A General Semantic Construction of Dependent Refinement Type Systems, Categorically

Satoshi Kura

National Institute of Informatics, Tokyo, Japan

The Graduate University for Advanced Studies (SOKENDAI), Kanagawa, Japan

Introduction

Dependent Refinement Type System

Dependent Refinement Type System (DRTS) [Flanagan, POPL'06] is used for verification.

Implementation: LiquidHaskell, F*, ...

A DRTS has

- refinement types,
- dependent types,
- subtyping relation.

Refinement Types

A DRTS has **refinement types** $\{v : A \mid p\}$.

$$\vdash 1 : \{v{:}\mathbf{int} \mid v \geq 0\}$$

$$\nvdash -1 : \{v{:}\mathbf{int} \mid v \geq 0\}$$

$$x : \{v{:}\mathbf{int} \mid v \geq 0\} \vdash 2x : \{w{:}\mathbf{int} \mid w \geq 0\}$$

DRTSs can specify pre-/postconditions.

Dependent Types

A DRTS is **dependently typed**.

$$\vdash \lambda x.x + 1 : (x{:}\mathbf{int}) \to \{v{:}\mathbf{int} \mid v = x + 1\}$$

Postconditions can depend on the input values.

Subtyping Relation

A DRTS has a **subtyping relation** $<:$ induced by logical implication.

$$\frac{v \geq 0 \implies \mathbf{true}}{\vdash \{v{:}\mathbf{int} \mid v \geq 0\} <: \{v{:}\mathbf{int} \mid \mathbf{true}\}}$$

DRTS Combines Type System and Logic

DRTS

$$x : \{v{:}\mathbf{int} \mid v \geq 0\}$$

$$\vdash x + 1 : \{w : \mathbf{int} \mid w = x + 1\}$$

Underlying Type System (UTS)

$$x : \mathbf{int} \vdash x + 1 : \mathbf{int}$$

DTT (or STT $\hookrightarrow$ DTT)

