A Standard Theory for the Pure Lambda-Value Calculus

Álvaro García-Pérez and Pablo Nogueira

1 IMDEA Software Institute, Madrid, Spain
2 Universidad Politécnica de Madrid, Spain

agarcia@babel.ls.fi.upm.es, pablo.nogueira@upm.es

1 Abstract

In the classical lambda calculus (i.e., pure untyped $\lambda K$, see [3]) a term $M$ is solvable iff there exists a head context $C[ ] \equiv (\lambda x_1 \ldots x_m.[] ) N_1 \ldots N_n$ such that $C[M] \rightarrow^*_{\beta} I$ [3]. Solvability characterises those terms that have a definite operational meaning, i.e., the terms that are operationally relevant.

In the definition of solvability, the role of the head context $C[ ]$ is to close an open $M$ if necessary (i.e., $\lambda x_1 \ldots x_m.M$) and to reveal the operational content of $\lambda x_1 \ldots x_m.M$ by providing arguments $N_1 \ldots N_n$ to it. The $I$ in the right-hand-side of the reduction judgement is just a simplification to keep things succinct. The definition above can be proven equivalent to the earliest—and original—definition in [4], which reads ‘$M$ is solvable iff there exists a head context $C[ ]$ such that $C[M] \rightarrow^*_{\beta} NF$’ [4, 5, 11].

In the light of the original definition, the meaning of ‘operational relevance’ can be better grasped. A program $P$ uses a sub-program $S$ effectively whenever $P$ has a definite result $R$, and a different result $R'$ is obtained when replacing $S$ by a non-equivalent sub-program $S'$ in $P$ [8]. In the classical lambda calculus, the head contexts belong to the programs that use the term in their hole effectively, $M$ is the sub-program, and the definite results are the normal forms. An operationally relevant term is a term that can be used effectively, i.e., a solvable term. Solvability gives rise to a very successful ‘standard theory’ [1] according to which all the unsolvables can be consistently equated (i.e., the lambda-theory $H$ in [3]) and where the concomitant notions of sensibility (i.e., satisfying $H$) and Böhm trees [3] connect theories and models with good operational behaviour [10, 11, 8].

The simplified definition of solvability above has been duly adapted in [8] to the pure version of the lambda-value calculus (i.e., $\lambda V$ in [9] without the primitive constants and $\delta$-rules). A term $M$ is $v$-solvable iff there exists a head context $C[ ] \equiv (\lambda x_1 \ldots x_m.[] ) N_1 \ldots N_n$ where the $N_i$ are closed values such that $C[M] \rightarrow^*_{\beta_v} I$. The notion of $v$-solvability captures operational relevance of terms, but only in the ‘weak’ version of the calculus in which the redices in abstractions are never contracted (i.e., what is known as the ‘lazy lambda-value calculus’ [6, 7]). This lazy lambda-value calculus generalises some of the results in the ‘lazy lambda calculus’ of [1] to a framework with a by-value calling convention. Both [1] and [6, 7] exhibit good model-theoretical properties, in particular the existence of initial models in any of the conventional categories for domains. However, the lazy lambda-value calculus does not answer in full whether a ‘standard theory’ exists for the unrestricted $\lambda V$. Unsurprisingly, $v$-solvability does not capture operational

---

*Research partially funded by the Spanish ‘Ministerio de Economía y Competitividad’ through projects DE- SAFIOS10 TIN2009-14599-C03-00 and STRONGSOFTWARE TIN2012-39391-C04-02, and by ‘Comunidad de Madrid’ through programme PROMETIDOS P2009/TIC-1405. The first author is supported by Comunidad de Madrid grant CPI/0622/2008 and by IMDEA Software Institute.
relevance in $\lambda_V$, nor shows how to equate the meaningless terms in that calculus in a consistent way. In fact, the syntactic characterisation of $v$-solvability is given in terms of $\lambda K$’s $\beta$-reduction, not in terms of $\beta_V$-reduction [8].

Inspired by the original definition in [4], we introduce the novel $\lambda_V$-solvability which characterises operational relevance in the unrestricted $\lambda_V$. A term is $\lambda_V$-solvable if there exists a head context $C[\ ]$ such that $C[M] \rightarrow^*_{\beta_v} \text{NF}_V$, where the $\text{NF}_V$’s are the full normal forms of $\lambda_V$. We generalise needed reduction [2] to $\lambda_V$, and introduce ribcage reduction, a sub-relation of $\rightarrow^*_{\beta_v}$ which is somewhat the $\lambda_V$ analogous of $\lambda K$’s head reduction. The terms with ribcage normal form are operationally relevant in $\lambda_V$, and hence having a ribcage normal form constitutes the syntactic definition of $\lambda_V$-solvability. Differently from [8], this syntactic definition can be given in terms of $\beta_V$-reduction. Our approach is enough to prove the Genericity Lemma and the ‘preservation of unsolvables’ [11, 3] in $\lambda_V$. We also provide a novel characterisation of all the complete strategies in $\lambda_V$.

A $\lambda_V$-unsolvable has an order $n$, in the spirit of [1], which informs about the maximum number of trailing lambdas in a term which is $\beta_V$-equivalent to the $\lambda_V$-unsolvable. The $\lambda_V$-solvability gives rise to the $\lambda_V$-theory $H_V$, which equates all the $\lambda_V$-unsolvables of the same order. We prove consistency of $H_V$, thus opening up for the definition of $\lambda_V$-Böhm trees. The concomitant notion of $\omega$-sensibility (i.e., satisfying $H_V$) characterises $\lambda_V$-theories and $\lambda_V$-models with good operational behaviour. Our contributions help to reinstate the validity of full normal forms in $\lambda_V$ and to lay the foundations of a ‘standard theory’ for the pure $\lambda_V$-calculus.

References


