The relation between Combinatory Logic and λ-calculus

In combinatory logic one can define abstraction operations that satisfy the β-scheme; such abstraction operations determine translations of the λ-calculus (***λβ*** or ***λβη***) into combinatory logic. Conversely the combinators correspond with λ-terms, so that *CL* lies embedded in ***λβ***. Because the combinators of the λ-calculus reduce stepwise, for instance, $\mathsf{K}x \longrightarrow \lambda y.x$ whereas ***K****x* is not a redex, and, more importantly, because λ-abstraction is *an operation* of the λ-calculus, these two translations do not produce a complete agreement between *CL* and ***λβ***. Generally speaking, combinatory logic is weaker. Curry [CL] conceived additional axioms that close the gap: a finite number (five, to be precise) of ground equations. (*Ground* means *variable-free*.)

Barendregt sketches in [LC] another, more direct approach to the problem. It has inspired the present account.

Let *P* be a term of *CL* (a *combinatory term*), and *M* a λ-term. We denote by V(*P*) the set of variables occurring in *P*, and by FV(*M*) the set of variables occurring free in *M*. If V(*P*) = ∅, we call *P* a *ground term*. We abbreviate ***SKK*** to ***I***, and note that $\boldsymbol{I}x \twoheadrightarrow_{\mathrm{w}} x$.

We use ≈ for formal equality and, informally, for provable equality (in a theory determined by the context); and for convertibility, where sometimes rules will be indicated by subscripts.

The deductive systems

In all the theories considered here, there is some syntax defining *terms*, including *application*, represented by juxtaposition; and terms **s**, **t** can be combined in *equations* **s** ≈ **t**. All theories contain the schemes

$\mathbf{t} \approx \mathbf{t}$ (identity),

$\mathbf{s} \approx \mathbf{t} \Rightarrow \mathbf{t} \approx \mathbf{s}$ (symmetry),

$\mathbf{r} \approx \mathbf{s}\ \&\ \mathbf{s} \approx \mathbf{t} \Rightarrow \mathbf{r} \approx \mathbf{t}$ (transitivity),

$\mathbf{r} \approx \mathbf{s}\ \&\ \mathbf{t} \approx \mathbf{u} \Rightarrow \mathbf{rt} \approx \mathbf{su}$ (application);

beyond this, in ***λβ*** we have $(\lambda x.M)N \approx [N/x]M$ (β-contraction) and the ξ-rule $M \approx N \Rightarrow \lambda x.M \approx \lambda x.N$, and additionally in ***λβη***: $\lambda x.Mx \approx M$, where $x \notin \mathrm{FV}(M)$ (η-contraction). Change of bound variables in λ-terms is considered part of the syntax, and in every context (in particular that of β-contraction), the bound variables are assumed to be distinct from the free. In *CL* we have the combinator schemes $\boldsymbol{KPQ} \approx P$ and $\boldsymbol{SPQR} \approx PR(QR)$.

Curry presents a list of defining clauses for abstraction in combinatory logic. Four are relevant to us:

(a)	$[x].P$	$= \boldsymbol{K}P$ if $x \notin \mathrm{V}(P)$
(b)	$[x].x$	$= \boldsymbol{I}$
(c)	$[x].Px$	$= P$ if $x \notin \mathrm{V}(P)$
(f)	$[x].PQ$	$= \boldsymbol{S}([x].P)([x].Q)$

Depending on which clauses we use, and the order in which we apply them, we get different definitions of abstraction. In particular, $\lambda^* x$ results from (abf) — so $\lambda^* x.P$ is $\boldsymbol{K}P$ if $x \notin \mathrm{V}(P)$, $\boldsymbol{I}$ if $P = x$, and $\boldsymbol{S}(\lambda^* x.P)(\lambda^* x.Q)$ otherwise —, $\lambda_1 x$ results from (bfa), and $\lambda_2 x$ from (abcf). Yet a fourth abstraction, $\lambda^+ x$, is described by Hindley and Seldin [LCCI]. Call a combinatory term P *functional* if Px, Pxy or $Pxyz$ is a redex. Consider the following clauses:

(c′) $\qquad [x].Px = P \qquad$ if $x \notin \mathrm{V}(P)$ and P is functional;

(f′) $\qquad [x].PQ = \boldsymbol{S}(\lambda_2 x.P)(\lambda_2 x.Q)$.

Then $\bar{\lambda}^{+}x$ results from (abc′f′).

