Contextual coercive subtyping

Tao Xue

Department of Computer Science
Royal Holloway, University of London

October 11, 2010

Outline

- Introduction
  - subtyping and coercive subtyping
  - example
- Some type systems with coercive subtyping
- Contextual coercive subtyping
  - the system
  - example
- Coercive subtyping in Coq
- Future work

Subtyping and coercive subtyping

Traditional: Subsumption rule

$$\frac{\Gamma \vdash a : A \quad \Gamma \vdash A \leq B}{\Gamma \vdash a : B}$$

Subtyping and coercive subtyping

Traditional: Subsumption rule

$$\frac{\Gamma \vdash a : A \quad \Gamma \vdash A \leq B}{\Gamma \vdash a : B}$$

Type systems with canonical objects: unique type

Subtyping and coercive subtyping

Traditional: Subsumption rule

$$\frac{\Gamma \vdash a : A \quad \Gamma \vdash A \leq B}{\Gamma \vdash a : B}$$

Type systems with canonical objects: unique type
Coercive subtyping

Subtyping and coercive subtyping

Traditional: Subsumption rule

$$\frac{\Gamma \vdash a : A \quad \Gamma \vdash A \leq B}{\Gamma \vdash a : B}$$

Type systems with canonical objects: unique type
Coercive subtyping
There is a unique coercion from A to B:

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) : C}$$

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

Subtyping and coercive subtyping

Traditional: Subsumption rule

$$\frac{\Gamma \vdash a : A \quad \Gamma \vdash A \leq B}{\Gamma \vdash a : B}$$

Type systems with canonical objects: unique type
Coercive subtyping
There is a unique coercion from A to B:

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) : C}$$

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

A mechanism of abbreviation

Example:

We could use type theory to interpret formal semantics,
e.g. **John runs**

Example:

We could use type theory to interpret formal semantics,
e.g. **John runs** (RO for RunningObject)
$[[run]] : RO \to prop$

Example:

We could use type theory to interpret formal semantics,
e.g. **John runs** (*RO* for RunningObject)
$[[run]] : RO \rightarrow prop$ $[[John]] : Human$

Example:

We could use type theory to interpret formal semantics,
e.g. **John runs** (RO for RunningObject)
$[[run]] : RO \rightarrow prop$ $[[John]] : Human$ $Human <_c RO$

Example:

We could use type theory to interpret formal semantics,
e.g. **John runs** (RO for RunningObject)
$[[run]] : RO \to prop$ $[[John]] : Human$ $Human <_c RO$

So **John runs**
$[[run]]([[John]])$ abbreviation of $[[run]](c[[John]])$

Some type systems with coercive subtyping

T[$\mathcal{R}$] ([Luo99] Zhaohui Luo)

	T	(type system)
+	$\Gamma \vdash A <_c B : \mathbf{Type}$	(subtyping judgements)
+	$\mathcal{R}$	(a set of coherent coercion rules)
+	general subtyping rules	
+	subkinding rules	
+	coercion application rules.	

Some type systems with coercive subtyping

T[$\mathcal{R}$] ([Luo99] Zhaohui Luo)

	T	(type system)
+	$\Gamma \vdash A <_c B :$ **Type**	(subtyping judgements)
+	$\mathcal{R}$	(a set of coherent coercion rules)
+	general subtyping rules	
+	subkinding rules	
+	coercion application rules.	

T[$\mathcal{C}$]([LL05] Yong Luo and Zhaoui Luo)

	T	(type system)
+	$\Gamma \vdash A <_c B :$ **Type**	(subtyping judgements)
+	$\mathcal{C}$	(well-defined coercions)
+	subkinding rules	
+	coercion application rules.	

Some type systems with coercive subtyping

T[$\mathcal{R}$] ([Luo99] Zhaohui Luo)

	T	(type system)
+	$\Gamma \vdash A <_c B :$ **Type**	(subtyping judgements)
+	$\mathcal{R}$	(a set of coherent coercion rules)
+	general subtyping rules	
+	subkinding rules	
+	coercion application rules.	

