A Combinatory Account of Internal Structure

Barry Jay and Thomas Given-Wilson
University of Technology, Sydney
{cbj,tgwilson}@it.uts.edu.au

September 4, 2009

Traditional combinatory logic is computationally equivalent to pure λ-calculus and able to represent all of the Turing computable functions on natural numbers, but there are effectively calculable functions on the combinators themselves that cannot be so represented, as they examine the internal structure of their arguments. This is consistent with the traditional theorems about computable functions of numbers, but imposes severe constraints upon the interpretation of Church's thesis, that all effectively calculable functions are general recursive. The practical consequence is that the expressive power of traditional combinatory logic can be increased by adding new combinators.

Consider the combinators built from two *atoms* (meta-variable A), namely the traditional S and a new, factorisation operator F. The *SF-matchable forms* are the combinators of the form S, SM, SMN, F, FM and FMN. Then the defining equations for S and F are

$$\begin{array}{rcll} SMNX & = & MX(NX) & \\ FAMN & = & M & \\ F(PQ)MN & = & NPQ & \text{if } PQ \text{ is } SF\text{-matchable.} \end{array}$$

Just as the combinators S and K are able to support λ-abstraction, the combinators S and F are able to support a larger class of pattern-matching functions. That is, SF-combinators are *structure complete*. SF-logic is the first such to support a combinator for generic equality of normal forms, obtained by comparing internal structures.

Further, one may define *generic queries* for selecting or updating structures that generalise the usual database queries from rows and tables to arbitrary structures. These queries are slightly more general than those of *pattern calculus* since they interact with arbitrary normal forms, while the latter are limited to *data structures*.

It is worth emphasising that the novelty lies in the genericity of the queries, since traditional combinators are able to exploit internal structure when this is fixed in advance. For example, the internal structure of a number in unary arithmetic is accessed by the predecessor function, though the latter is surprisingly complex.