NOVI SAD J. MATH.
VOL. 31, NO. 1, 2001, 27-32

REDUCIBILITY METHOD IN SIMPLY TYPED LAMBDA CALCULUS

Silvia Ghilezan[1], **Viktor Kunčak**[2]

Abstract. A general reducibility method for proving reduction properties of the simply typed lambda calculus is presented and sufficient conditions for its application are derived.

AMS Mathematics Subject Classification (1991): 03B40, 68N18

Key words and phrases: lambda calculus, simple types, reducibility method

1. Introduction

There has recently been a reawakening of interest in many aspects of realizability interpretation, in particular semantics of type theories for constructive reasoning and semantics of programming languages. The substantial idea of the reducibility method is to interpret types by suitable sets of lambda terms which satisfy certain realizability properties. The reducibility method, based on realizability interpretations, was introduced in [10] for proving the strong normalization property for the simply typed lambda calculus and further developed in [5] and [11] for proving the strong normalization property for polymorphic (second order) lambda calculus. There is an overiew of these proofs in [1].

In [9] and [3] the reducibility method is applied in order to characterize all strongly normalizing lambda terms in lambda calculus with intersection types. The reducibility method is also used in [2] for characterizing some special classes of (untyped) lambda terms such as strongly normalizing terms, normalizing terms, and terms having (weak) head-normal forms, by their typeability in the intersection type systems.

This work presents the reducibility method as a general framework for proving reduction properties of the simply typed lambda calculus (Section 3). The presented method leads to uniform proofs of the Church-Rosser property, the standardization property, and the strong normalization property of $\lambda\!\rightarrow$ in [4]. The simplicity and wide applicability of the method from Section 3 is a result of using simply typed lambda calculus. The basic idea of this method is inspired by the work of Koletsos and Stavrinos for lambda calculi with intersection types in [7], [8], and [6].

[1]Faculty of Engineering, University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Yugoslavia, e-mail: gsilvia@uns.ns.ac.yu

[2]Laboratory of Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, e-mail: vkuncak@mit.edu

2. Preliminary notions

First, we present some basic notions of the lambda calculus.

Definition 2.1. *The set* Λ *of lambda terms is defined by the following abstract syntax.*

$$\begin{array}{lcl} \Lambda & = & \mathtt{var} \mid \Lambda\Lambda \mid \lambda\mathtt{var}.\Lambda \\ \mathtt{var} & = & x \mid \mathtt{var}' \end{array}$$

We use $x, y, z, \ldots$ for arbitrary term variables and $M, N, P, Q, \ldots$ for arbitrary terms.

$FV(M)$ denotes the set of free variables of a term M. By $M[x:=N]$ we denote the term obtained by substituting the term N for all the free occurrences of the variable x in M, taking into account that free variables of N remain free in the term obtained.

Next we present the *simply typed lambda calculus*, $\lambda\!\rightarrow$.

Definition 2.2. *The set* **type** *of types is defined as follows.*

$$\begin{array}{lcl} \mathtt{type} & = & \mathtt{atom} \mid \mathtt{type} \rightarrow \mathtt{type} \\ \mathtt{atom} & = & \alpha \mid \mathtt{atom}' \end{array}$$

We use $\alpha, \beta, \ldots$ for arbitrary atoms and $\tau, \sigma, \ldots$ for arbitrary types.

A *type assignment* is an expression of the form $M : \varphi$, where $M \in \Lambda$ and $\varphi \in \mathtt{type}$. A *context* Γ is a set $\{x_1 : \sigma_1, \ldots, x_n : \sigma_n\}$ of type assignments with different term variables and $Dom\,\Gamma = \{x_1, \ldots, x_n\}$.

