Martin-Löf's type theory

Hugo Herbelin and Nicolas Tabareau

January 8, 2019

1 Core type theory

The core infrastructure of type theory is presented on Figure 1.

2 Universes

Extension with a hierarchy of universe is obtained with the rules on Figure 2.

3 Identity type

Extension with an identity type is obtained with the rules on Figure 3.

4 Dependent function type

Extension with a dependent function type is obtained with the rules on Figure 4. One assumes given a function $\Pi(l_1, l_2)$ on universe levels. We shall occasionally use the following syntactic abbreviations:

$$\begin{array}{lcll} A \to B & \triangleq & \Pi a\!:\!A.\,B & \text{for } a \text{ fresh variable} \\ A \Rightarrow B & \triangleq & \Pi a\!:\!A.\,B & \text{for } a \text{ fresh variable} \\ \forall a : A.B & \triangleq & \Pi a\!:\!A.\,B & \end{array}$$

5 Dependent sum type

Extension with a dependent sum type is obtained with the rules on Figure 5. One assumes given a function $\Sigma(l_1, l_2)$ on universe levels. We shall occasionally use the following syntactic abbreviations:

$$\begin{array}{lcll} A \times B & \triangleq & \Sigma a\!:\!A.\,B & \text{for } a \text{ fresh variable} \\ A \wedge B & \triangleq & \Sigma a\!:\!A.\,B & \text{for } a \text{ fresh variable} \\ \exists a : A.B & \triangleq & \Sigma a\!:\!A.\,B & \end{array}$$

6 Natural numbers

Extension with Peano natural numbers is obtained with the rules on Figure 6. One assumes given a universe level $l_{\mathbb{N}}$ where $\mathbb{N}$ lives.

syntax of contexts

$$\Gamma \quad ::= \quad \square \mid \Gamma, a : A$$

syntax of expressions

$$t, u, A, B \quad ::= \quad a \mid \mathsf{U}_l$$

where a ranges over a set of variables
and l ranges over a set of universe levels

context formers

$$\frac{}{\square\ \mathsf{wf}}\ \mathsf{Ctx}_\square \qquad \frac{\Gamma \vdash A : \mathsf{U}_l}{\Gamma, a : A\ \mathsf{wf}}\ \mathsf{Ctx}_{\mathsf{cons}}$$

axiom

$$\frac{\Gamma, a : A, \Gamma'\ \mathsf{wf}}{\Gamma, a : A, \Gamma' \vdash a : A}\ \mathsf{Ax}$$

conversion rule

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash A \equiv B : \mathsf{U}_l}{\Gamma \vdash t : B}\ \mathsf{Conv}$$

definitional equality

$$\frac{\Gamma \vdash t \ \triangleright\ u : A}{\Gamma \vdash t \equiv u : A}\ \mathsf{Conv}_{\triangleright}$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t \equiv t : A}\ \mathsf{Refl}_\equiv \qquad \frac{\Gamma \vdash t \equiv u : A}{\Gamma \vdash u \equiv t : A}\ \mathsf{Sym}_\equiv \qquad \frac{\Gamma \vdash t \equiv u : A \qquad \Gamma \vdash u \equiv v : A}{\Gamma \vdash t \equiv v : A}\ \mathsf{Trans}_\equiv$$

Figure 1: Core structure of type theory

type former

$$\frac{}{\Gamma \vdash \mathsf{U}_l : \mathsf{U}_{\mathcal{S}_l}}\ \mathsf{U}$$

Figure 2: Universes in type theory

extended syntax of expressions

$$t, u, v, A, B, p, q \quad ::= \quad \ldots \mid t =_A u \mid \mathsf{refl}\, t \mid \mathsf{subst}\, p\, \mathsf{in}\, v$$

type former

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash u : A}{\Gamma \vdash t =_A u : \mathsf{U}_l}$$

introduction rule

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash \mathsf{refl}\, t : t =_A t}$$

elimination rule

$$\frac{\Gamma \vdash p : t =_A u \qquad \Gamma, a : A, b : t =_A a \vdash P : \mathsf{U}_l \qquad \Gamma \vdash v : P[t/a][\mathsf{refl}\, t/b]}{\Gamma \vdash \mathsf{subst}\, p\, \mathsf{in}\, v : P[u/a][p/b]}$$

reduction rule

$$\frac{\Gamma \vdash t : A \qquad \Gamma, a : A, b : t =_A a \vdash B : \mathsf{U}_l \qquad \Gamma \vdash v : B[t/a][\mathsf{refl}\, t/b]}{\Gamma \vdash \mathsf{subst}\, \mathsf{refl}\, t\, \mathsf{in}\, v \ \triangleright\ v : B[t/a][\mathsf{refl}\, t/b]}$$

