$\lambda\rho$-calculus*

Yuichi Komori
komori@math.s.chiba-u.ac.jp
Department of Mathematics,
Faculty of Sciences, Chiba University

Arato Cho
aratoc@g.math.s.chiba-u.ac.jp
Department of Mathematics,
Faculty of Sciences, Chiba University

In [K02], one of the authors has posed a new system $\lambda\rho$-calculus and stated without proof that the strong normalization theorem hold. We will give a proof of it. While the type assignment system TA_λ gives a natural deduction for intuitionistic implicational logic (cf. [H97]), $TA_{\lambda\rho}$ gives a natural deduction for classical implicational logic. Our system is simpler than Parigot's $\lambda\mu$-calculus (cf. [P92]).

1 The type free $\lambda\rho$-calculus

Definition 1.1 ($\lambda\rho$-terms). An infinite sequence of *λ-variables* is assumed to be given, and an infinite sequence of *ρ-variables* is assumed to be given. Then linguistic expressions called *$\lambda\rho$-terms* are defined thus:

1. each λ-variable is a $\lambda\rho$-term, called an *atom* or *atomic term*;
2. if M and N are $\lambda\rho$-term then (MN) is a $\lambda\rho$-term called an *application*;
3. if M is a $\lambda\rho$-term and a is a ρ-variable then (aM) is a $\lambda\rho$-term called an *absurd*;
4. if M is a $\lambda\rho$-term and f is a λ-variable or a ρ-variable then $(\lambda f.M)$ is a $\lambda\rho$-term called an *abstract*. (If f is a λ-variable or a ρ-variable, then it called a λ-abstract or a ρ-abstract respectively.)

λ-variables are denoted by u, v, w, x, y, z, with or without number-subscripts. ρ-variables are denoted by a, b, c, d, with or without number-subscripts. A *term-variable* means a λ-variable or a ρ-variable. Term-variables are denoted by f, g,

*This article is an abstract and details will be published elsewhere.

h, with or without number-subscripts. Distinct letters denotes distinct variables unless otherwise stated.

A term $\lambda a.M$ is sometimes denoted by $\rho a.M$ if the variable a is a ρ-variable.

Arbitrary $\lambda\rho$-terms are denoted by L, M, N, P, Q, R, S, T, with or without number-subscripts. For $\lambda\rho$-term we shall say just term.

$FV(M)$ is the set of all variables free in M. For example, $FV(\lambda xb.a(x(by))) = \{a, y\}$.

DEFINITION 1.2 ($\beta\rho$-contraction). A *$\beta\rho$-redex* is any $\lambda\rho$-term of form $(aM)N$, $(\lambda x.M)N$ or $(\lambda a.M)N$; its contractum is (aM), $[N/x]M$ or $\lambda a.([\lambda x.a(xN)/a]M)N$ respectively. The re-write rules are

$$\begin{aligned} (aM)N &\rhd_{1\beta\rho} (aM), \\ (\lambda x.M)N &\rhd_{1\beta\rho} [N/x]M, \\ (\lambda a.M)N &\rhd_{1\beta\rho} \lambda a.([\lambda x.a(xN)/a]M)N. \end{aligned}$$

If P containd a $\beta\rho$-redex-occurence $\underline{R}$ and Q is the result of replacing this by its contractum, we say P *$\beta\rho$-contracts to* Q ($P \rhd_{1\beta\rho} Q$).

The notion of *$\beta\rho$-reduction* and the notation $P \rhd_{\beta\rho} Q$ are defined as usual.

THEOREM 1.3 (Church-Rosser threorem for $\beta\rho$-reduction). *If $M \rhd_{\beta\rho} P$ and $M \rhd_{\beta\rho} Q$, then there exists T such that*

$$P \rhd_{\beta\rho} T \text{ and } Q \rhd_{\beta\rho} T.$$

Proof. Simmlar to the case of β-reduction, see [HS86]. ■

2 Assgning types to terms

DEFINITION 2.1 (Types). An infinite sequence of *type-variables* is assumed to given, distinct from the term-variables. *Types* are linguistic expressions defined thus:

1. each type-variable is a type called an *atom*;
2. if σ and τ are types then $(\sigma \to \tau)$ is a type called a *composite type*.

Type-variables are denoted by p, q, r with or without number-subscripts, and distinct letters denote distinct variables unless otherwise stated.