Definition 2.3. *[Type assignment system* $\lambda\!\rightarrow$*] The type assignment* $P : \varphi$ *is derivable from the context* Γ *in* $\lambda\!\rightarrow$*, notation* $\Gamma \vdash P : \varphi$*, if* $\Gamma \vdash P : \varphi$ *can be generated by the following axiom-scheme and rules.*

$$(ax) \qquad \Gamma, x : \sigma \vdash x : \sigma$$

$$(\rightarrow E) \qquad \frac{\Gamma \vdash M : \sigma \rightarrow \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash MN : \tau}$$

$$(\rightarrow I) \qquad \frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash (\lambda x.M) : \sigma \rightarrow \tau}$$

3. Reducibility method for $\lambda\!\to$

Definition 3.1. *Let $A, B \subseteq \Lambda$, then $A \to B = \{M \in \Lambda \mid \forall N \in A \quad MN \in B\}$.*

Let us define the *interpretation of types* with respect to a fixed subset $\mathcal{P} \subseteq \Lambda$ in the following way.

Definition 3.2. *The map $[\![-]\!]_{\mathcal{P}} : \mathtt{type} \to 2^{\Lambda}$ is defined by:*

1. $[\![\alpha]\!]_{\mathcal{P}} = \mathcal{P}$, *$\alpha$ is an atom;*
2. $[\![\tau \to \sigma]\!]_{\mathcal{P}} = ([\![\tau]\!]_{\mathcal{P}} \to [\![\sigma]\!]_{\mathcal{P}}) \cap \mathcal{P}$.

We write simply $[\![-]\!]$ instead of $[\![-]\!]_{\mathcal{P}}$ when there is no place for confusion. The following statement can be derived immediately from Definition 3.2.

Proposition 3.3. *$[\![\sigma]\!]_{\mathcal{P}} \subseteq \mathcal{P}$, for every type σ.*

Let us further define the *valuation of terms* $[\![-]\!]_{\rho} : \Lambda \to \Lambda$ and the relation $\models$ which connects the type interpretation and the term valuation as follows.

Definition 3.4. *Let $[\![-]\!] : \mathtt{type} \to 2^{\Lambda}$ be a type interpretation and let $\rho : \mathtt{var} \to \Lambda$ be a valuation of term variables in Λ. Then*

1. *$[\![-]\!]_{\rho} : \Lambda \to \Lambda$ is defined by*
   $[\![M]\!]_{\rho} = M[x_1 := \rho(x_1), \ldots, x_n := \rho(x_n)]$ *where* $FV(M) = \{x_1, \ldots, x_n\}$;
2. $\rho \models M : \varphi \quad$ *iff* $\quad [\![M]\!]_{\rho} \in [\![\varphi]\!]$;
3. $\rho \models \Gamma \quad$ *iff* $\quad \forall (x : \varphi) \in \Gamma \quad \rho \models x : \varphi$;
4. $\Gamma \models M : \sigma \quad$ *iff* $\quad \forall \rho \models \Gamma \quad \rho \models M : \sigma$;
5. $\rho(x := N)(x) = N, \qquad \rho(x := N)(y) = \rho(y)$ *for* $x \neq y$.

The following conditions on $\mathcal{P} \subseteq \Lambda$ are sufficient to prove that all terms typeable in $\lambda\!\to$ are contained in $\mathcal{P}$.

Definition 3.5. *Let $\mathcal{P}$ be given. Then we define:*

($\mathcal{P}1$) $(\forall \varphi \in \mathtt{type})\ \mathtt{var} \subseteq [\![\varphi]\!]$;

($\mathcal{P}2$) $(\forall \varphi \in \mathtt{type})\ (\forall N \in \mathcal{P})\ M[x := N] \in [\![\varphi]\!] \Rightarrow (\lambda x.M)N \in [\![\varphi]\!]$;

($\mathcal{P}3$) $M \in \mathcal{P} \Rightarrow \lambda x.M \in \mathcal{P}$.

Now we can prove the following *realizability property.*

Proposition 3.6. *(Soundness) If $\mathcal{P}$ satisfies ($\mathcal{P}1$), ($\mathcal{P}2$), and ($\mathcal{P}3$), then*

$$\Gamma \vdash Q : \varphi \Rightarrow \Gamma \models Q : \varphi.$$

Proof. By induction on the derivation of $\Gamma \vdash Q : \varphi$.

Case 1. The last step applied is (ax), i.e. $\Gamma, x : \varphi \vdash x : \varphi$. Then obviously $\Gamma, x : \varphi \models x : \varphi$, by Definition 3.4 (2), (3) and (4).