T[$\mathcal{C}$]([LL05] Yong Luo and Zhaoui Luo)

	T	(type system)
+	$\Gamma \vdash A <_c B :$ **Type**	(subtyping judgements)
+	$\mathcal{C}$	(well-defined coercions)
+	subkinding rules	
+	coercion application rules.	

Well-Defined Coercions: a set of subtyping judgements which satisfies Coherence, Congruence, Transitivity, Substitution, and Weakening.

Coherence

Coherence: If $\Gamma \vdash A <_c B : \textbf{Type}$ and $\Gamma \vdash A <_{c'} B : \textbf{Type}$, then $\Gamma \vdash c = c' : (A)B$.

Coherence

Coherence: If $\Gamma \vdash A <_c B : \textbf{Type}$ and $\Gamma \vdash A <_{c'} B : \textbf{Type}$, then $\Gamma \vdash c = c' : (A)B$.

Why require coherence?
If we have two different coercions from A to B,

$$\Gamma \vdash A <_{c_1} B, \quad \Gamma \vdash A <_{c_2} B \quad \Gamma \nvdash c_1 = c_2$$

as the rule

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

Coherence

Coherence: If $\Gamma \vdash A <_c B : \mathbf{Type}$ and $\Gamma \vdash A <_{c'} B : \mathbf{Type}$, then $\Gamma \vdash c = c' : (A)B$.

Why require coherence?
If we have two different coercions from A to B,

$$\Gamma \vdash A <_{c_1} B, \quad \Gamma \vdash A <_{c_2} B \quad \Gamma \not\vdash c_1 = c_2$$

as the rule

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

$$\Gamma \vdash f(c_1(a)) = f(c_2(a))$$

Coherence

Coherence: If $\Gamma \vdash A <_c B : \mathbf{Type}$ and $\Gamma \vdash A <_{c'} B : \mathbf{Type}$, then $\Gamma \vdash c = c' : (A)B$.

Why require coherence?
If we have two different coercions from A to B,

$$\Gamma \vdash A <_{c_1} B, \quad \Gamma \vdash A <_{c_2} B \quad \Gamma \nvdash c_1 = c_2$$

as the rule

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

$$\Gamma \vdash c_1(a) = c_2(a)$$

Coherence

Coherence: If $\Gamma \vdash A <_c B : \mathbf{Type}$ and $\Gamma \vdash A <_{c'} B : \mathbf{Type}$, then $\Gamma \vdash c = c' : (A)B$.

Why require coherence?
If we have two different coercions from A to B,

$$\Gamma \vdash A <_{c_1} B, \quad \Gamma \vdash A <_{c_2} B \quad \Gamma \not\vdash c_1 = c_2$$

as the rule

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

$$\Gamma \vdash c_1 = c_2$$

Coherence

Coherence: If $\Gamma \vdash A <_c B : \textbf{Type}$ and $\Gamma \vdash A <_{c'} B : \textbf{Type}$, then $\Gamma \vdash c = c' : (A)B$.

Why require coherence?
If we have two different coercions from A to B,

$$\Gamma \vdash A <_{c_1} B, \;\; \Gamma \vdash A <_{c_2} B \;\; \Gamma \not\vdash c_1 = c_2$$

as the rule

$$\frac{\Gamma \vdash f : B \to C \quad \Gamma \vdash a : A \quad \Gamma \vdash A <_c B}{\Gamma \vdash f(a) = f(c(a)) : C}$$

$$\Gamma \vdash c_1 = c_2$$

not conservative

What is conservative

Consider two systems T_1 and T_2, T_2 is an extension of T_1. We call system T_2 is a conservative extension of system T_1, if for every judgement $\Gamma \vdash J$ in T_1,

$$\Gamma \vdash_{T_2} J \quad \Rightarrow \quad \Gamma \vdash_{T_1} J$$

What is conservative

Consider two systems T_1 and T_2, T_2 is an extension of T_1. We call system T_2 is a conservative extension of system T_1, if for every judgement $\Gamma \vdash J$ in T_1,

$$\Gamma \vdash_{T_2} J \quad \Rightarrow \quad \Gamma \vdash_{T_1} J$$

Why require conservative?

