Higher-Order Subtyping, Revisited

Syntactic Completeness Proofs for Algorithmic Judgements

Andreas Abel

TYPES Workshop, April 21, 2006

Contents

1 Higher-Order Subtyping

Subtyping for Collections

- When a Float is expected, an Int is acceptable.

$$\mathsf{Int} \leq \mathsf{Float}$$

- Read-only collections: a list of Ints passes for a list of Floats.

$$\frac{\mathsf{Int} \leq \mathsf{Float}}{\mathsf{List\ Int} \leq \mathsf{List\ Float}}$$

- Mutable collections: cannot store a Float into an Int cell.

$$\textit{not}\ \frac{\mathsf{Int} \leq \mathsf{Float}}{\mathsf{Array\ Int} \leq \mathsf{Array\ Float}}$$

Subtyping and Variance

- Distinguish type constructors by their *variance*

$$\begin{array}{lcll} \mathsf{Array} & : & * \xrightarrow{\circ} * & \text{mixed-variant} \\ \mathsf{List} & : & * \xrightarrow{+} * & \text{covariant} \\ \mathsf{Sink} & : & * \xrightarrow{-} * & \text{contravariant} \end{array}$$

- Subtyping applications:

$$\frac{F : * \xrightarrow{\circ} * \qquad A = B}{F\,A \leq F\,B}$$

$$\frac{F : * \xrightarrow{+} * \qquad A \leq B}{F\,A \leq F\,B} \qquad \frac{F : * \xrightarrow{-} * \qquad B \leq A}{F\,A \leq F\,B}$$

Polarized F^ω

- Polarities
$$p ::= \circ \mid + \mid -$$

- Kinds
$$\kappa ::= * \mid \kappa \xrightarrow{p} \kappa'$$

- Type constructors
$$F, G ::= C \mid X \mid \lambda X.F \mid F\,G$$

- Constants C, e.g.,
$$\begin{array}{lcl} \times & : & * \xrightarrow{+} * \xrightarrow{+} * \\ \rightarrow & : & * \xrightarrow{-} * \xrightarrow{+} * \\ \forall_\kappa & : & (\kappa \xrightarrow{\circ} *) \xrightarrow{+} * \end{array}$$

Polarized Kinding

- Polarized contexts
$$\Gamma ::= \diamond \mid \Gamma, X\!:\!p\kappa$$

- Polarized kinding
$$\Gamma \vdash F : \kappa$$

- E.g.,
$$\begin{array}{lcl} F\!:\!\circ(* \xrightarrow{+} *), & & \\ X\!:\!-*, & & \\ Y\!:\!+* & \vdash & F\,X \rightarrow F\,Y : * \end{array}$$

Declarative Equality and Subtyping

- Judgements
$$\begin{array}{ll} \Gamma \vdash F = F' : \kappa & \beta\eta\text{-equality} \\ \Gamma \vdash F \leq F' : \kappa & \text{polarized subtyping} \end{array}$$

- Subtyping axioms, e.g., $\Gamma \vdash \mathsf{Array} \leq \mathsf{List} : * \xrightarrow{+} *$.
- Axioms for β and η.

- Reflexivity, transitivity, (anti)symmetry.
- Closure under abstraction and *application*.

$$\frac{\Gamma \vdash F : \kappa \xrightarrow{+} \kappa' \qquad \Gamma \vdash G \leq G' : \kappa}{\Gamma \vdash F\,G \leq F\,G' : \kappa'} \qquad \frac{\Gamma \vdash F : \kappa \xrightarrow{\circ} \kappa' \qquad \Gamma \vdash G = G' : \kappa}{\Gamma \vdash F\,G = F\,G' : \kappa'}$$

Algorithmic Subtyping

- Judgement for *algorithmic subtyping*

$$\Gamma \vdash F \leq F' \leftleftarrows \kappa$$

- Steps

$$\begin{array}{llll}
\mathsf{Array} \leq (\lambda X.\, \mathsf{List}\, X) & \leftleftarrows & * \xrightarrow{+} * & \text{apply down to kind } *: \\
\mathsf{Array}\, Y \leq (\lambda X.\, \mathsf{List}\, X)\, Y & \leftleftarrows & * & \text{weak head normalize:} \\
\mathsf{Array}\, Y \leq \mathsf{List}\, Y & \leftleftarrows & * & \text{compare heads (axiom):} \\
\quad \mathsf{Array} \leq \mathsf{List} : * \xrightarrow{+} * & & & \text{continue with arguments:} \\
\quad Y \leq Y \leftleftarrows * & & &
\end{array}$$

Kind-directed Algorithmic Subtyping

- Weak head normal forms

$$\begin{array}{lcll}
N & ::= & C \mid X \mid N\,G & \text{neutral (atomic)} \\
W & ::= & N \mid \lambda X F & \text{weak head normal}
\end{array}$$

