an Intensional Dependent Type Theory with Type-in-Type and Recursion

February 23, 2021

Surface Language

l			position identifier
Γ	::=	$\lozenge \mid \Gamma, x : M$	var contexts
m, n, h, M, N, H, P	::=	x	expressions: variable
	$\mid$	$m ::_l M$	type annotation
	$\mid$	$\star$	type universe
	$\mid$	$\Pi x : M_l.N_{l'}$	function type
	$\mid$	$\mathsf{fun}\, f.\, x.\, m \mid m\,{}_l n$	function constructor, eliminator
v	::=	x	values
	$\mid$	$\star \mid \Pi x : M.N$	type values
	$\mid$	$\mathsf{fun}\, f.\, x.\, n$	function values

Cast Language

H	::=	$\lozenge \mid H, x : A$	var contexts
a_h, b_h, c_h	::=	x	
	$\mid$	$\star$	
	$\mid$	$\Pi x : A.B$	
	$\mid$	$\mathsf{fun}\, f.\, x.a \mid b\, a$	
e	::=	$A \mid e =_{l,o} A$	type equality chain
a, b, c, A, B, C	::=	$\star \mid \Pi x : A.B$	
	$\mid$	$a_h :: e$	
o	::=	$. \mid o.arg$	observation
	$\mid$	$o.bod[a]$	

Substitution

$$
\begin{array}{rll}
\star\,[x \coloneqq a] & = \star & b\,[x \coloneqq a] \to c \\
(\Pi x : A.B)\,[x \coloneqq a] & = \Pi x : A\,[x \coloneqq a]\,.B\,[x \coloneqq a] & \\
(x :: A =_{l,o} e)\,[x \coloneqq a_h :: e'] & = a_h :: e' =_{l,o} e\,[x \coloneqq a_h :: e'] & \\
(y :: e)\,[x \coloneqq a] & = y :: e\,[x \coloneqq a_h :: e'] & \\
(b_h :: e)\,[x \coloneqq a] & = b_h\,[x \coloneqq a] :: e\,[x \coloneqq a] & \\
\star\,[x \coloneqq a] & = \star & b_h\,[x \coloneqq a] \dashrightarrow c_h \\
(\Pi x : A.B)\,[x \coloneqq a] & = \Pi x : A\,[x \coloneqq a]\,.B\,[x \coloneqq a] & \\
(\mathsf{fun}\,f.\,y.b)\,[x \coloneqq a] & = \mathsf{fun}\,f.\,y.b\,[x \coloneqq a] & \\
(b\,c)\,[x \coloneqq a] & = b\,[x \coloneqq a]\ c\,[x \coloneqq a] & \\
(e =_{l,o} A)\,[x \coloneqq a] & e\,[x \coloneqq a] =_{l,o[x \coloneqq a]} A\,[x \coloneqq a] & e\,[x \coloneqq a] \to e' \\
B\,[x \coloneqq a] & B\,[x \coloneqq a] & \\
.\,[x \coloneqq a] & . & o\,[x \coloneqq a] \to o' \\
(o.arg)\,[x \coloneqq a] & o\,[x \coloneqq a]\,.arg & \\
(o.bod[b])\,[x \coloneqq a] & o\,[x \coloneqq a]\,.bod[b\,[x \coloneqq a]] &
\end{array}
$$

lookup

$$
\begin{array}{rll}
A\uparrow & = A & \text{apparent type} \\
e =_{l,o} A\uparrow & = A & \\
A\downarrow & = A & \text{constructor type} \\
e =_{l,o} A\downarrow & = e\downarrow &
\end{array}
$$

Casts

occasionally we will use the shorthand $A :: e$ to inject additional casts into A,

$$
\begin{array}{rl}
(A :: B) :: (B' =_{l,o} e) & = A :: B =_{l,o} e \\
(A :: e' =_{l,o} B) :: (B' =_{l,o} e) & = A :: e' =_{l,o} e
\end{array}
$$

Judgments

$$
\begin{array}{rl}
\Gamma \sim H \vdash n\,Elab\,a & \text{Infer cast} \\
\Gamma \sim H \vdash n\,Elab_{M,l}\,a & \text{Check cast*} \\
H \vdash a : A & \text{apparent type} \\
H \vdash a_h : A & \text{head type} \\
H \vdash a_h \Rrightarrow a : A & ** \\
H \vdash e \Rrightarrow e' : \overline{\star} & * \\
H \vdash a \Rrightarrow a' : A & * \\
H \vdash o \Rrightarrow o' & ** \\
e \sim e' & \text{same except for observations and evidence} \\
e\,Elim_{\star} & \text{concrete elimination} \\
e\,Elim_{\Pi} x : e_A.e_b &
\end{array}
$$