Let $\mathbf{G}_C$ be the term groupoid of combinatory logic, and $\mathbf{G}_\lambda$ the groupoid of λ-terms. We define homomorphisms λ: $\mathbf{G}_C \longrightarrow \mathbf{G}_\lambda$ en κ: $\mathbf{G}_\lambda \longrightarrow \mathbf{G}_C$ as follows:

$v_\lambda = v_\kappa = v$, for any variable v;

$\boldsymbol{K}_\lambda = \mathsf{K}$, $\boldsymbol{S}_\lambda = \mathsf{S}$;

$(\lambda x.M)_\kappa = [x].M_\kappa$.

Actually, the precise nature of κ depends on the details of abstraction; we might distinguish κ^*, κ_1, κ_2, and κ^+, corresponding with the choices $\bar{\lambda}^{*}x$, $\lambda_1 x$, $\lambda_2 x$, and $\bar{\lambda}^{+}x$. Observe that P and P_λ, and M and M_κ, contain the same variables free.

All these constructions fulfil the purpose for which they were designed:

Lemma 1 (Reduction Lemma). Suppose

$$Q \in \{\bar{\lambda}^{*}x.P, \lambda_1 x.P, \lambda_2 x.P, \bar{\lambda}^{+}x.P\}.$$

Then any combinatory term R satisfies $QR \twoheadrightarrow_{\mathrm{w}} [R/x]P$.

The abstraction λ_2 will not lead to a system that is equivalent to $\boldsymbol{\lambda\beta}$; if $y \neq x$, $\lambda_2 x.yx = y$, but the equality $\lambda x.yx \approx y$ is not valid in the $\lambda\beta$-calculus. It is *almost* valid, though:

Proposition 1. For each λ-term M, $\lambda x.(\lambda x.M)x \twoheadrightarrow_\beta \lambda x.M$.

This corresponds with the fact (to be established) that the abstraction $\bar{\lambda}^{+}$ fits $\boldsymbol{\lambda\beta}$.

Proposition 2. Let P be a combinatory term. If P is functional, then P_λ β-reduces to an abstraction.

Proof. Consider cases: P is of one of the forms $\boldsymbol{K}, \boldsymbol{K}A, \boldsymbol{S}, \boldsymbol{S}A$ or $\boldsymbol{S}AB$. ⊠

Corollary. For any combinatory term P, $(\bar{\lambda}^{+}x.P)_\lambda$ β-reduces to an abstraction.

Proof. By definition, $\bar{\lambda}^{+}x.P$ is functional. ⊠

For any set $\boldsymbol{A}$ of ground equations, let $CL + \boldsymbol{A}$ be the equational system that results from adding the axioms $\boldsymbol{A}$ to CL.

We fix (apart from $\boldsymbol{I} = \boldsymbol{SKK}$) the following abbreviations:

$\boldsymbol{F} := \boldsymbol{KI}$, so $\boldsymbol{F}x \longrightarrow \boldsymbol{I}$;

$\boldsymbol{X} := \boldsymbol{S}(\boldsymbol{KK})$, so that $\boldsymbol{X}xy \twoheadrightarrow \boldsymbol{K}(xy)$;

$\boldsymbol{Y} := \boldsymbol{S}(\boldsymbol{KS})$, hence $\boldsymbol{Y}xy \twoheadrightarrow \boldsymbol{S}(xy)$;

$\boldsymbol{U} := \boldsymbol{Y}(\boldsymbol{S}(\boldsymbol{KY})(\boldsymbol{S}(\boldsymbol{KX})\boldsymbol{S}))$, hence $\boldsymbol{U}x \twoheadrightarrow \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}(\boldsymbol{S}x)))$

and $\boldsymbol{U}xyz \twoheadrightarrow \boldsymbol{S}(\boldsymbol{K}(\boldsymbol{S}xz))(yz)$;

and $\boldsymbol{B} := \boldsymbol{YK}$, so that $\boldsymbol{B}xyz \twoheadrightarrow x(yz)$.