congruence rules

$$\frac{\Gamma \vdash A \equiv A' \qquad \Gamma \vdash t \equiv t' : A \qquad \Gamma \vdash u \equiv u' : A}{\Gamma \vdash (t =_A u) \equiv (t' =_{A'} u') : \mathsf{U}_l}$$

$$\frac{\Gamma \vdash t \equiv t' : A}{\Gamma \vdash \mathsf{refl}\, t \equiv \mathsf{refl}\, t' : t =_A t} \qquad \frac{\Gamma \vdash p \equiv p' : t =_A u \qquad \Gamma, a : A, q : t =_A a \vdash B : \mathsf{U}_l \qquad \Gamma \vdash v \equiv v' : B[t/a][\mathsf{refl}\, t/q]}{\Gamma \vdash \mathsf{subst}\, p\, \mathsf{in}\, v \equiv \mathsf{subst}\, p'\, \mathsf{in}\, v' : B[u/a][p/q]}$$

Figure 3: Identity type

extended syntax of expressions

$$t, u, v, A, B, p, q \quad ::= \quad \ldots \mid \Pi a{:}A.\, B \mid \lambda a{:}A.\, u \mid v\, t$$

type former

$$\frac{\Gamma \vdash A : \mathsf{U}_{l_1} \qquad \Gamma, a : A \vdash B : \mathsf{U}_{l_2}}{\Gamma \vdash \Pi a{:}A.\, B : \mathsf{U}_{\Pi(l_1, l_2)}}$$

introduction rule

$$\frac{\Gamma, a : A \vdash u : B}{\Gamma \vdash \lambda a{:}A.\, u : \Pi a{:}A.\, B}$$

elimination rule

$$\frac{\Gamma \vdash v : \Pi a{:}A.\, B \qquad \Gamma \vdash t : A}{\Gamma \vdash v\, t : B[t/a]}$$

reduction rule

$$\frac{\Gamma, a : A \vdash u : B \qquad \Gamma \vdash t : A}{\Gamma \vdash (\lambda a{:}A.\, u)\, t \,\triangleright\, u[t/a] : B[t/a]}\ \beta_\Pi$$

observational rule

$$\frac{\Gamma \vdash v : \Pi a{:}A.\, B}{\Gamma \vdash \lambda a{:}A.\, v\, a \,\triangleright\, v : \Pi a{:}A.\, B}\ \eta_\Pi$$

congruence rules

$$\frac{\Gamma \vdash A \equiv A' : \mathsf{U}_{l_1} \qquad \Gamma, a : A \vdash B \equiv B' : \mathsf{U}_{l_2}}{\Gamma \vdash \Pi a{:}A.\, B \equiv \Pi a{:}A'.\, B' : \mathsf{U}_{\Pi(l_1, l_2)}}$$

$$\frac{\Gamma \vdash A \equiv A' : \mathsf{U}_{l} \qquad \Gamma, a : A \vdash u \equiv u' : B}{\Gamma \vdash \lambda a{:}A.\, u \equiv \lambda a{:}A'.\, u' : \Pi a{:}A.\, B}$$

$$\frac{\Gamma \vdash v \equiv v' : \Pi a{:}A.\, B \qquad \Gamma \vdash t \equiv t' : A}{\Gamma \vdash v\, t \equiv v'\, t' : B[t/a]}$$

Figure 4: Typing and computational rules for Π

extended syntax of expressions

$$t, u, v, A, B, p, q \quad ::= \quad \ldots \mid \Sigma a\!:\!A.\,B \mid \langle t, u\rangle \mid v.1 \mid v.2$$

type former

$$\frac{\Gamma \vdash A : \mathsf{U}_{l_1} \qquad \Gamma, a : A \vdash B : \mathsf{U}_{l_2}}{\Gamma \vdash \Sigma a\!:\!A.\,B : \mathsf{U}_{\Sigma(l_1,l_2)}}$$

introduction rule

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash u : B[t/a]}{\Gamma \vdash \langle t, u\rangle : \Sigma a\!:\!A.\,B}$$

elimination rules

$$\frac{\Gamma \vdash v : \Sigma a\!:\!A.\,B}{\Gamma \vdash v.1 : A} \qquad \frac{\Gamma \vdash v : \Sigma a\!:\!A.\,B}{\Gamma \vdash v.2 : B[v.1/a]}$$

reduction rules

$$\frac{\Gamma \vdash t : A \qquad \Gamma \vdash u : B[t/a]}{\Gamma \vdash (\langle t, u\rangle).1 \,\triangleright\, t : A}\ \beta^1_\Sigma \qquad \frac{\Gamma \vdash t : A \qquad \Gamma \vdash u : B[t/a]}{\Gamma \vdash (\langle t, u\rangle).2 \,\triangleright\, u : B[t/a]}\ \beta^2_\Sigma$$