Aribitrary types are denoted by lower-case Greek letters except λ and ρ.

Parentheses will often (but not always) be omitted from types, and the reader should restore omitted ones in the way of association to the right.

DEFINITION 2.2 (Type-assignment). A *type-assignment* is any expression

$$M : \tau$$

where M is a $\lambda\rho$-term or a ρ-variable and τ is a type; we call M its *subject* and τ is its *predicate*.

Definition 2.3 (The system $TA_{\lambda\rho}$). $TA_{\lambda\rho}$ is a Natural Deduction system. Its formulas are type-assignments. $TA_{\lambda\rho}$ has no axioms and has four rules called $(\to E)$, $(\to I)$, $(Absurd)$ and $(Rati)$, as follows.

Deduction rules of $TA_{\lambda\rho}$:

$$\frac{P:\sigma\to\tau \quad Q:\sigma}{PQ:\tau}\ (\to E), \qquad \frac{\begin{array}{c}\overline{x:\sigma}\\ \Pi\\ P:\tau\end{array}}{\lambda x.P:\sigma\to\tau}\ (\to I),$$

$$\frac{a:\tau \quad P:\tau}{aP:\sigma}\ (Absurd), \qquad \frac{\begin{array}{c}\overline{a:\tau}\\ \Pi\\ P:\tau\end{array}}{\lambda a.P:\tau}\ (Rati).$$

Example 2.4 (Peirce's Law).

$$\dfrac{\dfrac{\overline{y:(\alpha\to\beta)\to\alpha} \quad \dfrac{\dfrac{\overline{a:\alpha} \quad \overline{x:\alpha}}{ax:\beta}\ (Absurd)}{\lambda x.ax:\alpha\to\beta}\ (\to I)}{\dfrac{y(\lambda x.ax):\alpha}{\lambda a.y(\lambda x.ax):\alpha}\ (Rati)}\ (\to E)}{\lambda ya.y(\lambda x.ax):((\alpha\to\beta)\to\alpha)\to\alpha}\ (\to I)$$

The above $TA_{\lambda\rho}$-deduction is writen in more compact style:

$$\dfrac{\dfrac{\overline{y:(\alpha\to\beta)\to\alpha} \quad \dfrac{\dfrac{\overline{a:\alpha} \quad \overline{x:\alpha}}{ax:\beta}}{\alpha\to\beta}\ \lambda x}{\dfrac{\alpha}{\alpha}\ \rho a}\ (\to E)}{((\alpha\to\beta)\to\alpha)\to\alpha}\ \lambda y$$

3 Strong Normalization Theorem for $TA_{\lambda\rho}$

$\beta\rho$-reductions of deductions of $TA_{\lambda\rho}$ correspond to $\beta\rho$-reductions of $\lambda\rho$-terms. We prove the strong normalization theorem for deductions of $TA_{\lambda\rho}$,that is, for every deduction of $TA_{\lambda\rho}$ Π, all reductions starting at Π are finite. To prove the theorem, we introduce $*$-expansion and use the strong normalization theorem for deductions of TA_{λ}.

Definition 3.1 ($\circ$-translation). For every deduction of $TA_{\lambda\rho}$, of which the last rule is *Rati*, we define $\circ$-*translation* as follows:

1. $\Pi^\circ \equiv \Pi$, where $\Pi \equiv \begin{array}{c}\Pi\\ M:\alpha\end{array}$ and α is an atomic type;

2. $\Pi^\circ \equiv \dfrac{\begin{array}{c}\overline{y:\beta}\\ (\Pi_2)^\circ\\ N:\gamma\end{array}}{\lambda y.N:\beta\to\gamma}$, where $\Pi \equiv \dfrac{\begin{array}{c}\overline{a:\beta\to\gamma}\\ \Pi_1\\ M:\beta\to\gamma\end{array}}{\lambda a.M:\beta\to\gamma}$ and

$$\Pi_2 \equiv \begin{array}{c} \dfrac{\dfrac{\dfrac{\dfrac{}{a:\gamma}\qquad \dfrac{\dfrac{}{x:\beta\to\gamma}\quad y:\beta}{xy:\gamma}}{a(xy):\delta}}{\lambda x.a(xy):(\beta\to\gamma)\to\delta}}{} \\ [\lambda x.a(xy)/a]\Pi_1 \\ \dfrac{\dfrac{[\lambda x.a(xy)/a]M:\beta\to\gamma \qquad y:\beta}{[\lambda x.a(xy)/a]My:\gamma}}{\lambda a.[\lambda x.a(xy)/a]My:\gamma} \end{array} .$$