$\boldsymbol{A}_\beta$ is the following set of axioms:

(A.1) $\boldsymbol{K} \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XK}))\boldsymbol{F}$

(A.2) $\boldsymbol{S} \approx \boldsymbol{SU}(\boldsymbol{KF})$

(A.3) $\boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XB}))(\boldsymbol{KK}) \approx \boldsymbol{X}$

(A.4) $\boldsymbol{YX} \approx \boldsymbol{X}(\boldsymbol{SBF})$

(A.5) $\boldsymbol{S}(\boldsymbol{KY})(\boldsymbol{YY}) \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}(\boldsymbol{Y}(\boldsymbol{S}(\boldsymbol{KY})\boldsymbol{S}))))(\boldsymbol{KS})$

Lemma 2. CL + (A.2) $\vdash \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}x))\boldsymbol{F} \approx \boldsymbol{S}(\boldsymbol{K}(\boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}x))\boldsymbol{F}))\boldsymbol{I}$.

Proof: $\boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}x))\boldsymbol{F} \approx \boldsymbol{SU}(\boldsymbol{KF})(\boldsymbol{Y}(\boldsymbol{X}x))\boldsymbol{F}$ by (A.2); now normalize. ☒

Lemma 3. Suppose P is functional. Then

$$CL + (\text{A.1}, 2) \vdash P \approx \boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I}.$$

Proof: There are five cases: P is of one of the five forms $\boldsymbol{K}, \boldsymbol{K}A, \boldsymbol{S}, \boldsymbol{S}A$ or $\boldsymbol{S}AB$.

(i) $\boldsymbol{K} \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XK}))\boldsymbol{F}$ by (A.1)
$\approx \boldsymbol{S}(\boldsymbol{K}(\boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XK}))\boldsymbol{F}))\boldsymbol{I}$ by Lemma 2, with $x = \boldsymbol{K}$
$\approx \boldsymbol{S}(\boldsymbol{KK})\boldsymbol{I} = \boldsymbol{XI}$ by (A.1).

(ii) Suppose $P = \boldsymbol{K}A$. Then $P \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XK}))\boldsymbol{F}A$ by (A.1)
$\approx \boldsymbol{Y}(\boldsymbol{XK})A(\boldsymbol{F}A) \approx \boldsymbol{S}(\boldsymbol{XK}A)\boldsymbol{I} \approx \boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I}$.

(iii) $\boldsymbol{S} \approx \boldsymbol{SU}(\boldsymbol{KF})\boldsymbol{U}(\boldsymbol{KF})$ (use (A.2) to substitute for the initial $\boldsymbol{S}$ in (A.2))
$\approx \boldsymbol{UU}(\boldsymbol{KFU})(\boldsymbol{KF}) \approx \boldsymbol{UUF}(\boldsymbol{KF}) \approx \boldsymbol{S}(\boldsymbol{K}(\boldsymbol{SU}(\boldsymbol{KF})))(\boldsymbol{F}(\boldsymbol{KF}))$
$\approx \boldsymbol{S}(\boldsymbol{KS})\boldsymbol{I} = \boldsymbol{YI}$ by (A.2).

(iv) Suppose $P = \boldsymbol{S}A$. Then $P \approx \boldsymbol{SU}(\boldsymbol{KF})A$ by (A.2)
$\approx \boldsymbol{U}A(\boldsymbol{KF}A) \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}(\boldsymbol{S}A)))\boldsymbol{F} \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}P))\boldsymbol{F}$
$\approx \boldsymbol{S}(\boldsymbol{K}(\boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}P))\boldsymbol{F}))\boldsymbol{I}$ by Lemma 2
$\approx \boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I}$ since $P \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}P))\boldsymbol{F}$.

(v) Suppose $P = \boldsymbol{S}AB$. Then $P \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}(\boldsymbol{S}A)))\boldsymbol{F}B$ by the proof of (iv)
$\approx \boldsymbol{Y}(\boldsymbol{X}(\boldsymbol{S}A))B(\boldsymbol{F}B) \approx \boldsymbol{S}(\boldsymbol{X}(\boldsymbol{S}A)B)\boldsymbol{I} \approx \boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I}$. ☒

Nested abstractions behave as in the λ-calculus:

Lemma 4. Let P, Q be combinatory terms, and x, y distinct variables; assume $y \notin \mathrm{V}(P)$. Then

(i) $[P/x](\lambda_2 y.Q) = \lambda_2 y.[P/x]Q$;

(ii) $CL + (\text{A.1}, 2) \vdash [P/x](\lambda^{+} y.Q) \approx \lambda^{+} y.[P/x]Q$.

Proof. (i) By induction on Q, following the various cases in the definition of λ_2-abstraction.