Predicate Logic

$$v \geq 0$$

$$w = x + 1$$

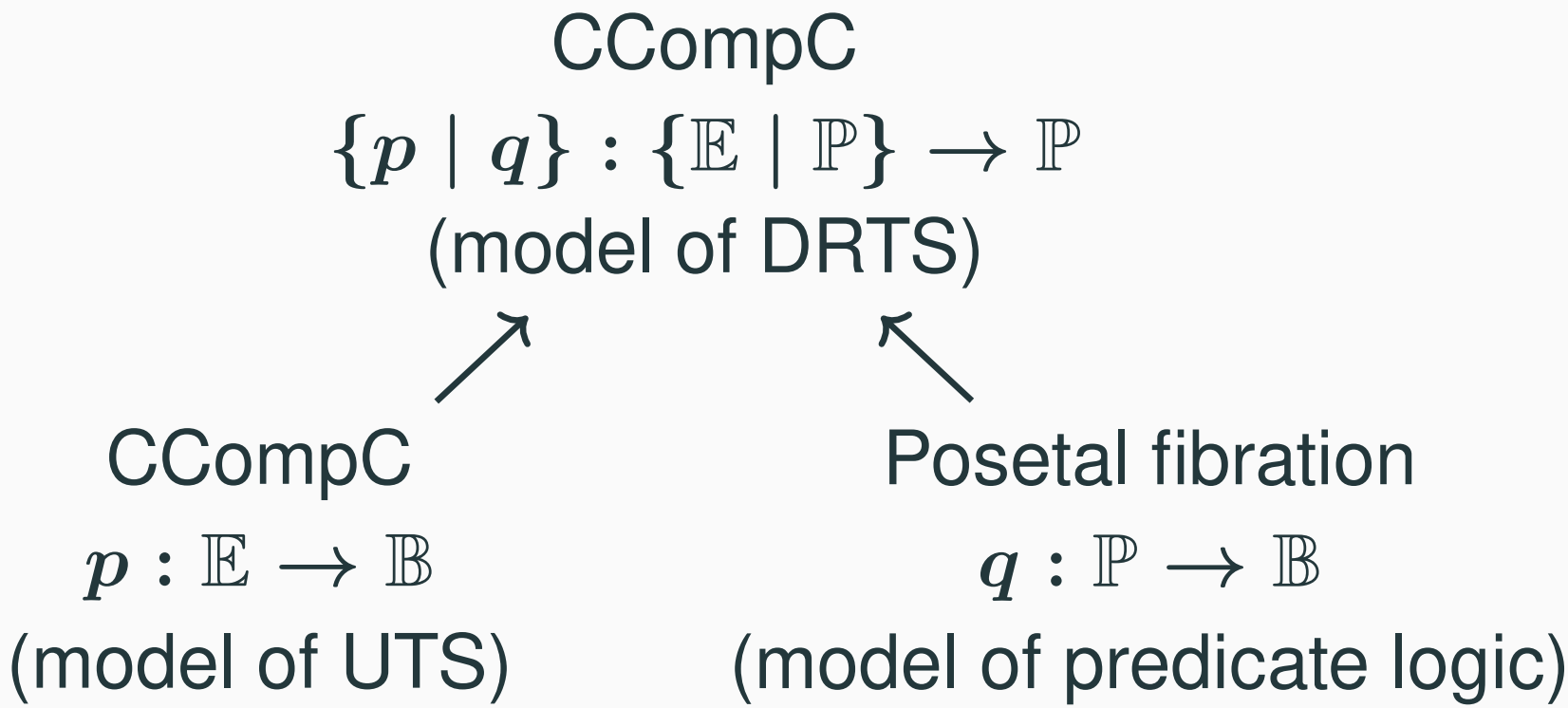
Our Question

How are UTS, predicate logic, and DRTS related from the viewpoint of categorical semantics?

- As a theoretical framework to handle them in a uniform manner.
- As a guideline to make new DRTSs.
  - With computational effects (= monads)
  - For relational verification

Our Answer: a General Construction

A categorical construction of DRTSs:



(CCompC: Closed Comprehension Category)

Refined Semantics

$\{p \mid q\}$ gives a (sound) interpretation of DRTS.

$$\text{context in } \{p \mid q\} = \text{context in } p \;\&\; \text{predicate in } q$$
$$\text{type in } \{p \mid q\} = \text{type in } p \;\&\; \text{predicate in } q$$
$$\text{term in } \{p \mid q\} = \text{term in } p \;\&\; \text{proof in } q$$

Interpretation of UTS

Interpretation of UTS

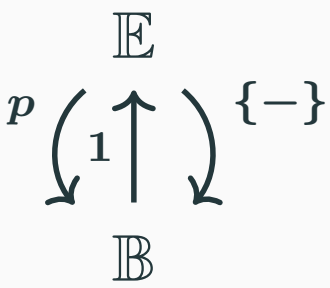
As an UTS, we consider a dependent type system with

- unit type 1,
- dependent product types $\Pi(x : A).B$, and
- dependent coproduct types $\Sigma(x : A).B$.

A (fibrational) model of the UTS is given by a **closed comprehension category (CCompC)** [Jacobs, TCS'93].

CCompC

A CCompC is a fibration $p : \mathbb{E} \to \mathbb{B}$ that has



- 1: fibred terminal object functor $p \dashv 1$,
- $\{-\}$: comprehension functor $1 \dashv \{-\}$,
- adjunctions for dependent products/coproducts,

satisfying some conditions (omitted in this talk).

