Fixed-Point Combinators

Suppose you have a lambda term \( g \), and you want another term \( k \) with the property that \( k \) is \( \beta \)-equivalent to \( gk \). Define terms

\[
\text{diag}(x) = xx
\]

and

\[
l(x) = g(\text{diag}(x))
\]

using our notational conventions; in other words, \( l \) is the term \( \lambda x. g(xx) \). Let \( k \) be the term \( ll \). Then we have

\[
k = (\lambda x. g(xx))(\lambda x. g(xx))
\]

\[
\triangleright g((\lambda x. g(xx))(\lambda x. g(xx)))
\]

\[
= gk.
\]

If one takes

\[
Y = \lambda g. ((\lambda x. g(xx))(\lambda x. g(xx)))
\]

then \( Yg \) and \( g(Yg) \) reduce to a common term; so \( Yg \equiv_{\beta} g(Yg) \). This is known as “Curry’s combinator.” If instead one takes

\[
Y = (\lambda xg. g(xxg))(\lambda xg. g(xxg))
\]

then in fact \( Yg \) reduces to \( g(Yg) \), which is a stronger statement. This latter version of \( Y \) is known as “Turing’s combinator.”

Photo Credits

Bibliography