1 Elaboration

1.1 Infer

$$\frac{x : A \in H}{\Gamma \sim H \vdash x \, Elab \, x :: A}$$

$$\frac{\Gamma \sim H \vdash M \, Elab_{\star,l} \, C \quad \Gamma \sim H \vdash m \, Elab_{C,l} \, a}{\Gamma \sim H \vdash m ::_l M \, Elab \, a}$$

$$\frac{???}{\Gamma \sim H \vdash \star \, Elab \, \star}$$

$$\frac{\Gamma \sim H \vdash M \, Elab_{\star,l} \, A \quad \Gamma, x : M \sim H, x : A \vdash N \, Elab_{\star,l'} \, B}{\Gamma \vdash \Pi x : M_l . N_{l'} \; Elab \; (\Pi x : A.B) :: \star}$$

$$\frac{\Gamma \sim H \vdash m \, Elab \, b :: e \quad (\Pi x : A.B) = e \uparrow \quad \Gamma \sim H \vdash n \, Elab_{A,l} \, a}{\Gamma \sim H \vdash m \;_l n \, Elab \, b \, a}$$

1.2 Check

TODO probably easiest to extend the elaboration checking judgment with the raw terms of the base lang, so everything can move in lock step

$$\frac{???}{\Gamma \sim H \vdash \star \, Elab_{\star,l} \, \star}$$

$$\frac{\Gamma, ??? \sim H, f : \Pi x : A.B, \; x : A \vdash m \, Elab_{B,l??} \, b}{\Gamma \sim H \vdash \mathsf{fun} \, f. \, x. \, m \, Elab_{\Pi x : A.B,l} \, \mathsf{fun} \, f. \, x.b}$$

$$\frac{\Gamma \sim H \vdash m \, Elab \, a_h :: e}{\Gamma \sim H \vdash m \, Elab_{A,l} \, a_h :: e =_{l,.} A}$$

2 Typing

2.1 Cast Typing

$$\frac{H \vdash A : \star}{H \vdash A : \overline{\star}} eq - ty - 1$$

$$\frac{H \vdash e : \overline{\star} \quad H \vdash A : \star}{H \vdash e =_{l,o} A : \overline{\star}} eq - ty - 2$$

2.2 Head Typing

$$\frac{x : A \in H}{H \vdash x : A} var - ty$$

$$\frac{???}{H \vdash \star : \star} \star - ty$$

$$\frac{H \vdash A : \star \quad H, x : A \vdash B : \star}{H \vdash \Pi x : A.B \ : \star} \Pi - ty$$

allow typing to uncast heads

$$\frac{H, f : \Pi x : A.B, x : A \vdash b : B}{H \vdash \mathsf{fun}\, f.\, x.b \ : \ \Pi x : A.B} \Pi - \mathsf{fun} - ty$$

$$\frac{H \vdash b : \Pi x : A.B \quad H \vdash a : A}{H \vdash b\, a \ : \ B\, [x \coloneqq a]} \Pi - app - ty$$

TODO: do I need a head converions rule? should I have a head conversion rule?

2.3 Cast Term Typing

$$\frac{???}{H \vdash \star : \star}$$

$$\frac{H \vdash A : \star \quad H, x : A \vdash B : \star}{H \vdash \Pi x : A.B \ : \star}$$

$$\frac{H \vdash a : A \quad H \vdash A \equiv A' : \star}{H \vdash a : A'} conv$$

$$\frac{H \vdash e : \overline{\star} \quad H \vdash a_h : B \downarrow}{H \vdash a_h :: e \quad : \quad e \uparrow} apparent$$

(TODO: may regret these as functions instead of relations)

3 Definitional Equality

$$\frac{H \vdash a \Rrightarrow_* b : A \quad H \vdash a' \Rrightarrow_* b' \ : A \quad b \sim b'}{H \vdash a \equiv a' : A}$$

