Problems with β-Conversion Rules in Type Theory with Quotation and Evaluation Terms

Petr Kuchyňka[1] and Jiří Raclavský[2,*]

[1] Seznam.Cz, Brno, the Czech Republic
p.kuchynka@gmail.com

[2] Dept. of Philosophy, Masaryk University, Arne Novaka 1, Brno, 602 00, the Czech Republic
raclavsky@phil.muni.cz

1. Extension of type theory by quotation and evaluation terms

To increase expressive power, some programming languages (e.g. Lisp, and its sequels, see [2]), extend the language $\mathscr{L}$ of *simple type theory* (STT)[1] by two new terms (here denoted by):

$\ulcorner X \urcorner$ – *quotation* of the *term* X

$[\![X]\!]$ – *evaluation* of the *term* X

Motivation: Following Farmer, we investigate these extensions for better understanding (from the logical point of view) of programming using such $\mathscr{L}$s, see his [2, 3] for more details.

Using the familiar *Henkin-style semantics* for STT, let $\mathscr{V}_v(X)$ be short for $[\![X]\!]^{\mathscr{M},v}$, where v is an *assignment*, $\mathscr{M}$ is a *model* $\langle \mathscr{F}, \mathscr{I} \rangle$, where $\mathscr{I}$ is an *interpretation function* from constants to objects of $\mathscr{F}$, and $\mathscr{F}$ is $\{\mathscr{D}_\tau \mid \tau \in \mathscr{T}\}$, where τ is a *type* belonging to the set of types $\mathscr{T}$ (forming the well-known hierarchy of function types), $\mathscr{D}_\tau$ is a *domain*, i.e. a set (of τ-objects) that interprets τ. Typical values $\mathscr{V}_v(X)$ are written as $\mathtt{X}$ etc., i.e. $\mathscr{V}_v(X) = \mathtt{X}$. The *evaluation rules* for $\ulcorner X \urcorner$ and $[\![X]\!]$ are:

$\mathscr{V}_v(\ulcorner X \urcorner) = X$, where X/τ (read: X stands for an object $\mathtt{X}$ of type τ, i.e. $\mathtt{X} \in \mathscr{D}_\tau$).

$\mathscr{V}_v([\![X]\!]) = \mathscr{V}_v(\mathtt{X})$, where $\mathtt{X} = \mathscr{V}_v(X)$ and (optionally) $\mathtt{X}$, $[\![X]\!]/\tau$

In other words, while X represents $\mathscr{V}_v(X)$ (i.e. $\mathtt{X}$), $\ulcorner X \urcorner$ represents X itself and $[\![X]\!]$ represents $\mathscr{V}_v(X)$. As noted in [2], employment of $\ulcorner X \urcorner$ and $[\![X]\!]$ necessiates TT with *partial functions*.

Aims of the paper. Several problems with i. and ii. have recently been observed (some of them solved) by Farmer [2, 3], Tichý and his followers (e.g. [7, 4, 5]). (We use here a partial TT called TT^* which lies between the systems in [2, 6, 7, 4].) We focus on various problems related to β-conversion rules, cf. the next section, and propose solutions to them.

2. Some problems with β-conversion of λ-abstracts containing $[\![X]\!]$

Following the ramified typing from [7], [4], $*^n$ is a type of nth-order computations X of objects $\mathtt{X}$ of various types $\tau_1, ..., \tau_m$. Let $x \in \mathscr{D}_{*^1}$ and $c \in \mathscr{D}_{*^2}$. However, $[\![c]\!]$ is *untypeable* [4], for e.g. $\mathscr{V}_{v_1}([\![c]\!]) = x$ and x/τ_1, but e.g. $\mathscr{V}_{v_2}([\![c]\!]) = y$ and y/τ_2, $\tau_1 \neq \tau_2$.

*Speaker.

[1] By Church, Andrews [1] and others ([2, 3]). $\mathscr{L}_{\mathsf{STT}}$: $\mathbf{a}$ (*constants*), x (*variables*), $Y(X)$ (*applications*), $\lambda x.Y$ (λ-*abstractions*); syncategorematic expressions: $(,), \lambda x.$ and for $\mathscr{L}_{\mathsf{STT}}$'s extensions by $\ulcorner X \urcorner$ and $[\![X]\!]$ also $\ulcorner, \urcorner, [\![,]\!]$.

