A Machine-Checked Correctness Proof of Normalization by Evaluation for Simply Typed Lambda Calculus

Author: András Kovács Advisor: Ambrus Kaposi

Budapest, 2016

Abstract

We present an executable implementation and correctness proof of normalization by evaluation for the simply typed lambda calculus, using the proof assistant Agda. First, we use a presheaf model over the category of order-preserving context embeddings to define the normalization function. Then we prove completeness by a Kripke logical relation between the term and presheaf models, and soundess by another Kripke logical relation on the presheaf model. Stability is proven by induction on normal forms. We follow Altenkirch, Hoffman and Streicher [1] in the usage of presheaf models, but unlike loc. cit. we keep the definition of normalization and its correctness proofs separate, and rely on direct type-theoretic rather than categorical constructions. Our formalization is lightweight in comparison to previous correctness proofs of normalization based on big-step evaluation [TODO] and hereditary substitution [TODO].

Contents

1 Introduction **4**
1.1 Normalization by evaluation 4
1.2 Related work 4

2 Metatheory **4**

3 Syntax **4**
3.1 Terms and contexts 4
3.2 Categories-with-families 4
3.3 Order-preserving embeddings 4
3.4 Substitution 4
3.5 Conversion 4

4 Preliminaries: normalization with a complete Kripke model **4**

5 Presheaf model **4**
5.1 Presheaves 4
5.2 Terms and contexts 4
5.3 Embeddings and substitutions 4
5.4 Normalization 4

6 Correctness **4**
6.1 Completeness 4
6.2 Soundness 4
6.3 Stability 4

7 Discussion **4**
7.1 Formal development 4
7.2 Efficiency 4
7.3 Comparison 4
7.3.1 Big-step normalization 4
7.3.2 Hereditary substitution 4
7.4 Future work 4

References **4**

1 Introduction

1.1 Normalization by evaluation

1.2 Related work

2 Metatheory

3 Syntax

3.1 Terms and contexts

3.2 Categories-with-families

3.3 Order-preserving embeddings

3.4 Substitution

3.5 Conversion

4 Preliminaries: normalization with a complete Kripke model

5 Presheaf model

5.1 Presheaves

5.2 Terms and contexts

5.3 Embeddings and substitutions

5.4 Normalization

6 Correctness

6.1 Completeness

6.2 Soundness

6.3 Stability

7 Discussion

7.1 Formal development

7.2 Efficiency

P (eds) Category theory and computer science. pp 182–199