Interpretation in CCompC

Context: object $[\![\Gamma]\!] \in \mathbb{B}$

Type in context: object $[\![\Gamma \vdash A]\!] \in \mathbb{E}_{[\![\Gamma]\!]}$

Term: morphism in $\mathbb{E}$ or $\mathbb{B}$

$$
\begin{aligned}
&[\![\Gamma \vdash M : A]\!] \\
&\in \mathbb{E}_{[\![\Gamma]\!]}\big(\, \mathbf{1}[\![\Gamma]\!],\ [\![\Gamma \vdash A]\!] \,\big) \\
&\cong \Big\{ f \in \mathbb{B}\big(\, [\![\Gamma]\!],\ \{[\![\Gamma \vdash A]\!]\} \,\big) \mid \pi \circ f = \mathbf{id} \Big\}
\end{aligned}
$$

(where $\pi = p\epsilon^{\mathbf{1} \dashv \{-\}}_{[\![\Gamma \vdash A]\!]}$ is the projection)

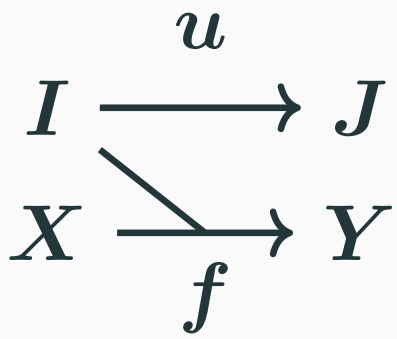
Example: Simple Fibration

$s(\mathbf{Set})$ is defined by

- object: (I, X) where $I, X \in \mathbf{Set}$
- morphism: $(u, f) : (I, X) \to (J, Y)$

where $u : I \to J$

and $f : I \times X \to Y$



The **simple fibration** $s_{\mathbf{Set}} : s(\mathbf{Set}) \to \mathbf{Set}$ defined by $(I, X) \mapsto I$ is a CCompC.

Example: Interpretation in s_{Set}

Let $\Gamma \coloneqq x : \mathbf{int}$, $M \coloneqq x + 1$, $A \coloneqq \mathbf{int}$.

$[\![\Gamma]\!] = [\![x : \mathbf{int}]\!]$:

$$\mathbb{Z} \in \mathbf{Set}$$

$[\![\Gamma \vdash A]\!] = [\![x : \mathbf{int} \vdash \mathbf{int}]\!]$:

$$(\mathbb{Z}, \mathbb{Z}) \in s(\mathbf{Set})$$

$[\![\Gamma \vdash M : A]\!] = [\![x : \mathbf{int} \vdash x + 1 : \mathbf{int}]\!]$:

- $(\mathrm{id}_{\mathbb{Z}}, \lambda(x, *).x + 1) \in s(\mathbf{Set})_{\mathbb{Z}}\big((\mathbb{Z}, 1), (\mathbb{Z}, \mathbb{Z})\big)$
- $\langle \mathrm{id}_{\mathbb{Z}}, \lambda x.x + 1 \rangle : \mathbb{Z} \to \mathbb{Z} \times \mathbb{Z}$ in $\mathbf{Set}$

Interpretation of Predicates

Interpreting Predicates

To interpret predicates, we consider a fibration $q : \mathbb{P} \to \mathbb{B}$ s.t.

- q is **posetal**, i.e., each fibre $\mathbb{P}_I$ is a poset,
- q is a **fibred-ccc** (admits $\top$, $\wedge$, $\Longrightarrow$),
- q admits **p-products** (universal quantifier $\forall x : A$) where $p : \mathbb{E} \to \mathbb{B}$ is a CCompC for UTS.

