Reducibility Method for Intersection Types

Silvia Ghilezan
Faculty of Engineering, University of Novi Sad, Novi Sad, Yugoslavia
Computing Science Department, Catholic University, Nijmegen, The Netherlands
e-mail: silviagh@cs.kun.nl

Viktor Kunčak
Laboratory of Computer Science, MIT, Cambridge, USA
e-mail: vkuncak@mit.edu

Silvia Likavec
Faculty of Engineering, University of Novi Sad, Novi Sad, Yugoslavia
e-mail: likavec@uns.ns.ac.yu

February 15, 2001

Abstract

A general reducibility method for proving reduction properties of lambda terms typeable with intersection types is presented. Sufficient conditions for its application are derived. This method leads to uniform proofs of the confluence, standardization, strong normalization property and some other reduction properties of the lambda terms typeable with intersection types. The method is based on Tait's reducibility method for the proof of the strong normalization of the simply typed lambda calculus, Krivine's extension of the same method for the strong normalization of intersection type system, and Statman-Mitchell's logical relation method for the proof of the confluence of the $\beta\eta$-reduction on the simply typed lambda terms.

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 Tait [13] for proving the strong normalization property for the simply typed lambda calculus and further developed in Girard [6] and Tait [14] for proving the strong normalization property for polymorphic (second order) lambda calculus. There is an overview of these proofs in Barendregt [2].

In Mitchell [10] and [11] this method is referred to as the logical relations and it is discussed that apart from the strong normalization this method can be used for the proof of the confluence (Church-Rosser property) and other basic results of the simply typed lambda calculus. The original proof of the Church-Rosser property of the simply typed lambda calculus using logical relations and the reducibility method is due to Statman [12] and Koletsos [7].

In Krivine [9] and later in Ghilezan [5] the reducibility method is applied in order to characterize all and only the strongly normalizing lambda terms in lambda calculus with intersection types. The reducibility method is also used in Gallier [4] for characterizing some special classes of lambda terms such as strongly normalizing terms, normalizing terms, head normalizing terms, and weak head normalizing terms by their typeability in the intersection type systems. In Dezani et al. [3]

the reducibility method is applied to characterizing both the mentioned terms and their persistent versions.

This work presents the reducibility method as a general framework for proving reduction properties of lambda terms typeable with intersection types. We distinguish two different kinds of type interpretation with respect to a given set $\mathcal{P} \subseteq \Lambda$ in the absence of η-conversion. Also, we distinguish two different types of conditions which the given set $\mathcal{P} \subseteq \Lambda$ has to satisfy. By combining different type interpretations with appropriate conditions on $\mathcal{P} \subseteq \Lambda$ we build up semantics and prove soundness in both cases. The method with weaker conditions on $\mathcal{P}$ and the corresponding stronger type interpretation leads to uniform proofs of the confluence of β-reduction on typeable lambda terms, the standardization property, the termination of the leftmost reduction, the uniqueness of (β-)normal form of typeable lambda terms and some other properties of typeable lambda terms. The method with stronger conditions on $\mathcal{P}$ and the corresponding type interpretation leads to uniform proofs of the confluence of $\beta\eta$-reduction on typeable lambda terms and the uniqueness of $\beta\eta$-normal form of typeable terms. The strong normalization of typeable lambda terms can be proved by both combinations.

The paper is organized as follows. Section 2 is an overview of some basic notions of the intersection types and the type assignment system. In Section 3 we present the general method. One part of the method is applied in Section 4 to the proofs of the confluence of β-reduction on typeable lambda terms, the existence of normal form, the termination of the leftmost reduction property of typeable lambda terms, the uniqueness of normal form and the standardization property of typeable terms. Section 5 is concerned with strong normalization and the complete method. The other part of the method is applied in Section 6 to the proofs of the confluence of $\beta\eta$-reduction on typeable lambda terms and the uniqueness of $\beta\eta$-normal form of typeable lambda terms.

2 Preliminary Notions

First, we present some preliminary notions of reductions on lambda terms, such as β-reduction, head reduction, and internal reduction. These notions can be found in [1].

Definition 2.1 *The set Λ of (type-free) lambda terms is defined by the following abstract syntax.*

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

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.

The axiom of the *β-reduction* is

$$(\lambda x.M)N \to_\beta M[x:=N].$$

The transitive reflexive closure of $\to_\beta$ is denoted by $\twoheadrightarrow_\beta$. The β-equality $=_{(\beta)}$ is the symmetric closure of $\twoheadrightarrow_\beta$.

A β-redex is a term of the form $(\lambda x.M)N$. If $M \equiv \lambda x_1 \ldots x_n.(\lambda x.M_0)M_1 \ldots M_m$, $n \geq 0$, $m \geq 1$, then $(\lambda x.M_0)M_1$ is called the *head-redex* of M ([1], p.173). We write $M \to_h M'$ if M' is obtained from M by reducing the head redex of M (head reduction). We write $M \to_i N$ if M' is obtained from M by reducing a redex other than the head redex (internal reduction). We also use the transitive closures of these relations, notation $\twoheadrightarrow_h$ and $\twoheadrightarrow_i$, respectively.

The axiom of the *η-reduction* is

$$\lambda x.Mx \to_\eta M, \;\; x \notin FV(M).$$

A term of the form $\lambda x.Mx$, provided $x \notin FV(M)$, is called an *η-redex.* The η-equality $=_\eta$ is the symmetric closure of $\twoheadrightarrow_\eta$.

Next we present the *pure intersection type assignment system* which will be denoted by $\lambda\cap$.

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

$$\begin{array}{|lcl|}\hline \texttt{type} & = & \texttt{atom} \mid \texttt{type} \to \texttt{type} \mid \texttt{type} \cap \texttt{type} \\ \texttt{atom} & = & \alpha \mid \texttt{atom}' \\ \hline \end{array}$$

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 $\varphi \in \texttt{type}$. A *context* Γ is the 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 (Preorder on type**)** *(i) The relation* $\leq$ *is defined on* type *by the following axioms and rules:*

1. $\sigma \leq \sigma$
2. $\sigma \leq \tau,\ \tau \leq \rho \Rightarrow \sigma \leq \rho$
3. $\sigma \cap \tau \leq \sigma,\ \sigma \cap \tau \leq \tau$
4. $\sigma \leq \tau,\ \sigma \leq \rho \Rightarrow \sigma \leq \tau \cap \rho$
5. $\sigma \leq \sigma',\ \tau \leq \tau' \Rightarrow \sigma' \to \tau \leq \sigma \to \tau'$

(ii) The induced equivalence relation is defined by:
$\sigma \sim \tau \Leftrightarrow \sigma \leq \tau \;\&\; \tau \leq \sigma.$

Definition 2.4 (Type assignment system $\lambda\cap$) *The type assignment* $P : \varphi$ *is derivable from the context* Γ *in* $\lambda\cap$*, notation* $\Gamma \vdash P : \varphi$*, if* $\Gamma \vdash P : \varphi$ *can be generated by the following axiom-scheme and rules.*

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

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

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

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

$$(\cap I) \qquad \frac{\Gamma \vdash M : \sigma \quad \Gamma \vdash M : \tau}{\Gamma \vdash M : \sigma \cap \tau}$$

$$(\leq) \qquad \frac{\Gamma \vdash M : \sigma, \quad \sigma \leq \tau}{\Gamma \vdash M : \tau}$$

3 Reducibility Method for $\lambda\cap$

The reducibility method is a generally accepted way for proving the *strong normalization property* of various type systems such as the simply typed lambda calculus in Tait [13], the polymorphic lambda calculus in Tait [14] and Girard [6], and the pure intersection type assignment system in Krivine [9]. This method was applied for the proof of the Church-Rosser property (confluence) of the simply typed lambda calculus in Statman [12], Koltesos [7], and Mitchell [10] and [11].

