TLCA List of Open Problems

http://tlca.di.unito.it/opltlca/

July 21, 2014

Contents

Open Problem List

Introduction

The TLCA List of Open Problems (modeled after the RTA LOOP) aims at collecting unresolved questions (and other relevant information, e.g. about solutions and related results) in the subject areas of the TLCA (Typed Lambda Calculi and Applications) series of conferences, in particular in the following topics:

- Typed and untyped lambda-calculi as models of computation.
- Proof-theory: Natural deduction, sequent calculi, cut elimination and normalization. Propositions as types, linear logic and proof nets.
- Semantics: Denotational semantics, game semantics, realizability, categorical models.
- Programming languages: Foundations of functional and object-oriented programming, proof search, logic programming, type checking.
- Implementation: Abstract machines, parallel execution, optimal reduction, program optimization.

Publication Remarks

1. Publication of problems, solutions and comments is subject to the approval of the Editorial Board of the TLCA Open Problems List. The Board reserves the right to edit all submissions.
2. The List is for informational purposes only. No guarantee is given that the summarized problem descriptions are accurate, complete and up to date. Always check the literature or contact the originator of a problem to be sure. The Editorial Board assume no liability for any inaccurate, delayed, or incomplete information, nor for any actions taken in reliance thereon.
3. The status of a solution announcement published on TLCA List of Open Problems is equivalent to that of a message to a moderated mailing list.

Submissions

Problem descriptions should be sent to tlca@mimuw.edu.pl or to any member of the Editorial Board (links to mail-addresses: Ryu Hasegawa, Luca Paolini, Pawel Urzyczyn) with subject "TLCA List of Open Problems".

The list is maintained as a web page, and also available as a (LaTeX-based) PDF file. Therefore, it is important that the submitted source material is provided in LaTeX, and it can be easily transformed into a html file (e.g. that it is typeset in basic LaTeX, with as few special symbols as possible). All references should be provided as BibTeX entries.

We would greatly appreciate, if submissions are done by sending us two files respecting the following constraints.

1. A BibTeX-file containing relevant references not yet included in opltlca.bib.
2. A LaTeX-file based on draft.tex (you need also tlcamacro.tex, if you want try a PdfLaTeX compilation) appropriately filled.

- A special LaTeX-command building the problem heading must be used. This command contains 4 fields. The first field can be used in order to describe the problem-origin, it is optional (as usual LaTeX-options, you can omit it and the square bracket surrounding it). Other fields respectively contain name of the submitter, a date and a statement.

```
\ProblemSummary[
        This field is optional.
        Insert here information around:
               - who posed first the problem
               - name-problem origin
               - related problems and conjectures
               - further historical remarks
]{
       YourFirstNAME YourLastNAME
       \INDEX{YourLastNAME, YourFirstNAME}
}
{PROBLEM BIRTH-DATE}
{STATEMENT SUMMARIZING THE OPEN PROBLEM}
```

  Here is a real example with origin remarks.

```
\ProblemSummary[
The problem was posed by
     \HREF{http://www.cs.ru.nl/\string ~henk/}
                             {Henk Barendregt}
                 \INDEX{Barendregt, Henk},
     \HREF{http://www.cs.ru.nl/\string ~herman/}
                                {Herman Geuvers}
                 \INDEX{Geuvers, Herman}
     and
     \HREF{http://www.cs.vu.nl/\string ~jwk/}
                            {Jan Willem Klop}
                 \INDEX{Klop, Jan Willem}.
]{
     \HREF{http://www.diku.dk/\string ~rambo/}
                    {Morten Heine S{\o}rensen}
         \INDEX{S{\o}rensen, Morten Heine}
}{1993}
{Is every weakly normalizing PTS also strongly normalizing?}
```

- Hyperlinks to relevant web pages where detailed information information can be given, by using the following LaTeX-command:

```
\HREF{...url...}{...label...}.
```

  Homepage can be added as follows,

```
\HREF{http://www.di.unito.it/\string ~paolini/}
                         {Luca Paolini HomePage}.
```

- To create index entries use

```
\INDEX{keyword_1}
   ....
```



```
\INDEX{keyword_n}.
```

  Please add keywords of your problem by using this feature.

- A brief and concise (up to 1/2 page) exposition of the problem.
- To cite references use

```
\cite{bibkey}
```

LaTeX and BibTeX Remarks

Please, use special symbols only in math-mode. For instance, if you need a section-symbol like § then write $\S$ instead of \S. To insert a tilde in the scope of \HREF, use "\string~".
A brief BibTeX introduction can be found here:
http://www.di.unito.it/ lambda/biblio/Doc/help.html.

Solutions and Comments

Comments related to problems published in the list, in particular pointers to solutions and related results, can be submitted by e-mail to tlca@mimuw.edu.pl or to any member of the Editorial Board (links to mail-addresses: Ryu Hasegawa, Luca Paolini, Pawel Urzyczyn). Each submission should contain:

- the number of the problem referred to;
- a brief statement of the comment;
- relevant references/links etc.

Do NOT submit full solutions, only references and/or stable web pointers.

Updates

Please notify us about any inaccuracies, provide updated references, links and other corrections. Send e-mail to tlca@mimuw.edu.pl.

The Editorial Board

Ryu Hasegawa, Luca Paolini, Paweł Urzyczyn.

Printable Full List

An updated version of the TLCA List of Open Problems can be downloaded from
http://tlca.di.unito.it/opltlca/opltlca.pdf.
Moreover, a two pages per sheet version can be downloaded from
http://tlca.di.unito.it/opltlca/opltlca2.pdf.

Alternative Resources

Further open problems published elsewhere can be found in many places.

- Many related interesting problems are presented in the RTA List of Open Problems.

  http://www.lsv.ens-cachan.fr/rtaloop/.

  In particular, look at problems 1,4,5,15,17,19,45,50,52-55,78,88,94,96,97.

- Some open problems collected by Henk Barendregt can be found in [Barendregt, 1975].
- A list of open problems collected at a 1993 Utrecht meeting of the Gentzen Working Group is available from http://www.mimuw.edu.pl/tlca/Gentzen93/.
- A lot of open problems can be found in some papers of Chantal Berline, in particular [Berline, 2000, Berline, 2006].

Problem Anthology

Problem # 1 [SOLVED]

Submitted by Roger Hindley *Date:* Known since 1958!
Statement. Is there a direct proof of the confluence of $\beta\eta$-strong reduction?
Problem Origin. First posed by Haskell Curry and Roger Hindley.

The $\beta\eta$-strong reduction is the combinatory analogue of $\beta\eta$-reduction in λ-calculus. It is confluent. Its only known confluence-proof is very easy, [Curry and Feys, 1958, § 6F, p. 221 Theorem 3], but it depends on the having already proved the confluence of $\lambda\beta\eta$-reduction. Thus the theory of combinators is not self-contained at present. Is there a confluence proof independent of λ-calculus?

Warning: Like a bog, the theory of strong reduction is messier than it looks at first; see [Curry and Feys, 1958, § 6F], [Curry et al., 1972, § 11E], [Hindley and Seldin, 1986, Ch. 9], and sources cited therein.

Solution: Two independent confluence proofs have been proposed in June 2008: one by René David [David, 2009], the other by Pierluigi Minari [Minari, 2009].

Problem # 2 [SOLVED]

Submitted by Roger Hindley *Date:* Known since 1973!
Statement. Is ticket entailment decidable?
Problem Origin. The problem was first posed by Robert Meyer.

