Comprehending types

L. G. Meredith
M. Stay

May 17, 2018

Abstract

Abstract.

1 Comprehension signatures

A **comprehension signature** Σ has a collection of sorts; the collection of types is generated inductively:

- There is a **unit type**, denoted by 1.
- Every sort is a type.
- Given two types A and B, there is a type $A \times B$.
- Given a type A, there is a **power type** PA.

The signature Σ also has a collection of **constant symbols,** a collection of **function symbols,** and a collection of **relation symbols.** To each constant symbol and to each relation symbol is assigned a type. To each function symbol is assigned a type $A \to B$, where A and B are types.

Formulae are terms of type $P1$; we can think of relation symbols as function symbols from A to $P1$. The well-formed formulae are defined recursively using the usual rules.

- If t is a term of type A and R is a relation symbol of type A, then $R(t)$ is an atomic formula with the same free variables as t.
- If z is a variable of type A, t is a term of type PA, and ϕ is a formula, then $\exists z \leftarrow t.\phi$ is a formula with free variables $\mathrm{FV}(\phi) \cup \mathrm{FV}(t) - \{z\}$.

Since well-formed formulae are well-typed, we can leave off the subscripts. Terms are generated recursively:

- The term $*$ is of type 1.
- Every constant of type A is a term of type A with no free variables.

- Every variable x of type A is a term of type A whose only free variable is x.
- If f is a function symbol of type $A \to B$ and t is a term of type A, then $f(t)$ is a term of type B with the same free variables as t.
- If s is a term of type A and t is a term of type B, then $\langle s, t\rangle$ is a term of type $A \times B$ with free variables $\mathrm{FV}(s) \cup \mathrm{FV}(t)$.
- If u is a term of type $A \times B$, then $\pi_1 u$ is a term of type A and $\pi_2 u$ is a term of type B, and both have the same free variables as u.
- If
  - $x_1, \ldots, x_n$ are variables of type $A_1 \times \cdots \times A_n$, respectively,
  - $t_1 \ldots, t_n$ are terms of type $PA_1, \ldots, PA_n$, respectively, where $\mathrm{FV}(t_i)$ may include any variable except those x_j such that $j \geq i$,
  - t is a term of type B, and
  - ϕ is a formula,

  then $[t \mid x_1 \leftarrow t_1 \ ; \ \cdots \ ; \ x_n \leftarrow t_n \text{ if } \phi]$ is a term of type PB with free variables $\bigcup_{i=1}^{n} FV(t_i) - \bigcup_{i=1}^{n}\{x_i\}$.

Note that the last rule means that term and formula formation is mutually recursive. A **suitable context** for a term or formula is an ordered list $\vec{y} = [y_1, \ldots, y_m]$ of variables, without repetitions and annotated with types, such that every free variable of that term or formula appears in $\vec{y}$.

2 Categorical semantics

An interpretation M of a comprehension signature Σ in a cartesian category V equipped with a strong monad (P, η, μ, σ) consists of the following, subject to the conditions on interpretations below:

- For each sort A in Σ, a chosen object MA in V.
- $M1$ is the terminal object 1 in V.
- If MA and MB have been defined, then $M(A \times B)$ is $MA \times MB$ in V.
- If MA has been defined, then $M(PA)$ is defined to be $P(MA)$ in V.
- For each constant symbol of type A, a chosen morphism $Ma\colon 1 \to MA$ in V.
- For each function symbol f of type $A \to B$, a chosen morphism $Mf\colon MA \to MB$ in V.
- For each relation symbol R of type A, a chosen morphism $MR\colon MA \to P1$ in V.

Given an interpretation M, the interpretation of a term t of type B in a suitable context $\vec{y}$ is a morphism $[\![\vec{y}.t]\!]_M \colon MA_1 \times \cdots \times MA_n \to MB$, where $A_1, \ldots, A_n$ are the types of $y_1, \ldots, y_n$, respectively. The interpretation of a formula ϕ in a suitable context $\vec{y}$ is a morphism $[\![\vec{y}.\phi]\!]_M \colon MA_1 \times \cdots \times MA_n \to P1$. We define these conditions on interpretations by recursion:

- $[\![\vec{y}.*]\!]_M$ is the unique morphism into 1.
- If a is a constant of type A, then $[\![\vec{y}.a]\!]_M$ is the composite $Ma \circ [\![\vec{y}.*]\!]_M$.
- $[\![\vec{y}.y_i]\!]_M$ is defined to be the ith projection $MA_1 \times \cdots MA_n \to MA_i$.
- Given a term t of type A and a function symbol f of type $A \to B$, if $[\![\vec{y}.t]\!]_M$ has been defined, then $[\![\vec{y}.f(t)]\!]_M$ is defined to be the composite $Mf \circ [\![\vec{y}.t]\!]_M$.
- If $[\![\vec{y}.s]\!]_M$ and $[\![\vec{y}.t]\!]_M$ have been defined, then $[\![\vec{y}.\langle s,t\rangle]\!]_M$ is defined to be $\langle [\![\vec{y}.s]\!]_M, [\![\vec{y}.t]\!]_M \rangle$.
- If u is a term of type $A \times B$ and $[\![\vec{y}.u]\!]_M$ has been defined, then $[\![\vec{y}.\pi_1 u]\!]_M$ is defined to be the composite $\pi_1 \circ [\![\vec{y}.u]\!]_M$, and similarly for π_2.
- If t is a term of type A, then $[\![\vec{y}.[t]]\!]_M$ is defined to be $\eta_A \circ [\![\vec{y}.t]\!]_M$.
- If
  - $\vec{y} = [y_1, \ldots, y_m]$ is a context and $A_1, \ldots, A_m$ are the types of $y_1, \ldots, y_n$, respectively,
  - $\vec{x} = [x_1, \ldots, x_n]$ is a context and $B_1, \ldots, B_n$ are the types of $x_1, \ldots, x_n$, respectively,
  - t is a term of type PC, and
  - $1 \le i \le n$ is such that $[y_1, \ldots, y_m, x_1, \ldots, x_i]$ is a suitable context for t,

