PFPL Supplement: Comparing fix and self

Robert Harper

November 14, 2019

General recursion, \( \text{fix} \, : \tau \, \text{is} \, e \), is only sensible in a by-name dynamics for \( \text{PCF} \), because it steps to \([\text{fix} \, : \tau \, \text{is} \, e/x]e\), which substitutes a non-value for the variable \( x \) in the expression \( e \). In a by-value dynamics general recursion is, for this reason, not sensible and must be replaced by type-specific forms of self-reference. For example, the self-referential function form \( \text{fun} \{ \tau_1; \tau_2 \}(f.x.e) \) is postulated to be a value of type \( \tau_1 \rightarrow \tau_2 \), under appropriate typing constraints. Such a function is unrolled on application, substituting the recursive function value itself for the recursive variable, \( f \), in the body of the function.

In \( \text{FPC} \) an alternative account of self-reference is provided by the recursive type \( \tau \, \text{self} \), the type of self-referential values \( \text{self} \, x \, \text{is} \, e \). Within \( e \) the self-reference, \( x \), must be unrolled, writing \( \text{unroll}(x) \), to unroll the recursion and access the underlying expression \( e \). More precisely,

\[
\text{unroll}(\text{self} \, x \, \text{is} \, e) \mapsto \text{[self} \, x \, \text{is} \, e/x]e,
\]

which makes sense in either a by-name or a by-value dynamics. For example, the recursive factorial function has type \( (\text{nat} \rightarrow \text{nat}) \, \text{self} \), revealing that it is self-referential. To call such a function, either from the outside or internally within its definition, it is necessary to \( \text{unroll} \) the self-reference before applying it to an argument; the self-referential variable is only ever replaced by a value.

Curiously, in \( \text{FPC} \) it is possible to define \( \text{fix} \) from \( \text{self} \), obtaining the expected dynamics, even under a by-value interpretation! Specifically, define \( \hat{e} \) to be \( [\text{unroll}(x)/x]e \), and then define \( \text{fix} \, : \tau \, \text{is} \, e \) to be the expression \( \text{unroll}(\text{self} \, x \, \text{is} \, \hat{e}) \). Then observe the following transition:

\[
\text{fix} \, : \tau \, \text{is} \, e \triangleq \text{unroll}(\text{self} \, x \, \text{is} \, \hat{e})
\]

\[
\mapsto \text{[self} \, x \, \text{is} \, \hat{e}/x] \hat{e}
\]

\[
\triangleq [\text{self} \, x \, \text{is} \, \hat{e}/x] \text{[unroll}(x)/x]e
\]

\[
= [\text{unroll}(\text{self} \, x \, \text{is} \, \hat{e})/x]e
\]

\[
= \text{[fix} \, : \tau \, \text{is} \, e/x]e.
\]

The penultimate line is the result of the iterated substitution; it does not arise in the dynamics as a substitution of a non-value for a variable, which would be disallowed in the by-value case, but is rather part of the protocol for using \( \text{self} \) types instead of implicit recursion.

References