The question is whether there is a decision-algorithm for the implicational fragment $T_\rightarrow$ of the propositional logic called *ticket entailment.* Equivalently, is there one for the simple type-theory of the restricted combinatory logic based on B, B', I, W? The logic $T_\rightarrow$ has just one deduction-rule (($\rightarrow$E) or *modus ponens*), and four axiom-schemes:

$$(\alpha \rightarrow \beta) \rightarrow ((\gamma \rightarrow \alpha) \rightarrow (\gamma \rightarrow \beta)), \quad \alpha \rightarrow \alpha,$$
$$(\alpha \rightarrow \beta) \rightarrow ((\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \gamma)), \quad (\alpha \rightarrow (\alpha \rightarrow \beta)) \rightarrow (\alpha \rightarrow \beta).$$

Alternatively, let $\mathrm{CL}_{B,B',I,W}$ be the system of combinatory logic whose *terms* are built by application from four basic combinators with reduction rules:

$$\mathsf{B}XYZ \rhd X(YZ), \quad \mathsf{B}'XYZ \rhd Y(XZ), \quad \mathsf{I}X \rhd X, \quad \mathsf{W}XY \rhd XYY.$$

(Abstraction in $\mathrm{CL}_{B,B',I,W}$ is much weaker than in full combinatory logic; see [Trigg et al., 1994, §3] for a characterization by P. Trigg.) Let *types* be built by the operation $(\sigma \rightarrow \tau)$ from type-variables $a, b, c, \ldots$, and let types be assigned to terms as usual, starting from these four axiom-schemes:

$$\mathsf{B} : (\alpha \rightarrow \beta) \rightarrow ((\gamma \rightarrow \alpha) \rightarrow (\gamma \rightarrow \beta)), \quad \mathsf{I} : \alpha \rightarrow \alpha,$$
$$\mathsf{B}' : (\alpha \rightarrow \beta) \rightarrow ((\beta \rightarrow \gamma) \rightarrow (\alpha \rightarrow \gamma)), \quad \mathsf{W} : (\alpha \rightarrow (\alpha \rightarrow \beta)) \rightarrow (\alpha \rightarrow \beta).$$

Is there an algorithm that, when applied to any type τ, will decide whether there exists a term X in this system such that $X : \tau$ is provable?

System $T_\rightarrow$ first appeared in print in [Anderson, 1960], although it dates back at least to work of Belnap in 1957. It was motivated and described in detail in [Anderson and Belnap, 1975, Chapter 1 §§ 6 and 8.3.2 (pp. 41–50 and 76)]. Its decidability question was first raised on

p. 69 of that book. Proofs of the decidability and undecidability of several related systems were given in [Anderson et al., 1992, §§ 60–67 (pp. 267–391)]; for example in § 65.2 the logic T of ticket entailment was shown to be undecidable, but the method did not apply to its implicational fragment $T_{\rightarrow}$. A decidability result for a restricted class of formulas can be found in [Broda et al., 2004].

Warnings: (1) In the 30 years since 1975 the $T_{\rightarrow}$ problem and its combinatory equivalent have been tried by several very able workers without success. For example some relevant results are in [Bimbó, 2005] and [Bimbó, 2006].

(2) In papers on entailment, omitted parentheses are usually restored by "association to the left", not "to the right" as in types in type theory!

Solution: Two independent confluence proofs have been proposed in 2010. The solution by Katalin Bimbó and J. Michael Dunn is published in [Bimbó and Dunn, 2013]. The solution by Vincent Padovani is published in [Padovani, 2013].

Problem # 3 [SOLVED]

Submitted by Richard Statman *Date:* 1981
Statement. Does the range property hold for the theory $\mathcal{H}$?
Problem Origin. The problem was first posed by Henk Barendregt.

Consider the least lambda theory $\mathcal{H}$ which equates all unsolvable terms. Take any combinator F. Prove that either $\mathcal{H} \vdash FM = FN$, for all closed M, N, or there are infinitely many closed M_i, such that no two of terms FM_i are equated under $\mathcal{H}$. This problem, known as the *range problem for* $\mathcal{H}$, is stated as Conjecture 20.2.8 in [Barendregt, 1984].

Comments by Richard Statman: Different researchers have different takes on this problem. Mine are the following.

(a) If we can compute functions recursive in the jump of 0 then the range property holds. The word problem for $\mathcal{H}$ is Σ_2-complete but there is no good notion of $\mathcal{H}$-computability that allows the "computation" of those number theoretic function recursive in jump of 0. If one knew such a notion of computability then one could copy the usual recursion theoretic proof of the range property for beta-eta.

(b) If we can construct solvable Plotkin terms then the range property fails. Take a Kleene enumerator E such that $E_0 = \Omega$ and E_{n+1} is the n-th solvable term. Then one can construct Plotkin terms F and G such that if $k > 0$ then

$$F_0(G_0)(E_0) \neq F_0(G_0)(E_k)$$

$$F_0(G_0)(E_k) = F_0(G_0)(E_{k+1})$$

This is for the beta-eta theory. If F and G can be constructed to be solvable then we would have a term with a range of size 2 for $\mathcal{H}$.

Solution: A negative solution has been presented by A. Polonsky in 2010 and published in [Polonsky, 2012], who constructs a term F with a two-element range. In contrast, Intrigila and Statman [Intrigila and Statman, 2011] showed that the range property for Combinatory Logic is true.

Problem # 4

Submitted by Richard Statman *Date:* 1985
Statement. A fixed point combinator from B and ω alone?
Problem Origin. The problem was first posed by Raymond Smullyan.

The combinator ω satisfies the reduction rule $\omega x \Rightarrow xx$ and the combinator B is as usual $\mathsf{B}xyz \Rightarrow x(yz)$. Is there an applicative combination of B and ω which is a fixed point combinator? The problem was originally proposed in [Smullyan, 1985].

Comments by Richard Statman: Note that $\omega(\mathsf{B}x\omega)$ is always a fixed point of x but it appears that x cannot be abstracted to give a fixed point combinator Y such that $\mathsf{Y}x = x(\mathsf{Y}x)$. If it is required that $\mathsf{Y}x \twoheadrightarrow x(\mathsf{Y}x)$ as with Turing's fixed point combinator then it can be shown [Statman, 1993, McCune and Wos, 1991] that no B,ω combination works.

Problem # 5

Submitted by Paweł Urzyczyn *Date:* 1993
Statement. Are there terms untypable in F_ω but typable with help of positive recursive types?

Consider an extension of simply typed lambda-calculus obtained by adding the construct $\mu p\,\tau$ for types τ containing only positive occurrences of p, together with the rules:

$$\frac{\Gamma \vdash M : \tau[p{:=}\mu p\,\tau]}{\Gamma \vdash M : \mu p\,\tau} \qquad \frac{\Gamma \vdash M : \mu p\,\tau}{\Gamma \vdash M : \tau[p{:=}\mu p\,\tau]}$$

This system can assign types for terms untypable in F, e.g. the combinator **22K** is typable, where **2** is the Church numeral [Urzyczyn, 1996]. Are there terms typable in this system that cannnot be typed in F_ω? Note that **22K** can be typed in F_ω [Urzyczyn, 1997]. Note also that $\lambda x.\,xx$, easily typable in system F, is clearly untypable with positive recursive types.

Problem # 6

Submitted by Richard Statman *Date:* 1993
Statement. Basis Decision Problem.

The *basis decision problem*, originally stated in [Statman, 1993], is to decide whether a given finite set of proper combinators is combinatorially complete. (A *proper* combinator [Barendregt, 1984, p. 184] is one of the form $\lambda\vec{x}.\,M$, where M contains no lambda, and $FV(M) \subseteq \vec{x}$.) It follows from Theorem 4 in [Statman, 1986] that the problem is undecidable for finite sets of normal (possibly improper) combinators. A good solution would be an abstraction algorithm, parameterized by a given set of combinators, which returned an error message on non-bases. Broda and Damas have proved that in the linear case the problem is decidable [Broda and Damas, 1997].

