Syntax and Semantics of Dependant Types

by Martin Hofmann

Louis Milhaud

January 29, 2024

Université Paris Saclay

Outline

Introduction

Definition

A dependant type is a family of types varying on the elements of another type.

Exemple:

$$Vec_\sigma(M), \quad M : \mathbb{N}$$

Built on:

- $nil_\sigma : Vec_\sigma(0)$
- $Cons_\sigma(U, V) : Vec_\sigma(Succ(M))$

with $U : \sigma$ and $V : Vec_\sigma(M)$

Syntax

$$\Gamma ::= \diamond$$

$$\mid \Gamma, x : \sigma \qquad \text{provided } x \text{ is not declared in } \Gamma$$

$$\sigma, \tau ::= \Pi x : \sigma.\tau \mid \Sigma x : \sigma.\tau \mid Id_\sigma(M, N) \mid \mathbb{N}$$

$$\begin{aligned} M, N, H, P ::= {} & x \mid \lambda x : \sigma.M^\tau \mid App_{[x:\sigma]\tau}(M, N) \mid Pair_{[x:\sigma]\tau}(M, N) \\ & \mid \boldsymbol{R}^{\Sigma}_{[z:\Sigma x:\sigma.\tau]\rho}([x : \sigma, y : \tau]H, M) \mid Refl_\sigma(M) \\ & \mid \boldsymbol{R}^{Id}_{[x:\sigma, y:\sigma, p:Id_\sigma(x,y)]\tau}([z : \sigma]H, M, N, P) \mid 0 \mid Suc(M) \mid \\ & \boldsymbol{R}^{\mathbb{N}}_{[n:\mathbb{N}]\sigma}(H_z, [n : \mathbb{N}, x : \sigma]H_s, M) \end{aligned}$$

Context morphisms

Let Γ and $\Delta \overset{\text{def}}{=} x_1 : \sigma_1, ..., x_n : \sigma_n$ be valid contexts.
$f \overset{\text{def}}{=} (M_1, ..., M_n)$ is a context morphism (we write $\Gamma \vdash f \implies \Delta$)
when the following n judgements hold:

$$\Gamma \vdash M_1 : \sigma_1$$

$$\Gamma \vdash M_2 : \sigma_2[M_1/x_1]$$

$$\dots$$

$$\Gamma \vdash M_n : \sigma_n[M_1/x_1][M_2/x_2] \dots [M_{n-1}/x_{n-1}]$$

Generalized substitution & Composition

If we have:

$$\vdash \Gamma, \Delta \ \text{context} \quad \Gamma \vdash \tau \ \text{type}$$

$$\Gamma \vdash f \implies \Delta$$

$$f \equiv (M_1, ..., M_n)$$

$$\Delta \equiv x_1 : \sigma_1, ..., x_n : \sigma_n$$

Then:

$$\tau[f/\Delta] \equiv \tau[M_1/x_1][M_2/x_2]...[M_n/x_n]$$

Thanks to **substitution** we can now define context morphism **composition**:

Let $\Delta \vdash g \implies \Theta$ a context morphism, with $g \equiv (N_1, ..., N_k)$

$$f \circ g \equiv (N_1[f/\Delta], ..., N_k[f/\Delta])$$

Semantic frameworks

Objects

Let's first define some data structures in our **semantic model**:

- $\mathcal{C}$ category of contexts and context morphisms
- for $\Gamma \in \mathcal{C}$ a collection $\mathit{Ty}_{\mathcal{C}}(\Gamma)$ of semantic types
- for $\Gamma \in \mathcal{C}$ and $\sigma \in \mathit{Ty}_{\mathcal{C}}$ a collection $\mathit{Tm}_{\mathcal{C}}(\Gamma, \sigma)$ of semantic terms

Context formation & type extension

Formation:

- $\top$ a **terminal** object in $\mathcal{C}$
- $\forall \Gamma \in \mathcal{C},\ \langle\rangle_\Gamma$ denotes the unique morphism from Γ to $\top$

Type Extension:

- $\forall (\Gamma, \sigma) \in \mathcal{C} \times \mathit{Ty}_{\mathcal{C}}(\Gamma),\ \Gamma.\sigma \in \mathcal{C}$ is the **comprehension** of σ
- in the term model:

$$\frac{\vdash \Gamma \text{ context} \quad \Gamma \vdash \sigma \text{ type}}{\Gamma, x : \sigma \text{ context}}$$

Substitution

Semantic substitution is described by one operation for types and one for terms.

