Basic Proof Theory

A. S. Troelstra and H. Schwichtenberg

December 27, 2019

Contents

1 Introduction

1.1 Simple type theories

Definition 1.1 (the set of simple types)**.** the set of **simple types** $\mathcal{T}_{\to}$ is constructed from a countable set of **type variables** $P_0, P_1, \dots$ by means of a type-forming operation (**function-type constructor**) $\to$

1. type variables belong to $\mathcal{T}_{\to}$
2. if $A, B \in \mathcal{T}_{\to}$, then $(A \to B) \in \mathcal{T}_{\to}$

A type of the form $A \to B$ is called a **function type**

Definition 1.2 (Terms of the simply typed lambda calculus $\lambda_{\to}$)**.** All terms appear with a type; for terms of type A we use t^A, s^A, r^A. The terms are generated by the following three clauses

1. For each $A \in T_{\to}$ there is a countably infinite supply of variables of type A; for arbitrary variables of type A we use $u^A, v^A, w^A, x^A, y^A, z^A$
2. if $t^{A\to B}, s^A$ are terms, then App$(t^{A\to B}, s^A)^B$ is a term of type B
3. if t^B is a term of type B and x^A a variable of type A, then $(\lambda x^A.t^B)^{A\to B}$

For App$(t^{A\to B}, s^A)^B$ we usually write simply $(t^{A\to B}s^A)^B$

Definition 1.3. The set FV(t) of variables free in t is specified by

$$\begin{aligned}
\mathrm{FV}(x^A) &:= x^A \\
\mathrm{FV}(ts) &:= \mathrm{FV}(t) \cup \mathrm{FV}(s) \\
\mathrm{FV}(\lambda x.t) &:= \mathrm{FV}(t) \backslash \{x\}
\end{aligned}$$

Definition 1.4 (Substitution)**.** The operation of substitution of a term s for a variable x in a term t (notation $t[x/s]$) may be defined by recursion on the complexity of t, as follows

$$\begin{aligned}
x[x/s] &:= s \\
y[x/s] &:= y \text{ for } y \not\equiv x \\
(t_1 t_2)[x/s] &:= t_1[x/s] t_2[x/s] \\
(\lambda x.t)[x/s] &:= \lambda x.t \\
(\lambda y.t)[x/s] &= \lambda y.t[x/s] \text{ for } y \not\equiv x; \text{ w.l.o.g. } y \notin \mathrm{FV}(s)
\end{aligned}$$

Lemma 1.5 (Substitution lemma)**.** *If $x \not\equiv y, x \notin FV(t_2)$, then*

$$t[x/t_1][y/t_2] \equiv t[y/t_2][x/t_1[y/t_2]]$$

Definition 1.6 (Conversion, reduction, normal form)**.** Let T be a set of terms, and let conv be a binary relation on T, written in infix notation: t conv s. If t conv s, we say that t **converts to** s; t is called a **redex** or **convertible** term and s the **conversum** of t. The replacement of a redex by its conversum is called a **conversion**. We write $t \succ_1 s$ (t **reduces in one step to** s) if s is obtained from t by replacement of a redex t' of t by a conversum t'' of t'. The relation $\succ$ (**properly reduces to**) is the transitive closure of $\succ_1$ and $\succeq$ (**reduces to**) is the reflexive and transitive closure of $\succ_1$. The relation $\succeq$ is said to be the notion of reduction **generated** by cont.

With the notion of reduction generated by cony we associate a relation on T called **conversion equality**: $t =_{\text{conv}} s$ (t is equal by conversion to s) if there is a sequence $t_0, \ldots, t_n$ with $t_0 \equiv t, t_n \equiv s$, and $t_i \preceq t_{i+1}$ or $t_i \succeq t_{i+1}$ for each $i, 0 \leq i < n$. The subscript "conv" is usually omitted when clear from the context

A term t is in **normal form**, or t is **normal**, if t does not contain a redex. t **has a normal form** if there is a normal s such that $t \succeq s$.