Interpretation of a predicate: $[\![\Gamma \vdash p]\!] \in \mathbb{P}_{[\![\Gamma]\!]}$

Example: Subobject Fibration

$\mathbf{Sub}(\mathbf{Set})$ is defined by

- object: (I, P)

  where $I \in \mathbf{Set}$ and $P \subseteq I$
- morphism: $f : (I, P) \to (J, Q)$

  s.t.
$$\begin{array}{ccc} P & \dashrightarrow & Q \\ \cap\!| & & \cap\!| \\ I & \xrightarrow{f} & J \end{array}$$

The **subobject fibration** is defined by

$$\mathbf{sub}_{\mathbf{Set}} : \mathbf{Sub}(\mathbf{Set}) \to \mathbf{Set} \quad \mathbf{sub}_{\mathbf{Set}}(I, P) \coloneqq I.$$

Example: Interpretation in $\mathrm{sub}_{\mathbf{Set}}$

In $\mathbf{sub_{Set}} : \mathbf{Sub(Set)} \to \mathbf{Set}$,

$$[\![x : \mathbf{int} \vdash x \geq 0]\!]$$
$$= (\mathbb{Z},\ \{x \in \mathbb{Z} \mid x \geq 0\})$$

The Construction

CCompC for DRTS

Given

- $s_{\mathbf{Set}} : s(\mathbf{Set}) \to \mathbf{Set}$
- $\mathbf{sub}_{\mathbf{Set}} : \mathbf{Sub}(\mathbf{Set}) \to \mathbf{Set}$,

we construct a CCompC for DRTS
whose total category consists of pairs of

- an underlying type in $s_{\mathbf{Set}} : s(\mathbf{Set}) \to \mathbf{Set}$
- a predicate in $\mathbf{sub}_{\mathbf{Set}} : \mathbf{Sub}(\mathbf{Set}) \to \mathbf{Set}$

Total Category of CCompC for DRTS

We define a category $\{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\}$ by the pullback.

$$\begin{array}{ccc} \{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\} & \longrightarrow & \mathbf{Sub}(\mathbf{Set})^{\to} \\ \downarrow & \lrcorner & \downarrow{\scriptstyle \mathbf{sub}^{\to}_{\mathbf{Set}}} \\ s(\mathbf{Set}) & \xrightarrow{\mathcal{P}} & \mathbf{Set}^{\to} \end{array}$$

where $\mathcal{P}(I, X) = \pi : I \times X \to I$ is the projection.

Objects in the Total Category

Objects in $\{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\}$:

$$((I, X), P, Q)$$

where $(I, X) \in s(\mathbf{Set})$ and $\begin{array}{ccc} Q & \dashrightarrow & P \\ \cap\!| & & \cap\!| \\ I \times X & \xrightarrow{\pi} & I \end{array}$.

- $(I, X) \in s(\mathbf{Set})$: underlying type
- $P \subseteq I$: predicate on the context
- $Q \subseteq I \times X$: predicate on the type

Definition of CCompC for DRTS

We define

$$\{s_{\mathbf{Set}} \mid \mathbf{sub}_{\mathbf{Set}}\} : \{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\} \to \mathbf{Sub}(\mathbf{Set})$$

by

$$((I, X), P, Q) \mapsto (I, P).$$

Then this gives a CCompC.

Example: Interpretation of Context

In the CCompC $\{s_{\mathbf{Set}} \mid \mathbf{sub}_{\mathbf{Set}}\}$,
a context is interpreted as an object in
the base category $\mathbf{Sub}(\mathbf{Set})$.

$$[\![x : \{x{:}\mathbf{int} \mid x \geq 0\}]\!]$$

$$= (\mathbb{Z}, \{x \mid x \geq 0\}) \in \mathbf{Sub}(\mathbf{Set})$$

Example: Interpretation of Type

A type is interpreted as an object in
the total category $\{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\}$.

