Draft

Methods in Structures

Luca Cardelli

There is a slight technical problem in adding methods to dependent structures. The “self” parameter of a method in a binding needs to know the final signature, since the method may want to refer to methods to its right through self. So the usual left-to-right checking of structures and signatures does not quite work. I don’t think we want to make the entire binding recursive. So here is a left-to-right solution that accounts for mutual method access through self.

A *structure* {B} contains a dependent *binding* B, and a *signature* {D} contains a dependent *declaration* D. We keep track at all times of the final declaration, via the judgment $\Gamma \vdash B : D \int D'$, meaning that B has declaration D, but must still be “integrated” with D’. Eventually we prove $\Gamma \vdash B : D \int ()$ and we are done.

See [Harper Lillibridge 1993] for the notation and the other necessary rules. Here $\tilde{D}$ is the label-stripping function, and $\bar{\xi}$ is a vector of variables/labels, where each ξ is a type or term variable/label.

Bindings: B
Structures: {B}
Declarations: D
Signatures: {D}

$\Gamma \vdash A :: K$	type A has kind K
$\Gamma \vdash a : A$	term a has type A
$\Gamma \vdash B : D \int D'$	binding B has declaration D, pending D’
$\Gamma \vdash \{B\} : \{D\}$	structure {B} has signature {D}

$x \div A$	method x has result type A (used in contexts and declarations)

(Empty binding)
$$\frac{\Gamma \vdash D}{\Gamma \vdash () : () \int D}$$

(Type binding)
$$\frac{\Gamma \vdash B : D \int b \triangleright X::K, D' \qquad \Gamma, \tilde{D} \vdash A::K \qquad X \notin dom(\Gamma, \tilde{D})}{\Gamma \vdash B, b \triangleright X::K=A \ : \ D, b \triangleright X::K \int D'}$$

(Term binding)
$$\frac{\Gamma \vdash B : D \int b \triangleright x:A, D' \qquad \Gamma, \tilde{D} \vdash a:A \qquad x \notin dom(\Gamma, \tilde{D})}{\Gamma \vdash B, b \triangleright x:A=a \ : \ D, b \triangleright x:A \int D'}$$

Draft

(Method binding) where $S = \{D, b \triangleright x \div A\{\bar{\xi}\}, D'\}$

$$\frac{\Gamma \vdash B : D \int b \triangleright x \div A, D' \qquad \Gamma, \tilde{D}, y{:}S \vdash a{:}(A\{\overline{y.\xi}\}) \qquad x,y \notin dom(\Gamma, \tilde{D}) \quad \bar{\xi} \in dom(\tilde{D})}{\Gamma \vdash B, b \triangleright x \div A\{\bar{\xi}\} = \varsigma(y{:}S)a \;:\; D, b \triangleright x \div A\{\bar{\xi}\} \int D'}$$

(Structure)

$$\frac{\Gamma \vdash B : D \int ()}{\Gamma \vdash \{B\}{:}\{D\}}$$

(Method invocation)

$$\frac{\Gamma \vdash a : \{b \triangleright x \div A\}}{\Gamma \vdash a.b : A}$$

(Method override) where $S = \{D, b \triangleright x \div A\{\bar{\xi}\}, D'\}$

$$\frac{\Gamma \vdash a : S \qquad \Gamma, \bar{D}, y{:}S \vdash a'{:}A\{\overline{y.\xi}\} \qquad y \notin dom(\Gamma, \bar{D}) \quad \bar{\xi} \in dom(\tilde{D})}{\Gamma \vdash a.b{:=}\varsigma(y{:}S)a' : S}$$

The substitution $A\{\overline{y.\xi}\}$ in the (Method binding) rule needs some explanations. At first I wrote:

(Method binding 0) where $S = \{D, b \triangleright x \div A, D'\}$

$$\frac{\Gamma \vdash B : D \int b \triangleright x \div A, D' \qquad \Gamma, \bar{D}, y{:}S \vdash a{:}A \qquad x,y \notin dom(\Gamma, \bar{D}) \quad y \notin A}{\Gamma \vdash B, b \triangleright x \div A = \varsigma(y{:}S)a \;:\; D, b \triangleright x \div A \int D'}$$

Where the restriction $y \notin A$ is similar to the usual restriction for the dot notation in function result types. But, according to (Method binding 0), the following does not typecheck (I am abbraviating $\xi \triangleright \xi$ as ξ):

$$\{X{::}Type = Int, \quad z \div X = \varsigma(y{:}\{X{::}Type, z \div X\})\, y.z\} \quad : \quad \{X{::}Type = Int, \; z \div X\}$$

However, we know that the current D in (Method binding 0) is a prefix of the final S, which is the type of y. Hence y.X is really the same as X in the current context., for any X declared in D. This is what (Method binding) is saying, and the example above is then typeable.

By the way, for a similar situation [Harper Lillibridge 1993] uses the following rules:

$$\frac{\Gamma, x{:}A \vdash a{:}A'}{\Gamma \vdash \lambda(x{:}A)a : \Pi(x{:}A)A'}$$

$$\frac{\Gamma \vdash a'{:}\Pi(x{:}A)A' \qquad \Gamma \vdash a{:}A \qquad x \notin A'}{\Gamma \vdash a'(a) : A'}$$

I don't quite understand why the side condition is placed on elimination, and not on introduction. Without subsumption, if x occurs in A' then $\lambda(x{:}A)a$ is unusable, and we are only delaying the error messages. Subsumption can eliminate occurrences of x in A', but is this really useful?.