Problem # 7

Submitted by Richard Statman *Date:* 1993
Statement. Word problem for combinators of orders less than 3.

The *word problem* for a set of combinators C is to determine if two applicative combinations of members of C are $\beta\eta$-equal. A proper combinator A, with reduction rule $\mathsf{A}x_1 \dots x_n \Rightarrow X$, where X is an applicative combination of $x_1, \dots, x_n$ is said to have *order n*. We ask if the word problem for all proper combinators of orders less than 3 is decidable. The problem was originally proposed in [Statman, 1989].

Comments by Richard Statman: It is easy to see that the word problem for combinators of order 1 is decidable. More can be shown [Statman, 1988a]. In [Statman, 1989] it is shown that the word problem for Smullyan's Lark combinator is solvable even though fixed points always exist. In [Statman, 2000] we prove that the word problem for all co-compositors and finitely many instances of (finitely many) compositors is solvable. We have been led to the conjecture: The word problem for all proper combinators of order less than 3 (taken together in one system) is decidable. Some combinator of order 3 or more is necessary for a basis of proper combinators [Statman, 1986] but slightly beyond order 2 [Statman, 1988b] leads to undecidabilty.

Added "in proof": This problem also occurs in the RTA LOOP as problem number 96.

Problem # 8

Submitted by Richard Statman *Date:* 1993
Statement. Which results on one-step effective strategies carry over from combinatory logic to lambda-calculus?

The following results on reduction and conversion strategies in combinatory logic are surveyed in [Statman, 1997]:

- There is an effective one-step cofinal reduction strategy.
- There is an effective one-step Church-Rosser strategy.
- There is an effective one-step enumeration strategy (a strategy which lists all the combinators convertible to a given one).
- There is no effective confluence function (computing a common reduct of any given terms M and N).

These results all apply to the combinator calculus and many seem to depend on the fact that residuals of disjoint redexes remain disjoint. The questions, however, appear to be more subtle for lambda terms where each one-step reduction can create a much more radical change in structure.

Problem # 9

Submitted by Morten Heine Sørensen *Date:* 1993
Statement. Is every weakly normalizing PTS also strongly normalizing?
Problem Origin. The problem was posed by Henk Barendregt, Herman Geuvers and Jan Willem Klop.

The conjecture that the above statement can be answered in the positive is known as the *Barendregt-Geuvers-Klop conjecture*. It was presented to the originator by Henk Barendregt some time in 1994. Is is stated as (part of) Conjecture 8.1.2 in [Geuvers, 1993].

A partial solution, confirming that the conjecture is true for a certain class of non-dependent systems, appeared in [Sørensen, 1997, Barthe et al., 2001].

It is natural to wonder how the result can be extended to larger classes of pure type systems. An obvious line of attack is to study translations that eliminate dependency, and to establish preservation of the necessary normalization properties (see Definition 6.5.23 in [Geuvers, 1993] for a translation eliminating dependency in the cube).

Another idea, which will probably only lead to marginal improvements, is to replace the CPS-translation of [Sørensen, 1997, Barthe et al., 2001] by a simpler thunkification translation or another translation with similar properties (see [Gørtz et al., 2003]).

A related conjecture is the so-called *K-conjecture* [Barthe, 1995], which is shown to follow from the Barendregt-Geuvers-Klop conjecture within the class of functional pure type systems [Barthe, 1998a].

See also [Ketema et al., 2005].

Problem # 10

Submitted by Richard Statman *Date:* 1993
Statement. Do uniform universal generators exist?

A term M is a *uniform universal generator* iff there is a non-trivial context $C[*]$ such that $M \twoheadrightarrow C[N]$ for each closed term N. The above question was first stated in [Statman, 1993], and it is open for either CL (with weak reduction) or lambda-calculus (with β- or $\beta\eta$-reduction). If the context $C[*]$ is required to have the form $P*$, so $C[N]$ is PN, then it can be shown [Statman, 1993] that uniform universal generators do not exist for combinatory logic.

Problem # 11

Submitted by Paweł Urzyczyn *Date:* 1993
Statement. Which equations on intersection types preserve normalization?
Problem Origin. Included in the list of open problems from the "Gentzen" meeting in 1993. Author unknown.

Let E be a finite set of equations between intersection types, and let $=_E$ be the congruence generated by E. Extend intersection type assignment by the rule:

$$\frac{\Gamma \vdash M : \tau}{\Gamma \vdash M : \sigma}\ (\tau =_E \sigma)$$

This generalizes the notion of a recursive type introduced via "type constraints" in the style of [Mendler, 1991].

Under what condition such a type assignment system has the (strong) normalization property? As an example consider $E = \{a_1 = (a_2 \to a_1) \wedge (a_1 \to a_1),\ a_2 = a_1 \to a_2\}$. This system is not "positive" or "monotone" (cf. Problem 1) yet it is believed to be strongly normalizing. Note: As shown in [Mendler, 1991], positivity is a necessary and sufficient condition for strong normalization of a system defined by equations between simple types (without intersections).

Problem # 12

Submitted by Paweł Urzyczyn *Date:* 1993

Statement. Does the typability hierarchy of F_ω collapse on F_1?

The type assignment system F_ω can be split into an infinite hierarchy of subsystems F_n (see [Girard, 1986]), where F_0 is the usual polymorphic lambda-calculus (system F), and F_n admits constructors of *order n*. (A type is a constructor of order zero, a constructor of order one is a function acting on types, and a constructor of order $n+1$ has arguments of order n.) It is conjectured [Urzyczyn, 1997] that every pure lambda-term typable in F_ω is typable already in F_1.

Remark: An incorrect proof of the conjecture appeared in [Malecki, 1997].

Problem # 13

Submitted by Mariangiola Dezani-Ciancaglini *Date:* 1993–2006

Statement. Inhabitation for intersection type systems

Problem Origin. Different variants of the problem were stated by Henk Barendregt, Mariangiola Dezani-Ciancaglini, Paula Severi, Paweł Urzyczyn, and others.

The inhabitation problem is to determine whether there exists a closed term of a given type. [Urzyczyn, 1999] shows the undecidability of inhabitation for the intersection type system of [Barendregt et al., 1983]. Decidable restrictions are discussed in [Kurata and Takahashi, 1995].

Many different intersection type systems have been introduced, mainly for describing λ-models: for a list see [Alessi et al., 2006] and the references there. A natural question is decidability of inhabitation for these systems. Using the notation of [Alessi et al., 2006], undecidability of the inhabitation for the system of [Barendregt et al., 1983] implies undecidability of the inhabitation for the systems $\mathcal{B}a$, $\mathcal{CDV}$, $\mathcal{E}n$. It is easy to show that all types are inhabited in $\mathcal{AO}$ and $\mathcal{P}a$ systems, since all terms inhabit the top type and all closed terms inhabit the bottom type [Honsell and Ronchi Della Rocca, 1992]. A similar argument shows decidability for $\mathcal{EHR}$ system. The question remains open for the systems $\mathcal{HL}$, $\mathcal{HR}$, $\mathcal{S}c$, $\mathcal{CDZ}$, $\mathcal{DHM}$.

A related issue is the inhabitation problem for intersection types of rank at most 3, according to the classification of [Leivant, 1983]. The undecidability proof in [Urzyczyn, 1999] works for rank 4, the problem for rank 2 is decidable, but EXPTIME-hard [Kuśmierek, 2007].

Yet another open question is the inhabitation problems for systems with recursive intersection types, cf. Problem 11.

