PFPL Supplement: Comparing fix and self

Robert Harper

October 2018

General recursion, \( \text{fix} \, x : \tau \rightarrow e \), is only sensible in a by-name dynamics for PCF, because it steps to \([\text{fix} \, x : \tau \rightarrow e \, / \, x] \, e\), 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, such as the self-referential function form \( \text{fun} \{ \tau_1 ; \tau_2 \}(f \cdot x \cdot e) \), which is postulated to be a value of type \( \tau_1 \rightarrow \tau_2 \). Such a function is a value that is unrolled on application, substituting the recursive function value itself for the recursive variable, \( f \), in the body of the function.

In FPC an alternative account of self-reference is provided by the recursive type \( \tau \text{self} \), the type of self-referential values \( \text{self} \, x \rightarrow e \). 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, we have the transition

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

which makes sense in either a by-name or a by-value dynamics.

Using self types the recursive factorial function has type \( (\text{nat} \rightarrow \text{nat}) \, \text{self} \), which reveals in its type that it is self-referential. To call such a function, either externally or internally within its definition, it is necessary to first \( \text{unroll} \) the self-reference and apply it to an argument, which may be either a value or a computation, depending on the dynamics. In any case the self-referential variable is only ever replaced by a value.

Curiously, in FPC we may define \( \text{fix} \) from self, and obtain the expected dynamics, even under a by-value dynamics! The “trick” is to anticipate the need to unroll any self-reference within \( \text{self} \, x \rightarrow e \) by forming \( \hat{e} \) to be \([\text{unroll} \, (x) / x] \, e\), and then defining \( \text{fix} \, x : \tau \rightarrow e \) to be the expression \( \text{unroll} \, (\text{self} \, x \rightarrow \hat{e}) \). Observe that

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

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

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

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

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

The penultimate line summarizes the result of the iterated substitution; it does not arise in the dynamics itself as a substitution of a non-value for a variable, which would be disallowed in the by-value case.
References