4 Consistent

A relation that equates terms except for source and observation information

$$\frac{}{\star \sim \star}$$

$$\frac{A \sim A' \quad B \sim B'}{\Pi x : A.B \sim \Pi x : A'.B'}$$

$$\frac{a_h \sim a'_h \quad e \sim e'}{a_h :: e \sim a'_h :: e'}$$

$$\frac{e \sim e' \quad A \sim A'}{e =_{l,o} A \sim e' =_{l',o'} A'}$$

$$\frac{a \sim a'}{\mathsf{fun}\, f.\, x.a \sim \mathsf{fun}\, f.\, x.a'}$$

$$\frac{b \sim b' \quad a \sim a'}{b\, a \sim b'\, a'}$$

5 Parallel Reductions

$$\frac{H \vdash a : A}{H \vdash a \Rrightarrow_* a : A}$$

$$\frac{H \vdash a \Rrightarrow_* b : A \quad H \vdash b \Rrightarrow c : A}{H \vdash a \Rrightarrow_* c : A}$$

6 Parallel Reduction

6.1 Cast Par reduction

$$\frac{H \vdash A \Rrightarrow A' : \star}{H \vdash A \Rrightarrow A' : \overline{\star}}$$

$$\frac{H \vdash e \Rrightarrow e' : \overline{\star} \quad H \vdash A \Rrightarrow A' : \star \quad H \vdash o \Rrightarrow o'}{H \vdash e =_{l,o} A \Rrightarrow e' =_{l,o'} A' : \overline{\star}}$$

annoyingly need to support observation reductions, to allow a direct substitution lemma to simplify the proof

6.2 Head Par reduction

$$\frac{H, f : \Pi x : A.B, x : A \vdash b \Rrightarrow b' : B \quad H \vdash a \Rrightarrow a' : A \quad e\, Elim_{\Pi}\, x : e_A.e_B \quad H \vdash e_B \Rrightarrow e'_B : \overline{\star}}{H \vdash (\mathsf{fun}\, f.\, x.b) :: e\, a \Rrightarrow (b' [f \coloneqq (\mathsf{fun}\, f.\, x.b')\,, x \coloneqq a' :: e_A] :: e'_B\, [x \coloneqq a'])\ :\ B\, [x \coloneqq a]}$$

$$\frac{x : A \in H}{H \vdash x \Rrightarrow x : A}$$

$$\frac{H \vdash}{H \vdash \star \Rrightarrow \star : \star}$$

$$\frac{H \vdash A \Rrightarrow A' : \star \quad H, x : A \vdash B \Rrightarrow B' : \star}{H \vdash \Pi x : A.B \Rrightarrow \Pi x : A'.B'\ :\ \star}$$

$$\frac{H, f : \Pi x : A.B, x : A \vdash b \Rrightarrow b'\ : B}{H \vdash \mathsf{fun}\, f.\, x.b \Rrightarrow \mathsf{fun}\, f.\, x.b'\ :\ \Pi x : A.B}$$

$$\frac{H \vdash b \Rrightarrow b' : \Pi x : A.B \quad H \vdash a \Rrightarrow b' : A}{H \vdash b\, a \Rrightarrow b'\, a'\ :\ B\, [x \coloneqq a]}$$

6.3 Cast Term Par reduction

$$\frac{H \vdash}{H \vdash \star \Rrightarrow \star : \star}$$

$$\frac{H \vdash A \Rrightarrow A' : \star \quad H, x : A \vdash B \Rrightarrow B' : \star}{H \vdash \Pi x : A.B \Rrightarrow \Pi x : A'.B'\ :\ \star}$$

$$\frac{H \vdash e \Rrightarrow e' : \star \quad H \vdash a \Rrightarrow a' : e \downarrow}{H \vdash a :: e \Rrightarrow a' :: e'\ :\ e \uparrow}$$

6.4 Observation Par reduction

$$\frac{H \vdash}{H \vdash .\, \Rrightarrow .}$$

$$\frac{H \vdash A \Rrightarrow A' : \star \quad H, x : A \vdash B \Rrightarrow B' : \star}{H \vdash \Pi x : A.B \Rrightarrow \Pi x : A'.B'\ :\ \star}$$

$$\frac{H \vdash e \Rrightarrow e' : \star \quad H \vdash a \Rrightarrow a' : e \downarrow}{H \vdash a :: e \Rrightarrow a' :: e'\ :\ e \uparrow}$$