A **reduction sequence** is a (finite or infinite) sequence of pairs $(t_0, \delta_0), (t_1, \delta_1), \ldots$ with δ_i an (occurrence of a) redex in t_i and $t_i \succ t_{i+1}$ by conversion of δ_i, for all i. This may be written as

$$t_0 \overset{\delta_0}{\succ}_1 t_1 \overset{\delta_1}{\succ}_1 t_2 \overset{\delta_2}{\succ}_1 \ldots$$

We often omit the δ_i, simply writing $t_0 \succ_1 t_1 \succ_1 t_2$

Finite reduction sequences are partially ordered under the initial part relation ("sequence σ is an initial part of sequence τ"); the collection of finite reduction sequences starting from a term g forms a tree, the **reduction tree** of t. The branches of this tree may be identified with the collection of all infinite and all terminating finite reduction sequences.

A term is **strongly normalizing** (is SN) if its reduction tree is finite

β-conversion:

$$(\lambda x^A.t^B)s^A \ \text{cont}_\beta \ t^B[x^A/s^A]$$

η-conversion:

$$\lambda x^A.tx \ \text{cont}_\eta \ t \quad (x \notin \text{FV}(t))$$

$\beta\eta$-conversion $\text{cont}_{\beta\eta}$ is $\text{cont}_\beta \cup \text{cont}_\eta$

Definition 1.7. A relation R is said to be **confluent**, or to have the **Church-Rosser property** (CR), if whenever $t_0 R t_1$ and $t_0 R t_2$, then there is a t_3 s.t. $t_1 R t_3$ and $t_2 R t_3$. A relation R is said to be **weakly confluent** or to have the **weak Church-Rosser property** if whenever $t_0 R t_1, t_0 R t_2$ there is a t_3 s.t. $t_1 R^* t_3$ and $t_2 R^* t_3$ where R^* is the reflexive and transitive closure of T

Theorem 1.8. *For a confluent reduction relation $\succeq$ the normal forms of terms are unique. Furthermore, if $\succeq$ is a confluent reduction relation we have $t = t'$ iff there is a term t'' s.t. $t \succ t''$ and $t' \succ t''$*

Theorem 1.9 (Newman's lemma). *Let $\succeq$ be the transitive and reflexive closure of $\succ_1$, and let $\succ_1$ be weakly confluent. Then the normal form w.r.t. $\succ_1$ of a strongly normalizing t is unique. Moreover, if all terms are strongly normalizing w.r.t. $\succ_1$ then the relation $\succeq$ is confluent.*

Proof. Assume WCR, and let write $s \in UN$ to indicate that s has a unique normal form. Assume $t \in SN, t \notin UN$. Then there are two reduction sequences $t \succ_1 t'_1 \cdots \succ_1 t'$ and $t \succ_1 T''_1 \succ_1 \cdots \succ_1 t''$ with $t' \not\equiv t''$. Then either $t'_1 = t''_1$ or $t'_1 \neq t''_1$

In the first case we can take $t_1 := t'_1 = t''_1$. In the second case, by WCR we can find a t^* s.t. $t^* \prec t'_1, t''_1$; $t \in SN$ hence $t^* \succ t'''$ for some normal t'''. Since $t' \neq t'''$ or $t'' \neq t'''$, either $t'_1 \notin UN$ or $t''_1 \notin UN$; so take $t_1 := t'_1$ if $t' \neq t'''$, $t_1 := t''_1$ otherwise.