(ii) The cases $Q = y$ and $y \notin \mathrm{V}(Q)$ are like (i). In the third case it may be that $Q = Ry$ with $y \notin \mathrm{V}(R)$ but R is not functional, whereas $[P/x]R$ *is* functional. Then we must show that $[P/x]\boldsymbol{S}(\boldsymbol{K}R)\boldsymbol{I} \approx [P/x]R$; we use Lemma 3. In the final case, use (i). ☒

Lemma 5. Let P be a combinatory term that does not contain x. Then

$$CL + (\text{A.3}) \vdash \lambda_1 x.P \approx \boldsymbol{K}P.$$

Proof. Induction on the complexity of P. In particular, suppose that P is composite, $P = P_1P_2$. Take $y \notin \mathrm{V}(Px)$. Then

$$\begin{aligned}\lambda_1 x.P &= \boldsymbol{S}(\lambda_1 x.P_1)(\lambda_1 x.P_2) \approx \boldsymbol{S}(\boldsymbol{K}P_1)(\boldsymbol{K}P_2) && \text{(ind. hyp.)}\\ &\approx (\lambda_2 xy.\boldsymbol{S}(\boldsymbol{K}x)(\boldsymbol{K}y))P_1P_2 && \text{(Lemmas 1 and 4(i))}\\ &= \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XB}))(\boldsymbol{KK})P_1P_2 \approx \boldsymbol{X}P_1P_2 && \text{(A.3)}\\ &= (\lambda_2 xy.\boldsymbol{K}(xy))P_1P_2 \approx \boldsymbol{K}P && \text{(Lemmas 1 and 4(i) again).}\end{aligned}$$ ☒

Lemma 6. Let P be a functional term that does not contain x. Then

$$CL + \text{(A.1-3)} \vdash \lambda_1 x.Px \approx P.$$

Proof. $\lambda_1 x.Px = \boldsymbol{S}(\lambda_1 x.P)\boldsymbol{I} \approx \boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I}$ by the previous lemma; by Lemma 3, $\boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I} \approx P$. ☒

A major difference between CL and $\boldsymbol{\lambda\beta}$ is the ξ-rule. We want to show that it holds for λ_1 in $CL + \boldsymbol{A}_\beta$.

Lemma 7. (i) $CL + \text{(A.4)} \vdash \lambda^+ yz.\boldsymbol{S}(\boldsymbol{X}y)z \approx \lambda^+ yzx.yx$;
(ii) $CL + \text{(A.5)} \vdash \lambda^+ yzw.\boldsymbol{S}(\boldsymbol{S}(\boldsymbol{Y}y)z)w \approx \lambda^+ yzw.\boldsymbol{S}(\boldsymbol{S}yw)(\boldsymbol{S}zw)$.

Proof. By applying the definition of λ^+-abstraction, we get the axiom on display. ☒

Observe that in (ii) and on the lefthand side of (i), λ^+ may be replaced by λ_2.

Theorem 1. If $CL + \boldsymbol{A} \vdash \text{(A.3-5)}$, then

$$CL + \boldsymbol{A} \vdash P \approx Q \Rightarrow CL + \boldsymbol{A} \vdash \lambda_1 x.P \approx \lambda_1 x.Q.$$

Proof. For ground terms P, Q: from $\boldsymbol{K} \approx \boldsymbol{K}$ and $P \approx Q$, by lemma 5. For the schema $\boldsymbol{K}PQ \approx P$, let $z \notin \mathrm{V}(P)$. Then

$$\begin{aligned}\lambda_1 x.\boldsymbol{K}PQ &= \boldsymbol{S}(\lambda_1 x.\boldsymbol{K}P)(\lambda_1 x.Q) = \boldsymbol{S}(\boldsymbol{X}(\lambda_1 x.P))(\lambda_1 x.Q) && \text{by definition}\\ &\approx (\lambda^+ yz.\boldsymbol{S}(\boldsymbol{X}y)z)(\lambda_1 x.P)(\lambda_1 x.Q) && \text{by Lemmas 1 and 4}\\ &\approx (\lambda^+ yzx.yx)(\lambda_1 x.P)(\lambda_1 x.Q) && \text{by Lemma 7(i)}\\ &\approx \lambda^+ x.(\lambda_1 x.P)x && \text{by Lemmas 1 and 4(ii)}\\ &= \lambda_1 x.P && \text{by definition.}\end{aligned}$$

