Noninvasive Polarized Subtyping for Inductive Types

Théo Laurent Kenji Maillard

Inria Paris, Prosecco Team

Inria Rennes-Bretagne Atlantique, Gallinette Team

TYPES 2022

Subtyping

$A_0 <: A_1$ (kind of) means $a : A_0 \implies a : A_1$

For example

- $\{\, \texttt{foo} : \texttt{bool};\ \texttt{bar} : \texttt{nat} \,\} <: \{\, \texttt{foo} : \texttt{bool} \,\}$
- $\{\, n : \texttt{nat} \mid 2 \leq n \,\} <: \texttt{nat}$

Subtyping

$A_0 <: A_1$ (kind of) means $a : A_0 \implies a : A_1$

For example

- $\{\, \texttt{foo} : \texttt{bool};\ \texttt{bar} : \texttt{nat} \,\} <: \{\, \texttt{foo} : \texttt{bool} \,\}$
- $\{\, n : \texttt{nat} \mid 2 \leq n \,\} <: \texttt{nat}$

Structural Subtyping

$\texttt{List}\ A_0 <: \texttt{List}\ A_1$ when $A_0 <: A_1$

$A_0 \to B_0 <: A_1 \to B_1$ when $A_0 :> A_1$ and $B_0 <: B_1$

Subtyping

$A_0 <: A_1$ (kind of) means $a : A_0 \implies a : A_1$

For example

- $\{\, \texttt{foo} : \texttt{bool};\ \texttt{bar} : \texttt{nat} \,\} <: \{\, \texttt{foo} : \texttt{bool} \,\}$
- $\{\, n : \texttt{nat} \mid 2 \leq n \,\} <: \texttt{nat}$

Structural Subtyping

$\texttt{List}\ A_0 <: \texttt{List}\ A_1$ when $A_0 <: A_1$

$A_0 \to B_0 <: A_1 \to B_1$ when $A_0 :> A_1$ and $B_0 <: B_1$

Polarities

$\texttt{List}\ {}^{+}\_ \qquad {}^{-}\_ \to {}^{+}\_$

Proof Assistants and Dependent Types

$\lambda(x : A).t : \Pi(x : A).B$	dependent functions
$A : \texttt{Type}$	type of types (universe)
$A \equiv B : \texttt{Type}$	computations in types

Proof Assistants and Dependent Types

$\lambda(x : A).t : \Pi(x : A).B$	dependent functions
$A : \texttt{Type}$	type of types (universe)
$A \equiv B : \texttt{Type}$	computations in types

Structural subtyping is requested!

e.g. F* github issue from 2014

Proof Assistants and Dependent Types

$\lambda(x : A).t : \Pi(x : A).B$	dependent functions
$A : \texttt{Type}$	type of types (universe)
$A \equiv B : \texttt{Type}$	computations in types

Structural subtyping is requested!

e.g. F* github issue from 2014

Metatheory is important

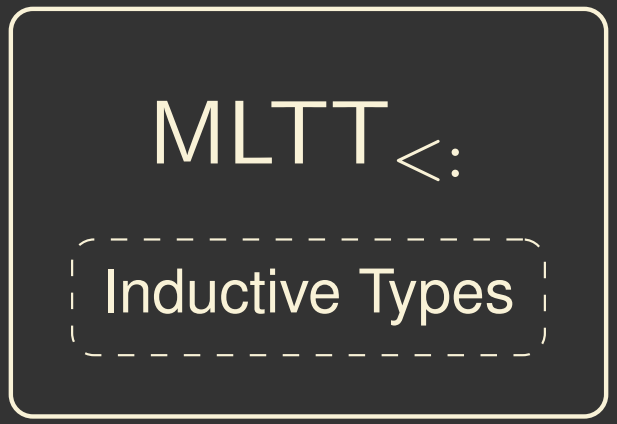
consistency, decidability of typechecking, etc.

Noninvasive Polarities: MLTT$_{<:}$

Directed type theories (Licata and Harper 2011, Nuyts 2015)

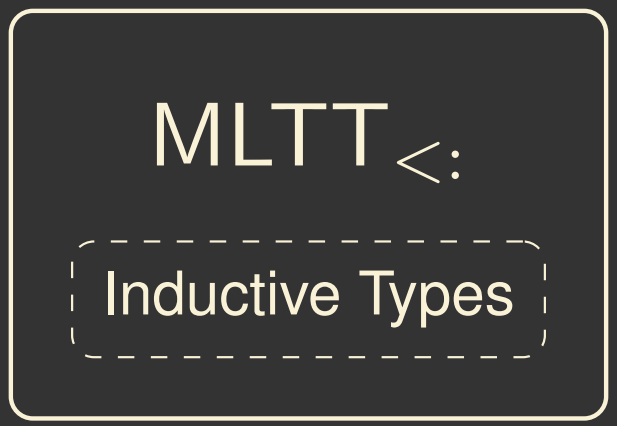
Noninvasive Polarities: $\mathrm{MLTT}_{<:}$

Directed type theories (Licata and Harper 2011, Nuyts 2015)