Problem 1. In [7], [4], body Y of the λ-*abstract* $\lambda x.Y$ must fulfil Y/τ, i.e. $Y := \llbracket c \rrbracket$ is excluded. Hence one avoids the following failure of β-*contraction rule*

$$\beta_c \qquad [\lambda x.Y](Z) \vdash Y_{(Z/x)}$$

where $Y_{(Z/x)} = \mathscr{V}_v(\mathbf{Sub}(\ulcorner Z \urcorner, \ulcorner x \urcorner, \ulcorner Y \urcorner))$. Let $\mathscr{V}_v(x) = \mathtt{X}$, x/τ (precisely, x/τ^1, so $x \in \mathscr{D}_{*^1}$) and $\mathscr{V}_v(c) = x$ (while $c/*^1$, so $c \in \mathscr{D}_{*^2}$) and $\mathscr{V}_v(Z) = \mathtt{Z}$, Z/τ; keeping it fixed below. Thus, $\mathscr{V}_v([\lambda x.\llbracket c \rrbracket](Z)) = \mathtt{Z}$, for $\mathscr{V}_v(\lambda x.\llbracket c \rrbracket) = \mathtt{Id}$ (the identity mapping for objects of type τ), but $\mathscr{V}_v(\llbracket c \rrbracket_{(Z/x)}) = \mathtt{X}$ (where $\mathtt{X} \neq \mathtt{Z}$), for $x \notin FV(c)$ (read: x is not a free variable in c) and so $\llbracket c \rrbracket_{(Z/x)} = \llbracket c \rrbracket$.

Problem 2. The above problem with β_c (re)appears in case (2.a) with $\lambda x.(\llbracket c \rrbracket = x)$ which is *typeable* according to [7], [4]; and in case (2.b) with $\llbracket X \rrbracket_\tau$ which is *restricted to* τ [5] (i.e. the above optional type condition for $\llbracket X \rrbracket_\tau$ is strictly required). Observe again that x that is not present/visible in $\llbracket c \rrbracket$ is 'activated' when evaluating the λ-abstract containing $\llbracket c \rrbracket$.

Solutions (S1) – (S3).

(S1) *Novel definition of evaluation of* λ*-abstracts*. On standard evaluation rules for e.g. $\lambda x.F(x)$ etc., one considers v and assignments v' such that for each v', v' is like v except for x's value. On *(A)-approach* based on standard approach, $\llbracket c \rrbracket_\tau$ is v'-evaluated in synchronicity with $v'(x)$. Thus, $\mathscr{V}_{v^{(\prime)}}(\lambda x.(\llbracket c \rrbracket_\tau = x))$ is the function which maps all objects from the range of x to $\mathtt{True}$ on any $v^{(\prime)}$; $\mathscr{V}_v(\lambda x.\llbracket c \rrbracket_\tau) = \mathtt{Id}$ as assumed above. But on *(B)-approach*, that synchronicity is broken, for one evaluates $\llbracket c \rrbracket_\tau$ w.r.t. v only. Thus, $\mathscr{V}_{v^{(\prime)}}(\lambda x.(\llbracket c \rrbracket_\tau = x))$ is a function whose values $\mathtt{True}$ and $\mathtt{False}$ vary; $\mathscr{V}_v(\lambda x.\llbracket c \rrbracket_\tau)$ is a constant mapping, not $\mathtt{Id}$. Works fine, perhaps not entirely intuitive.

(S2) *Deep substitution*: substitution is repeated and recursively changes variables obtained through the process of evaluation of X; it accommodates (A)-approach. Details complicated; not entirely intuitive either.

(S3) *Elimination of* $\llbracket X \rrbracket_\tau$ while one achieves the same effect by evaluation functions-as-mappings $Eval_\tau(\cdot)$. Inherently with (B)-approach; problems not known yet.

References

[1] Andrews, Peter B. (1986): *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof*, Academic Press.

[2] Farmer, William M. (2016): Incorporating Quotation and Evaluation into Church's Type Theory: Syntax and Semantics, In: *Intelligent Computer Mathematics. CICM 2016. LNCS, vol 9791*, M. Kohlhase, M. Johansson, B. Miller B., L. de Moura and F. Tompa (eds.), Springer, 83–98.

[3] Farmer, William M. (2017): Theory Morphisms in Church's Type Theory with Quotation and Evaluation, In: *Intelligent Computer Mathematics. CICM 2017. LNCS, vol 10383*, H. Geuvers, M. England, O. Hasan, F. Rabe and O. Teschke (eds.), Springer, 147–162.

[4] Raclavský, Jiří (2020): *Belief Attitudes, Fine-Grained Hyperintensionality and Type-Theoretic Logic*. Studies in Logic 88. College Publications.

[5] Raclavský, Jiří (2022): The Rule of Existential Generalisation and Explicit Substitution, *Logic and Logical Philosophy* 31(1), 105–141.

[6] Tichý, Pavel (1982): Foundations of Partial Type Theory. *Reports on Mathematical Logic* 14, 57–72.

[7] Tichý, Pavel (1988): *The Foundations of Frege's Logic*. Walter de Gruyter.