Principal Typing for the $\lambda\sigma_{dB}$-calculus

Kaliana dos Santos Dias de Freitas (kalianadias@gmail.com)
CIC - UnB

Abstract. The $\lambda\sigma_{dB}$-calculus is inspired on $\lambda\sigma$-calculus proposed in [1]. It is a calculus of explicit substitutions (CES) containing all De Bruijn indexes in its syntax. The CES's are extensions of the λ-calculus that implement concretely its main operation, which is the β-reduction since they contain the operation of substitutions as part of their language. The $\lambda\sigma_{dB}$-calculus is confluent (ground) and simulates the classical β-reduction. A version of $\lambda\sigma_{dB}$ with simple types was considered, which we called $B1$-system. The $B1$-system enjoys properties as subject reduction and soundness, and, in Church-style, $B1$ satisfies the type-uniqueness property. Terms may have many types depending on the context in Curry-style. Therefore, another important property to study is the *principal typing* (PT for short), which has also been called *principal pair* in [2]. The PT property in a type system responds to question whether for a given term its most general typing can be found [3]. A system-independent definition of PT due to Wells was proposed in [4]. He has proved that it generalizes previous system-specific definition. In this work, a new definition of PT concerning B1 is proposed and it is proven to be equivalent to that Wells' definition. Finally, we prove PT property for $B1$ according to the proposed definition from a type inference algorithm.

References

[1] M. Abadi, L. Cardelli, P.-L. Curien, and J. Lévy, *"Explicit substitutions,"* Journal of functional programming, vol. 1, no. 4, pp. 375–416, 1991.

[2] J. R. Hindley, *Basic simple type theory*, ser. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1997, vol. 42.

[3] D. L. Ventura, M. Ayala-Rincón, and F. Kamareddine, *"Principal typings for explicit substitutions calculi,"* in Conference on Computability in Europe. Springer, 2008, pp. 567–578.

[4] J. B. Wells, *"The essence of principal typings,"* in International Colloquium on Automata, Languages, and Programming. Springer-Velag, 2002, pp. 913–925.