Partial solution: The inhabitation problem turned out to be undecidable for types of rank 3 and EXPSPACE-complete for rank 2 [Urzyczyn, 2009].

Problem # 14

Submitted by Simona Ronchi Della Rocca *Date:* 1994

Statement. For every n, determine the set of terms which can be typed in the stratified polymorphic type assignment system of order n.

Problem Origin. First posed by Paola Giannini and Simona Ronchi Della Rocca

Giannini and Ronchi Della Rocca, in [Giannini and Ronchi Della Rocca, 1994], defined a complete stratification of the polymorphic type assignment system for λ-calculus, indexed by integers. The stratification of order $n \geq 0$ is obtained by restricting the rule for eliminating the

universal quantifier in the following way:

$$\frac{\Gamma \vdash_n M : \forall\alpha.\sigma \quad \text{the level of } \alpha \text{ in } \sigma \text{ is } \leq n}{\Gamma \vdash_n M : \sigma[\tau/\alpha]}$$

where the level of a variable in a type is defined as follows:

i) if α does not occur free in σ, then the level of α in σ is 0;

ii) the level of α in α is 1;

iii) the level of α in $\forall\beta.\tau$ is the level of α in τ;

iv) the level of α in $\tau_1 \to \tau_2$ is $max\{n_1, n_2\} + 1$, where n_1 and n_2 are respectively the levels of α in τ_1 and τ_2 respectively.

It turns out that $\vdash_0$ has the same typing power of the Curry type assignment system, and $\vdash_1$ assigns types to all and only the normal forms. Find a characterization of the set of terms typable in the system indexed by n, for $n \geq 2$.

Problem # 15

Submitted by Alex Simpson *Date:* 1995

Statement. Is the equational theory of typed λ-calculus with finite sums the maximum consistent typically-ambiguous congruence?

Problem Origin. The problem was posed by Alex Simpson and probably others.

Extend the simply-typed λ-calculus with syntax for finite product and sum types. This has a standard notion of β-equality, $=_\beta$. By $=_{\beta\eta}$ we mean the equational theory induced semantically by interpretations in bicartesian closed categories (cartesian closed categories with finite coproducts). It is not hard to axiomatize this equality. Following [Statman, 1983], a typed congruence relation $\sim$, is said to be *typically ambiguous* if it is preserved under type substitutions; that is: $s \sim t\colon \sigma$ implies $s^* \sim t^*\colon \sigma^*$, where the $(\cdot)^*$ transformation performs a substitution of types for atomic types. The main question is:

> *Is $=_{\beta\eta}$ maximum amongst consistent typically-ambiguous congruence relations containing $=_\beta$?*

A (possibly) weaker version of the question is simply: is it the case that the only typically-ambiguous congruence relation properly extending $=_{\beta\eta}$ is the inconsistent relation? There are various semantic equivalents to and consequences of a positive answer to this latter question. An equivalent statement is: given any bicartesian-closed category $\mathcal{C}$ that is not a preorder, the equational theory induced by all interpretations in $\mathcal{C}$ is exactly $=_{\beta\eta}$; see [Simpson, 1995] for a similar equivalence for the calculus without coproducts. In fact, it is enough to consider only the category of finite sets; that is, another equivalent is: the equational theory induced by interpretations in the category of finite sets is exactly $=_{\beta\eta}$. (To show that this implies that $=_{\beta\eta}$ is maximal amongst consistent typically-ambiguous congruences, one encodes the category of finite sets within the type theory using finite coproducts of the unit type.) The last equivalent above is simply the finite model property for $=_{\beta\eta}$, a standard consequence of which is decidability. A related result, the decidability of $=_{\beta\eta}$ in the case of non-empty coproducts is known from [Ghani, 1995] and [Altenkirch et al., 2001]. Again for non-empty sums only, the completeness of $=_{\beta\eta}$ with respect to interpretations in the category of sets was shown by [Dougherty and Subrahmanyam, 2000]; their argument makes essential use of infinite sets. A fragment of completeness relative to finite sets has been shown by [Altenkirch and Uustalu, 2004], who established completeness and decidability results for the extension of typed λ-calculus with a base type of booleans, but without other atomic types.

Problem # 16

Submitted by Jakob Rehof *Date:* 1996
Statement. Is the subtype entailment problem decidable?
Problem Origin. First stated in [Pottier, 1996] and [Trifonov and Smith, 1996]. The present formulation is from [Henglein and Rehof, 1998].

We ask if the entailment problem with simple subtyping constraints over non-structurally ordered trees is decidable. Non-structurally ordered trees have a least element, $\perp$, and a greatest element, $\top$, which can be compared to any tree regardless of its tree domain (shape). Simple type expressions, τ, are finite terms built from $\perp$, $\top$ and a binary constructor. Such expressions can be interpreted as denoting trees, and formal inequality constraints of the form $\tau \leq \tau'$ can consequently be valuated in the non-structural order on trees. For a finite set of constraints $C = \{\tau_i \leq \tau_i'\}_{i=1\ldots n}$ of constraints and terms τ and τ', we consider the entailment $C \models \tau \leq \tau'$, or, equivalently, validity of the first-order Horn implication $\forall\vec{\alpha}.(\bigwedge_{i=1}^{n} \tau_i \leq \tau_i') \rightarrow \tau \leq \tau'$, where $\vec{\alpha}$ are the variables occurring in C, τ and τ'.

The problem first appears in slightly different forms in the papers [Pottier, 1996] and [Trifonov and Smith, 1996] for the purpose of simplifying subtyping constraints. It was studied in the form presented here in [Henglein and Rehof, 1998] (following the formulation in [Henglein and Rehof, 1997] for simple types), where it was shown to be to be PSPACE-hard. The full first-order theory of subtyping constraints has been shown to be undecidable [Su et al., 2002], but the question of decidability of entailment remains open [Rehof, 1998]. Further references can be found in [Rehof, 1998] and [Niehren and Priesnitz, 2003].

Problem # 17

Submitted by Paweł Urzyczyn *Date:* 1999
Statement. Can recursive types be defined in system F?

It was shown by Geuvers and Spławski that inductive types in the style of [Mendler, 1991] are definable in the polymorphic $\beta\eta$-lambda-calculus. It is however an open question whether *recursive* types as in [Mendler, 1987] are also definable. In [Spławski and Urzyczyn, 1999] it is shown that system F with β-conversion only is not sufficient for this purpose. See also the paper [Regnier and Urzyczyn, 2001] and Problem 21.

Problem # 18

Submitted by Mariangiola Dezani-Ciancaglini *Date:* 2001–2005
Statement. Find trees representing contextual equivalences
Problem Origin. Stated by Mariangiola Dezani, Paula Severi and Fer-Jan de Vries.

If $\mathcal{P}$ is a set of λ-terms, the contextual equivalence $M \sim_{\mathcal{P}} N$ is defined by:

$$C[M] \longrightarrow M' \in \mathcal{P} \text{ if and only if } C[N] \longrightarrow N' \in \mathcal{P} \text{ for all contexts } C[\,],$$

where $\longrightarrow$ denotes β-reduction.

[Wadsworth, 1976] shows that $\sim_{\mathcal{H}}$, where $\mathcal{H}$ is the set of head normal forms, coincides with Böhm trees equality (up to infinite η's). [Hyland, 1976] shows that $\sim_{\mathcal{N}}$, where $\mathcal{N}$ is the set of normal forms, coincides with Böhm trees equality (up to finite η's).