observational rule

$$\frac{\Gamma \vdash v : \Sigma a\!:\!A.\,B}{\Gamma \vdash \langle v.1, v.2\rangle \,\triangleright\, v : \Sigma a\!:\!A.\,B}\ \eta_\Sigma$$

congruence rules

$$\frac{\Gamma \vdash A \equiv A' : \mathsf{U}_{l_1} \qquad \Gamma, a : A \vdash B \equiv B' : \mathsf{U}_{l_2}}{\Gamma \vdash \Sigma a\!:\!A.\,B \equiv \Sigma a\!:\!A'.\,B' : \mathsf{U}_{\Sigma(l_1,l_2)}}$$

$$\frac{\Gamma \vdash t \equiv t' : A \qquad \Gamma, a : A \vdash u \equiv u' : B}{\Gamma \vdash \langle t, u\rangle \equiv \langle t', u'\rangle : \Sigma a\!:\!A.\,B}$$

$$\frac{\Gamma \vdash v \equiv v' : \Sigma a\!:\!A.\,B}{\Gamma \vdash v.1 \equiv v'.1 : A} \qquad \frac{\Gamma \vdash v \equiv v' : \Sigma a\!:\!A.\,B}{\Gamma \vdash v.2 \equiv v'.2 : B[v.1/a]}$$

Figure 5: Typing and computational rules for Σ

extended syntax of expressions

$$t, u, v, A, B, p, q \quad ::= \quad \dots \mid \mathbb{N} \mid 0 \mid \mathsf{succ}\, t \mid \mathsf{rec}\, [0 \mapsto t \,|\, \mathsf{succ}\, a \mapsto_b u]\, v$$

type former

$$\frac{}{\Gamma \vdash \mathbb{N} : \mathsf{U}_{l_{\mathbb{N}}}}$$

introduction rules

$$\frac{}{\Gamma \vdash 0 : \mathbb{N}} \qquad \frac{\Gamma \vdash t : \mathbb{N}}{\Gamma \vdash \mathsf{succ}\, t : \mathbb{N}}$$

elimination rule

$$\frac{\Gamma, a : \mathbb{N} \vdash B : \mathsf{U}_l \qquad \Gamma \vdash t : B[0/a] \qquad \Gamma, a : \mathbb{N}, b : B[n/a] \vdash u : B[\mathsf{succ}\, n/a] \qquad \Gamma \vdash v : \mathbb{N}}{\Gamma \vdash \mathsf{rec}\, [0 \mapsto t \,|\, \mathsf{succ}\, a \mapsto_b u]\, v : B[v/a]}$$

reduction rules

$$\frac{\Gamma, a : \mathbb{N} \vdash B : \mathsf{U}_l \qquad \Gamma \vdash t : B[0/a] \qquad \Gamma, a : \mathbb{N}, b : B[n/a] \vdash u : B[\mathsf{succ}\, n/a]}{\Gamma \vdash \mathsf{rec}\, [0 \mapsto t \,|\, \mathsf{succ}\, a \mapsto_b u]\, 0 \rhd t : B[0/a]} \ \beta^0_{\mathbb{N}}$$

$$\frac{\Gamma, a : \mathbb{N} \vdash B : \mathsf{U}_l \qquad \Gamma \vdash t : B[0/a] \qquad \Gamma, a : \mathbb{N}, b : B[n/a] \vdash u : B[\mathsf{succ}\, n/a] \qquad \Gamma \vdash v : \mathbb{N}}{\Gamma \vdash \mathsf{rec}\, [0 \mapsto t \,|\, \mathsf{succ}\, a \mapsto_b u]\, \mathsf{succ}\, v \rhd u[v/a][\mathsf{rec}\, [0 \mapsto t \,|\, \mathsf{succ}\, a \mapsto_b u]\, v/b] : B[\mathsf{succ}\, v/a]} \ \beta^{\mathsf{succ}}_{\mathbb{N}}$$

observational rule

$$\frac{\Gamma, a : \mathbb{N} \vdash E[a] : A \qquad \Gamma \vdash v : \mathbb{N}}{\Gamma \vdash \mathsf{rec}\, [0 \mapsto E[0] \,|\, \mathsf{succ}\, a \mapsto_b E[\mathsf{succ}\, a]]\, v \rhd E[v] : A} \ \eta_{\mathbb{N}}$$

where $E[a]$ is made only from elimination rules applied to a

congruence rules

$$\frac{\Gamma \vdash t \equiv t' : \mathbb{N}}{\Gamma \vdash \mathsf{succ}\, t \equiv \mathsf{succ}\, t' : \mathbb{N}}$$

$$\frac{\Gamma \vdash t \equiv t' : B[0/a] \qquad \Gamma, a : \mathbb{N}, b : B[n/a] \vdash u \equiv u' : B[\mathsf{succ}\, n/a] \qquad \Gamma \vdash v \equiv v' : \mathbb{N}}{\Gamma \vdash \mathsf{rec}\, [0 \mapsto t \,|\, \mathsf{succ}\, a \mapsto_b u]\, v \equiv \mathsf{rec}\, [0 \mapsto t' \,|\, \mathsf{succ}\, a \mapsto_b u']\, v' : B[v/a]}$$