Let $f : \Gamma \to \Delta$, $g : \Delta \to \Theta$, $\sigma \in Ty(\Theta)$ and $M \in Tm(\Theta, \sigma)$

- $-\{g\} : Ty(\Theta) \to Ty(\Delta)$
- $-\{g\} : Tm(\Theta, \sigma) \to Tm(\Delta, \sigma\{g\})$
- compatible with composition and identities:

$$
\begin{aligned}
\sigma\{id_\Theta\} &= \sigma && \in Ty(\Theta) \\
\sigma\{g \circ f\} &= \sigma\{g\}\{f\} && \in Ty(\Gamma) \\
M\{id_\Theta\} &= M && \in Tm(\Theta, \sigma) \\
M\{g \circ f\} &= M\{g\}\{f\} && \in Tm(\Gamma, \sigma\{g \circ f\})
\end{aligned}
$$

p & v Morphisms

p *morphism:*

- $\mathrm{p}(\sigma) : \Gamma.\sigma \to \Gamma$ is the projection associated to σ
- in the term model:

$$\Gamma, x : \sigma \vdash \mathrm{p} \implies \Gamma$$

v *morphism:*

- $\mathrm{v}_\sigma \in Tm_{\mathcal{C}}(\Gamma.\sigma, \sigma\{\mathrm{p}(\sigma)\})$ is the second projection
- in the term model:

$$\Gamma, x : \sigma \vdash x : \sigma$$

Term Extension

Let $f : \Gamma \to \Delta,\ \sigma \in Ty(\Delta)$ and $M \in Tm(\Gamma, \sigma\{f\})$.

- $\langle f, M\rangle_\sigma : \Gamma \to \Delta.\sigma$ is the extension of f by M
- if $g : \Theta \to \Gamma$ then it satisfies the following:

$$
\begin{aligned}
\mathrm{p}(\sigma) \circ \langle f, M\rangle_\sigma &= f && : \Gamma \to \Delta \\
\mathrm{v}\{\langle f, M\rangle_\sigma\} &= M && \in Tm(\Gamma, \sigma\{f\}) \\
\langle f, M\rangle_\sigma \circ g &= \langle f \circ g, M\{g\}\rangle_\sigma && : \Theta \to \Delta.\sigma \\
\langle \mathrm{p}(\sigma), \mathrm{v}\rangle_\sigma &= id_{\Delta.\sigma} && : \Delta.\sigma \to \Delta.\sigma
\end{aligned}
$$

CwF: first definition

To recap, a **Category with families** is the following tuple:

$$(\mathcal{C}, \mathit{Ty}, \mathit{Tm}, -\{-\}, \top, \langle\rangle_{-}, -.-, \mathtt{p}, \mathtt{v}, \langle -, -\rangle_{-})$$

Definitions & $\mathcal{F}am$ Category

The category $\mathcal{F}am$ of families of sets has:

- as objects pairs $A = (A^0, A^1)$
- as arrows f between A and B a pair (f^0, f^1)

We also define the functor $\mathcal{F} : \mathcal{C}^{op} \to \mathcal{F}am$ such that:

$$\mathcal{F}(\Gamma) = (Ty_{\mathcal{C}}(\Gamma), (Tm_{\mathcal{C}}(\Gamma, \sigma))_{\sigma \in Ty_{\mathcal{C}}(\Gamma)})$$

CwF: second definition

We can now define a category with families with:

- a category $\mathcal{C}$ with terminal object
- a functor $\mathcal{F} = (\mathit{Ty}, \mathit{Tm}) : \mathcal{C}^{\mathit{op}} \to \mathcal{F}\mathit{am}$
- a comprehension for each $\Gamma \in \mathcal{C}$ and $\sigma \in \mathit{Ty}_{\mathcal{C}}(\Gamma)$

q morphism & Weakening

- *q morphism:*

Let $f : \Theta \to \Gamma$ and $\sigma \in Ty(\Gamma)$

$$\begin{aligned} \mathrm{q}(f, \sigma) &: \Theta.\sigma\{f\} \to \Gamma.\sigma \\ &\overset{\text{def}}{=} \langle f \circ \mathrm{p}(\sigma\{f\}), \mathrm{v}_{\sigma\{f\}} \rangle_\sigma \end{aligned}$$

- *Weakening maps:*

$$\begin{aligned} w := \quad & \mathrm{p}(\sigma) : \Gamma.\sigma \to \Gamma \\ \mid \quad & \mathrm{q}(w, \tau) \end{aligned}$$

Pullback property

Let $\mathcal{C}$ a CwF, $f : \Theta \to \Gamma$ and $\sigma \in Ty(\Gamma)$ the following diagram commutes:

$$\begin{tikzcd}
\Theta.\sigma\{f\} \arrow[r, "q(f,\sigma)"] \arrow[d, "p(\sigma\{f\})"'] & \Gamma.\sigma \arrow[d, "p(\sigma)"] \\
\Theta \arrow[r, "f"'] & \Gamma
\end{tikzcd}$$

CwA definition

A **category with attributes** consists of:

- A category $\mathcal{C}$ with terminal object $\top$
- A functor $Ty : \mathcal{C}^{op} \to Set$
- $\forall \sigma \in Ty(\Gamma)$ an object $\Gamma.\sigma$ and a morphism $\mathrm{p}(\sigma) : \Gamma.\sigma \to \Gamma$
- $\forall f : \Theta \to \Gamma$ and $\sigma \in Ty(\Gamma)$ a pullback diagram:

$$
\begin{tikzcd}
\Theta.\sigma\{f\} \arrow[r, "q(f,\sigma)"] \arrow[d, "p(\sigma\{f\})"'] & \Gamma.\sigma \arrow[d, "p(\sigma)"] \\
\Theta \arrow[r, "f"'] & \Gamma
\end{tikzcd}
$$

such that $\mathrm{q}(id_\Gamma, \sigma) = id_{\Gamma.\sigma}$ and $\mathrm{q}(f \circ g, \sigma) = \mathrm{q}(f, \sigma) \circ \mathrm{q}(g, \sigma\{f\})$.

Bonus

$\prod$ Type former

- Type former for the functions with return type depending on the parameter
- Set-theoretic equivalent:
  Cartesian product over a family of sets: $\Pi_{i \in I} B_i$
- preserved by definitional equality

$$\frac{\Gamma \vdash \sigma \text{ type} \quad \Gamma, x : \sigma \vdash \tau \text{ type}}{\Gamma \vdash \Pi x : \sigma . \tau \text{ type}} \mathit{Form}$$

Π Rules

$$\frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash \lambda x : \sigma . M^{\tau} : \Pi x : \sigma . \tau} \mathit{Intro}$$

$$\frac{\Gamma \vdash M : \Pi x : \sigma . \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash \mathit{App}_{[x:\sigma]\tau}(M, N) : \tau[N/x]} \mathit{Elim}$$

$$\frac{\Gamma \vdash \lambda x : \sigma . M^{\tau} : \Pi x : \sigma . \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash \mathit{App}_{[x:\sigma]\tau}(\lambda x : \sigma . M^{\tau}, N) = M[N/x] : \tau[N/x]} \mathit{Comp}$$

Π type former interpretation

For each $\sigma \in Ty(\Gamma)$, $\tau \in Ty(\Gamma.\sigma)$, $L \in Tm(\Gamma.\sigma, \tau)$, $M \in Tm(\Gamma, \Pi(\sigma, \tau))$ and $N \in Tm(\Gamma, \sigma)$ we can define:

- the type $\Pi(\sigma, \tau) \in Ty(\Gamma)$
- the term $\lambda_{\sigma,\tau}(L) \in Tm(\Gamma, \Pi(\sigma, \tau))$
- the term $App_{\sigma,\tau}(M, N) \in Tm(\Gamma, \tau\{\overline{M}\})$

such that

$$\begin{aligned} App_{\sigma,\tau}(\lambda_{\sigma,\tau}(M), N) &= M\{\overline{N}\} && \Pi - C \\ \Pi(\sigma, \tau)\{f\} &= \Pi(\sigma\{f\}, \tau\{\mathsf{q}(f, \sigma)\}) && \Pi - S \\ \lambda_{\sigma,\tau}(M)\{f\} &= \lambda_{\sigma\{f\},\tau\{\mathsf{q}(f,\sigma)\}}(M\{\mathsf{q}(f, \sigma)\}) && \lambda - S \\ App_{\sigma,\tau}(M, N)\{f\} &= App_{\sigma\{f\},\tau\{\mathsf{q}(f,\sigma)\}}(M\{f\}, N\{f\}) && App - S \end{aligned}$$

Interpretation function

Let $\llbracket - \rrbracket$ an interpretation function such that:

$$
\begin{aligned}
\llbracket \diamond \rrbracket &= \top \\
\llbracket \Gamma, x : \sigma \rrbracket &= \llbracket \Gamma \rrbracket . \llbracket \Gamma ; \sigma \rrbracket && \text{if } x \notin \Gamma \text{, undefined otherwise} \\
\llbracket \Gamma ; \Pi x : \sigma . \tau \rrbracket &= \Pi (\llbracket \Gamma ; \sigma \rrbracket , \llbracket \Gamma , x : \sigma ; \tau \rrbracket) \\
\llbracket \Gamma , x : \sigma ; x \rrbracket &= \mathtt{v}_{\llbracket \Gamma ; \sigma \rrbracket} \\
\llbracket \Gamma , x : \sigma , \Delta , y : \tau ; x \rrbracket &= \llbracket \Gamma , x : \sigma , \Delta ; x \rrbracket \{ \mathtt{p} (\llbracket \Gamma , x : \sigma , \Delta ; \tau \rrbracket) \} \\
\llbracket \Gamma ; App_{\sigma , [x : \sigma] \tau} (M , N) \rrbracket &= App_{\llbracket \Gamma ; \sigma \rrbracket , \llbracket \Gamma , x : \sigma ; \tau \rrbracket} (\llbracket \Gamma ; M \rrbracket , \llbracket \Gamma ; N \rrbracket) \\
\llbracket \Gamma ; \lambda x : \sigma . M^{\tau} \rrbracket &= \lambda_{\llbracket \Gamma ; \sigma \rrbracket , \llbracket \Gamma , x : \sigma ; \tau \rrbracket} (\llbracket \Gamma , x : \sigma ; M \rrbracket)
\end{aligned}
$$

Soundness properties

$$
\begin{aligned}
\Gamma \vdash \quad &\overset{\mathsf{S}}{\Longrightarrow} \quad [\![\Gamma]\!] \in \mathcal{C} \\
\Gamma \vdash \sigma \quad &\overset{\mathsf{S}}{\Longrightarrow} \quad [\![\Gamma; \sigma]\!] \in \mathsf{Ty}([\![\Gamma]\!]) \\
\Gamma \vdash \mathsf{M} : \sigma \quad &\overset{\mathsf{S}}{\Longrightarrow} \quad [\![\Gamma; \mathsf{M}]\!] \in \mathsf{Tm}([\![\Gamma; \sigma]\!]) \\
\vdash \Gamma = \Delta \text{ context} \quad &\overset{\mathsf{S}}{\Longrightarrow} \quad [\![\Gamma]\!] = [\![\Delta]\!] \\
\Gamma \vdash \sigma = \tau \text{ type} \quad &\overset{\mathsf{S}}{\Longrightarrow} \quad [\![\Gamma; \sigma]\!] = [\![\Gamma; \tau]\!] \\
\Gamma \vdash \mathsf{M} = \mathsf{N} : \sigma \quad &\overset{\mathsf{S}}{\Longrightarrow} \quad [\![\Gamma; \mathsf{M}]\!] = [\![\Gamma; \mathsf{N}]\!]
\end{aligned}
$$