Reduction Semantics

Matthias Felleisen

Reduction Semantics for All Languages

@inproceedings{ ffkd:reasoning1,
    author = {Matthias Felleisen and Daniel P. Friedman
    and Eugene E. Kohlbecker and Bruce F. Duba},
    title = {Reasoning with continuations},
    booktitle = {Proc. First Symposium on Logic
    in Computer Science},
    year = {1986},
    month = {June},
    pages = {131--141},
}

Content  The article adapts Plotkin’s “call-by-value vs. call-by-name”
program to a programming language with imperative control. It demonstr-
ates that the Church-Rosser theorem and the Curry-Feys Standard Red-
duction Theorem aren’t reserved for “purely” functional programming lan-
guages. As such it opens the door for all kinds of imperative extensions.

@article{fh:syntactic-control-state,
    author = "Matthias Felleisen and Robert Hieb",
    title = "The revised report on the syntactic theories of
    sequential control and state",
    booktitle= "Theoretical Computer Science",
    volume = {103},
    number = {2},
    year = "1992",
    pages="235--271"
}

Content  This article introduces the final form of reduction semantics for
imperative languages. It removes the wart of two-level calculi from the previ-
ous attempts (see above), and it demonstrates that reasoning for evaluation
can be completely local for imperative higher-order languages.

@book{ fff:redex,
  title = {Semantics Engineering with PLT Redex},
  author = "Matthias Felleisen and Robert Bruce Findler
           and Matthew Flatt",
  publisher = "MIT Press",
  year = 2009
}

Content  The book collects the major results of reduction semantics (the
above plus applications) and introduces a rapid-prototyping tool for working
with reduction semantics.