The question is to find tree representations of λ-terms whose equalities coincide with the following contextual equivalences:

1. $\sim_{\mathcal{W}}$ where $\mathcal{W}$ is the set of weak head normal forms,
2. $\sim_{\mathcal{T}}$, where $\mathcal{T}$ is the set of top normal forms,
3. $\sim_{\mathcal{SA}}$, where $\mathcal{SA}$ is the set of strongly active terms,
4. $\sim_{\mathcal{SA}_X}$, where X is a set of λ-terms and $\mathcal{SA}_X$ is the set of strongly active terms depending on X,
5. $\sim_{\mathcal{HA}}$, where $\mathcal{HA}$ is the set of head active terms.

The set $\mathcal{T}$ is defined in [Berarducci, 1996], and the sets $\mathcal{SA}$, $\mathcal{SA}_X$, $\mathcal{HA}$ are defined in [Severi and de Vries, 2005, Kennaway et al., 2005].

Problem # 19 [SOLVED]

Submitted by Mariangiola Dezani-Ciancaglini *Date:* 2002
Statement. Does easiness imply simple easiness?
Problem Origin. The problem was posed by Fabio Alessi and Mariangiola Dezani-Ciancaglini.

According to [Jacopini, 1975] a closed term E is *easy* if, for any other closed term M, the theory $\lambda\beta + \{M = E\}$ is consistent.

[Alessi and Lusin, 2002] introduce the notion of *simple easiness*: roughly a term M is simple easy if given an arbitrary intersection type τ one can find a suitable pre-order on types which allows to derive τ for M. In the same paper the authors show that for each simple easy term E and for each arbitrary closed term M it is possible to build a λ-model in which the interpretations of E and M coincide.

Clearly each simple easy term is easy, but the vice versa is open.

Remark: The content of [Alessi et al., 2004] are some applications of simple easiness.

Solution: Alberto Carraro and Antonino Salibra announced a solution in February 2010, the solution is published in [Carraro and Salibra, 2012].

Problem # 20 [SOLVED]

Submitted by Mariangiola Dezani-Ciancaglini *Date:* 2006
Statement. Type theoretic characterisation of hereditary permutations

According to [Bergstra and Klop, 1980, Barendregt, 1984] (Definition 21.2.9) a closed lambda-term M is a *hereditary permutation* if the Böhm tree $BT(M)$ has the following properties:

- the label $\perp$ does not occur in $BT(M)$;
- each variable occurs exactly once;

- the head variable is the first abstracted variable;
- all other variables occur at one level lower than their abstractions.

The problem is to find a type assignment system in which all and only the hereditary permutations get types of a fixed shape.
Many other sets of λ-terms have been characterised in this way, see for example [Kurata, 2002, Dezani-Ciancaglini et al., 2005] and the references there.

Solution: Makoto Tatsuta proved [Tatsuta, 2008] that the set of all hereditary permutations is not recursively enumerable, and therefore cannot be characterized by typability in a finitary type-assignment system. On the other hand, there is a system such that a term can be assigned a certain countably infinite family of types in that system if and only if it is a hereditary permutation.

Problem # 21

Submitted by Paweł Urzyczyn *Date:* 2007
Statement. Is higher-order matching decidable with many atoms?

It has recently been proved [Stirling, 2009] that the higher-order matching problem is decidable. However, the proof method only applies to the "classical" case of the problem: when all types are built from a single atom. Therefore, the problem remains open for the generalized case of types built from an arbitrary number of type variables.

A special case of the higher-order matching problem is the *retraction problem*, first mentioned in [de'Liguoro et al., 1992]. We say that a type ρ is a *retract* of a type τ (write $\rho \lhd \tau$) iff there exists a type environment Γ and terms

$$\Gamma \vdash F : \rho \to \tau \qquad \text{and} \qquad \Gamma \vdash G : \tau \to \rho$$

such that $G \circ F =_{\beta\eta} \mathbf{I}$. The problem to decide if $\rho \lhd \tau$ holds for given ρ and τ is decidable for a single atom [Padovani, 2001] but for many atoms it remains an open question. For a discussion see [Regnier and Urzyczyn, 2001].

Even less is known about polymorphic retractions. For instance, it is conjectured that $\rho \lhd \tau$ and $\tau \lhd \rho$ implies that ρ and τ are isomorphic. A confirmation of this conjecture yields a negative answer to Problem 17, see [Regnier and Urzyczyn, 2001].

Problem # 22

Submitted by Furio Honsell *Date:* April 20 2007
Statement. Is there a continuously complete CPO model of the λ-calculus whose theory is precisely $\lambda\beta\eta$ or $\lambda\beta$?
Problem Origin. I asked myself this question in 1983. In 1984, on different occasions I asked it to Dana Scott and Gordon Plotkin. Both told me that they had already thought about it.

A CPO model is continuously complete if all Scott-continuous self-maps are represented by a point in the model. Continuously complete models are sometimes called retract models or reflexive models.

The theory of a model is the set of all equations between closed λ-terms which hold in the model. The problem can be phrased equivalently as:

- Is $\lambda\beta$ (or $\lambda\beta\eta$) the only equations which hold in all (extensional) continuously complete CPO models of λ-calculus? I.e. Are retract models complete for λ-calclulus?
- Is there a filter model whose theory is precisely $\lambda\beta\eta$ or $\lambda\beta$?

There are many related results, e.g.:
There exists a ω_1-continuously complete ω_1-CPO model whose theory is precisely $\lambda\beta\eta$ [Di Gianantonio et al., 1995].
There exist theories which do not have continuously complete CPO-models [Honsell and Ronchi Della P

Problem # 23

Submitted by Robin Adams *Date:* 1992
Statement. Does Expansion Postponement hold for every PTS?
Problem Origin. This property was first conjectured by Pollack [Pollack, 1992] .

We can replace the usual conversion rule in a Pure Type System (PTS):

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash B : s \qquad A =_\beta B}{\Gamma \vdash M : B}$$

with the following two rules:

$$\text{(reduction)}\ \frac{\Gamma \vdash M : A \qquad A \twoheadrightarrow_\beta B}{\Gamma \vdash M : B}$$

$$\text{(expansion)}\frac{\Gamma \vdash M : B \qquad \Gamma \vdash B : s \qquad B \twoheadrightarrow_\beta A}{\Gamma \vdash M : A}$$

Expansion Postponement (EP) is the conjecture that every derivable judgement in a PTS has a derivation in which the expansion rule is used at most once, as the final step in the derivation.

This was proposed by Pollack in 1992 [Pollack, 1992], where he showed that it is a necessary condition for the type-checking algorithm given there to be complete. Gutiérrez and Ruiz [Gutiérrez and Ruiz, 2003] proved that it is a necessary condition for cut elimination in a sequent calculus formulation of PTSs.

It was shown by Erik Poll [Poll, 1998] to hold for every weakly normalising PTS, and by Barthe [Barthe, 1998b] to hold for injective PTSs.

Problem # 24

Submitted by Corrado Böhm *Date:* 2013
Statement. On the equational meaning of deeds
Problem Origin. The problem has been posed first by Corrado Böhm. It is added to the list to celebrate his 90th birthday.

A deed is a closed normal form with only one initial abstraction, namely a deed has shape $\lambda x.xP_1 \dots P_n$ where $n \geq 0$.

It is easy to verify that each equation of the shape

$$D_1 X = D_2 X,$$

where D_1, D_2 are deeds, can be solved, for instance by choosing a fixed point of the combinator K as X.

It it possible to define a one-one mapping between arbitrary normal forms and deeds, as follows. Given a normal form M with $m \geq 0$ initial abstractions, the deed corresponding to M is:

$$D^M \equiv \lambda x.\, M(xc_1) \ldots (xc_m),$$

where $c_1, \ldots, c_m$ are the Church numerals or any distinct closed beta-eta normal forms. Note that the mapping is injective by Böhm's theorem.