Case 2. The last step applied is $(\to E)$, i.e. $\Gamma \vdash M : \tau \to \varphi, \Gamma \vdash N : \tau \Rightarrow \Gamma \vdash MN : \varphi$. Then by the induction hypothesis $\Gamma \models M : \tau \to \varphi$ and $\Gamma \models N : \tau$. Let $\rho \models \Gamma$, then $[\![M]\!]_\rho \in [\![\tau \to \varphi]\!] \subseteq [\![\tau]\!] \to [\![\varphi]\!]$ and $[\![N]\!]_\rho \in [\![\tau]\!]$. Therefore $[\![MN]\!]_\rho \equiv [\![M]\!]_\rho [\![N]\!]_\rho \in [\![\varphi]\!]$.

Case 3. The last step applied is $(\to I)$, i.e. $\Gamma, x : \sigma \vdash M : \tau \Rightarrow \Gamma \vdash \lambda x.M : \sigma \to \tau$. By the induction hypothesis $\Gamma, x : \sigma \models M : \tau$. Let $\rho \models \Gamma$ and let $N \in [\![\sigma]\!]$. By the variable convention let us assume $x \notin FV(N)$ and $x \notin FV(T)$, whenever $\rho(x) \equiv T$ for $T \not\equiv x$. Then $\rho(x := N) \models \Gamma$ since $x \notin Dom\,\Gamma$ and $\rho(x := N) \models x : \sigma$ since $N \in [\![\sigma]\!]$. Therefore $\rho(x := N) \models M : \tau$, i.e. $[\![M]\!]_{\rho(x := N)} \in [\![\tau]\!]$, which means by Definition 3.4 (1) and (5) that $M[\vec{y} := \rho(\vec{y})][x := N] \in [\![\tau]\!]$. Here $\vec{y}$ are variables from $FV(M) \setminus \{x\}$ for which $\rho(y_i) \not\equiv y_i$. By $(\mathcal{P}2)$ we have $(\lambda x.M[\vec{y} := \rho(\vec{y})])N \in [\![\tau]\!]$. Then $[\![\lambda x.M]\!]_\rho N \in [\![\tau]\!]$ since $x \notin FV(\lambda x.M)$. We conclude that $[\![\lambda x.M]\!]_\rho \in [\![\sigma]\!] \to [\![\tau]\!]$ since $N \in [\![\sigma]\!]$ was arbitrary. It remains to show that $[\![\lambda x.M]\!]_\rho \in \mathcal{P}$. By $(\mathcal{P}1)$ we can take $N \equiv x$, so by repeating the previous argument it follows that $M[\vec{y} := \rho(\vec{y})] \in [\![\tau]\!] \subseteq \mathcal{P}$ by Proposition 3.3. Finally $[\![\lambda x.M]\!]_\rho \equiv \lambda x.M[\vec{y} := \rho(\vec{y})] \in \mathcal{P}$ by $(\mathcal{P}3)$. □

Proposition 3.7. *Let $\mathcal{P}$ satisfy $(\mathcal{P}1)$, $(\mathcal{P}2)$ and $(\mathcal{P}3)$. Then*

$$\Gamma \vdash M : \varphi \Rightarrow M \in \mathcal{P}.$$

Proof. Let $\Gamma \vdash M : \varphi$, then $\Gamma \models M : \varphi$ by Proposition 3.6. Let us take such a ρ that $\rho(y) \equiv y$ for all $y \in \mathbf{var}$. For every $(x : \sigma) \in \Gamma$ we have that $\rho \models x : \sigma$ since $x \in [\![\sigma]\!]$ by $(\mathcal{P}1)$. Therefore $\rho \models \Gamma$ and consequently $\rho \models M : \varphi$, which means that $M \equiv [\![M]\!]_\rho \in [\![\varphi]\!] \subseteq \mathcal{P}$. □

In order to prove that for a given $\mathcal{P} \subseteq \Lambda$ the properties $(\mathcal{P}1)$ and $(\mathcal{P}2)$ hold, we will proceed by induction on the construction of the type τ, but then we need stronger induction hypotheses. These stronger conditions actually unify the conditions for saturated and $\mathcal{P}$-saturated sets which are considered in reducibility methods in [9], [1], [2], [7], and [8].