Hence we can always find a $t_1 \prec t$ with $t_1 \notin UN$ and get an infinite sequence contradicting the SN of t □

Definition 1.10. The **simple typed lambda calculus** $\lambda_\to$ is the calculus of β-reduction and β-equality on the set of terms of $\lambda_\to$. $\lambda_\to$ has the term system as described with the following axioms and rules for $\prec$ ($\prec_\beta$) and $=$ (is $=_\beta$)

$$t \succeq t \quad (\lambda x^A.t^B)s^A \succeq t^B[x^A/s^A]$$
$$\frac{t \succeq s}{rt \succeq rs} \quad \frac{t \succ s}{tr \succ sr} \quad \frac{t \succeq s}{\lambda x.t \succeq \lambda x.s} \quad \frac{t \succeq s \quad s \succeq r}{t \succeq r}$$
$$\frac{t \succeq s}{t = s} \quad \frac{t = s}{s = t} \quad \frac{t = s \quad s = r}{t = r}$$

The **extensional simple typed lambda calculus** $\lambda\eta_\to$ is the calculus of $\beta\eta$-reduction and $\beta\eta$-equality and the ser of terms of $\lambda_\to$; in addition there is the axiom

$$\lambda x.tx \succeq t \quad (x \notin \mathrm{FV}(t))$$

Lemma 1.11 (Substitutivity of $\succ_\beta$ and $\succ_{\beta\eta}$). *For $\succeq$ either $\succeq_\beta$ or $\succ_{\beta\eta}$ we have*

$$\text{if } s \succeq s' \text{ then } s[y/s''] \succeq s'[y/s'']$$

Proof. By induction on the depth of a proof of $s \succeq s'$. It suffices to check the crucial basis step, where s is $(\lambda x.t)t'$ and s' is $t[x/t']$.

$$(\lambda x.t)t'[y/s''] = (\lambda x.(t[y/s''])t'[y/s'']) = t[y/s''][x/t'[y/s'']] = t[x/t'][y/s'']$$

□

Proposition 1.12. *$\succ_{\beta,1}$ and $\succ_{\beta\eta,1}$ are weakly confluent*

Proof. If the conversions leading from t to t' and t to t'' concern disjoint redexes, then t''' is simply obtained by converting both redexes

If $t \equiv \ldots (\lambda x.s)s' \ldots$, $t' \equiv \ldots s[x/s'] \ldots$ and $t'' \equiv \ldots (\lambda x.s)s'' \ldots$, $s' \succ_1 s''$, then $t''' \equiv \ldots s[x/s''] \ldots$

If $t \equiv \ldots (\lambda x.s)s' \ldots$, $t' \equiv \ldots s[x/s'] \ldots$ and $t'' \equiv \ldots (\lambda x.s'')s' \ldots$, $s \succ_1 s''$, then $t''' \equiv \ldots s''[x/s'] \ldots$

If $t \equiv \ldots (\lambda x.sx)s'$, $t' = \ldots (sx)[x/s'] \ldots$, $t'' = \ldots ss' \ldots$ □

Theorem 1.13. *The terms of $\lambda_\to, \lambda\beta_\to$ are SN for $\succeq_\beta$ and $\succeq_{\beta\eta}$ respectively, then hence the β- and $\beta\eta$-normal forms are unique*

Definition 1.14. $\succeq_p$ on $\lambda_\to$ is generated by the axiom and rules

$$(\mathrm{id}) x \succeq_p x$$

$$(\lambda\mathrm{mon})\frac{t \succeq_p t'}{\lambda x.t \succeq_p \lambda x.t'} \qquad (\mathrm{appmon})\frac{t \succeq_p t' \quad s \succeq_p s'}{ts \succeq_p t's'}$$

$$(\beta\mathrm{par})\frac{t \succeq_p t' \quad s \succeq_p s'}{(\lambda x.t)s \succeq_p t'[x/s']}(\eta\mathrm{par})\frac{t \succeq_p t'}{\lambda x.tx \succeq_p t'}(x \notin \mathrm{FV}(t))$$

Lemma 1.15 (Substitutivity of $\succ_p$)**.** *If $t \succ_p t'$, $s \succ_p s'$, then $t[x/s] \succ_p t'[x/s']$*

Proof. By induction on t.