The general idea of the reducibility method is to interpret types by suitable sets (saturated and stable sets in Tait [13] and Krivine [9] and admissible relations in Mitchell [10] and [11]) of lambda terms which satisfy the required property (e.g. strong normalization, confluence) and then to develop semantics in order to obtain the soundness of the type assignment. A consequence of soundness, the fact that every term typeable by a type in the type system belongs to the interpretations of that type, leads to the fact that terms typeable in the type system satisfy the required property, since the type interpretations are built up in that way.

We present the reducibility method as a general framework which leads to uniform proofs of the basic reduction properties of lambda terms typeable with intersection types. In order to develop the reducibility method we consider Λ as the *applicative structure* whose domain are lambda terms and where the application is just the application of terms. Let us define the *interpretation of types* with respect to a fixed subset $\mathcal{P} \subseteq \Lambda$ in the following way.

Definition 3.1 *Let $\mathcal{P} \subseteq \Lambda$. The map $[\![-]\!]_{\mathcal{P}} : \mathtt{type} \to 2^{\Lambda}$ is defined by:*

(I1) $[\![\alpha]\!]_{\mathcal{P}} = \mathcal{P}$, *$\alpha$ is an atom;*

(I2) $[\![\tau \cap \sigma]\!]_{\mathcal{P}} = [\![\tau]\!]_{\mathcal{P}} \cap [\![\sigma]\!]_{\mathcal{P}}$;

(I3) $[\![\tau \to \sigma]\!]_{\mathcal{P}} = [\![\tau]\!]_{\mathcal{P}} \to [\![\sigma]\!]_{\mathcal{P}} = \{M \mid \forall N \in [\![\tau]\!]_{\mathcal{P}} \quad MN \in [\![\sigma]\!]_{\mathcal{P}}\}$;

($I3^+$) $[\![\tau \to \sigma]\!]_{\mathcal{P}} = ([\![\tau]\!]_{\mathcal{P}} \to [\![\sigma]\!]_{\mathcal{P}}) \cap \mathcal{P}$.

We distinguish two different type interpretations: the *type interpretation* defined by $(I1)$, $(I2)$ and $(I3)$ will be denoted by $[\![-]\!]$, whereas the type interpretation defined by $(I1)$, $(I2)$ and $(I3^+)$ will be denoted by $[\![-]\!]_{\mathcal{P}}$ and called the *strong interpretation.*

Let us further define the *valuation of terms* $[\![-]\!]_{\rho} : \Lambda \to \Lambda$ and the *semantic satisfiability relations* $\models$ and $\models_{\mathcal{P}}$ which connects the type interpretations and the term valuations as follows.

Definition 3.2 *Let $[\![-]\!]_{(\mathcal{P})} : \mathtt{type} \to 2^{\Lambda}$ be the (strong) type interpretation for a given $\mathcal{P} \subseteq \Lambda$ 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. $\rho \models_{(\mathcal{P})} M : \varphi \quad$ *iff* $\quad [\![M]\!]_{\rho} \in [\![\varphi]\!]_{(\mathcal{P})}$;
3. $\rho \models_{(\mathcal{P})} \Gamma \quad$ *iff* $\quad (\forall (x : \varphi) \in \Gamma) \quad \rho \models_{(\mathcal{P})} x : \varphi$;
4. $\Gamma \models_{(\mathcal{P})} M : \sigma \quad$ *iff* $\quad (\forall \rho \models_{(\mathcal{P})} \Gamma) \quad \rho \models_{(\mathcal{P})} M : \sigma$;
5. $\rho[x := N](x) = N, \qquad \rho[x := N](y) = \rho(y)$ *for* $x \neq y$.

The relation between the two type interpretations and between the two satisfiability relations is given in the next lemma.

Lemma 3.3 *(i) For a given $\varphi \in \mathtt{type}$ we have that $[\![\varphi]\!]_{\mathcal{P}} \subseteq [\![\varphi]\!]$.*

(ii) $\Gamma \models_{\mathcal{P}} M : \sigma \Rightarrow \Gamma \models M : \sigma$.

(iii) $[\![\sigma]\!]_{\mathcal{P}} \subseteq \mathcal{P}$, *for every type σ.*

Proof. (i) and (ii) straightforward by Definition 3.1 and Definition 3.2.
(iii) For this property conditions $(I1)$ and $(I3^+)$ of Definition 3.1 are crucial.

Let us consider the following conditions on $\mathcal{P} \subseteq \Lambda$.

Definition 3.4 *Let $\mathcal{P} \subseteq \Lambda$ be given. Then we define:*

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

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

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

($\mathcal{P}3^+$) $Mx \in \mathcal{P} \Rightarrow M \in \mathcal{P}$.

In fact, the properties ($\mathcal{P}$3) and ($\mathcal{P}3^+$) are connected as follows, as noticed in Koletsos and Stavrinos [8].

Remark 3.5 *1. ($\mathcal{P}3^+$) $\Rightarrow$ ($\mathcal{P}$3).*
If $M \in \mathcal{P}$, then $(\lambda x.M)x \twoheadrightarrow M[x := x] \equiv M$. Hence, by ($\mathcal{P}$2) $(\lambda x.M)x \in \mathcal{P}$ and by ($\mathcal{P}3^+$) $\lambda x.M \in \mathcal{P}$.

2. ($\mathcal{P}$3) and η-equality $\Rightarrow$ ($\mathcal{P}3^+$).
If $Mx \in \mathcal{P}$ ($x \notin Fv(M)$), then by ($\mathcal{P}$3) $\lambda x.Mx \in \mathcal{P}$. Since $\lambda x.Mx =_\eta M$, it follows that $M \in \mathcal{P}$.

Therefore in the absence of η-equality it is necessary to distinguish these two conditions.

The preorder on types is interpreted as the set theoretic inclusion.

Lemma 3.6 *If $\tau \leq \sigma$ then $[\![\tau]\!]_{(\mathcal{P})} \subseteq [\![\sigma]\!]_{(\mathcal{P})}$.*

Proof. By induction on the length of the derivation of $\tau \leq \sigma$.

Now we can prove the following *realizability property*, which is referred to as the soundness or the adequacy.