$$[\![x : \{x{:}\mathbf{int} \mid x \geq 0\} \vdash \{v{:}\mathbf{int} \mid v = x + 1\}]\!]$$

$$= ((\mathbb{Z}, \mathbb{Z}), \{x \in \mathbb{Z} \mid x \geq 0\},$$
$$\{(x, v) \in \mathbb{Z}^2 \mid x \geq 0 \wedge v = x + 1\})$$

$$\in \{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\}$$

Example: Interpretation of Term

A term is interpreted as a morphism in $\mathbf{Sub}(\mathbf{Set})$ (or $\{s(\mathbf{Set}) \mid \mathbf{Sub}(\mathbf{Set})\}$)

$$A[\![x : \{x{:}\mathbf{int} \mid x \geq 0\} \vdash x+1 : \{v{:}\mathbf{int} \mid v = x+1\}]\!]$$

$$= \begin{array}{ccc} \{x \in \mathbb{Z} \mid x \geq 0\} & \dashrightarrow & \{(x,v) \in \mathbb{Z}^2 \mid \begin{array}{l} x \geq 0 \wedge \\ v = x+1 \end{array}\} \\ \cap\downarrow & & \\ \mathbb{Z} & \xrightarrow{\langle \mathbf{id},\ \lambda x.x+1 \rangle} & \cap\downarrow \\ & & \mathbb{Z}^2 \end{array}$$

Generalized Construction

Given

- a CCompC $p : \mathbb{E} \to \mathbb{B}$ and
- a posetal fibration $q : \mathbb{P} \to \mathbb{B}$,

we define $\{p \mid q\} : \{\mathbb{E} \mid \mathbb{P}\} \to \mathbb{P}$ by

$$\begin{array}{ccccc} \{\mathbb{E} \mid \mathbb{P}\} & \longrightarrow & \mathbb{P}^{\to} & \xrightarrow{\mathbf{cod}} & \mathbb{P} \\ \downarrow & \lrcorner & \downarrow{\scriptstyle q^{\to}} & & \\ \mathbb{E} & \xrightarrow{\mathcal{P}} & \mathbb{B}^{\to} & & \end{array}$$

where $\mathcal{P}X = p\epsilon_X^{1 \dashv \{-\}}$ is the projection.

Main Theorem

If $p : \mathbb{E} \to \mathbb{B}$ is a CCompC and $q : \mathbb{P} \to \mathbb{B}$ is a posetal fibration that is fibred-ccc and has p-products,
then $\{p \mid q\} : \{\mathbb{E} \mid \mathbb{P}\} \to \mathbb{P}$ is a CCompC.

Moreover, there is a morphism of CCompCs from $\{p \mid q\}$ to p.

$$\begin{array}{ccc} \{\mathbb{E} \mid \mathbb{P}\} & \longrightarrow & \mathbb{E} \\ {\scriptstyle \{p|q\}}\downarrow & & \downarrow{\scriptstyle p} \\ \mathbb{P} & \longrightarrow & \mathbb{B} \end{array}$$

Omitted in the Talk

In this talk, we consider UTS/DRTS with

- unit type
- dependent product/coproduct types.

We can also support UTS/DRTS with

- fibred coproduct types $A + B$
- computational effects (monads)
- recursion (but this is not completed yet)

Conclusions & Future Work

Conclusions

Given

- $\boldsymbol{p} : \mathbb{E} \to \mathbb{B}$ (CCompC for UTS)
- $\boldsymbol{q} : \mathbb{P} \to \mathbb{B}$ (posetal fibration for predicate logic),

we constructed a CCompC for DRTS

$$\{\boldsymbol{p} \mid \boldsymbol{q}\} : \{\mathbb{E} \mid \mathbb{P}\} \to \mathbb{P}.$$

context in $\{\boldsymbol{p} \mid \boldsymbol{q}\}$ = context in $\boldsymbol{p}$ & predicate in $\boldsymbol{q}$

type in $\{\boldsymbol{p} \mid \boldsymbol{q}\}$ = type in $\boldsymbol{p}$ & predicate in $\boldsymbol{q}$

term in $\{\boldsymbol{p} \mid \boldsymbol{q}\}$ = term in $\boldsymbol{p}$ & proof in $\boldsymbol{q}$

Future Work

- Complete treatment of recursion
  - Give concrete examples
- Algebraic effects & handlers
- Effect system