Figure 6: Typing and computational rules for $\mathbb{N}$

extended syntax of expressions

$$t, u, v, A, B, p, q \quad ::= \quad \ldots \mid \mathsf{Stream}\, A \mid \mathsf{hd}\, t \mid \mathsf{tl}\, t \mid \{\mathsf{hd} \mapsto t; \mathsf{tl} \mapsto_s u\}_c v$$

type former

$$\frac{\Gamma \vdash A : \mathsf{U}_l}{\Gamma \vdash \mathsf{Stream}\, A : \mathsf{U}_l}$$

introduction rule

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, c : C \vdash t : A \qquad \Gamma, s : C \to \mathsf{Stream}\, A, c : C \vdash u : \mathsf{Stream}\, A \qquad \Gamma \vdash v : C}{\Gamma \vdash \{\mathsf{hd} \mapsto t; \mathsf{tl} \mapsto_s u\}_c v : \mathsf{Stream}\, A[v/c]}$$

elimination rules

$$\frac{\Gamma \vdash A : \mathsf{U}_l \qquad \Gamma \vdash t : \mathsf{Stream}\, A}{\Gamma \vdash \mathsf{hd}\, t : A} \qquad \frac{\Gamma \vdash A : \mathsf{U}_l \qquad \Gamma \vdash t : \mathsf{Stream}\, A}{\Gamma \vdash \mathsf{tl}\, t : \mathsf{Stream}}$$

reduction rules

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, c : C \vdash t : A \qquad \Gamma, s : C \to \mathsf{Stream}\, A, c : C \vdash u : \mathsf{Stream}\, A \qquad \Gamma \vdash v : C}{\Gamma \vdash \mathsf{hd}\, \{\mathsf{hd} \mapsto t; \mathsf{tl} \mapsto_s u\}_c v \;\triangleright\; t[v/c] : A[v/c]}\ \beta^{\mathsf{hd}}_{\mathsf{Stream}}$$

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, c : C \vdash t : A \qquad \Gamma, s : C \to \mathsf{Stream}\, A, c : C \vdash u : \mathsf{Stream}\, A \qquad \Gamma \vdash v : C}{\Gamma \vdash \mathsf{tl}\, \{\mathsf{hd} \mapsto t; \mathsf{tl} \mapsto_s u\}_c v \;\triangleright\; u[v/c][\{\mathsf{hd} \mapsto t; \mathsf{tl} \mapsto_s u\}_c x / s\, x] : \mathsf{Stream}\, A}\ \beta^{\mathsf{tl}}_{\mathsf{Stream}}$$

observational rule

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, a : C \vdash t : \mathsf{Stream}\, A}{\Gamma \vdash \{\mathsf{hd} \mapsto \mathsf{hd}\, t; \mathsf{tl} \mapsto_s \mathsf{tl}\, t\}_a a \equiv t : \mathsf{Stream}\, A}\ \eta_{\mathsf{Stream}}$$

congruence rules

$$\frac{\Gamma \vdash A \equiv A' : \mathsf{U}_l}{\Gamma \vdash \mathsf{Stream}\, A \equiv \mathsf{Stream}\, A' : \mathsf{U}_l} \qquad \frac{\Gamma \vdash t \equiv t' : \mathsf{Stream}\, A}{\Gamma \vdash \mathsf{hd}\, t \equiv \mathsf{hd}\, t' : A} \qquad \frac{\Gamma \vdash t \equiv t' : \mathsf{Stream}\, A}{\Gamma \vdash \mathsf{tl}\, t \equiv \mathsf{tl}\, t' : A}$$

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma \vdash t \equiv t' : A \qquad \Gamma, s : C \to \mathsf{Stream}\, A, c : C \vdash u \equiv u' : \mathsf{Stream}\, A \qquad \Gamma \vdash v \equiv v' : C}{\Gamma \vdash \{\mathsf{hd} \mapsto t; \mathsf{tl} \mapsto_s u\}_c v \equiv \{\mathsf{hd} \mapsto t'; \mathsf{tl} \mapsto_s u'\}_c v' : \mathsf{Stream}\, A}$$

Figure 7: Typing and computational rules for streams

7 Streams

Extension with streams (infinite lists) is obtained with the rules on Figure 7.