7 Dynamic Check

$$\frac{???}{\star\, Elim_\star}$$

$$\frac{???}{\star :: \star\, Elim_\star}$$

$$\frac{e\, Elim_\star \quad A\, Elim_\star}{e =_{l,o} A\, Elim_\star}$$

$$\frac{???}{\Pi x : A.B\, Elim_\Pi\, x : A.B}$$

$$\frac{e\, Elim_\star}{\Pi x : A.B :: e\, Elim_\Pi\, x : A.B}$$

$$\frac{e\, Elim_\Pi\, x : e_A.e_B}{\Pi x : A.B =_{l,o} e\, Elim_\Pi\, x : A =_{l,o.arg} e_A.e_B\, [x \coloneqq x :: A =_{l,o.arg} A'] =_{l,o.bod[x]} B}$$

$$\frac{e\, Elim_\Pi\, x : e_A.e_B \quad e''\, Elim_\star}{(\Pi x : A.B :: e'') =_{l,o} e\, Elim_\Pi\, x : A =_{l,o.arg} e_A.e_B\, [x \coloneqq x :: A =_{l,o.arg} A'] =_{l,o.bod[x]} B}$$

8 Call-by-Value Small Step

$$\begin{array}{llll}
v & ::= & & \star \mid \Pi x : A.B \\
 & & & v_h :: v_{eq} \\
v_h & ::= & \mid & x \\
 & & \mid & \star \\
 & & \mid & \Pi x : A.B \\
 & & \mid & \mathsf{fun}\, f.\, x.a \\
v_{eq} & ::= & & v \\
 & & \mid & v_{eq} =_{l,o} v \\
v_{obs} & ::= & & . \\
 & & \mid & v_{obs}.arg \\
 & & \mid & v_{obs}.bod[v]
\end{array}$$

$$\frac{A \rightsquigarrow A'}{v_{obs}.bod[A] \rightsquigarrow v_{obs}.bod[A']}$$

$$\frac{O \rightsquigarrow O'}{v_{eq} =_{l,O} A \rightsquigarrow v_{eq} =_{l,O'} A}$$

$$\frac{A \rightsquigarrow A'}{v_{eq} =_{l,v_{obs}} A \rightsquigarrow v_{eq} =_{l,v_{obs}} A'}$$

$$\frac{e \rightsquigarrow e'}{e =_{l,o} A \rightsquigarrow e' =_{l,o} A}$$

$$\frac{e \rightsquigarrow e'}{a_h :: e \rightsquigarrow a_h :: e'}$$

$$\frac{a_h \rightsquigarrow a'_h}{a_h :: v_{eq} \rightsquigarrow a_h :: v_{eq}}$$

$$\frac{b \rightsquigarrow b'}{b\, a \rightsquigarrow b'\, a}$$

$$\frac{a \rightsquigarrow a'}{v\, a \rightsquigarrow v\, a'}$$

$$\frac{v_{eq}\, Elim_{\Pi}\, x : e_A . e_B}{(\mathsf{fun}\, f.\, x.b) :: v_{eq}\, v :: v'_{eq} \rightsquigarrow (b\, [f := (\mathsf{fun}\, f.\, x.b)\, , x := v :: e_A] :: e'_B\, [x := v])}$$

(this substitutes non-value casts into values, which is a little awkward but doesn't break anything)

Alt rules/ notation

$$\frac{\forall C \in B =_{l,o} e =_{l',o'} A,\ H \vdash C : \star \quad H \vdash a_h : B}{H \vdash a_h :: B =_{l,o} e =_{l',o'} A\ : A}\, apparent$$

$$\frac{v_{eq} Elim_{\Pi}\, v_{eq\,A}, e_B \quad v_{eq\,A}\, Elim_{\star} \quad v_{eq\,A}\, Elim_{\star}}{((\mathsf{fun}\, f.\, x.b) :: v_{eq}\, v) :: v'_{eq} \rightsquigarrow (\mathsf{fun}\, f.\, x.b) :: v_{eq}\, v}$$

.....

$$\frac{v_{eq} Elim_{\Pi}\, v_{eq\,A}, e_B \quad v_{eq\,A}\, Elim_{\star} \quad v_{eq\,A}\, Elim_{\star}}{(\mathsf{fun}\, f.\, x.b) :: v_{eq}\, v \rightsquigarrow b\, [f := (\mathsf{fun}\, f.\, x.b)\, , x := v]}$$