  then let $g_{\vec{y},\vec{x},i,t}$ be the composite

  $$\begin{aligned}&\mu_{A_1 \times \cdots \times A_n \times B_1 \times \cdots \times B_i} \circ P\,(\sigma_{A_1 \times \cdots \times A_n \times B_1 \times \cdots \times B_i, C} \\ &\quad \circ (A_1 \times \cdots \times A_n \times B_1 \times \cdots \times B_i \times [\![\vec{y}.t]\!]_M) \circ \Delta_{A_1 \times \cdots \times A_n \times B_1 \times \cdots \times B_i})\,.\end{aligned}$$

  If

  - $\vec{y} = [y_1, \ldots, y_m]$ is a context and $A_1, \ldots, A_m$ are the types of $y_1, \ldots, y_n$, respectively,
  - $\vec{x} = [x_1, \ldots, x_n]$ is a context and $B_1, \ldots, B_n$ are the types of $x_1, \ldots, x_n$, respectively,
  - t is a term of type C,
  - ϕ is a formula,
  - $\vec{y} + \vec{x}$ is a suitable context for t, and

– $t_1 \ldots, t_n$ are terms of type $PB_1, \ldots, PB_n$, respectively, where $\mathrm{FV}(t_i)$ may include any variable in $\vec{y}$ and any x_j such that $j < i$,

then $[\![\vec{y}.[t \mid x_1 \leftarrow t_1 \ ; \ \cdots \ ; \ x_n \leftarrow t_n \text{ if } \phi]]\!]_M$ is defined to be the composite

$$M[\![\vec{y}.t]\!]_M \circ g_{\vec{y},\vec{x},n,\phi} \circ \left(\bigcirc_{i=1}^{n} g_{\vec{y},\vec{x},i-1,t_i} \right) \circ \eta_{\vec{y}} \colon MA_1 \times \cdots MA_m \to PC$$

in V.

- If R is a relation symbol of type B and t is a term of type B, then $[\![\vec{y}.R(t)]\!]_M$ is defined to be the composite $MR \circ [\![\vec{y}.t]\!]_M$ in V.
- If z is a variable of type A, t is a term of type PA, ϕ is a formula, and $\vec{y}, z$ is a suitable context for ϕ and t, then $[\![\vec{y}.\exists z \leftarrow A.\phi]\!]_M$ is defined to be the morphism $[\![\vec{y}.[*|z \leftarrow t \text{ if } \phi]]\!]_M$ in V.

An interpretation M **satisfies** a formula θ with truth value $t \colon 1 \to P1$ if for some suitable context $\vec{y}$, the morphism $[\![\vec{y}.\theta]\!]_M$ factors through t. We write this symbolically as $M \models^t_{\vec{y}} \theta$.

3 Equality, inhabitation, and logical connectives

Notably absent from a comprehension signature is any notion of equality, inhabitation, or logical connectives. These concepts all depend on the specifics of the monad, and give us an opportunity to consider broader notions of truth. For instance, when $V = \mathrm{Set}$ and P is the list (or free monoid) monad, $P1$ is isomorphic to the natural numbers $\mathbb{N}$. In this situation, there are infinitely many ways to be "true", each more true than the last, so there's no clear answer to the question "How true is the statement $x = x$?" Neither is there a unique way for an item x to be an inhabitant of the list $[x, y, x]$; we could have $x \in [x, y, x]$ be 2, because it occurs twice; or 3 because it last appears at position 3 in the list; or 1, emulating those programming languages like C that use 1 to mean true and 0 to mean false. Logical connectives have similar problems: $\wedge$ and $\vee$ could just as easily be min and max as bitwise operators. When $V = \mathrm{Set}$ and P is the monad

In the case where P is finitary, we can take the operations of the corresponding Lawvere theory as primitive. For instance, the list monad has a distinguished element $[]$, so we have a distinguished term of type PA for any type A. We can ask whether an interpretation satisfies a formula with truth value $[]$. We can also concatenate lists: given any two terms t, u of type PA for any A, we can form the term $t \otimes u$ of type PA.

4 Comprehensional Mitchell–Bénabou Language

A cartesian category V equipped with a strong monad (P, η, μ, σ) gives rise to a comprehension signature in a fairly straightforward way. The basic sorts of the signature are the objects of V, the function symbols are the morphisms of V, the constant symbols are the morphisms out of 1, and the relation symbols are the morphisms into $P1$. We call this the **comprehensional Mitchell–Bénabou Language** of $(V, P, \eta, \mu, \sigma)$, and it has a tautological interpretation in $(V, P, \eta, \mu, \sigma)$.

5 Kripke–Joyal semantics

???