**ldf.1 Minimization**

The general recursive functions are those that can be obtained from the basic functions zero, succ, \( P^n \) by composition, primitive recursion, and regular minimization. To show that all general recursive functions are \( \lambda \)-definable we have to show that any function defined by regular minimization from a \( \lambda \)-definable function is itself \( \lambda \)-definable.

**Lemma ldf.1.** If \( f(x_1, \ldots, x_k, y) \) is regular and \( \lambda \)-definable, then \( g \) defined by
\[
g(x_1, \ldots, x_k) = \mu y \ f(x_1, \ldots, x_k, y) = 0
\]
is also \( \lambda \)-definable.

**Proof.** Suppose the lambda term \( F \) \( \lambda \)-defines the regular function \( f(\vec{x}, y) \). To \( \lambda \)-define \( h \) we use a search function and a fixpoint combinator:
\[
\text{Search} \equiv \lambda g. \lambda f \vec{x} y. \text{IsZero}(f \vec{x} y) y (g \vec{x}(\text{Succ } y))
\]
\[
H \equiv \lambda \vec{x}. (Y \text{ Search}) F \vec{x} 0
\]
where \( Y \) is any fixpoint combinator. Informally speaking, Search is a self-referencing function: starting with \( y \), test whether \( f \vec{x} y \) is zero: if so, return \( y \), otherwise call itself with Succ \( y \). Thus \((Y \text{ Search}) F \vec{n_1} \ldots \vec{n_k}0\) returns the least \( m \) for which \( f(n_1, \ldots, n_k, m) = 0 \).

Specifically, observe that
\[
(Y \text{ Search}) F \vec{n_1} \ldots \vec{n_k} \overset{\#}{\rightarrow} m
\]
if \( f(n_1, \ldots, n_k, m) = 0 \), or
\[
\overset{\#}{\rightarrow} (Y \text{ Search}) F \vec{n_1} \ldots \vec{n_k} m + 1
\]
otherwise. Since \( f \) is regular, \( f(n_1, \ldots, n_k, y) = 0 \) for some \( y \), and so
\[
(Y \text{ Search}) F \vec{n_1} \ldots \vec{n_k} 0 \overset{\#}{\rightarrow} h(n_1, \ldots, n_k).
\]

**Proposition ldf.2.** Every general recursive function is \( \lambda \)-definable.

**Proof.** By ??, all basic functions are \( \lambda \)-definable, and by ??, ??, and Lemma ldf.1, the \( \lambda \)-definable functions are closed under composition, primitive recursion, and regular minimization.

---

**Photo Credits**

**Bibliography**

1