What is conservative

Consider two systems T_1 and T_2, T_2 is an extension of T_1. We call system T_2 is a conservative extension of system T_1, if for every judgement $\Gamma \vdash J$ in T_1,

$$\Gamma \vdash_{T_2} J \quad \Rightarrow \quad \Gamma \vdash_{T_1} J$$

Why require conservative?
We take coercive subtyping as a mechanism of abbreviation, which means it should not increase the original type system!

T[$\mathcal{R}$] is not conservative

	T[$\mathcal{R}$]	
	T	(type system)
+	$\Gamma \vdash A <_c B$: **Type**	(subtyping judgements)
+	$\mathcal{R}$	(a set of coherent coercion rules)
+	general subtyping rules	T[R]$_0$
+	subkinding rules	T[R]$_{0K}$
+	coercion application rules	T[$\mathcal{R}$].

$\mathsf{T}[\mathcal{R}]$ is not conservative

If rule set $\mathcal{R}$ contains the following two rules.

$$\frac{\Gamma \vdash f : Nat \to Nat \quad \Gamma \vdash e_0 \in Even \quad \Gamma \vdash f(e_0) : Nat}{\Gamma \vdash Int <_{c_1} Nat : \textbf{Type}}$$

$$\frac{}{\vdash Even <_{c_2} Nat : \textbf{Type}}$$

$\mathrm{T}[\mathcal{R}]$ is not conservative

If rule set $\mathcal{R}$ contains the following two rules.

$$\frac{\Gamma \vdash f : \mathit{Nat} \to \mathit{Nat} \quad \Gamma \vdash e_0 \in \mathit{Even} \quad \Gamma \vdash f(e_0) : \mathit{Nat}}{\Gamma \vdash \mathit{Int} <_{c_1} \mathit{Nat} : \mathbf{Type}}$$

$$\frac{}{\vdash \mathit{Even} <_{c_2} \mathit{Nat} : \mathbf{Type}}$$

In $\mathrm{T}[\mathcal{R}]$

$$\frac{\Gamma \vdash f : \mathit{Nat} \to \mathit{Nat} \quad \Gamma \vdash e0 \in \mathit{Even} \quad \vdash \mathit{Even} <_{c_2} \mathit{Nat} : \mathbf{Type}}{\Gamma \vdash f(e_0) : \mathit{Nat}}$$

T[$\mathcal{R}$] is not conservative

If rule set $\mathcal{R}$ contains the following two rules.

$$\frac{\Gamma \vdash f : Nat \rightarrow Nat \quad \Gamma \vdash e_0 \in Even \quad \Gamma \vdash f(e_0) : Nat}{\Gamma \vdash Int <_{c_1} Nat : \textbf{Type}}$$

$$\frac{}{\vdash Even <_{c_2} Nat : \textbf{Type}}$$

In T[$\mathcal{R}$]

$$\Gamma \vdash Int <_{c_1} Nat : \textbf{Type}$$

T[$\mathcal{R}$] is not conservative

If rule set $\mathcal{R}$ contains the following two rules.

$$\frac{\Gamma \vdash f : Nat \to Nat \quad \Gamma \vdash e_0 \in Even \quad \Gamma \vdash f(e_0) : Nat}{\Gamma \vdash Int <_{c_1} Nat : \mathbf{Type}}$$

$$\frac{}{\vdash Even <_{c_2} Nat : \mathbf{Type}}$$

In T[$\mathcal{R}$]

$$\Gamma \vdash Int <_{c_1} Nat : \mathbf{Type}$$

Could not be derived in T$[\mathcal{R}]_0$ and T$[\mathcal{R}]_{0K}$, we could not use application rule of coercive subtyping.