Definition 3.8. *Let $\mathcal{P}$, $X \subseteq \Lambda$ be given. Then*

$$\mathcal{P}VAR(X) \;\; \textit{means} \;\; (\forall x \in \mathbf{var})\ (\forall n \geq 0)\ (\forall M_1, \ldots, M_n \in \mathcal{P})\ \ xM_1 \ldots M_n \in X.$$

Lemma 3.9. $\mathcal{P}VAR(\mathcal{P}) \Rightarrow (\forall \varphi \in \mathbf{type})\ \mathcal{P}VAR([\![\varphi]\!])$.

Proof. By induction on the construction of φ. Let us assume $\mathcal{P}VAR(\mathcal{P})$.

Case $\varphi \equiv \alpha$ is an atom. Since $[\![\alpha]\!] = \mathcal{P}$, the statement holds by the assumption.

Case $\varphi \equiv \tau \to \sigma$. Let $M_1, \ldots, M_n \in \mathcal{P}$. Then $xM_1 \ldots M_n \in \mathcal{P}$ by the assumption. It remains to prove that $xM_1 \ldots M_n \in [\![\tau]\!] \to [\![\sigma]\!]$ and this holds by

Definition 3.1, since for any $M_{n+1} \in [\![\tau]\!] \subseteq \mathcal{P}$, we have that $xM_1 \ldots M_n M_{n+1} \in [\![\sigma]\!]$ by the induction hypothesis. □

An immediate consequence of Lemma 3.9 is the following statement.

Corollary 3.10. $\mathcal{P}VAR(\mathcal{P}) \Rightarrow (\mathcal{P}1)$.

Proof. If $\mathcal{P}VAR(\mathcal{P})$ holds, then according to Lemma 3.9 $\mathcal{P}VAR([\![\varphi]\!])$ holds for every $\varphi \in \mathtt{type}$. Obviously, $\mathcal{P}VAR([\![\varphi]\!])$ implies that $\mathtt{var} \subseteq [\![\varphi]\!]$. |*Box*

We proceed similarly for $(\mathcal{P}2)$.

Definition 3.11. *Let $\mathcal{P} \subseteq \Lambda$ be given. Then $\mathcal{P}SAT(X)$ means*

$$(\forall M, N \in \mathcal{P})\,(\forall n \geq 0)\,(\forall M_1, \ldots, M_n \in \mathcal{P})$$
$$M[x := N]M_1 \ldots M_n \in X \;\Rightarrow\; (\lambda x.M)NM_1 \ldots M_n \in X.$$

Lemma 3.12. $\mathcal{P}SAT(\mathcal{P}) \Rightarrow (\forall \varphi \in \mathtt{type})\mathcal{P}SAT([\![\varphi]\!])$.

Proof. By induction on the construction of φ. Let us assume $\mathcal{P}SAT(\mathcal{P})$.

Case $\varphi \equiv \alpha \in \mathtt{atom}$. Since $[\![\alpha]\!] = \mathcal{P}$, the property holds by the assumption.

Case $\varphi \equiv \tau \to \sigma$. Let $M, N, M_1, \ldots, M_n \in \mathcal{P}$. Suppose

$$M[x := N]M_1 \ldots M_n \in ([\![\tau]\!] \to [\![\sigma]\!]) \cap \mathcal{P}.$$

By $\mathcal{P}SAT(\mathcal{P})$ we have that $(\lambda x.M)NM_1 \ldots M_n \in \mathcal{P}$. Let $M_{n+1} \in [\![\tau]\!]$ be arbitrary. Since $[\![\tau]\!] \subseteq \mathcal{P}$, we have that $M[x := N]M_1 \ldots M_n M_{n+1} \in [\![\sigma]\!]$. Therefore by the induction hypothesis $(\lambda x.M)NM_1 \ldots M_n M_{n+1} \in [\![\sigma]\!]$. Since M_{n+1} was arbitrary, we obtain $(\lambda x.M)NM_1 \ldots M_n \in [\![\tau]\!] \to [\![\sigma]\!]$. □

Corollary 3.13. $\mathcal{P}SAT(\mathcal{P}) \Rightarrow (\mathcal{P}2)$.