Proposition 3.7 *(Soundness)*

(i) If $\mathcal{P} \subseteq \Lambda$ satisfies ($\mathcal{P}$1), ($\mathcal{P}$2), and ($\mathcal{P}$3), then

$$\Gamma \vdash Q : \varphi \Rightarrow \Gamma \models_{\mathcal{P}} Q : \varphi.$$

(ii) If $\mathcal{P} \subseteq \Lambda$ 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 $\Gamma \vdash Q : \varphi$.

(i) *Case 1.* The last step applied is (ax), i.e. $\Gamma, x : \varphi \vdash x : \varphi$. Then obviously $\Gamma, x : \varphi \models_{\mathcal{P}} x : \varphi$, by Definition 3.2 (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 $\Gamma \models_{\mathcal{P}} M : \tau \to \varphi$ and $\Gamma \models_{\mathcal{P}} N : \tau$. Let $\rho \models_{\mathcal{P}} \Gamma$, then $[\![M]\!]_\rho \in [\![\tau \to \varphi]\!]_{\mathcal{P}} \subseteq [\![\tau]\!]_{\mathcal{P}} \to [\![\varphi]\!]_{\mathcal{P}}$ and $[\![N]\!]_\rho \in [\![\tau]\!]_{\mathcal{P}}$. Therefore $[\![MN]\!]_\rho \equiv [\![M]\!]_\rho [\![N]\!]_\rho \in [\![\varphi]\!]_{\mathcal{P}}$.

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 $\Gamma, x : \sigma \models_{\mathcal{P}} M : \tau$. Let $\rho \models_{\mathcal{P}} \Gamma$ and let $N \in [\![\sigma]\!]_{\mathcal{P}}$. Then $\rho[x := N] \models_{\mathcal{P}} \Gamma$ since $x \notin Dom\,\Gamma$ and $\rho[x := N] \models_{\mathcal{P}} x : \sigma$ since $N \in [\![\sigma]\!]_{\mathcal{P}}$. Therefore $\rho[x := N] \models_{\mathcal{P}} M : \tau$, i.e. $[\![M]\!]_{\rho[x := N]} \in [\![\tau]\!]_{\mathcal{P}}$, which means by Definition 3.2 (1) and (5) that $M[\vec{y} := \rho(\vec{y})][x := N] \in [\![\tau]\!]_{\mathcal{P}}$, where $\vec{y} \subseteq FV(M) \setminus \{x\}$ for which $\rho(y_i) \not\equiv y_i$. By ($\mathcal{P}$2) we have $(\lambda x.M[\vec{y} := \rho(\vec{y})])N \in [\![\tau]\!]_{\mathcal{P}}$. Then $[\![\lambda x.M]\!]_\rho N \in [\![\tau]\!]_{\mathcal{P}}$ since $x \notin FV(\lambda x.M)$. We conclude that $[\![\lambda x.M]\!]_\rho \in [\![\sigma]\!]_{\mathcal{P}} \to [\![\tau]\!]_{\mathcal{P}}$ since $N \in [\![\sigma]\!]_{\mathcal{P}}$ 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 $M[\vec{y} := \rho(\vec{y})] \equiv [\![M]\!]_\rho \in [\![\tau]\!]_v p \subseteq \mathcal{P}$. Therefore $[\![\lambda x.M]\!]_\rho \equiv \lambda x.[\![M]\!]_\rho \in \mathcal{P}$ by ($\mathcal{P}$3).

Case 4. The last step applied is $(\cap E)$, i.e. $\Gamma \vdash M : \sigma \cap \tau \Rightarrow \Gamma \vdash M : \sigma,\ \Gamma \vdash M : \tau$. By the induction hypothesis $\Gamma \models_{\mathcal{P}} M : \sigma \cap \tau$. Let $\rho \models_{\mathcal{P}} \Gamma$. Then $[\![M]\!]_\rho \in [\![\sigma \cap \tau]\!]_{\mathcal{P}} = [\![\sigma]\!]_{\mathcal{P}} \cap [\![\tau]\!]_{\mathcal{P}}$. Therefore $[\![M]\!]_\rho \in [\![\sigma]\!]_{\mathcal{P}}$ and $[\![M]\!]_\rho \in [\![\tau]\!]_{\mathcal{P}}$ i.e. $\Gamma \models_{\mathcal{P}} M : \sigma$ and $\Gamma \models_{\mathcal{P}} M : \tau$.

Case 5. The last step applied is $(\cap I)$, i.e. $\Gamma \vdash M : \sigma,\ \Gamma \vdash M : \tau \Rightarrow \Gamma \vdash M : \sigma \cap \tau$. Then by the induction hypothesis $\Gamma \models_{\mathcal{P}} M : \sigma$ and $\Gamma \models_{\mathcal{P}} M : \tau$. Let $\rho \models_{\mathcal{P}} \Gamma$, then $[\![M]\!]_\rho \in [\![\sigma]\!]_{\mathcal{P}}$ and $[\![M]\!]_\rho \in [\![\tau]\!]_{\mathcal{P}}$. Therefore $[\![M]\!]_\rho \in [\![\sigma \cap \tau]\!]_{\mathcal{P}}$ i.e. $\Gamma \models_{\mathcal{P}} M : \sigma \cap \tau$.

Case 6. The last step applied is $(\leq)$, i.e. $\Gamma \vdash M : \sigma,\ \sigma \leq \tau \Rightarrow \Gamma \vdash M : \tau$. By the induction hypothesis $\Gamma \models_{\mathcal{P}} M : \sigma$. Let $\rho \models_{\mathcal{P}} \Gamma$, then $[\![M]\!]_\rho \in [\![\sigma]\!]_{\mathcal{P}}$. According to Lemma 3.6 $[\![\sigma]\!]_{\mathcal{P}} \subseteq [\![\tau]\!]_{\mathcal{P}}$ so it follows that $[\![M]\!]_\rho \in [\![\tau]\!]_{\mathcal{P}}$, i.e. $\Gamma \models_{\mathcal{P}} M : \tau$.

(ii) Let $\Gamma \vdash Q : \varphi$ and let $(\mathcal{P}1)$, $(\mathcal{P}2)$, and $(\mathcal{P}3^+)$ hold. Then $(\mathcal{P}3)$ holds by Remark 3.5. Therefore by (i) we have that $\Gamma \models_{\mathcal{P}} Q : \varphi$ and hence $\Gamma \models Q : \varphi$ according to Lemma 3.3(ii).

An immediate consequence of soundness is the following property.

Proposition 3.8 *(i) Let $\mathcal{P}$ satisfy $(\mathcal{P}1)$, $(\mathcal{P}2)$ and $(\mathcal{P}3)$. Then $\Gamma \vdash M : \varphi \Rightarrow M \in \mathcal{P}$.*

(ii) 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. (i) Let $\Gamma \vdash M : \varphi$, then $\Gamma \models_{\mathcal{P}} M : \varphi$ by Proposition 3.7(i). Let us take such a ρ that $\rho(y) \equiv y$ for all $y \in \mathtt{var}$. For every $(x : \sigma) \in \Gamma$ we have that $\rho \models_{\mathcal{P}} x : \sigma$ since $x \in [\![\sigma]\!]_{\mathcal{P}}$ by $(\mathcal{P}1)$. Therefore $\rho \models_{\mathcal{P}} \Gamma$ and consequently $\rho \models_{\mathcal{P}} M : \varphi$, which means that $M \equiv [\![M]\!]_\rho \in [\![\varphi]\!]_{\mathcal{P}}$. According to Lemma 3.3(iii) we have that $[\![\varphi]\!]_{\mathcal{P}} \subseteq \mathcal{P}$.