...

$$\frac{v_{eq}\, Elim_{\Pi}\, x, -, v_{eq\,A}, e_B \quad v_{eq\,A}\, Elim_{\star}}{((\mathsf{fun}\, f.\, x.b) :: v_{eq}\, v) :: v'_{eq} \rightsquigarrow (b :: e_B)\, [f := (\mathsf{fun}\, f.\, x.b) :: v_{eq}, x := v] :: v'_{eq}}$$

$$\frac{H, f : \Pi x : A.B, x : A \vdash b \Rrightarrow b'\ : B}{H \vdash ((\mathsf{fun}\, f.\, x.b) :: e)\ a \Rrightarrow ((\mathsf{fun}\, f.\, x.b) :: e)\ a\ :\ B\, [x := a]}$$

$$\underset{\rightarrow}{A} \quad = A$$

$$\underset{\longrightarrow}{e =_{l,o} A} \quad = A$$

a

$$\frac{\Gamma \sim H \vdash m\, Elab\, b :: e \quad (\Pi x : A.B) :: e' \in e \quad \Gamma \sim H \vdash n\, Elab_{A,l}\, a}{\Gamma \sim H \vdash m\, {}_l n\, Elab\, b\, a}$$

$$\frac{???}{H \vdash \star\, Elim_\star}$$

$$\frac{???}{H \vdash \star :: \star\, Elim_\star}$$

$$\frac{H \vdash e\, Elim_\star \quad H \vdash A\, Elim_\star}{H \vdash e =_{l,o} A\, Elim_\star}$$

$$\frac{???}{H \vdash \Pi x : A.B\, Elim_\Pi\, x : A.B}$$

$$\frac{H \vdash e\, Elim_\star}{H \vdash \Pi x : A.B :: e\, Elim_\Pi\, x : A.B}$$

$$\frac{H \vdash e\, Elim_\Pi\, x : A'.e'}{H \vdash \Pi x : A.B =_{l,o} e\, Elim_\Pi\, x : A.e'\, [x \coloneqq x :: A =_{l,o.arg} A'] =_{l,o.bod} B}$$

$$\frac{H \vdash e\, Elim_\Pi\, x : A'.e' \quad H \vdash e''\, Elim_\star}{H \vdash (\Pi x : A.B :: e'') =_{l,o} e\, Elim_\Pi\, x : A.e'\, [x \coloneqq x :: A =_{l,o.arg} A'] =_{l,o.bod} B}$$

TODO this is too strict a notion of equality! Evidences should be irrelevant!

$$\frac{H \vdash a \Rrightarrow_* b_h :: e\ : A \quad H \vdash a' \Rrightarrow_* b_h :: e'\ : A}{H \vdash a \equiv a' : A}$$

not enough since it may share an internal cast

$$\frac{H \vdash |a| \Rrightarrow_* |b|\ : A \quad H \vdash |a'| \Rrightarrow_* |b|\ : A}{H \vdash a \equiv a' : A}$$

Erased Language

$$\begin{array}{lcll} r, r', R, R' & ::= & & \text{x} \\ & & | & \star \\ & & | & \Pi x : R.R' \\ & & | & \textsf{fun}\, f.\, x.r \mid r\, r' \end{array}$$

Erasure

$$
\begin{array}{rll}
|\star| & = \star & |a| \to r \\
|\Pi x : A.B| & = \Pi x : |A|.|B| & \\
|A :: e| & = |A| & \\
|x| & = x & |a_h| \to r \\
|\star| & = \star & \\
|\Pi x : A.B| & = \Pi x : |A|.|B| & \\
|\mathsf{fun}\, f.\, x.b| & = \mathsf{fun}\, f.\, x.|b| & \\
|b\, c| & = |b|\, |c| & \\
|x| & = x & |m| \to r \\
|m ::_l M| & = |m| & \\
|\star| & = \star & \\
|\Pi x : M_l.N_{l'}| & = \Pi x : |M|.|N| & \\
|\mathsf{fun}\, f.\, x.m| & = \mathsf{fun}\, f.\, x.|m| & \\
|m\, {}_l n| & = |m|\, |n| &
\end{array}
$$

$$
\frac{|a| \Rrightarrow_* r \quad |a'| \Rrightarrow_* r \quad H \vdash a : A \quad H \vdash a' : A}{H \vdash a \equiv a' : A}
$$