For the schema $\boldsymbol{S}PQR \approx PR(QR)$, let $z, w \notin \mathrm{Var}(P)$ and $w \notin \mathrm{Var}(Q)$, then

$$\begin{aligned}\lambda_1 x.\boldsymbol{S}PQR &= \boldsymbol{S}(\lambda_1 x.\boldsymbol{S}PQ)(\lambda_1 x.R) = \boldsymbol{S}(\boldsymbol{S}(\lambda_1 x.\boldsymbol{S}P)(\lambda_1 x.Q))(\lambda_1 x.R)\\ &= \boldsymbol{S}(\boldsymbol{S}(\boldsymbol{Y}(\lambda_1 x.P))(\lambda_1 x.Q))(\lambda_1 x.R) && \text{by definition}\\ &\approx (\lambda^+ yzw.\boldsymbol{S}(\boldsymbol{S}(\boldsymbol{Y}y)z)w)(\lambda_1 x.P)(\lambda_1 x.Q)(\lambda_1 x.R) && \text{by Lemmas 1, 4}\\ &\approx \lambda^+ yzw.\boldsymbol{S}(\boldsymbol{S}yw)(\boldsymbol{S}zw)(\lambda_1 x.P)(\lambda_1 x.Q)(\lambda_1 x.R) && \text{by Lemma 7(ii)}\\ &\approx \boldsymbol{S}(\boldsymbol{S}(\lambda_1 x.P)(\lambda_1 x.R))(\boldsymbol{S}(\lambda_1 x.Q)(\lambda_1 x.R)) && \text{by Lemmas 1, 4}\\ &= \lambda_1 x.PR(QR) && \text{by definition.}\end{aligned}$$

If $P = P_1P_2$ and $Q = Q_1Q_2$ and the last step in the deduction of $P \approx Q$ was

$$\frac{P_1 \approx Q_1 \qquad P_2 \approx Q_2}{P \approx Q}$$

then by induction hypothesis $\lambda_1 x.P_i \approx \lambda_1 x.Q_i$ is provable ($i = 1, 2$); so

$$\boldsymbol{S}(\lambda_1 x.P_1)(\lambda_1 x.P_2) \approx \boldsymbol{S}(\lambda_1 x.Q_1)(\lambda_1 x.Q_2),$$

i.e. $\lambda_1 x.P \approx \lambda_1 x.Q$. ☒

Corollary I. Let λ be λ^+ or λ^*, P a combinatory term. Then

$$CL + \boldsymbol{A}_\beta \vdash \lambda x.P \approx \lambda_1 x.P.$$

Proof. By the Reduction Lemma, $(\lambda x.P)x \approx P$. So by the theorem,

$$\lambda_1 x.(\lambda x.P)x \approx \lambda_1 x.P.$$

Since $\lambda x.P$ is functional, by Lemma 6 we have $\lambda_1 x.(\lambda x.P)x \approx \lambda x.P$. ☒

Corollary II. Let λ be λ^+ or λ^*. Then

$$CL + \boldsymbol{A}_\beta \vdash P \approx Q \;\Rightarrow\; CL + \boldsymbol{A}_\beta \vdash \lambda x.P \approx \lambda x.Q.$$

Lemma 8. For any combinatory term P,
(i) $(\lambda_2 x.P)_\lambda x \twoheadrightarrow_\beta P_\lambda$;
(ii) $(\lambda^+ x.P)_\lambda x \twoheadrightarrow_\beta P_\lambda$.

Proof. By induction on P, and using (i) for (ii). ☒

Lemma 9. For any combinatory term P, $(\lambda^+ x.P)_\lambda \approx_\beta \lambda x.P_\lambda$.

Proof. By the previous lemma, $(\lambda^+ x.P)_\lambda x \approx_\beta P_\lambda$. So by Rule (ξ),

$$\lambda x.(\lambda^+ x.P)_\lambda x \approx_\beta \lambda x.P_\lambda.$$

Now apply Proposition 1 and the corollary to Proposition 2. ☒

The next lemma and theorem are easiest if we take κ to mean κ^+.

