Subtyping in Type Theory: Coercion Contexts and Local Coercions

Zhaohui Luo* and Fedor Part

Department of Computer Science
Royal Holloway, University of London
Egham, Surrey TW20 0EX, U.K.

The notion of subtyping is better understood for type assignment systems (those in programming languages) than for type theories with canonical objects (those in proof assistants) and, in this work, we are studying subtyping for the latter. As pointed out in [10, 8] among others, subsumptive subtyping, the traditional notion of subtyping with the subsumption rule, is incompatible with the notion of canonical object for inductive types in the sense that some key properties such as canonicity and subject reduction would fail to hold. It may be argued that coercive subtyping provides a more adequate alternative framework [10].

In this talk, we shall study two related constructs in coercive subtyping: coercion contexts and local coercions. These were introduced, and proved to be useful, in the context of employing type theories in linguistic semantics (see, for example, [9]). A *coercion context* is a context whose entries may be of the form $A <_c B$ as well as the usual form $x : A$:[1]

$$\frac{\Gamma \vdash A : Type \quad \Gamma \vdash B : Type \quad \Gamma \vdash c : (A)B}{\Gamma,\ A <_c B\ \ valid} \qquad \frac{\Gamma,\ A <_c B,\ \Gamma'\ \ valid}{\Gamma,\ A <_c B,\ \Gamma' \vdash A <_c B : Type}$$

A *local coercion* is a subtyping assumption localised in terms (or judgements). For instance:

$$\frac{\Gamma,\ A <_c B \vdash k : K}{\Gamma \vdash (\mathbf{coercion}\ A <_c B\ \mathbf{in}\ k)\ :\ (\mathbf{coercion}\ A <_c B\ \mathbf{in}\ K)}$$

Note that the above constructs are the two sides of the same coin: subtyping relations can be assumed in a coercion context and they can be moved to the right of the $\vdash$-symbol to form terms with local coercions (otherwise, without local coercions, a subtyping entry in a context would block entries to its left from such moves[2]).

A formal treatment of coercion contexts and local coercions involves several technical issues. For example, validity of a coercion context is not enough anymore for it

*This work is partially supported by the grant F/07-537/AJ of the Leverhulme Trust in U.K.

[1] This is similar to coercion declarations in the proof assistants like Coq [3].

[2] In order to move subtyping assumptions to the right, one might consider an alternative approach: employing the so-called *bounded quantification* [1] that extends a higher-order calculus where quantification over (classes of) types are possible. However, the notion of bounded quantification is not well-understood and causes problems such as undecidability.

to be legal: instead, one needs to make sure that the context is *coherent*, guaranteeing the uniqueness of coercions. Also, the key word **coercion** distributes through the components of a judgement. For example, the conclusion judgement of the above rule should be identified with $\Gamma \vdash$ **coercion** $A <_c B$ **in** $(k : K)$. We shall give a formal presentation of coercion contexts and local coercions based on which a comparison to the coercive subtyping extension with global coercions [10] will be made. In particular, we shall prove that such an extension of type theory T with coercion contexts and local coercions is conservative in the sense that, if $\Gamma \vdash J$ is a judgement in T, then if $\Gamma \vdash J$ is derivable in the extension of T, it is derivable in T.

We shall also study the model-theoretic semantics of subtyping. Categorical semantics of dependent type theories have been studied (see, for example, [2, 5, 4] and more recently [7] for univalent foundations) and there has been research on models of subtyping for non-dependent type theories (see, for instance, [6]). We shall study categorical semantics of type theories with canonical objects extended by subtyping and, in particular, coercion contexts and local coercions.

This is work in progress.

References

[1] L. Cardelli and P. Wegner. On understanding types, data abstraction and polymorphism. *Computing Surveys*, 17, 1985.

[2] J. Cartmell. Generalized algebraic theories and contextual category. *Annals of Pure and Applied Logic*, 32, 1986.

[3] Coq Development Team. *The Coq Proof Assistant Reference Manual (Version 8.1), INRIA*, 2007.

[4] M. Hofmann. Syntax and semantics of dependent types. In *Semantics and Logics of Computation*, pages 79–130. Cambridge University Press, 1997.

[5] M. Hyland and A. Pitts. The theory of constructions: categorical semantics and topos-theoretic models. *Categories in Computer Science and Logic, Boulder*, 1987.

[6] B. Jacobs. Subtypes and bounded quatification from a fibred perspective. *Electronic Notes in Theoretical Computer Science*, 1, 1995.

[7] C. Kapulkin, P. L. Lumsdaine, and V. Voevodsky. The simplicial model of univalent foundations. Manuscript, 2012.

[8] Z. Luo. Notes on universes in type theory. Lecture notes for a talk at Institute for Advanced Study, Princeton (URL: `http://www.cs.rhul.ac.uk/home/zhaohui/universes.pdf`), 2012.

[9] Z. Luo. Formal semantics in modern type theories with coercive subtyping. *Linguistics and Philosophy*, 2013. (in press).

[10] Z. Luo, S. Soloviev, and T. Xue. Coercive subtyping: theory and implementation. *Information and Computation*, 223, 2013.