8 Generic positive types

We give a syntax for arbitrary forms of (non-recursive) positive type, as a (non-recursive) generalization of the type $\mathbb{N}$. For that purpose, we introduce a couple of auxiliary structures.

We introduce a class of positive types, denoted by the letter P and we reuse for that purpose the notation $\otimes$ of linear logic, but this time in a dependent form (i.e. the type on the right can depend on the inhabitant of the type of the left), and in an intuitionistic setting (i.e. with contraction and weakening allowed).

We introduce a class w of inhabitants of such positive types and a class ρ of patterns for matching inhabitants of such positive types. These patterns can be declared in the context.

A positive type has the form $(c_1 : P_1 \oplus ... \oplus c_n : P_n)^{w}_{\rho:P}$ where w are the parameters of the type and the c_i are the names of constructors (assumed all distinct).

A constructor of this type has the form $c_i\, w$. A destructor has the form $\mathsf{case}\; t\; \mathsf{of}\; [c_1\rho \mapsto t|...|c_n\rho \mapsto t]$.

Substitution of ρ by w is as expected. Note that the axiom rule needs to be generalized so as to extract variables of a pattern.

9 Generic negative types

We give a syntax for arbitrary forms of (non-recursive) negative type, as a (non-recursive) generalization of the type $\mathsf{Stream}\, A$.

A negative type has the form $\{d_1 : A \,\&\, ... \,\&\, d_n : A\}^{w}_{\rho:P}$ where w are the parameters of the type and the d_i are the names of destructors (assumed all distinct). A constructor of this type has the form $\{d_1 \mapsto t; ...; d_n \mapsto t\}^{v}_{c}$. A destructor has the form $t\,$.

10 Recursive types

We give a syntax for recursive types, i.e. for types defined as smallest type generated by its constructors. Recursion is expected to occur only in *strictly positive* position, as e.g. in $\mu X.(1 \oplus X)$ (which is isomorphic to $\mathbb{N}$) or $\mu X.(1 \oplus A \otimes X)_{A:=\mathbb{N}}$ (which denotes the type of lists of natural numbers), or $\mu X.\{\mathsf{hd} : \mathbb{N} \,\&\, \mathsf{tl} : (1 \oplus X)\}$ (which denotes the negative presentation of lists).

We restricted the rules to the case of recursion on a variable $X : \mathsf{U}_l$. This could be extended to mutual recursion on a tuple of type variable. This could be extended as well to a recursion on arities, i.e. on variables X of type $\Pi a_1 : A_1. ... \Pi a_n : A_n. \mathsf{U}_l$ (in which case, one would also provide an instance for the arity).

The property of x guarded from a in t informally means that the recursive call x can be applied to an element of $\mu X.A$ which comes by steps of destruction of a without ever using enter.

Note that the observational rule is one among other variants.

11 Co-recursive types

We give a syntax for co-recursive types, i.e. for types defined as greatest type generated by its constructors.

The property of x guarded in t informally means that the path to the occurrences of x in t never meet an out.

Note that the observational rule is one among other variants.

extended syntax of expressions

$$\begin{array}{lll} t,u,v,A,B,p,q & ::= & \ldots \mid (c_1 : P \oplus \ldots \oplus c_n : P)^w_{\rho:P} \mid \mathsf{case}\ t\ \mathsf{of}\ [c_1\rho \mapsto t|\ldots|c_n\rho \mapsto t] \mid c_i w \\ P & ::= & 1 \mid (a:A) \otimes P \\ w & ::= & () \mid (t,w) \\ \rho & ::= & () \mid (a,\rho) \\ \Gamma & ::= & \ldots \mid \rho : P \end{array}$$

type formers

$$\frac{}{\Gamma \vdash_P 1 : \mathsf{U}_l} \qquad \frac{\Gamma \vdash A : \mathsf{U}_l \qquad \Gamma, a : A \vdash_P P : \mathsf{U}_l}{\Gamma \vdash_P (a:A) \otimes P : \mathsf{U}_l} \qquad \frac{\Gamma \vdash_P P : \mathsf{U}_l \qquad \Gamma, \rho : P \vdash_P P_i : \mathsf{U}_l \quad (1 \le i \le n) \qquad \Gamma \vdash w : P \qquad \text{names } d_i \text{ disjoint}}{\Gamma \vdash (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho:P} : \mathsf{U}_l}$$

typing rules for instances

$$\frac{}{\Gamma \vdash_p () : 1} \qquad \frac{\Gamma \vdash t : A \qquad \Gamma \vdash_p w : P[t/a]}{\Gamma \vdash_p (t,w) : (a:A) \otimes P}$$