Definition 3.2 ($*$-expansion). For every deduction of $TA_{\lambda\rho}$, we define $*$-*expansion* as follows:

1. $(x:\alpha)^* \equiv x:\alpha$;

2. $\Pi^* \equiv \dfrac{\begin{array}{c}(\Pi_1)^*\\ M^*:\alpha\to\beta\end{array}\quad \begin{array}{c}(\Pi_2)^*\\ N^*:\alpha\end{array}}{M^*N^*:\beta}$, where $\Pi \equiv \dfrac{\begin{array}{c}\Pi_1\\ M:\alpha\to\beta\end{array}\quad \begin{array}{c}\Pi_2\\ N:\alpha\end{array}}{MN:\beta}$;

3. $\Pi^* \equiv \dfrac{\begin{array}{c}\overline{x:\alpha}\\ (\Pi_1)^*\\ M^*:\beta\end{array}}{\lambda x.M^*:\alpha\to\beta}$, where $\Pi \equiv \dfrac{\begin{array}{c}\overline{x:\alpha}\\ \Pi_1\\ M:\beta\end{array}}{\lambda x.M:\alpha\to\beta}$;

4. $\Pi^* \equiv \dfrac{a:\alpha \quad \begin{array}{c}(\Pi_1)^*\\ M^*:\alpha\end{array}}{aM^*:\beta}$, where $\Pi \equiv \dfrac{a:\alpha \quad \begin{array}{c}\Pi_1\\ M:\alpha\end{array}}{aM:\beta}$;

5. $\Pi^* \equiv (\Pi_1)^\circ$, where $\Pi \equiv \dfrac{\begin{array}{c}\overline{a:\alpha}\\ \Pi_2\\ M:\alpha\end{array}}{\lambda a.M:\alpha}$ and $\Pi_1 \equiv \dfrac{\begin{array}{c}\overline{a:\alpha}\\ (\Pi_2)^*\\ M^*:\alpha\end{array}}{\lambda a.M^*:\alpha}$.

Definition 3.3 (βa-reduction). A *βa-reduction* is a $\beta\rho$-reduction which does not allow a contruction $(\lambda a.M)N \rhd_{1\beta\rho} \lambda a.([\lambda x.a(xN)/a]M)N$.

Theorem 3.4 (Strong normalization theorem for βa-reduction). *For every deduction Π of $TA_{\lambda\rho}$, all βa-reductions starting at Π are finite.*

Proof. Similar to the case of TA_λ, see [HS86]. ■

Theorem 3.5 (Strong normalization theorem for $TA_{\lambda\rho}$). *For every deduction Π of $TA_{\lambda\rho}$, all $\beta\rho$-reductions starting at Π are finite.*

Proof. We can prove that $(\Pi_1)^* \rhd_{1\beta a} (\Pi_2)^*$ if $\Pi_1 \rhd_{1\beta\rho} \Pi_2$. So we get a infinite sequence of βa-reductions from a infinite sequence of $\beta\rho$-reductions. That is, strong normalization theorem for βa-reduction leads strong normalization theorem for $TA_{\lambda\rho}$. ■

References

[H97] J. Roser Hindley. *Basic Simple Type Theory*, Vol. 42 of *Cambridge Tracts in Theoretical Computer Science*, Cambridge University Press, 1997.

[HS86] J. Roger Hindley and Jonathan P. Seldin. *Introduction to combinators and λ-calculus*, Vol. 1 of *London Mathematical Society Student Texts*, Cambridge University Press, 1986.

[P92] Michel Parigot. *$\lambda\mu$-CALCULUS: AN ALGORITHMIC INTERPRETATION OF CLASSICAL NATURAL DEDUCTION*, Lecture Notes in Computer Science 624, 190-201, 1992.

[K02] Yuichi Komori. *$\lambda\rho$-Calculus: A Natural Deduction for Classical Logic*, BULLETIN OF THE SECTION OF LOGIC, VOL. 31, No. 2, 65-70, 2002.