Proof. By Lemma 3.12 and by Definition 3.5 of $(\mathcal{P}2)$. □

Consequently, the conditions $\mathcal{P}VAR(\mathcal{P})$ and $\mathcal{P}SAT(\mathcal{P})$ are generalizations of $(\mathcal{P}1)$ and $(\mathcal{P}2)$, respectively.

The following statement presents the general reducibility method.

Proposition 3.14. *Let $\mathcal{P} \subseteq \Lambda$ be such that $\mathcal{P}VAR(\mathcal{P})$, $\mathcal{P}SAT(\mathcal{P})$, and $(\mathcal{P}3)$ hold. Then $\Gamma \vdash M : \varphi \Rightarrow M \in \mathcal{P}$.*

Proof. According to Proposition 3.7 and Corollaries 3.10 and 3.13. □

4. Discussion

This method is general, since a suitable choice for the subset $\mathcal{P}$ of Λ for which three properties $\mathcal{P}VAR(\mathcal{P})$, $\mathcal{P}SAT(\mathcal{P})$, and $(\mathcal{P}3)$ hold provides that a term typeable in $\lambda\!\to$ belongs to $\mathcal{P}$. In [4] it is shown that $\mathcal{P}$ satisfying the required properties can be:

- the set CR of all lambda terms having the Church-Rosser property,
- the set SN of all strongly normalizing lambda terms,
- the set ST of all lambda terms having the standardization property.

It remains to investigate whether the method presented here can be extended in order to prove more properties of the (untyped) lambda calculus using simply typed lambda calculus instead of intersection type systems from [9]. The other direction in future work can deal with the question whether this method can be applied to some other type systems as well.

Acknowledgment. The authors thank George Stavrinos for valuable remarks.

References

[1] Barendregt, H.P., Lambda calculi with types. In: Abramsky, S., Gabbay, D.M., Maibaum, T.S.E. (eds.): Handbook of Logic in Computer Science, Vol. 2. Oxford University Press, Oxford (1992), 117-309

[2] Gallier, J., Typing untyped λ-terms, or reducibility strikes again! Annals of Pure and Applied Logic 91(1998), 231-270

[3] Ghilezan, S., Strong normalization and typability with intersection types. Notre Dame Journal of Formal Logic 37(1996), 44-53

[4] Ghilezan, S., Kunčak, V., General reducibility method in lambda calculus (submitted for publication)

[5] Girard, J.-Y., Une extension de l'interprétation de Gödel à l'analyse, et son application à l'elimination des coupures dans l'analyse et la théorie des types. In: Fenstad, J.E. (ed.): Proceedings of the 2nd Scandinavian Logic Symposium. North–Holland, Amsterdam (1971), 63-92

[6] Koletsos, G., Church-Rosser theorem for typed functionals. Journal of Symbolic Logic 50(1985), 782-790

[7] Koletsos, G., Stavrinos, G., Church-Rosser theorem for conjunctive type systems. In: Kakas, A.K, Sinachopoulos, A. (eds.): Proceedings of the First Panhellenic Logic Symposium. University of Cyprus (1997), 25-37

[8] Koletsos, G., Stavrinos, G., The structure of reducibility proofs. In: Kolaitis, Ph., Koletsos, G. (eds.): Proceedings of the Second Panhellenic Logic Symposium Delphi (1999), 138-144

[9] Krivine, J.L., Lambda-calcul types et modèles. Masson, Paris (1990)

[10] Tait, W.W., Intensional interpretation of functionals of finite type I. Journal of Symbolic Logic 32(1967), 198-212

[11] Tait, W.W., A realizability interpretation of the theory of species. In: Logic Colloquium (Boston). Lecture Notes in Mathematics, Vol. 453. Springer-Verlag, Berlin (1975), 240-251