(ii) By induction on the construction of φ. The interesting case is when $\varphi \equiv \sigma \to \tau$. Take $M \in [\![\sigma \to \tau]\!]$ and show that $M \in \mathcal{P}$. By the definition of the type interpretation $[\![-]\!]$ we have that $M \in [\![\sigma]\!] \to [\![\tau]\!]$. By $(\mathcal{P}1)$ let us take $x \in [\![\sigma]\!]$. Then $Mx \in [\![\tau]\!]$. By the induction hypothesis $[\![\tau]\!] \subseteq \mathcal{P}$. Hence $Mx \in \mathcal{P}$. Then $M \in \mathcal{P}$ by $(\mathcal{P}3^+)$.

Remark 3.9 *Let us notice here that in the previous proposition the required property, which states that a typeable term belongs to $\mathcal{P}$, is provided on the one hand in (i) by the condition $(I3^+)$ of the strong type interpretation, whereas on the other hand in (ii) it is provided by the condition $(\mathcal{P}3^+)$ on the subset $\mathcal{P} \subseteq \Lambda$.*

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 which are easier to prove. These stronger conditions actually unify the conditions for saturated and $\mathcal{P}$-saturated sets which are considered in reducibility methods in Krivine [9], Barendregt [2], Gallier [4], and Koletsos and Stavrinos [8].

Definition 3.10 *Let $\mathcal{P}$, $X \subseteq \Lambda$ be given. Then*

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

Lemma 3.11 $\mathcal{P}VAR(\mathcal{P}) \ \Rightarrow \ (\forall \varphi \in \mathtt{type})\ \mathcal{P}VAR([\![\varphi]\!]_{(\mathcal{P})})$.

Proof. We prove the statement for the stronger type interpretation by induction on the construction of φ. According to Lemma 3.3(i) it immediately follows that it holds for the type interpretation as well. Let us assume $\mathcal{P}VAR(\mathcal{P})$.

Case $\varphi \equiv \alpha$ is an atom. Since $[\![\alpha]\!]_{\mathcal{P}} = \mathcal{P}$, 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 $xM_1 \ldots M_n \in [\![\tau]\!]_{\mathcal{P}} \to [\![\sigma]\!]_{\mathcal{P}}$ and this holds by Definition 3.1$(I3^+)$, since for any $M_{n+1} \in [\![\tau]\!]_{\mathcal{P}} \subseteq \mathcal{P}$, we have that $xM_1 \ldots M_n M_{n+1} \in [\![\sigma]\!]_{\mathcal{P}}$ by the induction hypothesis.

Case $\varphi \equiv \tau \cap \sigma$. Let $M_1, \ldots, M_n \in \mathcal{P}$. Then $xM_1 \ldots M_n \in \mathcal{P}$. By the induction hypothesis $xM_1 \ldots M_n \in [\![\tau]\!]_{\mathcal{P}}$ and $xM_1 \ldots M_n \in [\![\sigma]\!]_{\mathcal{P}}$. Obviously, $xM_1 \ldots M_n \in [\![\tau]\!]_{\mathcal{P}} \cap [\![\sigma]\!]_{\mathcal{P}}$.

An immediate consequence of Lemma 3.11 is the following statement.

Corollary 3.12 $\mathcal{P}VAR(\mathcal{P}) \Rightarrow (\mathcal{P}1)$.

Proof. If $\mathcal{P}VAR(\mathcal{P})$ holds, then according to Lemma 3.11 $\mathcal{P}VAR([\![\varphi]\!]_{(\mathcal{P})})$ holds for every $\varphi \in \texttt{type}$. Obviously, $\mathcal{P}VAR([\![\varphi]\!]_{(\mathcal{P})})$ implies that $\texttt{var} \subseteq [\![\varphi]\!]_{(\mathcal{P})}$.

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

Definition 3.13 *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 3.14 $\mathcal{P}SAT(\mathcal{P}) \ \Rightarrow\ (\forall \varphi \in \texttt{type})\mathcal{P}SAT([\![\varphi]\!]_{(\mathcal{P})})$.

Proof. By induction on the construction of φ. For the same reasons as in the previous lemma we proceed with the proof for the strong type interpretation. Let us assume $\mathcal{P}SAT(\mathcal{P})$.

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

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

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

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

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

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

Then $M[x := N]M_1 \ldots M_n \in [\![\tau]\!]_{\mathcal{P}}$ and $M[x := N]M_1 \ldots M_n \in [\![\sigma]\!]_{\mathcal{P}}$. By the induction hypothesis $(\lambda x.M)NM_1 \ldots M_n \in [\![\tau]\!]_{\mathcal{P}}$ and $(\lambda x.M)NM_1 \ldots M_n \in [\![\sigma]\!]_{\mathcal{P}}$, therefore $(\lambda x.M)NM_1 \ldots M_n \in [\![\tau]\!]_{\mathcal{P}} \cap [\![\sigma]\!]_{\mathcal{P}}$.

Corollary 3.15 $\mathcal{P}SAT(\mathcal{P}) \Rightarrow (\mathcal{P}2)$.

Proof. By Lemma 3.14 and by Definition 3.4 of $(\mathcal{P}2)$.

Consequently, 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 which leads to uniform proofs of various reduction properties of the lambda terms typeable in $\lambda\cap$ which will be presented in the forthcoming sections.

Proposition 3.16 *(i) Let* $\mathcal{P} \subseteq \Lambda$ *be such that* $\mathcal{P}VAR(\mathcal{P})$, $\mathcal{P}SAT(\mathcal{P})$ *and* $(\mathcal{P}3)$ *hold. Then* $\Gamma \vdash M : \varphi \Rightarrow M \in \mathcal{P}$.

(ii) 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 3.8 and Corollaries 3.12 and 3.15.

4 First Part of the Method

In this section we show that the method given in Proposition 3.16(i) is applicable when $\mathcal{P}$ is:

4.1 $\mathcal{P} = C = \{M \in \Lambda \mid \beta\text{-reduction is confluent on } M\}$,

4.2 $\mathcal{P} = N = \{M \in \Lambda \mid M \text{ is normalizing}\}$,

4.3 $\mathcal{P} = L = \{M \in \Lambda \mid \text{the leftmost reduction of M terminates}\}$,

4.4 $\mathcal{P} = U = \{M \in \Lambda \mid M \text{ has a unique } (\beta\text{-})\text{normal form}\}$, 4.5 $\mathcal{P} = ST = \{M \mid$ every reduction of M can be done in a standard way$\}$.