Lemma 10. Let $\kappa = \kappa^+$. For all λ-terms M, N and for all variables x,

$$CL + (\text{A}.1, 2) \vdash [N_\kappa/x]M_\kappa \approx ([N/x]M)_\kappa.$$

Proof. By induction on M. The least trivial case is abstraction. If $M = \lambda y.P$ (where by convention $y \neq x$ and $y \notin \text{FV}(N)$), then

$$[N_\kappa/x]M_\kappa = [N_\kappa/x](\lambda^+ y.P_\kappa) \approx \lambda^+ y.[N_\kappa/x]P_\kappa$$

by Lemma 4(ii), for $y \notin \text{V}(N_\kappa)$; so by induction hypothesis

$$[N_\kappa/x]M_\kappa \approx \lambda^+ y.([N/x]P)_\kappa = ([N/x]M)_\kappa.$$ ☒

Theorem 2. $\boldsymbol{\lambda\beta}$ and $CL + \boldsymbol{A}_\beta$ are equivalent, in the following sense: for all λ-terms M, N and combinatory terms P, Q, and with $\kappa = \kappa^+$,
(i) $\boldsymbol{\lambda\beta} \vdash M_{\kappa\lambda} \approx M$;
(ii) $P_{\lambda\kappa} = P$;
(iii) $\boldsymbol{\lambda\beta} \vdash M \approx N \Leftrightarrow CL + \boldsymbol{A}_\beta \vdash M_\kappa \approx N_\kappa$;
(iv) $CL + \boldsymbol{A}_\beta \vdash P \approx Q \Leftrightarrow \boldsymbol{\lambda\beta} \vdash P_\lambda \approx Q_\lambda$.

Proof. (i) By induction on M; use Lemma 9.
(ii) By induction on P; observe that $\mathsf{K}_\kappa = \boldsymbol{K}$ and $\mathsf{S}_\kappa = \boldsymbol{S}$.
(iii) ($\Rightarrow$) By induction on the length of the proof of $M \approx N$. Identity axioms translate to identity axioms, and instances of the application rule to instances of the application rule. For β-axioms $(\lambda x.M_1)M_2 \approx [M_2/x]M_1$ we get

$$((\lambda x.M_1)M_2)_\kappa = (\lambda^+ x.M_{1\kappa})M_{2\kappa} \approx [M_{2\kappa}/x]M_{1\kappa} = ([M_2/x]M_1)_\kappa$$

by Lemma 10. If $M \approx N$ is the conclusion of an instance of the ξ-rule, say $M = \lambda x.M_0$ and $N = \lambda x.N_0$, then by induction hypothesis

$$CL + \boldsymbol{A}_\beta \vdash M_{0\kappa} \approx N_{0\kappa};$$

so by Corollary II of Theorem 1 we have

$$CL + \boldsymbol{A}_\beta \vdash \bar{\lambda}^+ x.M_{0\kappa} \approx \bar{\lambda}^+ x.N_{0\kappa},$$

which is to say that $M_\kappa \approx N_\kappa$ is deducible.
(iv) (⇒) By induction on the length of the proof of $P \approx Q$. Identity axioms translate into identity axioms, and instances of the application rule into instances of the application rule. The combinator schemes of CL correspond to the β-reductions $\mathsf{K}MN \twoheadrightarrow_\beta M$ and $\mathsf{S}MNL \twoheadrightarrow_\beta ML(NL)$. The translations of the $\boldsymbol{A}_\beta$-axioms are seen to be valid by straightforward calculation.
(iii) (⇐) If $CL + \boldsymbol{A}_\beta \vdash M_\kappa \approx N_\kappa$, then by the half of (iv) we just proved,

$$\boldsymbol{\lambda\beta} \vdash M_{\kappa\lambda} \approx N_{\kappa\lambda};$$

so by (i), $\boldsymbol{\lambda\beta} \vdash M \approx N$.
(iv) (⇐) If $\boldsymbol{\lambda\beta} \vdash P_\lambda \approx Q_\lambda$, then by (iii), $CL + \boldsymbol{A}_\beta \vdash P_{\lambda\kappa} \approx Q_{\lambda\kappa}$. So by (ii), $CL + \boldsymbol{A}_\beta \vdash P \approx Q$. ⊠

Remark. By Theorem 1, Corollary I, the theorem holds for $\kappa \in \{\kappa^*, \kappa_1\}$ as well, if we replace (ii) by

(ii′) $CL + \boldsymbol{A}_\beta \vdash P_{\lambda\kappa} \approx P$.

