Circular Proofs for Subtyping and Termination

Rodolphe Lepigre, Christophe Raffalli

LIPN, séminaire de l'équipe LCR, 22/09/2017

Our goal: a classical, Curry-style proof assistant (PML)

An ML-like language with support for proofs of programs

List of features (not exhaustive):

- Call-by-value, general recursion, polymorphism, effects, Curry style
- First order layer with (untyped) terms as individuals
- Testriction type $A \wedge p$ where p is a “semantic predicate”
- Membership type $t \in A$ for linking the worlds of types and terms
- Inductive and coinductive types, with sizes
- Termination checking (only required for proofs)

There are many different forms of subtyping:

- Subtyping on sums (variants) and products (records)
- Mitchell's subtyping $\forall X.A \Rightarrow B \subseteq (\forall X.A) \Rightarrow (\forall X.B)$
- Restriction type subtyping $A \wedge P \subseteq A$
- Membership type subtyping $t \in A \subseteq A$
- Subtyping on (sized) inductive types $\mu_\tau X.A \subseteq \mu_\kappa X.A$ if $\tau \leqslant \kappa$

Subtyping on inductive type is handled using circular proofs

System F à la Church

$$\frac{}{\Gamma, x : A \vdash x : A} \qquad \frac{\Gamma \vdash t : A \Rightarrow B \quad \Gamma \vdash u : A}{\Gamma \vdash t\ u : B}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x {\color{red}: A}.t : A \Rightarrow B}$$

$$\frac{\Gamma \vdash t : A \quad X \notin \Gamma}{\Gamma \vdash {\color{red}\Lambda X.}t : \forall X.A} \qquad \frac{\Gamma \vdash t : \forall X.A}{\Gamma \vdash t\ {\color{red}B} : A[X := B]}$$

System F à la Curry

$$\frac{}{\Gamma, x : A \vdash x : A}$$

$$\frac{\Gamma \vdash t : A \Rightarrow B \qquad \Gamma \vdash u : A}{\Gamma \vdash t\ u : B}$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \Rightarrow B}$$

$$\frac{\Gamma \vdash t : A \qquad X \notin \Gamma}{\Gamma \vdash t : \forall X.A}$$

$$\frac{\Gamma \vdash t : \forall X.A}{\Gamma \vdash t : A[X := B]}$$

Curry-style, simply-typed λ-calculus (ε version)

Curry-style, simply-typed λ-calculus (ε version)

$$\frac{}{\varepsilon_{x\in A}(t \notin B) : A}$$

$$\frac{t : A \Rightarrow B \qquad u : A}{t\ u : B}$$

$$\frac{t[x := \varepsilon_{x\in A}(t \notin B)] : B}{\lambda x.t : A \Rightarrow B}$$

Curry-style, simply-typed λ-calculus (ε version), with subtyping

$$\frac{\varepsilon_{x\in A}(t \notin B) : A \subseteq C}{\varepsilon_{x\in A}(t \notin B) : C}$$

$$\frac{t : A \Rightarrow B \qquad u : A}{t\ u : B}$$

$$\frac{\lambda x.t : A \Rightarrow B \subseteq C \qquad t[x := \varepsilon_{x\in A}(t \notin B)] : B}{\lambda x.t : C}$$

$$\frac{}{t : A \subseteq A}$$

Curry-style System F (ε version), with subtyping

$$\frac{\varepsilon_{x\in A}(t \notin B) : A \subseteq C}{\varepsilon_{x\in A}(t \notin B) : C} \qquad \frac{t : A \Rightarrow B \quad u : A}{t\ u : B}$$

$$\frac{\lambda x.t : A \Rightarrow B \subseteq C \qquad t[x := \varepsilon_{x\in A}(t \notin B)] : B}{\lambda x.t : C}$$

$$\frac{}{t : A \subseteq A} \qquad \frac{\varepsilon_{x\in C}(t\ x \notin D) : C \subseteq A \qquad t\ \varepsilon_{x\in C}(t\ x \notin D) : B \subseteq D}{t : A \Rightarrow B \subseteq C \Rightarrow D}$$

Curry-style System F (ε version), with subtyping

$$\frac{\varepsilon_{x\in A}(t \notin B) : A \subseteq C}{\varepsilon_{x\in A}(t \notin B) : C} \qquad \frac{t : A \Rightarrow B \quad u : A}{t\ u : B}$$

$$\frac{\lambda x.t : A \Rightarrow B \subseteq C \qquad t[x := \varepsilon_{x\in A}(t \notin B)] : B}{\lambda x.t : C}$$

$$\frac{}{t : A \subseteq A} \qquad \frac{\varepsilon_{x\in C}(t\ x \notin D) : C \subseteq A \qquad t\ \varepsilon_{x\in C}(t\ x \notin D) : B \subseteq D}{t : A \Rightarrow B \subseteq C \Rightarrow D}$$

$$\frac{t : A \subseteq B[X := \varepsilon_X(t \notin B)]}{t : A \subseteq \forall X.B} \qquad \frac{t : A[X := C] \subseteq B}{t : \forall X.A \subseteq B}$$

Can be implemented with standard unification techniques !

$$\frac{\varepsilon_{x\in A}(t \notin B) : A \subseteq C}{\varepsilon_{x\in A}(t \notin B) : C} \qquad \frac{t : U \Rightarrow B \quad u : U}{t\ u : B}$$

$$\frac{\lambda x.t : U \Rightarrow V \subseteq C \quad t[x := \varepsilon_{x\in U}(t \notin V)] : V}{\lambda x.t : C}$$

$$\frac{A = B}{t : A \subseteq B} \qquad \frac{\varepsilon_{x\in C}(t\ x \notin D) : C \subseteq A \quad t\ \varepsilon_{x\in C}(t\ x \notin D) : B \subseteq D}{t : A \Rightarrow B \subseteq C \Rightarrow D}$$

$$\frac{t : A \subseteq B[X := \varepsilon_X(t \notin B)]}{t : A \subseteq \forall X.B} \qquad \frac{t : A[X := U] \subseteq B}{t : \forall X.A \subseteq B}$$

Adding a least fixpoint constructor (inductive types)

Adding a least fixpoint constructor (inductive types)

$$\frac{t : A \subseteq B[X{:=}\mu_\infty X.B]}{t : A \subseteq \mu_\infty X.B} \qquad \frac{t : A \subseteq B[X{:=}\mu_\tau X.B] \qquad \tau < \kappa}{t : A \subseteq \mu_\kappa X.B}$$

$$\frac{t : A[X{:=}\mu_\tau X.A] \subseteq B \qquad \tau = \varepsilon_{\alpha<\kappa}(t \in A[X{:=}\mu_\alpha X.A])}{t : \mu_\kappa X.A \subseteq B}$$

Introducing a cyclic structure (generalisation, induction)

$$\frac{\forall \alpha \ \forall x \ (x : A \subseteq B)}{t : A[\alpha := \kappa] \subseteq B[\alpha := \kappa]}$$

$$\dfrac{\dfrac{\dfrac{[\forall \alpha \ \forall x \ (x : A \subseteq B)]_i}{\vdots}}{A[\alpha := \tau] \subseteq B[\alpha := \tau]} \qquad \tau = \varepsilon_\alpha(A \not\subseteq B)}{\forall \alpha \ \forall x \ (x : A \subseteq B)}\,i$$

Thanks!