The question is as follows: What can we understand about the solution of the equation $MX = NX$, where M, N are arbitrary normal forms, by looking at the equation $D^M X = D^N X$?

Problem # 25

Submitted by Benedetto Intrigila *Date:* 2000
Statement. How many fixed points can a combinator have?
Problem Origin. First posed in [Intrigila and Biasone, 2000].

The question is how many fixed points can a combinator (*i.e.* a closed term) have in the $\lambda\beta$-calculus. In [Intrigila and Biasone, 2000] it is proved that, if a combinator has a fixed point in normal form then, it has either an infinite number of fixed points or exactly one fixed point.

Problem # 26

Submitted by Henk Barendregt *Date:* 2014
Statement. Assign (in an 'easy' way) ordinals to terms of the simply typed lambda calculus such that reduction of the term yields a smaller ordinal.
Problem Origin. First posed by Kurt Gödel.

Construct an 'easy' assignment of (possibly trans-finite) ordinals to terms of the simply typed lambda calculus, i.e. a map

$$\#\colon \Lambda_{\to} \to \{\alpha \mid \alpha \text{ is an ordinal}\}, \tag{1}$$

such that

$$\forall M, N \in \Lambda_{\to} [\, M \to_{\beta} N \;\Rightarrow\; \#M > \#N \,]. \tag{2}$$

By the fact that the ordinals are well-ordered, this immediately shows that β-reduction on simply typed lambda terms is strongly normalizing (SN).
Comments.

1. This item in the list of open problems is actually not a problem but a (mathematical) 'koan'. A *problem*, according to [Pólya, 2004], should have the property that a candidate solution is clearly recognizable as such in a well-defined way. The proof of correctness of the candidate solution still may be hard. A *koan*, according to Jan Willem Klop, is a non-precise question for which the space of solutions is *a priori* not clear[1]. But once a right solution is given, it should be recognizable as such. For example, two centuries ago it was a mathematical koan what is the right definition of the notion 'continuous function' on the real numbers. Or one century ago, there was the koan 'Why do some isomorphisms feel to be more natural than other ones?' The pondering over this question by Eilenberg and MacLane led to the introduction of the notion of a category. The present item is a koan, because the notion of 'easy' in its formulation is not well-defined.

2. Once we know SN($\to_{\beta}$) on $\Lambda_{\to}$, we can define

$$\#M = \text{the length of the longest path from } M \text{ to normal form.}$$

This assignment satisfies (+) and even assigns finite ordinals to terms. But the purpose of this 'koan' is to prove SN in an easy way and this 'solution' presupposes that SN holds.

3. In [de Vrijer, 1987] the value $\#M$ of (ii) is defined by an analysis of M. But the construction is not 'easy'.

4. Alan Turing has given a simple map (1) such that for all M not in β normal form there exists a reduct N such that $\#M > \#N$. This establishes weak normalization (WN), but not SN. In "*Alan Turing: His Work and Impact*", see [Barendregt and Manzonetto, 2013].

5. W.A. Howard (see [Howard, 1970]), to whom (personal communication) Gödel had originally asked the present koan for the system T (also including numerals and the recursor) and [Wilken and Weiermann, 2012], made significant steps (but not a final one) towards solving this item. Howard's construction works only for combinatory terms. Wilken and Weiermann, extending Howard's work, assigns ordinals to terms and their previous reduction past; this establishes SN, but doesn't solve Gödel's koan.

[1] Klop was inspired by the Zen Buddhist Rinzai, who introduced Sanzen, the practice in which the master asks the trainee a seemingly impossible question, called *koan*, that nevertheless has a convincing but unexpected answer. The intention is to help the student 'to think outside the box'.

Full Bibliography

[Alessi et al., 2006] Alessi, F., Barbanera, F., and Dezani-Ciancaglini, M. (2006). Intersection types and lambda models. *Theoretical Computer Science*, 355(2):108–126.

[Alessi et al., 2004] Alessi, F., Dezani-Ciancaglini, M., and Lusin, S. (2004). Intersection types and domain operators. *Theoretical Computer Science*, 316(1–3):25–47.

[Alessi and Lusin, 2002] Alessi, F. and Lusin, S. (2002). Simple easy terms. In van Bakel, S., editor, *Intersection Types and Related Systems*, volume 70 of *Electronic Notes in Computer Science*. Elsevier.

[Altenkirch et al., 2001] Altenkirch, T., Dybjer, P., Hofmann, M., and Scott, P. (2001). Normalization by evaluation for typed lambda calculus with coproducts. In *Logic in Computer Science*, pages 303–310. IEEE Computer Society Press.

[Altenkirch and Uustalu, 2004] Altenkirch, T. and Uustalu, T. (2004). Normalization by evaluation for $\lambda^{\to 2}$. In *Functional and Logic Programming*, volume 2998 of *Lecture Notes in Computer Science*, pages 260–275.

[Anderson, 1960] Anderson, A. R. (1960). Entailment shorn of modality. *Journal of Symbolic Logic*, 25:388. (Abstract).

[Anderson and Belnap, 1975] Anderson, A. R. and Belnap, N. D. (1975). *Entailment. The Logic of Relevance and Necessity, Volume 1.* Princeton University Press, U.S.A.

[Anderson et al., 1992] Anderson, A. R., Belnap, N. D., and Dunn, J. M. (1992). *Entailment. The Logic of Relevance and Necessity, Volume 2.* Princeton University Press, U.S.A.

[Barendregt, 1975] Barendregt, H. (1975). Open problems. In [Böhm, 1975], pages 367–370.

[Barendregt, 1984] Barendregt, H. (1984). *The Lambda Calculus. Its Syntax and Semantics*. North-Holland, second edition.

[Barendregt et al., 1983] Barendregt, H., Coppo, M., and Dezani-Ciancaglini, M. (1983). A filter lambda model and the completeness of type assignment. *Journal of Symbolic Logic*, 48(4):931–940.

[Barendregt and Manzonetto, 2013] Barendregt, H. and Manzonetto, G. (2013). Turing's contributions to lambda calculus. In Cooper, S. B. and van Leeuwen, J., editors, *Alan Turing: His Work and Impact*, pages 139–143. Elsevier.

[Barthe, 1995] Barthe, G. (1995). Extensions of Pure Type Systems. In [Dezani-Ciancaglini and Plotkin, 1995], pages 16–31.

[Barthe, 1998a] Barthe, G. (1998a). The semi-full closure of Pure Type Systems. In Brim, L., Gruska, J., and Zlatuska, J., editors, *Proceedings of MFCS'98*, volume 1450 of *Lecture Notes in Computer Science*, pages 316–325. Springer-Verlag.

[Barthe, 1998b] Barthe, G. (1998b). Type-checking injective pure type systems. *Journal of Functional Programming*, 9:675–698.

[Barthe et al., 2001] Barthe, G., Hatcliff, J., and Sørensen, M. H. (2001). Weak normalization implies strong normalization in a class of non-dependent pure type systems. *Theoretical Computer Science*, 269(1–2):317–361.

[Berarducci, 1996] Berarducci, A. (1996). Infinite λ-calculus and non-sensible models. In Ursini, A. and Aglianò, P., editors, *Logic and Algebra (Pontignano, 1994)*, pages 339–377. Dekker.

[Bergstra and Klop, 1980] Bergstra, J. A. and Klop, J. W. (1980). Invertible terms in the lambda calculus. *Theoretical Computer Science*, 11:19–37.

[Berline, 2000] Berline, C. (2000). The lambda-calculus and its webbed models. *Theoretical Computer Science*, 249(1):81–161.

[Berline, 2006] Berline, C. (2006). Graph models of lambda-calculus at work, and variations. *Mathematical Structures in Computer Science*, 16:185–221.