4.1 Confluence of $\twoheadrightarrow_\beta$ on lambda terms typeable in $\lambda\cap$

Let C be the set of all lambda terms on which β-reduction is confluent (has the Church-Rosser property). We shall prove that $CVAR(C)$, $CSAT(C)$, and $(C3)$ hold. Then the confluence of β-reduction on lambda terms typeable in $\lambda\cap$ is a direct consequence of the method presented in the previous section in Proposition 3.16(i). For the sake of simplicity in this section we write $\rightarrow$ instead of $\rightarrow_\beta$ and $\twoheadrightarrow$ instead of $\twoheadrightarrow_\beta$.

Definition 4.1

$$C = \{M \in \Lambda \mid M_1 \twoheadleftarrow M \twoheadrightarrow M_2 \Rightarrow (\exists M_3 \in \Lambda)\ \ M_1 \twoheadrightarrow M_3 \twoheadleftarrow M_2\}.$$

Lemma 4.2 $CVAR(C)$.

Proof. Let $xM_1' \dots M_n' \twoheadleftarrow xM_1 \dots M_n \twoheadrightarrow xM_1'' \dots M_n''$. The only possibility for the reductions is $M_i' \twoheadleftarrow M_i \twoheadrightarrow M_i''$ for $1 \leq i \leq n$. Since $M_i \in C$, there is M_i''' such that $M_i' \twoheadrightarrow M_i''' \twoheadleftarrow M_i''$. But then
$xM_1' \dots M_n' \twoheadrightarrow xM_1''' \dots M_n''' \twoheadleftarrow xM_1'' \dots M_n''$.

Lemma 4.3 $CSAT(C)$.

Proof. Let $M, N, M_1, \dots, M_n \in C$ and $M[x := N]M_1 \dots M_n \in C$. Let $P \equiv (\lambda x.M)NM_1 \dots M_n$ and suppose $R \twoheadleftarrow P \twoheadrightarrow S$. Depending on whether the head redex of P is reduced we consider the following cases.

Case $(\lambda x.M')N'M_1' \dots M_n' \twoheadleftarrow P \twoheadrightarrow (\lambda x.M'')N''M_1'' \dots M_n''$ with $M' \twoheadleftarrow M \twoheadrightarrow M''$, $N' \twoheadleftarrow N \twoheadrightarrow N''$, and $M_i' \twoheadleftarrow M_i \twoheadrightarrow M_i''$ for $1 \leq i \leq n$. Then $M' \twoheadrightarrow M''' \twoheadleftarrow M''$, $N' \twoheadrightarrow N''' \twoheadleftarrow N''$ and $M_i' \twoheadrightarrow M_i''' \twoheadleftarrow M_i''$ for $1 \leq i \leq n$, so $(\lambda x.M')N'M_1' \dots M_n' \twoheadrightarrow (\lambda x.M''')N'''M_1''' \dots M_n''' \twoheadleftarrow (\lambda x.M'')N''M_1'' \dots M_n''$.

Case $R \twoheadleftarrow M'[x := N']M_1' \dots M_n' \twoheadleftarrow P \twoheadrightarrow M''[x := N'']M_1'' \dots M_n'' \twoheadrightarrow S$ with $M' \twoheadleftarrow M \twoheadrightarrow M''$, $N' \twoheadleftarrow N \twoheadrightarrow N''$ and $M_i' \twoheadleftarrow M_i \twoheadrightarrow M_i''$ for $1 \leq i \leq n$. Then
$R \twoheadleftarrow M'[x := N']M_1' \dots M_n' \twoheadleftarrow M[x := N]M_1 \dots M_n \twoheadrightarrow M''[x := N'']M_1'' \dots M_n'' \twoheadrightarrow S$, so the result follows from $P \twoheadrightarrow M[x := N]M_1 \dots M_n \in C$.

Case $R \twoheadleftarrow M'[x := N']M_1' \dots M_n' \twoheadleftarrow P \twoheadrightarrow (\lambda x.M'')N''M_1'' \dots M_n''$ with $M' \twoheadleftarrow M \twoheadrightarrow M''$, $N' \twoheadleftarrow N \twoheadrightarrow N''$ and $M_i' \twoheadleftarrow M_i \twoheadrightarrow M_i''$ for $1 \leq i \leq n$. Let $M' \twoheadrightarrow M''' \twoheadleftarrow M''$, $N' \twoheadrightarrow N''' \twoheadleftarrow N''$ and $M_i' \twoheadrightarrow M_i''' \twoheadleftarrow M_i''$ for $1 \leq i \leq n$. Then
$R \twoheadleftarrow M'[x := N']M_1' \dots M_n' \twoheadleftarrow M[x := N]M_1 \dots M_n \twoheadrightarrow M'''[x := N''']M_1''' \dots M_n'''$ so by
$M[x := N]M_1 \dots M_n \in C$ there is $Z \in \Lambda$ such that $R \twoheadrightarrow Z \twoheadleftarrow M'''[x := N''']M_1''' \dots M_n'''$. But then also $R \twoheadrightarrow Z \twoheadleftarrow M'''[x := N''']M_1''' \dots M_n''' \twoheadleftarrow (\lambda x.M'')N''M_1'' \dots M_n''$.

Lemma 4.4 **$(C3)$** $M \in C \Rightarrow \lambda x.M \in C$.

Proof. Let $M \in C$. Assume $R \twoheadleftarrow \lambda x.M \twoheadrightarrow S$. Then $R \equiv \lambda x.R'$ and $S \equiv \lambda x.S'$ with $R' \twoheadleftarrow M \twoheadrightarrow S'$. Hence there is a $Z \in \Lambda$ such that $R' \twoheadrightarrow Z \twoheadleftarrow S'$. Thus $\lambda x.R' \twoheadrightarrow \lambda x.Z \twoheadleftarrow \lambda x.S'$.

Let us notice here that $(C3^+)$ cannot be proved without η-reduction.

Proposition 4.5 $\Gamma \vdash M : \tau \Rightarrow$ *β-reduction is confluent on M.*

Proof. By Proposition 3.16(i) and Lemmas 4.2, 4.3, and 4.4.

4.2 Existence of normal form in $\lambda\cap$

Let N be the set of all lambda terms that have a normal form. We shall prove that $NVAR(N)$, $NSAT(N)$, and $(N3)$ hold. Then the existence of normal forms of lambda terms typeable in $\lambda\cap$ is a direct consequence of the method presented in the previous section in Proposition 3.16(i). For the sake of simplicity in this section we write $\rightarrow$ instead of $\rightarrow_\beta$ and $\twoheadrightarrow$ instead of $\twoheadrightarrow_\beta$.

Definition 4.6

$$N = \{M \in \Lambda \mid \exists N \in \Lambda,\ M \twoheadrightarrow N,\ N \textit{ is a normal form}\}.$$

Lemma 4.7 *$NVAR(N)$.*

Proof. Let $M_1, \dots, M_n \in N$. Then there exist M_i' such that $M_i \twoheadrightarrow M_i'$ and M_i' are normal forms, $1 \leq i \leq n$. Then $xM_1' \dots M_n'$ is also a normal form. But $xM_1 \dots M_n \twoheadrightarrow xM_1' \dots M_n'$ so $xM_1 \dots M_n \in N$.