T[$\mathcal{R}$] is not conservative

If rule set $\mathcal{R}$ contains the following two rules.

$$\frac{\Gamma \vdash f : Nat \to Nat \quad \Gamma \vdash e_0 \in Even \quad \Gamma \vdash f(e_0) : Nat}{\Gamma \vdash Int <_{c_1} Nat : \mathbf{Type}}$$

$$\frac{}{\vdash Even <_{c_2} Nat : \mathbf{Type}}$$

In T[$\mathcal{R}$]

$$\Gamma \vdash Int <_{c_1} Nat : \mathbf{Type}$$

Could not be derived in T$[\mathcal{R}]_0$ and T$[\mathcal{R}]_{0K}$, we could not use application rule of coercive subtyping.
"$T[\mathcal{R}]$" $= T[\mathcal{C}_{\mathcal{R}}]$

$$\mathcal{C}_{\mathcal{R}} = \{\Gamma \vdash A <_c B : \mathbf{Type} | \Gamma \vdash_{T[\mathcal{R}]_{0K}} A <_c B : \mathbf{Type}\}$$

All the subtyping judgements which could be derived in $T[\mathcal{R}]_{0K}$

T[$\mathcal{R}$] is not conservative

If rule set $\mathcal{R}$ contains the following two rules.

$$\frac{\Gamma \vdash f : Nat \to Nat \quad \Gamma \vdash e_0 \in Even \quad \Gamma \vdash f(e_0) : Nat}{\Gamma \vdash Int <_{c_1} Nat : \mathbf{Type}}$$

$$\frac{}{\vdash Even <_{c_2} Nat : \mathbf{Type}}$$

In T[$\mathcal{R}$]

$$\Gamma \vdash Int <_{c_1} Nat : \mathbf{Type}$$

Could not be derived in $\mathrm{T}[\mathcal{R}]_0$ and $\mathrm{T}[\mathcal{R}]_{0K}$, we could not use application rule of coercive subtyping.
$''T[\mathcal{R}]'' = T[\mathcal{C}_{\mathcal{R}}] \quad \leftrightsquigarrow \mathrm{T}$

$$\mathcal{C}_{\mathcal{R}} = \{\Gamma \vdash A <_c B : \mathbf{Type} | \Gamma \vdash_{T[\mathcal{R}]_{0K}} A <_c B : \mathbf{Type}\}$$

All the subtyping judgements which could be derived in $T[\mathcal{R}]_{0K}$

Coerceive subtyping in global

But sometimes, we only want some coercions to be meaningful in a specially local context.

Coercive subtyping in global

But sometimes, we only want some coercions to be meaningful in a specially local context.

E.g., **seven runs**

$$[[seven]] : Number, \quad [[run]] : RO \rightarrow Prop$$

Coercive subtyping in global

But sometimes, we only want some coercions to be meaningful in a specially local context.
E.g., **seven runs**

$$[[seven]] : Number, \quad [[run]] : RO \rightarrow Prop$$

in a football match, should mean a man runs.

$$Number <_{c_2} Human, \quad Human <_{c_1} RO$$

$$Number <_{c_1 \circ c_2} RO$$

$$[[run]]([[seven]]) = [[run]](c_1 \circ c_2[[seven]]) : Prop$$

Coercive subtyping in global

But sometimes, we only want some coercions to be meaningful in a specially local context.
E.g., **seven runs**

$$[[seven]] : Number, \quad [[run]] : RO \to Prop$$

in a football match, should mean a man runs.

$$Number <_{c_2} Human, \quad Human <_{c_1} RO$$

$$Number <_{c_1 \circ c_2} RO$$

$$[[run]]([[seven]]) = [[run]](c_1 \circ c_2[[seven]]) : Prop$$

in a car racing match, should refer a car runs.

$$Number <_{c_4} Car, \quad Car <_{c_3} RO$$

$$Number <_{c_3 \circ c_4} RO$$

Put coercions into context!

Informally, we may derive things like

$$...Number <_c Human... \vdash \quad seven\ runs$$

$$...Number <_{c'} \quad Car \quad ... \vdash \quad seven\ runs$$

Put coercions into context!

Informally, we may derive things like

$$...Number <_c Human... \vdash \quad seven\ runs = Beckham\ runs$$

$$...Number <_{c'} \quad Car \quad ... \vdash \quad seven\ runs$$

Put coercions into context!

Informally, we may derive things like

$$...Number <_c Human... \vdash \quad seven\ runs = Beckham\ runs$$

$$...Number <_{c'} \quad Car \quad ... \vdash \quad seven\ runs = Ferrari\ runs$$

Contextual coercive subtyping: System $T^0_<$

Definition (system $T^0_<$)

Given a type system T, System $T^0_<$ is an extension of System T with the following:

- Context Γ is a finite sequence of expressions of form $a : M$ and $A <_c B$.
- Extended with judgements of form $\Gamma \vdash A <_c B : \mathbf{Type}$.
- Extended with the following rules .

