a solution of our problem.

Semantic Approach. Denotational semantics aims at formalising the 'meanings' of programming languages by constructing suitable objects called *models*, which describe the meanings of expressions from the languages. Using this approach it is possible to capture operational properties of programs by means of more abstract models, on which a broader range of tools and proof techniques are available. In particular, in the relational models of the so called *differential λ-calculus* [ER03], an element living in the interpretation of a term gives enough information to compute the number of linear head reduction steps needed from such a term to its normal form. These kind of models have already been used to build proofs of strong normalisation [BL13, dCdF13] and we believe that they can give fruitful insights for the definition of an easy norm for the simply typed λ-calculus.

Practical aspects and scientific framework

The Logic, Computation and Reasoning (LCR) team of LIPN has a well-established expertise in Linear Logic, and on its applications to several domains of foundations of logic and computation: proof theory, λ-calculus and functional programming, denotational semantics, computational complexity. At the moment (December 2015), 3 full professors, 5 assistant professors, 2 CNRS assistant researchers, and 1 postdoctoral researchers actively work on these thematics; moreover, 6 PhD thesis are currently under development on logic and computation thematics (among which 1 in co-tutorship with the University of Bologna, Italy). The researchers of the LCR team lead or contribute to national and international research projects, and have research collaborations with Italy, United Kingdom, The Netherlands, Japan, and United States. Each year, the LCR team hosts leading international researchers for short and long periods.

References

[Bar14] Henk Barendregt. Problem # 26 of the TLCA list of open problems, 2014.

[BDS13] Hendrik Pieter Barendregt, Wil Dekkers, and Richard Statman. *Lambda Calculus with Types*. Perspectives in logic. Cambridge University Press, 2013.

[BL13] Alexis Bernadet and Stéphane Lengrand. Non-idempotent intersection types and strong normalisation. *Logical Methods in Computer Science*, 9(4), 2013.

[BS11] Pierre Bourreau and Sylvain Salvati. Game semantics and uniqueness of type inhabitance in the simply-typed λ-calculus. In C.-H. Luke Ong, editor, *Typed Lambda Calculi and Applications - 10th International Conference, TLCA 2011, Novi Sad, Serbia, June 1-3, 2011. Proceedings*, volume 6690 of *Lecture Notes in Computer Science*, pages 61–75. Springer, 2011.

[Chu32] Alonzo Church. A set of postulates for the foundation of logic. *The Annals of Mathematics*, 33(2):346–366, April 1932.

[Chu40] Alonzo Church. A formulation of the simple theory of types. *The Journal of Symbolic Logic*, 5(2):56–68, June 1940.

[dCdF13] Daniel de Carvalho and Lorenzo Tortora de Falco. A semantic account of strong normalization in linear logic. *CoRR*, abs/1304.6762, 2013.

[DM07] Nachum Dershowitz and Georg Moser. Rewriting computation and proof. chapter The Hydra Battle Revisited, pages 1–27. Springer-Verlag, Berlin, Heidelberg, 2007.

[dV87] R de Vrijer. Exactly estimating functionals and strong normalization. *Indagationes Mathematicae*, 49:479–493, 1987.

[ER03] Thomas Ehrhard and Laurent Regnier. The differential lambda-calculus. *Theor. Comput. Sci.*, 309(1-3):1–41, 2003.

[Gir87] Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50:1–102, 1987.

[Gue04] Stefano Guerrini. Proof nets and the lambda-calculus. In Thomas Ehrhard, Paul Ruet, Jean-Yves Girard, and Phil Scott, editors, *Linear Logic in Computer Science*, volume 316 of *London Mathematical Society Lecture Notes Series*, pages 65–118. Cambridge University Press, 2004.

[How70] W Howard. Assignment of ordinals to terms for primitive recursive functionals of finite type. In J. M. A. Kino and R. Vesley, editors, *Intuitionism and Proof Theory: Proceedings of the Summer Conference at Buffalo N.Y. 1968, volume 60 of Studies in Logic and the Foundations of Mathematics*, pages 443 – 458, 1970.

[Loa98] Ralph Loader. Notes on simply typed lambda calculus. Technical report, 1998.

[Pol04] George Polya. *How to solve it : a new aspect of mathematical method*. Princeton University Press, Princeton, Oxford, 2004.

[Sal09] Sylvain Salvati. Recognizability in the simply typed lambda-calculus. In Hiroakira Ono, Makoto Kanazawa, and Ruy J. G. B. de Queiroz, editors, *Logic, Language, Information and Computation, 16th International Workshop, WoLLIC 2009, Tokyo, Japan, June 21-24, 2009. Proceedings*, volume 5514 of *Lecture Notes in Computer Science*, pages 48–60. Springer, 2009.

[SMGB12] Sylvain Salvati, Giulio Manzonetto, Mai Gehrke, and Henk Barendregt. Loader and urzyczyn are logically related. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part II*, volume 7392 of *Lecture Notes in Computer Science*, pages 364–376. Springer, 2012.

[WW12] G. Wilken and A Weiermann. Derivation lengths classification of goedel's t extending howard?s assignment. *Logical Methods in Computer Science*, 8(1):44, 2012.