Lemma 4.8 *$NSAT(N)$.*

Proof. Let $M, N, M_1, \dots, M_n \in N$ and let $M[x := N]M_1 \dots M_n \in N$. When reducing $(\lambda x.M)NM_1, \dots, M_n$, the head redex $(\lambda x.M)N$ must be reduced at some point. Since $M, N, M_1, \dots, M_n$ have normal forms they can be reduced to their normal forms $M', N', M_1', \dots, M_n'$, respectively. Then
$(\lambda x.M)NM_1, \dots, M_n \twoheadrightarrow (\lambda x.M')N'M_1', \dots, M_n' \twoheadrightarrow M'[x := N']M_1' \dots M_n'$. But
$M[x := N]M_1 \dots M_n \twoheadrightarrow M'[x := N']M_1' \dots M_n'$ and $M[x := N]M_1 \dots M_n \in N$, therefore
$(\lambda x.M)NM_1, \dots, M_n \in N$.

Lemma 4.9 *$(N3)$.*

Proof. Let $M \in N$ and let N be its normal form. Then $\lambda x.N$ is a normal form as well, and it is a normal form of $\lambda x.M$, which means $\lambda x.M \in N$

Proposition 4.10 *$\Gamma \vdash M : \tau \Rightarrow M$ has a normal form.*

Proof. By Proposition 3.16(i) and Lemmas 4.7, 4.8, and 4.9.

4.3 Termination of the leftmost reduction in $\lambda\cap$

The definition of the leftmost reduction can be found in [1].

Definition 4.11 *A leftmost reduction is the reduction where all contractions proceed from left to right, i.e. no redex is ever contracted which is a residual of a redex to the left of one already contracted.*

Definition 4.12 *We define L by:*

$$L = \{M \in \Lambda \mid \textit{ the leftmost reduction of } M \textit{ terminates}\}.$$

Lemma 4.13 *$LVAR(L)$.*

Proof. Let $M_1, \dots, M_n \in L$. The only way to reduce $xM_1 \dots M_n$ using the leftmost reduction strategy is $xM_1 \dots M_n \twoheadrightarrow xM_1' \dots M_n'$, where $M_i \twoheadrightarrow M_i'$ are the leftmost reductions for each M_i, $1 \leq i \leq n$. Since $M_i \in L$, $1 \leq i \leq n$, it follows that all these reductions terminate, therefore the first reduction terminates as well, i.e. $xM_1 \dots M_n \in L$.

Lemma 4.14 *LSAT*(L).

Proof. Let $M, N, M_1, \ldots, M_n \in L$, $M[x := N]M_1 \ldots M_n \in L$. We have to show that $(\lambda x.M)NM_1 \ldots M_n \in L$. The leftmost reduction path of the term $(\lambda x.M)NM_1 \ldots M_n$ has to start with the reduction of the head redex $(\lambda x.M)N$. Therefore, the first step in the reduction is the following: $(\lambda x.M)NM_1 \ldots M_n \twoheadrightarrow M[x := N]M_1 \ldots M_n$. Since, $M[x := N]M_1 \ldots M_n \in L$, it follows that $(\lambda x.M)NM_1 \ldots M_n \in L$.

Lemma 4.15 $(L3)$.

Proof. If $M \in L$, then $\lambda x.M \in L$, since no new redexes are created by this lambda abstraction.

Proposition 4.16 $\Gamma \vdash M : \tau \Rightarrow$ *the leftmost reduction of* M *terminates.*

Proof. By Proposition 3.16(i) and Lemmas 4.13, 4.14, and 4.15.

4.4 Uniqueness of normal form for $\lambda\cap$

In this section we consider the set U of all the lambda terms that have a unique normal form. For this set we prove that *UVAR*(U), *USAT*(U), and $(U3)$ hold. Then a consequence of the presented method is the fact that every lambda term typeable in $\lambda\cap$ has a unique normal form. We also write $\rightarrow$ instead of $\rightarrow_\beta$ and $\twoheadrightarrow$ instead of $\twoheadrightarrow_\beta$.

Definition 4.17

$$U = \{M \in \Lambda \mid M \textit{ has a unique normal form}\}.$$

Lemma 4.18 *UVAR*(U).

Proof. Let $M_1, \ldots, M_n \in U$ and suppose that P_1 and P_2 are two different normal forms of $P \equiv xM_1 \ldots M_n$. Then $P_1 \twoheadleftarrow P \twoheadrightarrow P_2$ and P_1 and P_2 must be of the form $P_1 \equiv xM_1' \ldots M_n'$ and $P_2 \equiv xM_1'' \ldots M_n''$, where M_i' and M_i'' are normal forms of M_i, $1 \leq i \leq n$. Since $M_i \in U$, $1 \leq i \leq n$, it follows that $M_i' \equiv M_i''$. Therefore $P_1 \equiv P_2$, i.e. $P \in U$.

Lemma 4.19 *USAT*(U).

Proof. Let $M, N, M_1, \ldots, M_n \in U$ and let $M[x := N]M_1 \ldots M_n \in U$. Denote by P_1 and P_2 two different normal forms of $P \equiv (\lambda x.M)NM_1, \ldots, M_n$. Then $P_1 \twoheadleftarrow P \twoheadrightarrow P_2$, $P_1 \equiv M'[x := N']M_1' \ldots M_n'$ and $P_2 \equiv M''[x := N'']M_1'' \ldots M_n''$, where M', M'', N', N'', M_i', M_i'' are normal forms of M, N, M_i, $1 \leq i \leq n$ respectively. The head redex $(\lambda x.M)N$ must be reduced in order to obtain normal forms. But
$M'[x := N']M_1' \ldots M_n' \twoheadleftarrow M[x := M]M_1 \ldots M_n \twoheadrightarrow M''[x := N'']M_1'' \ldots M_n''$ and
$M[x := N]M_1 \ldots M_n \in U$ so it follows that $P_1 \equiv P_2$, i.e. $P \in U$.

Lemma 4.20 $(U3)$.

Proof. Let $M \in U$ and let P_1 and P_2 be two different normal forms of $P \equiv \lambda x.M$. The only way to reduce P is $P_1 \equiv \lambda x.M_1 \twoheadleftarrow \lambda x.M \twoheadrightarrow \lambda x.M_2 \equiv P_2$, where $M_1 \twoheadleftarrow M \twoheadrightarrow M_2$ and M_1 and M_2 are normal forms. But $M \in U$, hence $M_1 \equiv M_2$. Therefore $\lambda x.M_1 \equiv \lambda x.M_2$ and $\lambda x.M \in U$.

Let us notice here that $(U3^+)$ cannot be proved without η-reduction.

Proposition 4.21 $\Gamma \vdash M : \tau \Rightarrow M$ *has a unique normal form.*

Proof. By Proposition 3.16 and Lemmas 4.18, 4.19, and 4.20.