[Bimbó, 2005] Bimbó, K. (2005). Types of I-free hereditary right maximal terms. *Journal of Philosophical Logic*, 34:607–620.

[Bimbó, 2006] Bimbó, K. (2006). Relevance logics. In Jacquette, D., editor, *Philosophy of Logic*, volume 5 of *Handbook of the Philosophy of Science*, pages 723–789. North-Holland/ Elsevier, Amsterdam.

[Bimbó and Dunn, 2013] Bimbó, K. and Dunn, J. M. (2013). On the decidability of implicational ticket entailment. *Journal of Symbolic Logic*, 78(1):214–236.

[Böhm, 1975] Böhm, C., editor (1975). *λ-Calculus and Computer Science Theory*, volume 37 of *Lecture Notes in Computer Science*. Springer-Verlag.

[Broda and Damas, 1997] Broda, S. and Damas, L. (1997). On combinatory complete sets of proper combinators. *Journal of Functional Programming*, 7(6):593–612.

[Broda et al., 2004] Broda, S., Damas, L., Finger, M., and e Silva, P. S. (2004). The decidability of a fragment of BB'IW-logic. *Theoretical Computer Science*, 318(3):373–408.

[Carraro and Salibra, 2012] Carraro, A. and Salibra, A. (2012). Easy lambda-terms are not always simple. *RAIRO - Theoretical Informatics and Applications*, 46:291–314.

[Curry and Feys, 1958] Curry, H. B. and Feys, R. (1958). *Combinatory Logic, Volume I*. North-Holland, Amsterdam. (3rd edition 1974).

[Curry et al., 1972] Curry, H. B., Hindley, J. R., and Seldin, J. P. (1972). *Combinatory Logic, Volume II*. North-Holland, Amsterdam.

[David, 2009] David, R. (2009). A direct proof of the confluence of combinatory strong reduction. *Theoretical Computer Science*, 410(42):4204–4215.

[de Vrijer, 1987] de Vrijer, R. (1987). Exactly estimating functionals and strong normalization. *Indagationes Mathematicae*, 49:479–493.

[de'Liguoro et al., 1992] de'Liguoro, U., Piperno, A., and Statman, R. (1992). Retracts in simply typed lambda-beta-eta-calculus. In *Logic in Computer Science*, pages 461–469. IEEE Computer Society Press.

[Dezani-Ciancaglini et al., 2005] Dezani-Ciancaglini, M., Honsell, F., and Motohama, Y. (2005). Compositional characterization of λ-terms using intersection types. *Theoretical Computer Science*, 340(3):459–495.

[Dezani-Ciancaglini and Plotkin, 1995] Dezani-Ciancaglini, M. and Plotkin, G., editors (1995). *Typed Lambda Calculi and Applications*, volume 902 of *Lecture Notes in Computer Science*. Springer-Verlag.

[Di Gianantonio et al., 1995] Di Gianantonio, P., Honsell, F., and Plotkin, G. (1995). Uncountable limits and the lambda calculus. *Nordic Journal of Computing*, 2(2):126–145.

[Dougherty and Subrahmanyam, 2000] Dougherty, D. and Subrahmanyam, R. (2000). Equality between functionals in the presence of coproducts. *Information and Computation*, 157:52–83.

[Geuvers, 1993] Geuvers, H. (1993). *Logics and Type Systems*. PhD thesis, Nijmegen University.

[Ghani, 1995] Ghani, N. (1995). Beta-eta equality for coproducts. In [Dezani-Ciancaglini and Plotkin, 1995], pages 171–185.

[Giannini and Ronchi Della Rocca, 1994] Giannini, P. and Ronchi Della Rocca, S. (1994). A type inference algorithm for a complete stratification of the polymorphic type discipline. *Information and Computation*, 110:115 – 173.

[Girard, 1986] Girard, J.-Y. (1986). The system F of variable types, fifteen years later. *Theoretical Computer Science*, 45:159–192.

[Gørtz et al., 2003] Gørtz, I. L., Reuss, S., and Sørensen, M. H. (2003). Strong normalization from weak normalization by translation into the lambda-I-calculus. *Higher Order Symbol. Comput.*, 16(3):253–285.

[Gutiérrez and Ruiz, 2003] Gutiérrez, F. and Ruiz, B. (2003). Cut elimination in a class of sequent calculi for pure type systems. *Electronic Notes in Computer Science*, 84:105–116.

[Henglein and Rehof, 1997] Henglein, F. and Rehof, J. (1997). The complexity of subtype entailment for simple types. In *Logic in Computer Science*, pages 352–361. IEEE Computer Society Press.

[Henglein and Rehof, 1998] Henglein, F. and Rehof, J. (1998). Constraint automata and the complexity of recursive subtype entailment. In *International Colloquium on Automata, Languages, and Programming*, volume 1443 of *Lecture Notes in Computer Science*, pages 616–627. Springer-Verlag.

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

[Honsell and Ronchi Della Rocca, 1992] Honsell, F. and Ronchi Della Rocca, S. (1992). An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus. *Journal of Computer and System Sciences*, 45(1):49–75.

[Howard, 1970] Howard, W. (1970). Assignment of ordinals to terms for primitive recursive functionals of finite type. In A. Kino, J. M. and Vesley, R., editors, *Intuitionism and Proof Theory: Proceedings of the Summer Conference at Buffalo N.Y. 1968*, volume 60 of *Studies in Logic and the Foundations of Mathematics*, pages 443 – 458. Elsevier.

[Hyland, 1976] Hyland, M. (1976). A syntactic characterization of the equality in some models for the lambda calculus. *J. London Math. Soc. (2)*, 12(3):361–370.

[Intrigila and Biasone, 2000] Intrigila, B. and Biasone, E. (2000). On the number of fixed points of a combinator in lambda calculus. *Mathematical Structures in Computer Science*, 10(5):595–615.

[Intrigila and Statman, 2011] Intrigila, B. and Statman, R. (2011). Solution to the range problem for combinatory logic. *Fundamenta Informaticae*, 111(2):203–222.

[Jacopini, 1975] Jacopini, G. (1975). A condition for identifying two elements of whatever model of combinatory logic. In [Böhm, 1975], pages 213–219.

[Kennaway et al., 2005] Kennaway, J., Severi, P., Sleep, M., and de Vries, F. J. (2005). Infinite rewriting: from syntax to semantics. In *Processes, Terms and Cycles: Steps on the Road to Infinity: Essays Dedicated to Jan Willem Klop on the Occasion of His 60th Birthday*, volume 3838 of *Lecture Notes in Computer Science*, pages 148–172. Springer-Verlag.

[Ketema et al., 2005] Ketema, J., Klop, J. W., and van Oostrom, V. (2005). Vicious circles in orthogonal term rewriting systems. *Electronic Notes in Computer Science*, 124(2):65–77.

[Kurata, 2002] Kurata, T. (2002). Intersection and singleton type assignment characterizing finite Böhm-trees. *Information and Computation*, 178(1):1–11.

[Kurata and Takahashi, 1995] Kurata, T. and Takahashi, M. (1995). Decidable properties of intersection type systems. In [Dezani-Ciancaglini and Plotkin, 1995], pages 297–311.

[Kuśmierek, 2007] Kuśmierek, D. (2007). The inhabitation problem for rank two intersection types. In Ronchi Della Rocca, S., editor, *Typed Lambda Calculi and Applications*, volume 4583 of *Lecture Notes in Computer Science*, pages 240–254. Springer-Verlag.

[Leivant, 1983] Leivant, D. (1983). Polymorphic type inference. In *Symposium on Principles of Programming Languages*, pages 88–98. ACM Press.