The extensional case

$\boldsymbol{A}_\eta$ is the following quartet of axioms:

(A.3) $\boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{XB}))(\boldsymbol{KK}) \approx \boldsymbol{X}$

(A.4) $\boldsymbol{YX} \approx \boldsymbol{X}(\boldsymbol{SBF})$

(A.5) $\boldsymbol{S}(\boldsymbol{KY})(\boldsymbol{YY}) \approx \boldsymbol{S}(\boldsymbol{Y}(\boldsymbol{X}(\boldsymbol{Y}(\boldsymbol{S}(\boldsymbol{KY})\boldsymbol{S}))))(\boldsymbol{KS})$

(A.6) $\boldsymbol{SBF} \approx \boldsymbol{I}$

Lemma 11. $CL + \text{(A.6)} \vdash x \approx \boldsymbol{S}(\boldsymbol{K}x)\boldsymbol{I}$.

Proof. By (A.6), $x \approx \boldsymbol{I}x \approx \boldsymbol{SBF}x \approx \boldsymbol{B}x(\boldsymbol{F}x) \approx \boldsymbol{S}(\boldsymbol{K}x)\boldsymbol{I}$. ⊠

So (A.6) makes all terms functional, up to provable identity. Then Lemma 6 implies:

Lemma 12. Let P be a combinatory term that does not contain x. Then

$$CL + \text{(A.6)} \vdash \lambda_1 x.Px \approx P.$$

Lemma 13. For any combinatory term P, $CL + \boldsymbol{A}_\eta \vdash \lambda_2 x.P \approx \lambda_1 x.P$.

Proof. By Theorem 1, the ξ-rule holds for λ_1 in $CL + \boldsymbol{A}_\eta$. So from

$$(\lambda_2 x.P)x \approx P \qquad \text{(Reduction Lemma)}$$

we get $\lambda_1 x.(\lambda_2 x.P)x \approx \lambda_1 x.P$; and by Lemma 12, $\lambda_1 x.(\lambda_2 x.P)x \approx \lambda_2 x.P$. ⊠

Combining Theorem 1 with Lemma 13, we obtain:

Lemma 14. $CL + \boldsymbol{A}_\eta \vdash P \approx Q \Rightarrow \boldsymbol{CL} + \boldsymbol{A}_\eta \vdash \lambda_2 x.P \approx \lambda_2 x.Q$.

Lemma 15. For any combinatory term P, $(\lambda_2 x.P)_\lambda \approx_{\beta\eta} \lambda x.P_\lambda$.

Proof. By Lemma 8(i), $(\lambda_2 x.P)_\lambda x \twoheadrightarrow_\beta P_\lambda$. Then by the ξ-rule,

$$\lambda x.(\lambda_2 x.P)_\lambda x \approx_\beta \lambda x.P_\lambda.$$

By the (η)-scheme, $\lambda x.(\lambda_2 x.P)_\lambda x \approx (\lambda_2 x.P)_\lambda$. ☒

Lemma 16. Let $\kappa = \kappa_2$. For all λ-terms M, N and any variable x,

$$[N_\kappa/x]M_\kappa = ([N/x]M)_\kappa.$$

Proof. As Lemma 10; use (i) of Lemma 4 instead of (ii). ☒

Theorem 3. $\boldsymbol{\lambda\beta\eta}$ and $CL + \boldsymbol{A}_\eta$ are equivalent, in the following sense: for all λ-terms M, N and combinatory terms P, Q, and with $\kappa = \kappa_2$,

(i) $\boldsymbol{\lambda\beta\eta} \vdash M_{\kappa\lambda} \approx M$;

(ii) $P_{\lambda\kappa} = P$;

(iii) $\boldsymbol{\lambda\beta\eta} \vdash M \approx N \Leftrightarrow CL + \boldsymbol{A}_\eta \vdash M_\kappa \approx N_\kappa$;

(iv) $CL + \boldsymbol{A}_\eta \vdash P \approx Q \Leftrightarrow \boldsymbol{\lambda\beta\eta} \vdash P_\lambda \approx Q_\lambda$.

Proof. Similar to the proof of Theorem 2; use Lemmas 5 and 17-19. ☒

Remark. The argument of Lemma 13 applies to λ^+ or λ^* as well. Hence the theorem holds for $\kappa \in \{\kappa^*, \kappa_1, \kappa^+\}$ as well; only for κ^* and κ_1 we must replace (ii) by

(ii′) $CL + \boldsymbol{A}_\eta \vdash P_{\lambda\kappa} \approx P$.

