Reasoning about Programs in Continuation-Passing Style*

AMR SABRY†
MATTHIAS FELLEISEN†

Department of Computer Science, Rice University, Houston, TX 77251-1892

Keywords: \(\lambda\)-calculus, \(\lambda_v\)-calculus, continuation-passing style, CPS transformations, inverse CPS transformations, \(\lambda_v\)-C-calculus, IOCC

Abstract. Plotkin's \(\lambda_v\)-calculus for call-by-value programs is weaker than the \(\lambda\beta\eta\)-calculus for the same programs in continuation-passing style (CPS). To identify the call-by-value axioms that correspond to \(\beta\eta\) on CPS terms, we define a new CPS transformation and an inverse mapping, both of which are interesting in their own right. Using the new CPS transformation, we determine the precise language of CPS terms closed under \(\beta\eta\)-transformations, as well as the call-by-value axioms that correspond to the so-called administrative \(\beta\eta\)-reductions on CPS terms. Using the inverse mapping, we map the remaining \(\beta\) and \(\eta\) equalities on CPS terms to axioms on call-by-value terms. On the pure (constant free) set of \(\Lambda\)-terms, the resulting set of axioms is equivalent to Moggi's computational \(\lambda\)-calculus. If the call-by-value language includes the control operators \texttt{abort} and \texttt{call-with-current-continuation}, the axioms are equivalent to an extension of Felleisen et al.'s \(\lambda_v\)-C-calculus and to the equational subtheory of Talcott's logic IOCC.

Contents

1 Compiling with and without Continuations 292

2 \(\Lambda\): Calculi and Semantics 295

3 The Origins and Practice of CPS 298
   3.1 The Original Encoding 298
   3.2 The Universe of CPS Terms 299

4 A Compacting CPS Transformation 301

---

*This article is a revised and extended version of the conference paper with the same title [42]. The technical report of the same title contains additional material.
†The authors were supported in part by NSF grant CCR 89-17022 and by Texas ATP grant 91-003604014.