1. $t \equiv (\lambda y.t_1)t_2$, then

$$t \succeq_p t_1'[y/t_2']$$

$$t[x/s] \equiv (\lambda y.t_1[x/s])t_2[x/s] \succeq_p t_1'[x/s'][y/t_2'[x/s']] \succeq_p t_1'[y/t_2'][x/s']$$

□

Lemma 1.16. *$\succeq_p$ is confluent*

Proof. Induction on t □

Theorem 1.17. *β- and $\beta\eta$-reduction are confluent*

Proof. The reflexive closure of $\succ_1$ for $\beta\eta$-reduction is contained in $\succeq_p$, and $\succeq$ is therefore the transitive closure of $\succeq_p$. Write $t \succeq_{p,n} t'$ if there is a chain $t \equiv t_0 \succeq_p t_1 \succeq_p \cdots \succeq_p t_n \equiv t'$. Then we show by induction on $n + m$ using the preceding lemma, that if $t \succeq_{p,n} t', t \succeq_{p,m} t''$ then there is a t''' s.t. $t' \succeq_{p,m} t''', t'' \succeq_{p,n} t'''$

$$
\begin{tikzcd}
t \arrow[r, "\alpha-1"] \arrow[dr, "n+m+1-\alpha"'] & t'_0 \arrow[r, "1"] \arrow[dr] & t' \arrow[dr, "n+m+1-\alpha"] & \\
& t'' \arrow[r, "\alpha-1"] & t'''_0 \arrow[r] & t'''
\end{tikzcd}
$$

$\square$

Definition 1.18 (Terms of typed combinatory logic $\mathbf{CL}_{\to}$)**.** The terms are inductive defined as in the case of $\lambda_{\to}$, but now with the clauses

1. For each $A \in \mathcal{T}_{\to}$ there is a countably infinite supply of variables of type A; for arbitrary variables of type A we use $u^A, v^A, w^A, x^A, y^A, z^A$
2. for each $A, B, C \in \mathcal{T}$ there are constant terms

$$
\begin{aligned}
&\boldsymbol{k}^{A,B} \in A \to (B \to A) \\
&\boldsymbol{s}^{A,B,C} \in (A \to (B \to C)) \to ((A \to B) \to (A \to C))
\end{aligned}
$$

3. if $t^{A,B}, s^A$ are terms, then so is $t^{A,B}s$

$\mathrm{FV}(\boldsymbol{k}) = \mathrm{FV}(\boldsymbol{s}) = \emptyset$

Definition 1.19. The **weak reduction** relation $\succeq_w$ on the terms of $\mathbf{CL}_{\to}$ is generated by a conversion relation cont_w consisting of the following pairs

$$\boldsymbol{k}^{A,B}x^Ay^B \ \ \mathrm{cont}_w \ \ x, \quad \boldsymbol{s}^{A,B,C}x^{A\to(B\to C)}y^{A\to B}z^A \ \ \mathrm{cont}_w \ \ xz(yz)$$

In otherwords, $\mathbf{CL}_{\to}$ is the term system defined above with the following axioms and rules for $\succeq_w$ and $=_w$

$$
\begin{array}{lll}
t \succeq t & \boldsymbol{k}xy \succeq x & \boldsymbol{s}xyz \succeq xz(yz) \\
\dfrac{t \succeq s}{rt \succeq rs} & \dfrac{t \succeq s}{tr \succeq sr} & \dfrac{t \succeq s \quad s \succeq r}{t \succeq r} \\
\dfrac{t \succeq s}{t = s} & \dfrac{t = s}{s = t} & \dfrac{t = s \quad s = r}{t = r}
\end{array}
$$

Theorem 1.20. *The weak reduction relation in $\mathbf{CL}_{\to}$, is confluent and strongly normalizing, so normal forms are unique.*

Theorem 1.21. *To each term t in $\mathbf{CL}_{\to}$, there is another term $\lambda^* x^A.t$ such that*

1. $x^A \notin FV(\lambda^* x^A.t)$
2. $(\lambda^* x^A.t)s^A \succ_w t[x^A/s^A]$

Proof.