Curry's axioms

Curry does not use the axioms (A.1, 2); instead he has

(C.1) $\boldsymbol{K} \approx \boldsymbol{B}(\boldsymbol{SBF})\boldsymbol{K}$,

(C.2) $\boldsymbol{S} \approx \boldsymbol{B}(\boldsymbol{B}(\boldsymbol{SBF}))\boldsymbol{S}$.

These do the same job as (A.1, 2). Abbreviate $\boldsymbol{SBF}$ to $\boldsymbol{I}_\eta$.

Lemma 17. CL + (C.2) $\vdash \boldsymbol{B}xy \approx \boldsymbol{B}(\boldsymbol{B}xy)\boldsymbol{I}$.

Proof: $\boldsymbol{B}xy \approx \boldsymbol{S}(\boldsymbol{K}x)y \approx \boldsymbol{B}(\boldsymbol{BI}_\eta)\boldsymbol{S}(\boldsymbol{K}x)y$ by (C.2)

$\approx \boldsymbol{BI}_\eta(\boldsymbol{S}(\boldsymbol{K}x))y \approx \boldsymbol{I}_\eta(\boldsymbol{S}(\boldsymbol{K}x)y) \approx \boldsymbol{B}(\boldsymbol{S}(\boldsymbol{K}x)y)(\boldsymbol{F}(\boldsymbol{S}(\boldsymbol{K}x)y))$

$\approx \boldsymbol{B}(\boldsymbol{B}xy)\boldsymbol{I}$. ☒

Lemma 18. Suppose P is functional. Then

$$CL + (\text{C.1}, 2) \vdash P \approx \boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I}.$$

Proof: We have five cases as in Lemma 3. Observe that $\boldsymbol{S}(\boldsymbol{K}P)\boldsymbol{I} \approx \boldsymbol{B}P\boldsymbol{I}$.

(i) $\boldsymbol{K} \approx \boldsymbol{BI}_\eta\boldsymbol{K}$ by (C.1)

$\approx \boldsymbol{B}(\boldsymbol{BI}_\eta\boldsymbol{K})\boldsymbol{I}$ by Lemma 17, with $x = \boldsymbol{I}_\eta$ and $y = \boldsymbol{K}$

$\approx \boldsymbol{BKI}$.

(ii) Suppose $P = \boldsymbol{K}A$. Then $P \approx \boldsymbol{BI}_\eta\boldsymbol{K}A$ by (C.1)

$\approx \boldsymbol{I}_\eta P \approx \boldsymbol{B}P\boldsymbol{I}$.

(iii) $\boldsymbol{S} \approx \boldsymbol{B}(\boldsymbol{BI}_\eta)\boldsymbol{S}$ by (C.2)

$\approx \boldsymbol{B}(\boldsymbol{B}(\boldsymbol{BI}_\eta)\boldsymbol{S})\boldsymbol{I}$ by Lemma 17

$\approx \boldsymbol{BSI}$.

(iv) Suppose $P = \boldsymbol{S}A$. Then $P \approx \boldsymbol{B}(\boldsymbol{BI}_\eta)\boldsymbol{S}A$ by (C.2)

$\approx \boldsymbol{BI}_\eta P \approx \boldsymbol{B}(\boldsymbol{BI}_\eta P)\boldsymbol{I}$ by lemma 17

$\approx \boldsymbol{B}P\boldsymbol{I}$.

(v) Suppose $P = \boldsymbol{S}AB$. Then $P \approx \boldsymbol{B}(\boldsymbol{BI}_\eta)\boldsymbol{S}AB$ by (C.2)

$\approx \boldsymbol{BI}_\eta(\boldsymbol{S}A)B \approx \boldsymbol{I}_\eta P \approx \boldsymbol{B}P\boldsymbol{I}$. ⊠

References

[CL] Haskell B. Curry, Robert Feys & William Craig: Combinatory Logic. Amsterdam, 1958.

[LC] Henk Barendregt: The λ-calculus. 2nd edition, Amsterdam 1984.

[LCCI] J. Roger Hindley & Jonathan P. Seldin: Lambda-calculus and combinators, an introduction. Cambridge University Press, 2008.

Further References

D.A. Turner: Another algorithm for bracket abstraction. Afdruk van JSL XLIV, 267-70. A4(2r) Λ.

Gebruikt combinator $\boldsymbol{S}'$ gedefinieerd door $\boldsymbol{S}'xyzw \approx x(yw)(zw)$.