Higher Order β Matching is Undecidable

RALPH LOADER, *58A Dryden St., Grey Lynn, Auckland 1002, New Zealand. E-Mail: rcl@ihug.co.nz*

Abstract

We show that the solvability of matching problems in the simply typed λ-calculus, up to β equivalence, is not decidable. This decidability question was raised by Huet [4].

Note that there are two variants of the question: that concerning β equivalence (dealt with here), and that concerning $\beta\eta$ equivalence.

The second of these is perhaps more interesting; unfortunately the work below sheds no light on it, except perhaps to illustrate the subtlety and difficulty of the problem.

Keywords: unification, matching, higher-order, lambda-calculus, types.

1 Introduction

Equation solving problems where the equations to be solved are between expressions in some formal language are generically referred to as *unification problems*. One is given two terms $s[X]$ and $t[X]$, with a free variable X. A solution to the problem is a term that when substituted for X, makes the two terms identical, up to an appropriate syntactical equivalence.

The classic result concerning these is the Herbrand-Robinson theorem, which shows the decidability of *first-order unification*: unification in languages given by a set of operation symbols, each denoting an operation of some finite arity, and using syntactical equality as the equivalence.

An important subclass of unification problems is comprised of *matching problems*. These are unification problems where the variable to be solved for occurs only on one side: $s[X] = t$. They arise naturally in computer implementations of formal languages. Typically, $s[X]$ is a *pattern* indicating a class of expressions on which some action can be performed, and t is an expression for which one wishes to find out if that action is appropriate.

Huet [3] showed that unification problems in the simply typed λ calculus (*higher order unification*) have no general algorithm for their solution. Later, he gave an efficient search algorithm [4] that always finds solutions when they exist and that terminates in many cases—in particular, Huet and Lang showed the algorithm decides second order matching [5]. At the same time, Huet posed the question of the decidability of matching problems in the simply typed λ calculus (*higher order matching*). This question was asked independently by Statman [12], there referred to as the *range problem*.

Higher order matching and unification problems may be classified by the order of the type of the variable to be solved for. Goldfarb [2] improved Huet's undecidability proof to show that second order unification is undecidable. Further decidability results were given by Dowek [1] for third order matching and by Padovani [9] for fourth order matching. The latter is recast in model theoretic terms by Schubert [10].

Padovani's construction of the minimal model [8] gives a decision procedure for another class of higher order matching problems—essentially those for which the RHS is a closed term (without constants) of order two. Schubert [11] extends this to order three.

For syntactical questions concerning the simply typed λ calculus ($\lambda^{\rightarrow}$), there are two relevant syntactical equivalences that can be considered: β-equivalence, which is generated by the equation[1]

$$(\lambda \mathsf{x} \,.\, b)(a) = b[\mathsf{x}/a],$$

and $\beta\eta$-equivalence, which adds the equation

$$\lambda \mathsf{x} \,.\, f\,\mathsf{x} = f.$$

The notion of $\beta\eta$-equivalence is the more interesting one; it is the syntactical equality one obtains from considering equality of representations in obvious models, and it is more robust under minor reformulations of the language. This article concerns only β-equivalence.

We consider $\lambda^{\rightarrow}$ with a single ground type 0, and no constant terms. Any type can be expressed in the form $A_1 \Rightarrow \ldots \Rightarrow A_m \Rightarrow 0$. We shall write this as $[A_1 \ldots A_m]$ when convenient. A digit n will be used to refer to the type $[0 \ldots 0]$ with 0 repeated n times.

Formally, a *higher order matching problem* is a quadruple (A, B, f, b) where:

- A and B are types of $\lambda^{\rightarrow}$.
- $f : A \Rightarrow B$ and $b : B$ are closed terms of the given types.

We shall usually write a higher order matching problem as an equation

$$f\,\mathsf{X} = b$$

with the types implicit. The *type* of the problem is A. We shall write a matching problem $(A, B, \lambda\mathsf{X}.\, a[\mathsf{X}], b)$ as an equation $a[\mathsf{X}] = b$. The resulting ambiguity in our notation is of no consequence.

A *solution* to a problem (A, B, f, b) is a closed term a of type A such that $f(a) \equiv_\beta b$. A problem is *solvable* if it has a solution. There is a corresponding notion with $\equiv_{\beta\eta}$ instead of $\equiv_\beta$, which we refer to as *higher order $\beta\eta$ matching*. We also refer to *higher order β matching* when we wish to emphasize the role of β equivalence.

