数理解析研究所講究録
976 巻 1997 年 70-76

A system of $\lambda\mu$-calculus proper to the implicational fragment of classical natural deduction with one conclusion

ANDOU, Yuuki*
norakuro@fujimi.hosei.ac.jp
(安東祐希)

Abstract

A modified version of lambda-mu-calculus is defined. It corresponds with the implicational fragment of the system of classical natural deduction with one conclusion. Strong normalization theorem is proved for the modified version of lambda-mu-calculus.

1 Introduction

In [1], we defined a reduction for the system of first order classical natural deduction which contains all logical symbols primitively. The reduction is the natural extension of Prawitz's one ([7] [8]) for the intuitionistic case. Concerning our reduction, we proved weak normalization theorem in [1] and Church-Rosser property in [2]. But the notations used in the proof in [2] is so much complicated. In some sense, systems of typed terms is more suitable than those of proof-figures to denote reductions and prove their properties. Could we rewrite our complicated proof in [2] to a simple one ? This question has motivated us to investigate systems of typed terms suitable to work for the theorems about reduction in the systems of classical natural deduction. In the intuitionistic case, there is a well-known correspondence between typed terms of λ-calculus and proof-figures of natural deduction called Curry-Howard isomorphism. For the classical logic, this correspondence is extended to the one between Parigot's $\lambda\mu$-calculus ([4] [5] [6]) and a system of second order classical natural deduction. However, the system of classical natural deduciton which corresponds directly with Parigot's $\lambda\mu$-calculus is slightly different from the usual one we want to investigate, since in the former system derivations are allowed to have more than one conclusion. The essence of our reduction in [1] is its treatment of

*Hosei University, Fujimi, Chiyoda-ku, Tokyo 102, Japan.

redexes followed by classical absurdity rule, which is used in the system of classical natural deduction with one conclusion. It is possible to modify Parigot's $\lambda\mu$-calculus to suitable one for our classical natural deduciton system. Such modified system of $\lambda\mu$-calculus is rather simple in comparison with Parigot's original one. Namely, it is not necessary to divide variables into λ-variables and μ-variables. Only one sort of variables is sufficient. But a simple proof of Church-Rosser property of our modified system of $\lambda\mu$-calculus has not been proved yet. In this paper, we comment a fact that has been obtained in the investigation of the modified system. That is, if the modified $\lambda\mu$-calculus is restricted to the implicational fragment (in the sense of natural deduction), it enjoys strong normalization theorem. Strong normalization theorem for our system of classical natural deduction is still a conjecture.

2 Classical natural deduction (CND$_\supset$)

CND$_\supset$is a system for classical natural deduction which contains only implication as logical connectives primitively. Formulae of **CND**$_\supset$are composed from propositional variables, the propositional constant $\perp$ for false, and a logical symbol $\supset$ for implication. A formula of the form $A \supset \perp$ is abbreviated as $\neg A$. Inference rules of **CND**$_\supset$are introduction and elimination rules for $\supset$, and classical absurdity rule.

Introduction and elimination rules for $\supset$:

$$\frac{\begin{array}{c}[A]\\ B\end{array}}{A \supset B}\,(\supset I) \qquad \frac{A \supset B \quad A}{B}\,(\supset E)$$

Classical absurdity rule:

$$\frac{\begin{array}{c}[\neg A]\\ \perp\end{array}}{A}\,(\perp_c)$$

Regularity of $(\perp_c)$. It is assumed that any assumption formula discharged by any application of $(\perp_c)$ in a derivation is the major premiss of an application of $(\supset E)$. Notice that if a derivation which does not satisfy the regularity of $(\perp_c)$, then we can easily transform it to a regular one [1].

In [1] we define our reduction rules for the system of classical natural deduction with full logical symbols, and prove its weak normalization theorem.

3 $\lambda\mu 1$-calculus

We define $\lambda\mu 1$-calculus, which is a modified version of Parigot's $\lambda\mu$-calculus ([4][5][6]). Types are formulae of $\mathbf{CND}_{\supset}$. $\lambda\mu 1$-variables x^A,y^A,... are available for each type A.

3.1 Definition (Terms)

Terms are defined inductively as follows.