Contextual coercive subtyping: Basic rules

Coercion in context

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

$$\frac{\Gamma, A <_c B, \Gamma' \ \mathbf{valid}}{\Gamma, A <_c B, \Gamma' \vdash A <_c B : \mathbf{Type}}$$

Congruence rule

$$\frac{\Gamma \vdash A <_c B : \mathbf{Type} \quad \Gamma \vdash A = A' : \mathbf{Type} \quad \Gamma \vdash B = B' : \mathbf{Type} \quad \Gamma \vdash c = c' : (A)B}{\Gamma \vdash A' <_{c'} B' : \mathbf{Type}}$$

Transitivity rule

$$\frac{\Gamma \vdash A <_c B : \mathbf{Type} \quad \Gamma \vdash B <_{c'} C : \mathbf{Type}}{\Gamma \vdash A <_{c' \circ c} C : \mathbf{Type}}$$

Substitution rule

$$\frac{\Gamma, x : K, \Gamma' \vdash A <_c B : \mathbf{Type} \quad \Gamma \vdash k : K}{\Gamma, [k/x]\Gamma' \vdash [k/x]A <_{[k/x]c} [k/x]B : \mathbf{Type}}$$

Contextual coercive subtyping

Definition (coherence of coercion in contexts)

Let Γ be a context with coercive subtyping, we say Γ is coherent if it has the following coherence properties in system $T^0_<$.

- If $\Gamma \vdash A <_c B : \mathbf{Type}$, then $\Gamma \vdash A : \mathbf{Type}$, $\Gamma \vdash B : \mathbf{Type}$, and $\Gamma \vdash c : (A)B$.
- $\Gamma \not\vdash A <_c A : \mathbf{Type}$ for any Γ, A, and c.
- $\Gamma \vdash c = c' : (A)B$ whenever $\Gamma \vdash A <_c B$ and $\Gamma \vdash A <_{c'} B$.

We will only focus on the coherent contexts.

Contextual coercive subtyping

Definition (system $\mathrm{T}_{<}^{0K}$)

System $T_{<}^{0K}$ is an extension of system T, obtained from $T_{<}^{0}$ by adding the following subkinding inference rules

Contextual coercive subtyping: Subkinding rules

Basic subkinding rule

$$\frac{\Gamma \vdash A <_c B : \textbf{Type}}{\Gamma \vdash El(A) <_c El(B)}$$

Subkinding for dependent product kinds

$$\frac{\Gamma \vdash K_1' <_{c_1} K_1 \quad \Gamma, x' : K_1' \vdash [c_1(x')/x]K_2 = K_2' \quad \Gamma, x : K_1 \vdash K_2 : \textbf{kind}}{\Gamma \vdash (x : K_1)K_2 <_c (x' : K_1')K_2'}$$

where $c \equiv [f : (x : K_1)K_2][x' : K_1']f(c_1(x'))$;

$$\frac{\Gamma \vdash K_1 = K_1 \quad \Gamma, x' : K_1' \vdash K_2 <_{c_2} K_2' \quad \Gamma, x : K_1 \vdash K_2 : \textbf{kind}}{\Gamma \vdash (x : K_1)K_2 <_c (x' : K_1')K_2'}$$

where $c \equiv [f : (x : K_1)K_2][x' : K_1']c_2 f(x')$;

$$\frac{\Gamma \vdash K_1' <_{c_1} K_1 \quad \Gamma, x' : K_1' \vdash [c_1(x')/x]K_2 <_{c_2} K_2' \quad \Gamma, x : K_1 \vdash K_2 : \textbf{kind}}{\Gamma \vdash (x : K_1)K_2 <_c (x' : K_1')K_2'}$$

where $c \equiv [f : (x : K_1)K_2][x' : K_1']c_2 f(c_1(x'))$.