[Malecki, 1997] Malecki, S. (1997). Proofs in system F_ω can be done in system F_ω^1. In van Dalen, D. and Bezem, M., editors, *Computer Science Logic*, volume 1258 of *Lecture Notes in Computer Science*, pages 297–315. Springer-Verlag.

[McCune and Wos, 1991] McCune, W. and Wos, L. (1991). The absence and the presence of fixed point combinators. *Theoretical Computer Science*, 87:221–228.

[Mendler, 1987] Mendler, N. (1987). Recursive types and type constraints in second-order lambda calculus. In *Logic in Computer Science*, pages 30–36. IEEE Computer Society Press.

[Mendler, 1991] Mendler, N. (1991). Inductive types and type constraints in the second-order lambda calculus. *Annals of Pure and Applied Logic*, 51(1-2):159–172.

[Minari, 2009] Minari, P. (2009). A solution to Curry and Hindley's problem on combinatory strong reduction. *Archive for Mathematical Logic*, 43(2):159–184.

[Niehren and Priesnitz, 2003] Niehren, J. and Priesnitz, T. (2003). Non-structural subtype entailment in automata theory. *Information and Computation*, 186(2):319–354.

[Padovani, 2001] Padovani, V. (2001). Retracts in simple types. In Abramsky, S., editor, *Typed Lambda Calculi and Applications*, volume 2044 of *Lecture Notes in Computer Science*, pages 376–384. Springer-Verlag.

[Padovani, 2013] Padovani, V. (2013). Ticket entailment is decidable. *Mathematical Structures in Computer Science*, 23(3):568–607.

[Poll, 1998] Poll, E. (1998). Theoretical pearl: Expansion postponement for normalising pure type systems. *Journal of Functional Programming*, 8:89–96.

[Pollack, 1992] Pollack, R. (1992). Typechecking in Pure Type Systems. In *Informal Proceedings of the 1992 Workshop on Types for Proofs and Programs, Bastad, Sweden*, pages 271–288.

[Polonsky, 2012] Polonsky, A. (2012). The range property fails for H. *Journal of Logic and Computation*, 77(4):1195–1210.

[Pólya, 2004] Pólya, G. (2004). *How to Solve It: A New Aspect of Mathematical Method*. Penguin mathematics. Princeton University Press. Expanded version of the 1988 edition, with a new foreword by John H. Conway.

[Pottier, 1996] Pottier, F. (1996). Simplifying subtyping constraints. In *International Conference on Functional Programming*, pages 122–133. ACM Press.

[Regnier and Urzyczyn, 2001] Regnier, L. and Urzyczyn, P. (2001). Retractions of types with many atoms. Technical Report 29, IML Marseille. `http://arxiv.org/abs/cs/0212005`.

[Rehof, 1998] Rehof, J. (1998). *The Complexity of Simple Subtyping Systems*. PhD thesis, Department of Computer Science, University of Copenhagen.

[Severi and de Vries, 2005] Severi, P. and de Vries, F. J. (2005). Order structures for Böhm-like models. In *Computer Science Logic*, volume 3634 of *Lecture Notes in Computer Science*, pages 103–116. Springer-Verlag.

[Simpson, 1995] Simpson, A. (1995). Categorical completeness results for simply-typed λ-calculus. In [Dezani-Ciancaglini and Plotkin, 1995], pages 414–427.

[Smullyan, 1985] Smullyan, R. (1985). *To Mock a Mockingbird*. Knopf.

[Sørensen, 1997] Sørensen, M. (1997). *Normalization in λ-Calculus and Type Theory*. PhD thesis, Department of Computer Science, University of Copenhagen. Available as DIKU Rapport 97/27.

[Spławski and Urzyczyn, 1999] Spławski, Z. and Urzyczyn, P. (1999). Type fixpoints: Iteration vs. recursion. In *International Conference on Functional Programming*, pages 102–113. ACM Press.

[Statman, 1983] Statman, R. (1983). λ-definable functionals and $\beta\eta$ conversion. *Archiv für Mathematische Logik und Grundlagenforschung*, 23:21–26.

[Statman, 1986] Statman, R. (1986). On translating lambda terms into combinators. In *Logic in Computer Science*, pages 378–382. IEEE Computer Society Press.

[Statman, 1988a] Statman, R. (1988a). Combinators hereditarily of order one. Technical Report 88-32, Department of Mathematics, Carnegie Mellon University.

[Statman, 1988b] Statman, R. (1988b). Combinators hereditarily of order two. Technical Report 88-33, Department of Mathematics, Carnegie Mellon University.

[Statman, 1989] Statman, R. (1989). The word problem for Smullyan's Lark combinator is decidable. *Journal of Symbolic Computation*, 7(2):103–112.

[Statman, 1993] Statman, R. (1993). Some examples of non-existent combinators. *Theoretical Computer Science*, 121:441–448.

[Statman, 1997] Statman, R. (1997). Effective reduction and conversion strategies for combinators. In *Rewriting Techniques and Applications*, volume 1232 of *Lecture Notes in Computer Science*, pages 299–307. Springer-Verlag.

[Statman, 2000] Statman, R. (2000). On the word problem for combinators. In *Rewriting Techniques and Applications*, volume 1833 of *Lecture Notes in Computer Science*, pages 203–213. Springer-Verlag.

[Stirling, 2009] Stirling, C. (2009). Decidability of higher-order matching. *Logical Methods in Computer Science*, 5(3).

[Su et al., 2002] Su, Z., Aiken, A., Niehren, J., Priesnitz, T., and Treinen, R. (2002). The first-order theory of subtyping constraints. In *Symposium on Principles of Programming Languages*, pages 203–216. ACM Press.

[Tatsuta, 2008] Tatsuta, M. (2008). Types for hereditary permutators. In *Proceedings of the Twenty-Third Annual IEEE Symposium on Logic in Computer Science, LICS 2008, 24-27 June 2008, Pittsburgh, PA, USA*. http://nii.ac.jp/TechReports/07-010E.html.

[Trifonov and Smith, 1996] Trifonov, V. and Smith, S. (1996). Subtyping constrained types. In *Static Analysis Symposium*, volume 1145 of *Lecture Notes in Computer Science*, pages 349–365. Springer-Verlag.

[Trigg et al., 1994] Trigg, P., Hindley, J. R., and Bunder, M. W. (1994). Combinatory abstraction using B, B' and friends. *Theoretical Computer Science*, 135:405–422.

[Urzyczyn, 1996] Urzyczyn, P. (1996). Positive recursive type assignment. *Fundamenta Informaticae*, 28(1-2):197–209.

[Urzyczyn, 1997] Urzyczyn, P. (1997). Type reconstruction in F_ω. *Mathematical Structures in Computer Science*, 7(4):329–358.

[Urzyczyn, 1999] Urzyczyn, P. (1999). The emptiness problem for intersection types. *Journal of Symbolic Logic*, 64(3):1195–1215. (Preliminary version in *Proc. LICS 94*).

[Urzyczyn, 2009] Urzyczyn, P. (2009). Inhabitation of low-rank intersection types. In Curien, P.-L., editor, *Typed Lambda Calculi and Applications*, volume 5608 of *Lecture Notes in Computer Science*, pages 356–370. Springer-Verlag.

[Wadsworth, 1976] Wadsworth, C. P. (1976). The relation between computational and denotational properties for Scott's D_∞-models of the lambda-calculus. *SIAM Journal of Computing*, 5(3):488–521.

[Wilken and Weiermann, 2012] Wilken, G. and Weiermann, A. (2012). Derivation lengths classification of Gödel's T extending Howard's assignment. *Logical Methods in Computer Science*, 8(1):44.