4.5 Standardization for $\lambda\cap$

The property of lambda terms that each reduction can be decomposed into head reductions followed by internal reductions (these notions are mentioned in Section 2) is refered to as the *standardization*. Let ST denote the set of all lambda terms that satisfy the standardization property. We prove $STVAR(ST)$, $STSAT(ST)$, and $(ST3)$. In this section we write $\rightarrow$ instead of $\rightarrow_\beta$ and $\twoheadrightarrow$ instead of $\twoheadrightarrow_\beta$.

Definition 4.22

$$ST = \{M \mid M \twoheadrightarrow Z \Rightarrow (\exists N \in \Lambda)\ \ M \underset{h}{\twoheadrightarrow} N \underset{i}{\twoheadrightarrow} Z\}.$$

Lemma 4.23 $STVAR(ST)$.

Proof. If $xM_1 \dots M_n \twoheadrightarrow Z$ then $xM_1 \dots M_n \twoheadrightarrow_i Z$ since the term has no head redexes.

Lemma 4.24 $STSAT(ST)$.

Proof. Let $M, N, M_1 \dots M_n \in ST$ and let $M[x := N]M_1 \dots M_n \in ST$.
Suppose $P \equiv (\lambda x.M)NM_1 \dots M_n \twoheadrightarrow Z$.

Case $Z \equiv (\lambda x.M')N'M_1' \dots M_n'$, $M \twoheadrightarrow M'$, $N \twoheadrightarrow N'$, $M_i \twoheadrightarrow M_i'$ for $1 \leq i \leq n$. Then the reduction is internal: $P \twoheadrightarrow_h P \twoheadrightarrow_i Z$.

Case $P \twoheadrightarrow (\lambda x.M')N'M_1' \dots M_n' \rightarrow M'[x := N']M_1' \dots M_n' \twoheadrightarrow Z$, $M \twoheadrightarrow M'$, $N \twoheadrightarrow N'$, and $M_i \twoheadrightarrow M_i'$ for $1 \leq i \leq n$. Then $M[x := N]M_1 \dots M_n \twoheadrightarrow M'[x := N']M_1' \dots M_n'$, and, since $M[x := N]M_1 \dots M_n \in ST$, $(\lambda x.M)NM_1 \dots M_n \twoheadrightarrow_h M[x := N]M_1 \dots M_n \twoheadrightarrow_h Z' \twoheadrightarrow_i Z$, which means that $(\lambda x.M)NM_1 \dots M_n \in ST$.

Lemma 4.25 **$(ST3)$** $M \in ST \Rightarrow \lambda x.M \in ST$.

Proof. Suppose $M \in ST$ and $\lambda x.M \twoheadrightarrow Z$. Then $Z \equiv \lambda x.M'$ with $M \twoheadrightarrow M'$, so $M \twoheadrightarrow_h N \twoheadrightarrow_i M'$. But the head redex of M is also a head redex of $\lambda x.M$ and vice versa, so $\lambda x.M \twoheadrightarrow_h \lambda x.N \twoheadrightarrow_i \lambda x.M'$, which means that $\lambda x.M \in ST$.

Again, let us notice here that $(ST3^+)$ cannot be proved without η-reduction.

Proposition 4.26 $\Gamma \vdash M : \tau \Rightarrow M \in ST$.

Proof. By Proposition 3.16(i) and Lemmas 4.23, 4.24, and 4.25.

5 Strong Normalization for $\lambda\cap$ and the Complete Method

We show that in the case of strong normalization both parts of the reducibility method Proposition 3.16 can be applied. Actually, we prove for the set SN of all strongly normalizing lambda terms SN that $SNVAR(SN)$, $SNSAT(SN)$ and both $(SN3)$ and $(SN3^+)$ hold. Then the strong normalization property of terms typeable in $\lambda\cap$ is a consequence of Proposition 3.16. We also write $\rightarrow$ instead of $\rightarrow_\beta$ and $\twoheadrightarrow$ instead of $\twoheadrightarrow_\beta$ in this section.

Definition 5.1

$$SN = \{M \in \Lambda \mid \neg(\exists M_1, M_2, \dots \in \Lambda)\ M \rightarrow M_1 \rightarrow M_2 \rightarrow \dots\}.$$

Lemma 5.2 *SNVAR(SN).*

Proof. Let $x \in \mathtt{var}$ and $M_1, \ldots, M_n \in SN$. If $xM_1 \ldots M_n \twoheadrightarrow Z$, then it must be $Z \equiv xM'_1 \ldots M'_n$. Since reductions $M_i \twoheadrightarrow M'_i$ are finite, so is $xM_1 \ldots M_n \twoheadrightarrow Z$.

Lemma 5.3 *SNSAT(SN).*

Proof. Let $M, N, M_1 \ldots M_n \in SN$ and $M[x := M]M_1 \ldots M_n \in SN$. Suppose that there is an infinite reduction of $(\lambda x.M)NM_1 \ldots M_n$. Then after a finite number of steps the head redex has to be reduced:

$$(\lambda x.M)NM_1 \ldots M_n \twoheadrightarrow (\lambda x.M')N'M'_1 \ldots M'_n \to M'[x := N']M'_1 \ldots M'_n,$$

where $M \twoheadrightarrow M', M \twoheadrightarrow N'$, and $M_i \twoheadrightarrow M'_i$ for $1 \leq i \leq n$. Hence, we have that the term $M'[x := N']M'_1 \ldots M'_n$ has an infinite reduction path, and so does the term $M[x := N]M_1 \ldots M_n$, since $M[x := N]M_1 \ldots M_n \twoheadrightarrow M'[x := N']M'_1 \ldots M'_n$. This contradicts $M[x := N]M_1 \ldots M_n \in SN$.

Lemma 5.4 **(***SN***3)** $M \in SN \Rightarrow \lambda x.M \in SN.$

Proof. If $\lambda x.M$ had an infinite reduction path, it would be of the form $\lambda x.M \to \lambda x.M_1 \to \lambda x.M_2 \to \ldots$, with an infinite reduction path $M \to M_1 \to M_2 \to \ldots$, contradicting $M \in SN$.

Lemma 5.5 **(***SN***3$^+$)** $Mx \in SN \Rightarrow M \in SN.$

Proof. Suppose that $Mx \in SN$ and $M \notin SN$. Then there exists an infinite reduction path $M \to M_1 \to M_2 \to \ldots$. As a consequence of this, $Mx \to M_1x \to M_2x \to \ldots$ is also an infinite reduction which contradicts the fact that $Mx \in SN$.

Proposition 5.6 $\Gamma \vdash M : \tau \Rightarrow M$ *is strongly normalizing.*

Proof. By Proposition 3.16(i) and Lemmas 5.2, 5.3, and 5.4, or by Proposition 3.16(ii) and Lemmas 5.2, 5.3, and 5.5.