- Weak head evaluation

$$F \searrow W$$

- Kind-directed algorithmic subtyping

$$\begin{array}{ll}
\Gamma \vdash F \leq F' \leftleftarrows \kappa & \text{checking mode} \\
\Gamma \vdash N \leq N' \rightrightarrows \kappa & \text{inference mode}
\end{array}$$

- (Analogously for algorithmic equality)

Rules for Algorithmic Subtyping

- Checking mode

$$\frac{\Gamma, X\!:\!p\kappa \vdash F\,X \leq F'\,X \leftleftarrows \kappa'}{\Gamma \vdash F \leq F' \leftleftarrows p\kappa \to \kappa'}$$

$$\frac{F \searrow N \qquad F' \searrow N' \qquad \Gamma \vdash N \leq N' \rightrightarrows *}{\Gamma \vdash F \leq F' \leftleftarrows *}$$

- Inference mode: Axioms +

$$\frac{(X : p\kappa) \in \Gamma \qquad p \in \{\circ, +\}}{\Gamma \vdash X \leq X \rightrightarrows \kappa}$$

$$\frac{\Gamma \vdash N \leq N' \rightrightarrows +\kappa \to \kappa' \qquad \Gamma \vdash G \leq G' \leftleftarrows \kappa}{\Gamma \vdash N\, G \leq N'\, G' \rightrightarrows \kappa'}$$

Completeness of Algorithmic Subtyping

- Soundness of algorithmic judgements easy
- Transitivity, (anti)symmetry easy
- Completeness hard: *Closure under application?*
- Alternatives:
  1. From strong normalization (Aspinall Hofmann 2005; Goguen 2005)
  2. Model (e.g., Harper Pfenning 2004)
  3. *Direct, syntactically*

From a Bird's Perspective

- Type language of F^ω is weak (no recursion)
- Roughly simply-typed λ-calculus
- Proof theory says: there is an elementary meta theory
- *How* to construct this elementary proof?
- *Technical skill* required

Main Lemma: Application and Substitution

- Let $\Gamma \vdash G \leq G' \leftleftarrows \kappa$. Prove simultaneously:
  1. If $\Gamma \vdash F \leq F' \leftleftarrows +\kappa \to \kappa'$ then $\Gamma \vdash F\, G \leq F'\, G' \leftleftarrows \kappa'$.
  2. If $\Gamma, X : +\kappa \vdash N \leq N' \rightrightarrows \kappa'$ then
     - either $\Gamma \vdash [G/X]N \leq [G/X]N' \rightrightarrows \kappa'$,
     - or $\Gamma \vdash [G/X]N \leq [G/X]N' \leftleftarrows \kappa'$ and $|\kappa'| \leq |\kappa|$.
  3. If $\Gamma, X : +\kappa \vdash F \leq F' \leftleftarrows \kappa'$ then $\Gamma \vdash [G/X]F \leq [G'/X]F' \leftleftarrows \kappa'$.
- Lexicographic induction on $|\kappa|$ and derivation length.

1. ...
2. Case $N = N' = Y \neq X$. Case $N = N' = X$. Case

$$\frac{\Gamma \vdash M \leq M' \rightrightarrows +\kappa'' \to \kappa' \qquad \Gamma \vdash H \leq H' \leftleftarrows \kappa''}{\Gamma \vdash M\,H \leq M'\,H' \rightrightarrows \kappa'}$$

Consequences and Evaluation

Consequences of Main Lemma:

- Closure under β and application.
- Reflexivity.
- Completeness.

Evaluation of proof:

- Short, direct
- Purely syntactical
- Avoiding logical relations and models
- Well-suited for formalization (e.g., in Twelf)

Applicability of Proof Technique

- Normalization of simply-typed lambda-calculus (Joachimski Matthes 2003)
- Algorithmic equality for LF
- Other logical frameworks (LLF, CLF)
- Predicative polymorphism!?
- Languages of low proof-theoretical complexity
- POPLmark challenges
- Limitations
  - Impredicativity
  - Inductive types

Related Work

- Cut elimination for FOL
- Troelstra 1973: Syntactical normalization proof
- Joachimski Matthes 2003: λ + permutative conversions
- Hereditary substitutions:
  - Watkins Cervesato Pfenning Walker 2003: Concurrent LF
  - Nanevski Pfenning Pientka 2005: Contextual Modal Type Theory
  - Adams (PhD 2005): λ-free LF
- Goguen 1995-2005: Typed Operational Semantics

Conclusions

- Purely syntactical approach to meta theory
- Does not work for CC or inductive types
- But applicable to many logical frameworks
- Proofs suited for formalization (HOAS)
- Should be in your tool box!