typing rules for patterns

$$\frac{}{\Gamma \vdash_{pat} () : 1} \qquad \frac{\Gamma \vdash A : \mathsf{U}_l \qquad \Gamma, a : A \vdash_{pat} \rho : P}{\Gamma \vdash_{pat} (a,\rho) : (a:A) \otimes P} \qquad \frac{\Gamma\ \mathsf{wf} \qquad \Gamma \vdash_{pat} \rho : P}{\Gamma, \rho : P\ \mathsf{wf}}$$

introduction rules

$$\frac{\Gamma \vdash w' : P \qquad \Gamma, \rho : P \vdash P_j : \mathsf{U}_l \quad (1 \le j \le n) \qquad \Gamma \vdash w : P_i[w'/\rho]}{\Gamma \vdash c_i w : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^{w'}_{\rho:P}}\ (1 \le i \le n)$$

elimination rule

$$\frac{\Gamma \vdash v : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho:P} \qquad \Gamma, a : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho:P} \vdash B : \mathsf{U}_l \qquad \Gamma, \rho : P_i[w/\rho] \vdash t_i : B[c_i\,\rho/a] \quad (1 \le i \le n)}{\Gamma \vdash \mathsf{case}\ v\ \mathsf{of}\ [c_1\rho \mapsto t_1|\ldots|c_n\rho \mapsto t_n] : B[v/a]}$$

reduction rules

$$\frac{\begin{array}{c}\Gamma \vdash_P P : \mathsf{U}_l \qquad \Gamma, \rho' : P \vdash_P P_i : \mathsf{U}_l \quad (1 \le i \le n) \qquad \Gamma \vdash w' : P \qquad d_i \text{ disjoint} \\ \Gamma, a : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^{w'}_{\rho:P'} \vdash B : \mathsf{U}_l \qquad \Gamma, \rho : P_i[w'/\rho'] \vdash t_i : B[c_i\,\rho/a] \quad (1 \le i \le n) \qquad \Gamma \vdash_p w : P_i[w'/\rho']\end{array}}{\Gamma \vdash \mathsf{case}\ c_i\,w\ \mathsf{of}\ [c_1\rho \mapsto t_1|\ldots|c_n\rho \mapsto t_n] \ \triangleright\ t_i[w/\rho] : B[c_i\,w/a]}\ \beta^i_{pos}$$

observational rule

$$\frac{\Gamma \vdash v : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho:P} \qquad \Gamma, a : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho:P} \vdash E[a] : A}{\Gamma \vdash \mathsf{case}\ v\ \mathsf{of}\ [c_1\rho \mapsto E[c_1\,\rho]|\ldots|c_n\rho \mapsto E[c_n\,\rho]] \ \triangleright\ E[v] : B[v/a]}\ \eta_{pos}$$

where $E[a]$ is made only from elimination rules applied to a

congruence rules

$$\frac{\Gamma \vdash P \equiv P' : \mathsf{U}_l \qquad \Gamma, \rho : P \vdash P_i \equiv P_i' : \mathsf{U}_l \quad (1 \le i \le n) \qquad \Gamma \vdash w \equiv w' : P}{\Gamma \vdash (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho:P} \equiv (c_1 : P_1' \oplus \ldots \oplus c_n : P_n')^{w'}_{\rho:P'} : \mathsf{U}_l}$$

$$\frac{\Gamma \vdash P : \mathsf{U}_l \qquad \Gamma, \rho : P \vdash P_i : \mathsf{U}_l \quad (1 \le i \le n) \qquad \Gamma \vdash w'' : P \qquad \Gamma \vdash w \equiv w' : P_i[w''/\rho]}{\Gamma \vdash c_i\,w \equiv c_i\,w' : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^{w''}_{\rho:P}}$$

$$\frac{\Gamma \vdash v \equiv v' : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho':P} \qquad \Gamma, a : (c_1 : P_1 \oplus \ldots \oplus c_n : P_n)^w_{\rho':P} \vdash B : \mathsf{U}_l \qquad \Gamma, \rho : P_i[w/\rho'] \vdash t_i \equiv t_i' : B[c_i\,\rho/a] \quad (1 \le i \le n)}{\Gamma \vdash \mathsf{case}\ v\ \mathsf{of}\ [c_1\rho \mapsto t_1|\ldots|c_n\rho \mapsto t_n] \equiv \mathsf{case}\ v'\ \mathsf{of}\ [c_1\rho \mapsto t_1'|\ldots|c_n\rho \mapsto t_n'] : B[v/a]}$$

