MSCS 12/12/2016

CT Basics → CT applications in CS

- λ-cal. & CCC Cartesian closed categories
- Automata & Coalgebras
- (Categorical logic & fibration)

Functional Programming

- Haskell
- OCaml
- F#
- (Scheme)
- ⋮

e.g. - succinctness
- correctness
- types

ICFP

typed λ-calculus

λ-calculus is a prototype

a model of computation

- While languages
- Turing machines
- recursive functions
- untyped λ-calculus
- ⋮

⟸ all equivalent

⟹ Church's thesis

- If $\sim$ were defined coinductively ...

$s \sim t \overset{\text{def.}}{\iff} \exists$

<u>Prop.</u> We have $s \sim t$ for any s, t.

<u>Proof.</u>

$$\dfrac{\dfrac{\dfrac{\vdots}{s \sim t}\,\text{Sym}}{t \sim s}\,\text{Sym}}{s \sim t}\,\text{Sym}$$

<u>Def.</u> Let G be a group.

Let $v : \mathrm{Var} \longrightarrow |G|$ be a function. "valuation"

$\mathrm{Var} = \{\text{all the variables}\}$, $|G|$ = {the underlying set of G}

For each term t, we define $[\![t]\!]_{G,v} \in |G|$ by induction

$t ::= x \mid e \mid t \cdot t \mid t^{-1}$, $x \in \mathrm{Var}$

the interpretation of t under G, v

on the construction of the term t, as follows.

<u>Base</u> $[\![x]\!]_{G,v} = v(x)$ $(x \in \mathrm{Var})$

$[\![\underline{e}]\!]_{G,v} = \underline{e_G}$

a symbol — syntactic

the unit element of the group G — semantical object

Step $[\![t_1 \cdot t_2]\!]_{G,v} = [\![t_1]\!]_{G,v} \cdot_G [\![t_2]\!]_{G,v}$

symbol

multiplication
$\cdot_G : |G|^2 \longrightarrow |G|$
in G

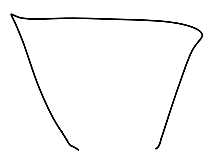
these elem.
in $|G|$ are
already defined
by induction

$[\![t_1^{-1}]\!]_{G,v} = \left([\![t_1]\!]_{G,v}\right)^{-1}$

the inverse
operation
in G

Thm (Soundness)

Proof. Assume $s \approx t$ holds,
that is, $\exists$



$s \approx t$

$\boxed{\text{Base 1}}$ If the pf tree is of the form

$$\frac{}{s \approx t}\ (\text{Associativity})$$

The last applied rule is
(Assoc)

By the shape of the (Assoc) rule we must have

$$s \equiv (t_1 \cdot t_2) \cdot t_3$$
$$t \equiv t_1 \cdot (t_2 \cdot t_3)$$

the syntactic equality
(the same as seq. of symbols)

[Goal: $[\![(t_1 \cdot t_2) \cdot t_3]\!]_{G,\upsilon} = [\![t_1 \cdot (t_2 \cdot t_3)]\!]_{G,\upsilon}$]

goal

by def. of $[\![-]\!]_{G,\upsilon}$ by def. of $[\![-]\!]_{G,\upsilon}$

$$\left([\![t_1]\!] \cdot_G [\![t_2]\!] \right) \cdot_G [\![t_3]\!] = [\![t_1]\!] \cdot_G \left([\![t_2]\!] \cdot_G [\![t_3]\!] \right)$$

[Since G is a group and $\cdot_G$ is associative.]

$\boxed{\text{Base } 2, 3, \ldots}$ Skip!

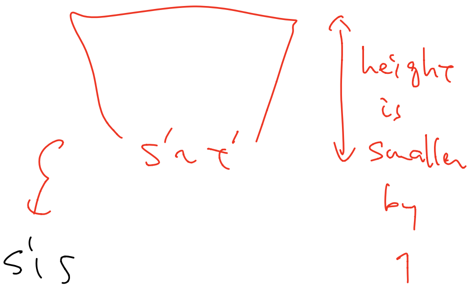
$\boxed{\text{Step } 1}$ If the pf tree is of the form

$$\dfrac{\;\vdots\; \\ s' \sim t'}{s \sim t} \; ((-)^{-1}\text{-cong.})$$

By inspecting the rule we see that

$$s \equiv (s')^{-1}$$

$$t \equiv (t')^{-1}$$



By the induction hypothesis we have $[\![s']\!]_{G,v} = [\![t']\!]_{G,v}$. — ☺

Now

$$[\![s]\!] \underset{s \equiv (s')^{-1}}{=} [\![(s')^{-1}]\!] \underset{\substack{\text{by def. of} \\ [\![\_]\!]_{G,v}}}{=} \left([\![s']\!] \right)^{-1}$$

$$\underset{\text{☺}}{=} \left([\![t']\!] \right)^{-1} \underset{\substack{\text{by def.} \\ \text{of } [\![\_]\!]}}{=} [\![(t')^{-1}]\!] \underset{t \equiv (t')^{-1}}{=} [\![t]\!].$$

$\boxed{\text{Step 2, 3, ...}}$ Similar. ∎

Summary

$\boxed{\text{Syntactic system for groups}} \Longrightarrow \boxed{\text{Algebraic semantics}}$

- terms
- rules for deriving $\sim$

Thm (soundness)

$$s \sim t \Longrightarrow [\![s]\!]_{G,v} = [\![t]\!]_{G,v} \quad (\forall G, v)$$

Now:

- the language for groups

  $\Longrightarrow$ that for (typed) λ-calculus

- groups / algebras

  $\Longrightarrow$ categories