- The $\lambda\mu 1$-variables x^A,... are terms of type A.
- If x is a $\lambda\mu 1$-variable of type A and u is a term of type B, then $\lambda x.u$ is a term of type $A \supset B$.
- If t and u are terms of type $A \supset B$ and A respectively, then tu is term of type B.
- If x is a $\lambda\mu 1$-variables of type $\neg A$ and u a term of type $\perp$, then $\mu x.u$ is a term of type A.

Curry-Howard isomorphism can be easily extended to the correspondence between terms of $\lambda\mu 1$-calculus and derivations of $\mathbf{CND}_{\supset}$without restriction of regularity on $(\perp_c)$. The regularity on $(\perp_c)$ corresponds with the notion μ-regular which will be defined later in this section.

3.2 Definition(μ-nice)

Let u be a term and x a $\lambda\mu 1$-variable of type $\neg A$. x is μ-nice in u if the following conditions hold:

- x is not bound in u.
- u is not x itself.
- For any occurrence of x in u, the smallest subterm of u including properly the occurrence of x is of the form xw for some term w.

3.3 Definition(μ-regular)

A term t is μ-regular; if for any subterm $\mu x.u$ of t, x is μ-nice in u.

Hereafter, we assume that all terms are μ-regular.

4 Reduction

4.1 Definition($[/^*]$)

Let u and v be terms of type C and A respectively, x a $\lambda\mu 1$-variable of type $\neg(A \supset B)$ which is μ-nice in u, and y a $\lambda\mu 1$-variable of type $\neg B$ not occurring in u nor v. Then $u[v/^*x, y]$ is the term of type C defined inductively over the construction of u as follows:

- $z[v/^*x, y] = z$ if z is a $\lambda\mu 1$-variable.
- $(\lambda z.t)[v/^*x, y] = \lambda z.(t[v/^*x, y])$
- $(st)[v/^*x, y] = (s[v/^*x, y])(t[v/^*x, y])$ if s is not x.
- $(xt)[v/^*x, y] = y(t[v/^*x, y]v)$

Notice that z is not x since x is μ-nice in u.

4.2 Definition(Reduction relations)

Basic reduction relations (denoted $\rhd_c$)are defined as follows.

- $(\lambda x.u)v \rhd_c u[v/x]$
- $(\mu x.u)v \rhd_c \mu y.(u[v/^*x, y])$, where y is a $\lambda\mu 1$-variable not occurring in u nor v.

The one-step reduction relation (denoted $\rhd_1$) is defined as the compatible closure of the basic reduction relation. The reduction relation (denoted $\rhd$) is defined as the reflexive and transitive closure of the one-step reduction relation.

4.3 Definition(Strong normalizability)

A term u is strongly normalizable (denoted $SN(u)$) if there is no infinite sequence $(u_i)_{i<\omega}$ such that $u_0 = u$ and $u_i \rhd_1 u_{i+1}$.

5 A proof of SN

In this section, we prove the storng normalization theorem of $\lambda\mu 1$-calculus or that of $\mathbf{CND}_{\supset}$.

5.1 Definition(Strongly computability)

For a term u, the predicate "u is strongly computable", denoted $SC(u)$, is defined as follows.

- For a term u of atomic type, $SC(u)$ if $SN(u)$.
- For a term u of type $A \supset B$, $SC(u)$ if for all term w of type A, $SC(w)$ imples $SC(uw)$.

5.2 Lemma

Let T be any type.

- *Every term $(xu_1 \dots u_n)$ of type T, where $SN(u_i)$ for all i, is strongly computable.*
- *For any term u of type T, $SC(u)$ implies $SN(u)$.*

5.3 Lemma

If $SC(u[v/x])$ and $SC(v)$, then $SC((\lambda x.u)v)$.

These two lemmata above are proved similarly in the case of typed λ-calculus ([3]).

5.4 Notations

Let u, v_1, and v_2 be terms of type C, A_1, and A_2 respectively. Let x_1, x_2, and y be $\lambda\mu$1-variables of type $\neg(A_1 \supset A_2 \supset B)$, $\neg(A_2 \supset B)$, and $\neg B$ respectively such that x_2 and y do not occur in u, v_1, nor v_2. We use the notation $u[v_1, v_2/^{**}x_1, y]$ to denote the term $u[v_1/^{*}x_1, x_2][v_2/^{*}x_2, y]$. Notice that the term is independent of the choice of x_2. Similarly, we use the notation $u[v_1, \dots, v_n/^{**}x_1, y]$ for the term $u[v_1/^{*}x_1, x_2] \dots [v_n/^{*}x_n, y]$. If $n = 1$, $u[v_1, \dots, v_n/^{**}x, y]$ means $u[v_1/^{*}x, y]$.