6 Second Part of the Method

The advantage of the second part of the method presented in Proposition 3.16(ii) is in the definition of the type interpretation (Definition 3.1. It does not require any restrictions with respect to the considered $\mathcal{P} \subseteq \Lambda$. However, then we have to impose a stronger condition ($\mathcal{P}3^+$) on $\mathcal{P} \subseteq \Lambda$ in order to achieve the desired consequence that a term typeable with intersection types belongs to $\mathcal{P}$, as noticed in Remark 3.9. In this section we show that the method given in Proposition 3.16(ii) is applicable when $\mathcal{P}$ is:

6.1 $\mathcal{P} = CE = \{M \in \Lambda \mid \beta\eta\text{-reduction is confluent on } M\}$,

6.2 $\mathcal{P} = UE = \{M \in \Lambda \mid M \text{ has a unique } \beta\eta\text{-normal form}\}$.

In Koletsos and Stavrinos [8] this method is applied to prove the conservation theorem that all normalizing Λ_I terms are strongly normalizing.

6.1 Confluence of $\twoheadrightarrow_{\beta\eta}$ on lambda terms typeable in $\lambda\cap$

Let us consider the set CE of lambda terms on which the $\beta\eta$-reduction is confluent. In order to show that $\beta\eta$-reduction is confluent on all typeable lambda terms we prove $CEVAR(CE)$, $CESAT(CE)$, and ($CE3^+$) and apply Proposition 3.16(ii). This is proved in Mitchell [10], [11] and Statman [12] for lambda terms typeable by simple types.

Definition 6.1

$$CE = \{M \in \Lambda \mid P \underset{\beta\eta}{\twoheadleftarrow} M \underset{\beta\eta}{\twoheadrightarrow} R \Rightarrow \exists S \in \Lambda \ (P \underset{\beta\eta}{\twoheadrightarrow} S \underset{\beta\eta}{\twoheadleftarrow} R)\}.$$

Lemma 6.2 *CEVAR*(*CE*).

Proof. The proof is similar to the proof of Lemma 4.2.

Lemma 6.3 *CESAT*(*CE*).

Proof. The proof is similar to the proof of Lemma 4.3.

Lemma 6.4 ($CE3^+$).

Proof. First, we prove that if $P \in CE$, then $\lambda x.P \in CE$. This part of the proof is similar to the proof of Lemma 4.4. Further assume $Mx \in CE$ and $x \notin FV(M)$. Then $\lambda x.Mx \in CE$ as shown above. Finally $\lambda x.Mx \to_\eta M$, which means that $M \in CE$.

Proposition 6.5 $\Gamma \vdash M : \tau \Rightarrow$ *$\beta\eta$-reduction is confluent on M.*

Proof. By Proposition 3.16(ii) and Lemmas 6.2, 6.3, and 6.4.

6.2 Uniqueness of $\beta\eta$-normal form for $\lambda\cap$

In this section we consider the set *UE* of all lambda terms that have a unique $\beta\eta$-normal. For this set we prove that *UEVAR*(*UE*), *UESAT*(*UE*), and ($UE3^+$) hold. Then a consequence of the presented method is the fact that all typeable lambda terms have a unique $\beta\eta$-normal form.

Definition 6.6

$$UE = \{M \in \Lambda \mid M \ \text{ has a unique } \beta\eta\text{-normal form}\}.$$

Lemma 6.7 *UEVAR*(*UE*).

Proof. Similar to proof of Lemma 4.18.

Lemma 6.8 *UESAT*(*UE*).

Proof. Similar to proof of Lemma 4.19.

Lemma 6.9 ($UE3^+$).

Proof. First, we prove that if $P \in UE$, then $\lambda x.P \in UE$. This part of the proof is similar to the proof of Lemma 4.20. Further assume $Mx \in UE$ and $x \notin FV(M)$. Then $\lambda x.Mx \in UE$ as shown above. Finally, $\lambda x.Mx \to_\eta M$, which means that $M \in UE$.

Proposition 6.10 $\Gamma \vdash M : \tau \Rightarrow M$ *has a unique $\beta\eta$-normal form.*

Proof. By Proposition 3.16(ii) and Lemmas 6.7, 6.8, and 6.9.

7 Concluding remarks

This method is based on the property that terms typeable in $\lambda\cap$ belong to $\mathcal{P}$, which is provided by the duality of the conditions $(I3^+)$ of the strong type interpretation and $(\mathcal{P}3^+)$ on the subset $\mathcal{P} \subseteq \Lambda$, as noticed in Remark 3.9.

Let us discuss what is the largest possible meaning of the conditions $\mathcal{P}VAR(\mathcal{P})$ and $(\mathcal{P}3)$ from the point of view of reduction properties of lambda terms. A term is said to be weakly head normalizing if it reduces to a term starting with a variable ($\mathcal{P}VAR(\mathcal{P})$) or to an abstraction (($\mathcal{P}3$)). Consequently, from the point of view of reduction properties the largest set $\mathcal{P}$ of lambda terms which satisfies the conditions $\mathcal{P}VAR(\mathcal{P})$ and $(\mathcal{P}3)$ is the set *WN* of weakly head normalizing lambda terms. According to this the method is also applicable to the set *HN* of head normalizing (solvable) lambda terms.

In the presented work we did not consider the type system with the inevitable feature related to intersection types, the universal type ω, which types every lambda term from Λ. In accordance with its role the interpretation of ω should be the whole set Λ. In order to have the condition $(I3^+)$ or $(P3^+)$ fulfilled we have to abandon the axiom $\omega \leq \omega \to \omega$ which provides the equivalence $\omega \sim \omega \to \omega$ of trivial types. Then the strong type interpretation satisfies $[\![\omega \to \omega]\!]_{WN} = WN$. If we take $\omega \leq \omega \to \omega$ into consideration then it seems that the proof does not apply to all types but to nontrivial types, types not equivalent to ω. Correspondingly, $\mathcal{P}$ cannot be *WN*, but *HN*.

The method developed here can be applied to the simply typed lambda calculus as well, since it can be seen as a restriction of $\lambda\cap$ and therefore the application of the method is straightforward. It remains to investigate whether the method presented here can be extended in order to prove more properties of lambda terms typeable with intersection types. The other direction in future work can deal with the question whether this method can be extended to some other type systems.

Acknowledgment. The authors thank George Stavrinos for valuable remarks on some previous versions of the paper.

References

[1] Barendregt, H.P.: *The lambda calculus - Its Syntax and Semantics.* North-Holland, Amsterdam (1984).

[2] 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.

[3] Dezani-Ciancaglini, M., Honsell, F., Motohama, Y.: Compositional Characterizations of λ-terms using Intersection Types. Preprint submitted to Elsevier Preprint.

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

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

[6] 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.

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

[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] Mitchell, J.C.: Type Systems for Programming Languages. In: van Leeuwen, J. (ed.): *Handbook of Theoretical Computer Science*, Vol. B. Elsevire Science Publishers B.V., (1990) 415–431.

[11] Mitchell, J.C.: *Foundation of Programming Lamguages*. MIT Press (1996).

[12] Statman, R.: Logical relations and the simply typed lambda calculus. *Information and Control* **65** (1985) 85–97.

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

[14] 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.