$+$ judgemental equality and congruence rules for $\Gamma \vdash w : P$

Figure 8: General typing and computational rules for positive types

extended syntax of expressions

$$\begin{array}{lll} t, u, v, A, B, p, q & ::= & \dots \mid \{d_1 : N \,\&\, ... \,\&\, d_n : N\}^w_{\rho:P} \mid \{d_1 \mapsto t; ...; d_n \mapsto t\}^v_c \mid d_i\, t \\ N & ::= & A \mid \Pi a\!:\!A.\, N \qquad \text{(i.e. a distinguished subset of the grammar of terms)} \end{array}$$

type formers

$$\frac{\Gamma \vdash_p w : P \qquad \Gamma, \rho : P \vdash N_i : \mathsf{U}_l \quad (1 \leq i \leq n) \qquad \text{names } d_i \text{ disjoint}}{\Gamma \vdash \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}}$$

introduction rule

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma \vdash_p w : P \qquad \Gamma, \rho : P \vdash N_i : \mathsf{U}_l \quad (1 \leq i \leq n) \qquad \Gamma, c : C \vdash t_i : N_i[w/\rho] \quad (1 \leq i \leq n) \qquad \Gamma \vdash v : C}{\Gamma \vdash \{d_1 \mapsto t_1; ...; d_n \mapsto t_n\}^v_c : \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}}$$

elimination rules

$$\frac{\Gamma \vdash t : \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}}{\Gamma \vdash d_i\, t : N_i[w/\rho]} \; (1 \leq i \leq n)$$

reduction rules

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma \vdash_p w : P \qquad \Gamma, \rho : P \vdash N_i : \mathsf{U}_l \quad (1 \leq i \leq n) \qquad \Gamma, c : C \vdash t_i : N_i[w/\rho] \quad (1 \leq i \leq n) \qquad \Gamma \vdash v : C}{\Gamma \vdash d_i\, \{d_1 \mapsto t_1; ...; d_n \mapsto t_n\}^v_c \;\triangleright\; t_i[v/c] : N_i[w/\rho]} \; \beta^i_{neg}$$

observational rule

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, c : C \vdash t : \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}}{\Gamma \vdash \{d_1 \mapsto d_i\, t; ...; d_n \mapsto d_n\, t\}^c_c \;\triangleright\; t : \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}} \; \eta_{neg}$$

congruence rules

$$\frac{\Gamma \vdash_p w \equiv w' : P \qquad \Gamma, \rho : P \vdash N_i \equiv N'_i : \mathsf{U}_l \quad (1 \leq i \leq n) \qquad \Gamma \vdash P \equiv P' : \mathsf{U}_l}{\Gamma \vdash \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P} \equiv \{d_1 : N'_1 \,\&\, ... \,\&\, d_n : N'_n\}^{P'}_{w'::}\mathsf{U}_l}$$

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma \vdash_p w : P \qquad \Gamma, \rho : P \vdash N_i : \mathsf{U}_l \quad (1 \leq i \leq n) \qquad \Gamma, c : C \vdash t_i \equiv t'_i : N_i[w/\rho] \quad (1 \leq i \leq n) \qquad \Gamma \vdash v \equiv v' : C}{\Gamma \vdash \{d_1 \mapsto t_1; ...; d_n \mapsto t_n\}^v_c \equiv \{d_1 \mapsto t'_1; ...; d_n \mapsto t'_n\}^{v'}_c : \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}}$$

$$\frac{\Gamma \vdash t \equiv t' : \{d_1 : N_1 \,\&\, ... \,\&\, d_n : N_n\}^w_{\rho:P}}{\Gamma \vdash d_i\, t \equiv d_i\, t' : N_i[w/\rho]}$$

Figure 9: General typing and computational rules for negative types

extended syntax of expressions

$$
\begin{array}{lcl}
t, u, v, A, B, p, q & ::= & \dots \mid \mu X.A \mid \mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; t]\, \mathsf{in}\, f\, t \\
P & ::= & \dots \mid (a : X) \otimes P \\
N & ::= & \dots \mid X
\end{array}
$$

type former

$$
\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad A \text{ is a } (\ldots \oplus \ldots) \text{ or } \{\ldots \,\&\, \ldots\} \text{ type}}{\Gamma \vdash \mu X.A : \mathsf{U}_l}
$$

introduction rule

$$
\frac{\Gamma \vdash t : A[\mu X.A/X]}{\Gamma \vdash \mathsf{enter}\, t : \mu X.A}
$$

elimination rule

$$
\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad \Gamma, b : \mu X.A \vdash P : \mathsf{U}_l \qquad \Gamma \vdash v : \mu X.A \qquad \Gamma, f : \Pi b {:} \mu X.A.\, P, a : A[\mu X.A/X] \vdash t : P[\mathsf{enter}\, a/b] \qquad x \text{ guarded from } a \text{ in } t}{\Gamma \vdash \mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; t]\, \mathsf{in}\, f\, v : P[v/b]}
$$