5.5 Lemma

Let $(\mu x.u)v_1 \dots v_n$ be a term of an atomic type. If $SC(u[v_1, \dots, v_n/^{**}x, y])$ and $SC(v_i)$ for each i, then $SC((\mu x.u)v_1 \dots v_n)$.

Proof. By lemma 5.2, $SC(u[v_1, \dots, v_n/^{**}x, y])$ leads $SN(u[v_1, \dots, v_n/^{**}x, y])$. It implies $SN((\mu x.u)v_1 \dots v_n)$ since $SN(v_i)$ for each i. The type of $(\mu x.u)v_1 \dots v_n$ is atomic, so we have the result. □

5.6 Notations

In the case that it is inessential that which variable has been chosen for y in the term $u[v_1, \ldots, v_n/^{**}x, y]$, we denote it $u[v_1, \ldots, v_n/^{**}x]$. We also use the notation $u[\vec{v}/^{(*)}x]$ which stands for $u[\vec{v}/x]$ or $u[\vec{v}/^{**}x]$ where $\vec{v}$ is a sequence of terms and its length is equal to 1 in the case of $u[\vec{v}/x]$.

5.7 Theorem

Let u be any term and $x_1, \ldots, x_n$ mutually distinct $\lambda\mu 1$-variables which do not bound in u. Let $\vec{v}_1, \ldots, \vec{v}_n$ be sequences of terms and $z_1, \ldots, z_n$ mutually distinct $\lambda\mu 1$-variables not occurring in u, $\vec{v}_1, \ldots,$ nor $\vec{v}_n$ such that $u[z_1/x_1] \ldots [z_n/x_n][\vec{v}_1/^{(*)}z_1] \ldots [\vec{v}_n/^{(*)}z_n]$ becomes a term. Then, $SC(\vec{v}_i)$ for each i implies $SC(u[z_1/x_1] \ldots [z_n/x_n][\vec{v}_1/^{(*)}z_1] \ldots [\vec{v}_n/^{(*)}z_n])$ where $SC(v^1, \ldots, v^l)$ means $SC(v^k)$ for all k.

Proof. By induction over the construction of u.

- $u \equiv x_i$: Trival.
- $u \equiv y$ where y is not in $x_1, \ldots, x_n$: Use lemma 5.2.
- $u \equiv ts$: By definition of strong computability.
- $u \equiv \lambda x.t$: Use lemma 5.3.
- $u \equiv \mu x.t$: Use lemma 5.5. $\square$

From the theorem and lemma 5.2, it immediately follows that every $\lambda\mu 1$-term (i.e. every deduction in $\mathbf{CND}_{\supset}$) is strongly normalizable.

References

[1] Andou Y., A normalization-procedure for the first order classical natural deduction with full logical symbols, Tsukuba J. Math. 19 (1995), 153-162.

[2] Andou Y., CR of a reduction for classical natural deduction, RIMS Kokyuroku 912 (1995), 1-21.

[3] Hindley, J. R., J. P. Seldin, Introduction to combinators and λ-calculus, (Cambridge University Press, 1986)

[4] Parigot, M., Classical proofs as programs, in: Gottlob, G., Leitsch, A., and Mundici, D., editors, Computational Logic and Proof Theory, 263-277, (Springer, Berlin, 1993),LNCS 713.

[5] Parigot, M., $\lambda\mu$-Calculus: an algorithmic interpretation of classical natural deduction, in: Voronkov, A., editor, Logic Programming and Automated Reasoning, 190-201, (Springer, Berlin, 1992),LNAI 624.

[6] Parigot, M., Storng normalization for second order classical natural deduction, in: Logic in Computer Science, 39-46, (IEEE Computer Society Press, 1993), LICS 1993.

[7] Prawitz, D., Natural deduction - A proof theoretical study, (Almqvist & Wiksell, Stokholm, 1965)

[8] Prawitz, D., Ideas and results in proof theory, in: Proceedings of the second Scandinavian logic symposium, 235-307, (North-Holland, Amsterdam, 1971)