Contextual coercive subtyping: Subkinding rules

Congruence for subkinding

$$\frac{\Gamma \vdash K_1 <_c K_2 \quad \Gamma \vdash K_1 = K_1' \quad \Gamma \vdash K_2 = K_2' \quad \Gamma \vdash c = c' : (K_1)K_2}{\Gamma \vdash K_1' <_c K_2'}$$

Transitivity for subkinding

$$\frac{\Gamma \vdash K <_c K' \quad \Gamma \vdash K' <_c' K'}{\Gamma \vdash K <_{c' \circ c} K''}$$

Substitution for subkinding

$$\frac{\Gamma, x : K, \Gamma' \vdash K_1 <_c K_2 \quad \Gamma \vdash k : K}{\Gamma, [k/x]\Gamma' \vdash [k/x]K_1 <_{[k/x]c} [k/x]K_2}$$

Rules in UTT system ([Luo94])

Contextual coercive subtyping: System $\mathrm{T}_<$

Definition (system $\mathrm{T}_<$)

System $\mathrm{T}_<$ is an extension of system T, obtained from $T^{0K}_<$ by adding the following rules.

New rules for application

$$\frac{\Gamma \vdash f : (x : K)K' \quad \Gamma \vdash k_0 : K_0 \quad \Gamma \vdash K_0 <_c K}{\Gamma \vdash f(k_0) : [c(k_0)/x]K'}$$

Coercive definition rule

$$\frac{\Gamma \vdash f : (x : K)K' \quad \Gamma \vdash k_0 : K_0 \quad \Gamma \vdash K_0 <_c K}{\Gamma \vdash f(k_0) = f(c(K_0)) : [c(k_0)/x]K'}$$

Example

Let's consider the example of **seven runs** in both football match and car racing match again.

First, we take some types, and a function.

$$RO : \mathbf{Type}, \quad Human : \mathbf{Type}$$

$$Car : \mathbf{Type}, \quad Number : \mathbf{Type}$$

$$[[run]] : RO \to Prop$$

In the context of football match,

$$c_1 : Human \to RO, \quad Human <_{c_1} RO$$

$$c_2 : Number \to Human, \quad Number <_{c_2} Human$$

In the context of car racing match,

$$c_3 : Car \to RO, \quad Car <_{c_3} RO$$

$$c_4 : Number \to Car, \quad Number <_{c_4} Car$$

In both the football match and car racing match, **seven** is a object of type Number.

$$[[seven]] : Number$$

Example

We use Γ_1 for context of football match and Γ_2 for the context of car racing match, then we have:

$\Gamma = RO : \textbf{Type}, Car : \textbf{Type}, Human : \textbf{Type}, Number : \textbf{Type}, [[run]] : RO \to Prop$

$\Gamma_1 = \Gamma, c_1 : Human \to RO, Human <_{c_1} RO, c_2 : Number \to Human, Number <_{c_2} Human$

$\Gamma_2 = \Gamma, c_3 : Car \to RO, Car <_{c_3} RO, c_4 : Number \to Car, Number <_{c_4} Car$

Example

Then we could derive:

$$\frac{\Gamma_1 \vdash \mathit{Human} <_{c_1} \mathit{RO} : \mathbf{Type} \quad \Gamma_1 \vdash \mathit{Number} <_{c_2} \mathit{Human} : \mathbf{Type}}{\Gamma_1 \vdash \mathit{Number} <_{c_2 \circ c_1} \mathit{RO} : \mathbf{Type}}$$

$$\frac{\Gamma_1 \vdash \mathit{Number} <_{c_2 \circ c_1} \mathit{RO} : \mathbf{Type} \quad \Gamma_1 \vdash \mathit{run} : \mathit{RO} \rightarrow \mathit{Prop}}{\Gamma_1 \vdash [[\mathit{run}]]([[\mathit{seven}]]) = [[\mathit{run}]](c_2 \circ c_1([[\mathit{seven}]]) : \mathit{Prop}}$$

$$\frac{\Gamma_2 \vdash \mathit{Car} <_{c_3} \mathit{RO} : \mathbf{Type} \quad \Gamma_2 \vdash \mathit{Number} <_{c_4} \mathit{Car} : \mathbf{Type}}{\Gamma_2 \vdash \mathit{Number} <_{c_4 \circ c_3} \mathit{RO} : \mathbf{Type}}$$

$$\frac{\Gamma_2 \vdash \mathit{Number} <_{c_4 \circ c_3} \mathit{RO} : \mathbf{Type} \quad \Gamma_2 \vdash \mathit{run} : \mathit{RO} \rightarrow \mathit{Prop}}{\Gamma_2 \vdash [[\mathit{run}]]([[\mathit{seven}]]) = [[\mathit{run}]](c_4 \circ c_3([[\mathit{seven}]])) : \mathit{Prop}}$$