reduction rule

$$
\frac{: \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad \Gamma, b : \mu X.A \vdash P : \mathsf{U}_l \qquad \Gamma \vdash v : A[\mu X.A/X] \qquad \Gamma, f : \Pi b {:} \mu X.A.\, P, a : A[\mu X.A/X] \vdash t : P[\mathsf{enter}\, a/b] \qquad x \text{ guarded from } a \text{ in}}{\Gamma \vdash \mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; t]\, \mathsf{in}\, f\, (\mathsf{enter}\, v) \;\triangleright\; t[v/a][\mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; t]\, \mathsf{in}\, f\, a/f a] : P[\mathsf{enter}\, v/b]}
$$

observational rule

$$
\frac{\Gamma \vdash t : \mu X.A \qquad \Gamma, a : \mu X.A \vdash E[a] : B}{\Gamma \vdash \mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; E[\mathsf{enter}\, a]]\, \mathsf{in}\, f\, t \;\triangleright\; E[t] : B[t/a]}
$$

congruence rules

$$
\frac{\Gamma, X : \mathsf{U}_l \vdash A \equiv A' : \mathsf{U}_l}{\Gamma \vdash \mu X.A \equiv \mu X.A' : \mathsf{U}_l}
$$

$$
\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad \Gamma, a : \mu X.A \vdash P : \mathsf{U}_l \qquad \Gamma \vdash v : \mu X.A \qquad \Gamma, f : \Pi a {:} \mu X.A.\, P, a : A[\mu X.A/X] \vdash t : P \qquad x \text{ guarded from } a \text{ in } t}{\Gamma \vdash \mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; t]\, \mathsf{in}\, f\, v \equiv \mathsf{fix}\, f\, [\mathsf{enter}\, a \;\mapsto\; t']\, \mathsf{in}\, f\, v' : P[v/a]}
$$

Figure 10: General typing and computational rules for recursive types

extended syntax of expressions

$$t, u, v, A, B, p, q \quad ::= \quad \dots \mid \nu X.A \mid \mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto t\}\, \mathsf{in}\, f\, v$$

type formers

$$\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad A \text{ is a } (\dots \oplus \dots) \text{ or } \{\dots \,\&\, \dots\} \text{ type}}{\Gamma \vdash \nu X.A : \mathsf{U}_l}$$

introduction rule

$$\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad \Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, f : C \to \nu X.A, c : C \vdash t : A[\nu X.A/X] \qquad \Gamma \vdash v : C \qquad x \text{ guarded in } t}{\Gamma \vdash \mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto t\}\, \mathsf{in}\, f\, v : \nu X.A}$$

elimination rule

$$\frac{\Gamma \vdash t : \nu X.A}{\Gamma \vdash \mathsf{out}\, t : A[\nu X.A/X]}$$

reduction rule

$$\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad \Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, f : C \to \nu X.A, c : C \vdash t : A[\nu X.A/X] \qquad \Gamma \vdash v : C \qquad x \text{ guarded in } t}{\Gamma \vdash \mathsf{out}\,(\mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto t\}\, \mathsf{in}\, f\, v) \;\rhd\; t[v/c][\mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto t\}\, \mathsf{in}\, f\, c/f\, c] : A[\nu X.A/X]}\ \beta_\nu$$

observational rule

$$\frac{\Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, c : C \vdash t : \nu X.A \qquad \Gamma \vdash v : C}{\Gamma \vdash \mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto \mathsf{out}\, t\}\, \mathsf{in}\, f\, v \;\rhd\; t[v/c] : \nu X.A}\ \eta_{neg}$$

congruence rules

$$\frac{\Gamma, X : \mathsf{U}_l \vdash A \equiv A' : \mathsf{U}_l}{\Gamma \vdash \nu X.A \equiv \nu X.A' : \mathsf{U}_l}$$

$$\frac{\Gamma, X : \mathsf{U}_l \vdash A : \mathsf{U}_l \qquad \Gamma \vdash C : \mathsf{U}_l \qquad \Gamma, f : C \to \nu X.A, c : C \vdash t \equiv t' : A[\nu X.A/X] \qquad \Gamma \vdash v \equiv v' : C \qquad x \text{ guarded in } t}{\Gamma \vdash \mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto t\}\, \mathsf{in}\, f\, v \equiv \mathsf{cofix}\, f\, c \;=\; \{\mathsf{out} \mapsto t'\}\, \mathsf{in}\, f\, v' : \nu X.A}$$

Figure 11: General typing and computational rules for co-recursive types