Noninvasive Polarities: $\text{MLTT}_{<:}$

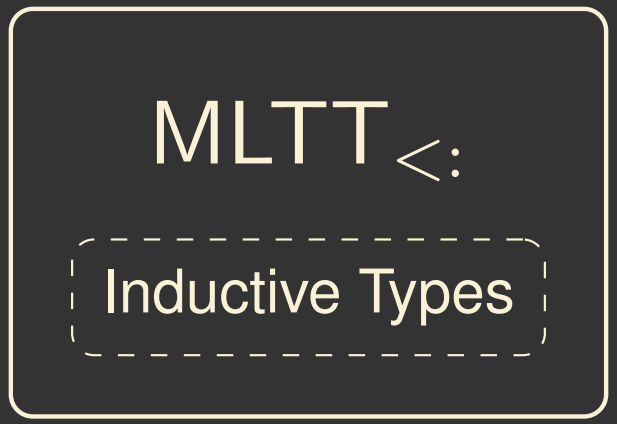
Directed type theories (Licata and Harper 2011, Nuyts 2015)



$\Gamma \vdash A_0 <: A_1$ $\qquad$ $A :_+ \texttt{Type} \vdash \texttt{List}\ A :_+ \texttt{Type}$

Noninvasive Polarities: $\mathsf{MLTT}_{<:}$

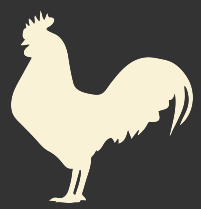
Directed type theories (Licata and Harper 2011, Nuyts 2015)



$$\frac{\Gamma \vdash A_0 <: A_1 \qquad\qquad A :_+ \mathtt{Type} \vdash \mathtt{List}\ A :_+ \mathtt{Type}}{\Gamma \vdash \mathtt{List}\ A_0 <: \mathtt{List}\ A_1}$$

Constructing a Model in the Coq Proof Assistant

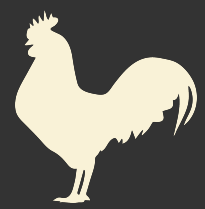
Constructing a Model in the Coq Proof Assistant

$\mathsf{MLTT}_{<:}$ $\xrightarrow{[\![\_]\!]}$ 

$\mathtt{U} : \mathtt{Type} \qquad \mathtt{El} : \mathtt{U} \to \mathtt{Type} \qquad \mathtt{sub} : \mathtt{U} \to \mathtt{U} \to \mathtt{SProp}$

$$\mathtt{coe} : \Pi(A\ B : \mathtt{U}).\ \mathtt{sub}\ A\ B \to \mathtt{El}\ A \to \mathtt{El}\ B$$

Constructing a Model in the Coq Proof Assistant

$\mathsf{MLTT}_{<:}$ $\xrightarrow{[\![\_]\!]}$ 

$$\texttt{U} : \texttt{Type} \qquad \texttt{El} : \texttt{U} \to \texttt{Type} \qquad \texttt{sub} : \texttt{U} \to \texttt{U} \to \texttt{SProp}$$

$$\texttt{coe} : \Pi(A\ B : \texttt{U}).\ \texttt{sub}\ A\ B \to \texttt{El}\ A \to \texttt{El}\ B$$

$$\begin{array}{llll}
\texttt{coe} & \texttt{codeNat} & \texttt{codeNat} & _ \quad := \texttt{id} \\
\texttt{coe} & (\texttt{codePi}\ A\ B) & \texttt{codeNat} & \texttt{wit} := (* \ \texttt{exfalso}\ \texttt{wit}\ *) \\
\texttt{coe} & (\texttt{codePi}\ A\ B) & (\texttt{codePi}\ A\ B) & \texttt{wit} := \ldots
\end{array}$$

Coercions for Inductive Types

$$[\![\, A :_+ \texttt{Type} \vdash \texttt{List}\ A :_+ \texttt{Type} \,]\!]$$

Coercions for Inductive Types

$$[\![\ A :_+ \texttt{Type} \vdash \texttt{List}\ A :_+ \texttt{Type}\ ]\!]$$

$$\texttt{List}\ (A : \texttt{Type}) : \texttt{Type}$$

$$\texttt{List}_\varepsilon\ (A_0 : \texttt{Type})(A_1 : \texttt{Type})(A_\varepsilon : \texttt{Rel}\ A_0\ A_1)$$
$$: \texttt{Rel}\ (\texttt{List}\ A_0)\ (\texttt{List}\ A_1)$$

Coercions for Inductive Types

$$[\![\ A :_+ \texttt{Type} \vdash \texttt{List}\ A :_+ \texttt{Type}\ ]\!]$$

$$\begin{aligned}
&\texttt{List}\ (A : \texttt{Type}) : \texttt{Type} \\
&\texttt{List}_\varepsilon\ (A_0 : \texttt{Type})(A_1 : \texttt{Type})(A_\varepsilon : \texttt{Rel}\ A_0\ A_1) \\
&\qquad : \texttt{Rel}\ (\texttt{List}\ A_0)\ (\texttt{List}\ A_1)
\end{aligned}$$

$$\begin{aligned}
&\Pi(A_0 : \texttt{Type})(A_1 : \texttt{Type})(A_\varepsilon : \texttt{Rel}\ A_0\ A_1). \\
&\quad \texttt{IsFun}\ A_\varepsilon \to \texttt{IsFun}\ (\texttt{List}_\varepsilon\ A_0\ A_1\ A_\varepsilon)
\end{aligned}$$

Takeaway

We present MLTT$_{<:}$:

- a dependent type theory with structural subtyping
- polarities restricted to inductive types
- aiming at extending established implementations

Takeaway

We present MLTT$_{<:}$

- a dependent type theory with structural subtyping
- polarities restricted to inductive types
- aiming at extending established implementations

Future work

- mechanizing metatheory on top of MetaCoq