Coercive subtyping systems

T

Coercive subtyping systems

$$T$$

$$T[\mathcal{R}] \qquad T[\mathcal{C}]$$

Coercive subtyping systems

$$T$$

$$T[\mathcal{R}] = T[\mathcal{C}_{\mathcal{R}}] \qquad T[\mathcal{C}]$$

Coercive subtyping systems

$$T$$

$$T[\mathcal{R}] = T[\mathcal{C}_{\mathcal{R}}] \qquad T[\mathcal{C}]$$

$$T_{<}$$

Coercive subtyping systems

$$T$$

$$T[\mathcal{R}] = T[\mathcal{C}_{\mathcal{R}}] \qquad T[\mathcal{C}]$$

$$T_<$$

$$T_<[\mathcal{R}] = T_<[\mathcal{C}_{\mathcal{R}}] \qquad T_<[\mathcal{C}]$$

Coercive subtyping systems

$$T$$

$$T[\mathcal{R}] = T[\mathcal{C}_{\mathcal{R}}] \qquad T[\mathcal{C}]$$

$$T_{<}$$

$$T_{<}[\mathcal{R}] = T_{<}[\mathcal{C}_{\mathcal{R}}] \qquad T_{<}[\mathcal{C}]$$

Coercive subtyping in Coq

Coercive subtyping could be defined as followed in Coq

$$\textit{Variable} \;\; c : A \rightarrow B.$$

$$\textit{Coercion} \;\; c : A \rightarrowtail B.$$

It uses a graph to store the coercions (keep coherence) And coercive subtyping could be defined in **Section**, to be contextual.

Coercive subtyping in Coq

Coercive subtyping could be defined as followed in Coq

$$\textit{Variable} \;\; c : A \to B.$$

$$\textit{Coercion} \;\; c : A \rightarrowtail B.$$

It uses a graph to store the coercions (keep coherence) And coercive subtyping could be defined in **Section**, to be contextual.

However, Coq has a finite graph of the coercions, and the $\mathcal{R}$ and $\mathcal{C}$ are infinite.

Future work

Coercions may be introduced into terms, with the following rule:

$$\frac{\Gamma, A <_c B \vdash J}{\Gamma \vdash \mathbf{coercion} A <_c B \;\; \mathbf{in} \;\; J}$$

J could be any judgement, if $J \equiv t : T$, **coercion** should distribute through J, $\mathbf{coercion} A <_c B \;\; \mathbf{in} \;\; (t : T) = (\mathbf{coercion} A <_c B \;\; \mathbf{in} \;\; t) : (\mathbf{coercion} A <_c B \;\; \mathbf{in} \;\; T)$

$$\frac{\Gamma, A <_c B \vdash t : T}{\Gamma \vdash (\mathbf{coercion} A <_c B \;\; \mathbf{in} \;\; t) : (\mathbf{coercion} A <_c B \;\; \mathbf{in} \;\; T)}$$

Reference

- [Luo99] Z. Luo. Coercive subtyping. *Journal of Logic and Computation.* 9(1), 105-130, 1999.
- [SL00] S. Soloviev, Z. Luo. Coercion completion and conservativity in ceorcive subtyping. *ANNALS OF PURE AND APPLIED LOGIC*, 2000
- [LL05] Z. Luo, Y. Luo. Transitivity in coercive subtyping. *Information and Computation* 197, 122-144, 2005.
- [Luo10] Z. Luo. Type-Theoretical Semantics with Coercive Subtyping. Proceedings of SALT Volume 20, 1-15, 2010.