$$\lambda^* x^A.x := \boldsymbol{s}^{A,A\to A,A}\boldsymbol{k}^{A,A\to A}\boldsymbol{k}^{A,A}$$
$$\lambda^* x^A.y^B := \boldsymbol{k}^{B,A}y^B \text{ for } y \not\equiv x$$
$$\lambda^* x^A.t_1^{B\to C}t_2^B := \boldsymbol{s}^{A,B,C}(\lambda^* x.t_1)(\lambda^* x.t_2)$$

□

Corollary 1.22. ***CL***$_\to$ *is **combinatorially complete**, i.e. for every applicative combination t of $\boldsymbol{k}, \boldsymbol{s}$ and variables $x_1, x_2, \ldots x_n$ there is a closed term s s.t. in* ***CL***$_\to \vdash sx_1 \ldots x_n =_w t$*, in fact even* ***CL***$_\to \vdash sx_1 \ldots x_n \succeq_w t$

Remark. Note that: it's not true that if $t = t'$ then $\lambda^* x.t = \lambda^* x.t'$. $\boldsymbol{k}x\boldsymbol{k} = x$ but $\lambda^* x.\boldsymbol{k}x\boldsymbol{k} = \boldsymbol{s}(\boldsymbol{s}(\boldsymbol{kk})(\boldsymbol{skk}))(\boldsymbol{kk})$, $\lambda^* x.x = \boldsymbol{skk}$

Definition 1.23. The **Church numerals** of type A are β-normal terms $\bar{n}_A$ of type $(A \to A) \to (A \to A), n \in \mathbb{N}$, defined by

$$\bar{n}_A := \lambda f^{A\to A}\lambda x^A.f^n(x)$$

where $f^0(x) := x, f^{n+1}(x) := f(f^n(x))$. $N_A = \{\bar{n}_A\}$

N.B. If we want to use $\beta\eta$-normal terms, we must use $\lambda f^{A\to A}.f$ instead of $\lambda fx.fx$ for $\bar{1}_A$

Definition 1.24. A function ﬀ$f : \mathbb{N}^k \to \mathbb{N}$ is said to be **A-representable** if there is a term F of $\lambda_\to$ s.t. (abbreviating $\bar{n}_A$ as $\bar{n}$)

$$F\bar{n}_1 \ldots \bar{n}_k = \overline{f(n_1, \ldots, n_k)}$$

for all $n_1, \ldots, n_k \in \mathbb{N}, \bar{n}_i = (\bar{n}_i)_A$

Definition 1.25. Polynomials, extended polynomials

1. The n-argument **projections** $\boldsymbol{p}_i^n$ are given by $\boldsymbol{p}_i^n(x_1, \ldots, x_n) = x_i$, the unary constant functions $\boldsymbol{c}_m$ by $\boldsymbol{c}_m(x) = m$, and sg, $\bar{sg}$ are unary functions which satisfy $\text{sg}(S_n) = 1, \text{sg}(0) = 0$, where S is the successor function.
2. The n-argument function f is the **composition** of m-argument g, n-argument $h_1, \ldots, h_m$ if f satisfies $f(\bar{x}) = g(h_1(\bar{x}), \ldots, h_m(\bar{x}))$
3. The **polynomials** in n variables are generated from $\boldsymbol{p}_i^n, \boldsymbol{c}_m$, addition and multiplication by closure under composition. The **extended polynomials** are generated from $\boldsymbol{p}_i^n, \boldsymbol{c}_m$, sg, $\bar{sg}$, addition and multiplication by closure under proposition

Exercise 1.1.1. Show that all terms in β-normal form of type $(P \to P) \to (P \to P)$, P a propositional variable, are either of the form $\bar{n}_P$ or of the form $\lambda f^{P \to P}.f$

Proof. 1. $\lambda f^{P \to P}.g^{P \to P}$, if $g \neq f$, then g is of the form $\lambda x^P.y^P$ and hence $\lambda f^{P \to P} \lambda x^P.y^P$

$\square$