The goal of this article is to prove that the solvability of higher order β matching problems is undecidable. The proof will be based on an encoding of definability problems for finite models of $\lambda^{\rightarrow}$ (see [7]).

In formulating our definition of higher order matching problems and their solutions, we made several arbitrary choices. We included only one ground type, we did not include any constants in our language, we required all terms to be closed, and we considered problems with only a single equation and a single variable. These choices make no difference to the decidability of the solvability of matching problems. The appropriate equivalences are somewhat tedious to state and prove. We limit ourselves to proving one which we actually need:

---

[1] We follow the common convention of including α conversion, but completely omitting all mention of it.

Theorem 1.1 1. For any finite set S of matching equations of some type A, there is a single matching equation $f\,\mathsf{X} = b$ which has exactly the same solutions as S. 2. The solvability of finite sets of matching equations is decidable iff higher order matching is decidable.

As is usual, we define a *solution* to a set S of equations to be a value which solves every member of S.

PROOF. Suppose that $S = \{g_i\mathsf{X} = c_i \mid 1 \leqslant i \leqslant n\}$ with $c_i : B_i$ for each i. Let $B = [B_1 \ldots B_n]$ and, following the usual λ-calculus encoding of n-tuples, let

$$f = \lambda\mathsf{X}\colon A\,.\,\lambda\mathsf{y}\colon B\,.\,\mathsf{y}(g_1\mathsf{X})\ldots(g_n\mathsf{X})$$

and

$$b = \lambda\mathsf{y}\colon B\,.\,\mathsf{y}\,c_1\ldots c_n.$$

Clearly $f\,s \equiv_\beta b$ iff $g_i\,s \equiv_\beta c_i$ for all i. This shows the first part. The second follows immediately, noting that the construction above is effective. ∎

2 Finite models and Type lifting

In [7], an undecidability result was given for definability in finite models. In what follows, we shall relate matching problems to such definability problems.

Following [7], the particular model $\mathcal{M}$ we consider has $\mathcal{M}_0$ defined to be an arbitrary 7-element set, and $\mathcal{M}_{A\Rightarrow B}$ defined to be the set of all functions from $\mathcal{M}_A$ to $\mathcal{M}_B$. Refer to [12] for background such as the interpretation $[\![a]\!] \in \mathcal{M}_A$ of a closed term a of type A.

An element $\alpha \in \mathcal{M}_A$ is *definable* if $\alpha = [\![a]\!]$ for some closed a of type A.

We also make use of $\lambda^{\rightarrow}$ extended with a single constant $\perp$ of type 0. Terms of this calculus are called $\perp$-*terms*. We extend $[\![\cdot]\!]$ to $\perp$-terms by making an arbitrary choice of $[\![\perp]\!] \in \mathcal{M}_0$. An element $\alpha \in \mathcal{M}_A$ is $\perp$-*definable* if there is a closed $\perp$-term $a : A$ such that $\alpha = [\![a]\!]$.

The word "term" is always used to refer to terms of the unextended calculus; terms cannot contain $\perp$. Where $\perp$-terms are considered, this is always mentioned explicitly.

The theorem below is the result of [7], but extended to $\perp$-definability as well as definability. The extended result can also be derived from [6].

Theorem 2.1 Definability and $\perp$-definability are each undecidable.

PROOF. The argument for definability is given in [7]. We indicate how to extend this to $\perp$-definability. Take $[\![\perp]\!]$ to be the $N \in \mathcal{M}_0$ of that paper, and modify the definition of preword-term to allow occurrences of $\perp$.

An induction over preword-terms then shows that if t is a preword-term containing $\perp$, then $[\![t]\!]_v = N$ for any valuation v. This shows that one only need consider preword-terms not containing $\perp$, and the rest of the argument is unchanged. ∎

The notion of type lifting plays an important part in the encoding of the model $\mathcal{M}$ in matching problems. For types A and D we define A^D (the *lifting by D of A*) by $0^D = D$ and $(A \Rightarrow B)^D = A^D \Rightarrow B^D$. Given a term t of type A, we also define t^D, the *lifting by type D*, to be the term of type A^D given by the lifting by type D all types within t. If A' (or t') is the lifting by type D of some type (or term), then we say that A' (or t') is *D-lifted*.

Lifting commutes with all the usual syntactical operations of $\lambda^{\rightarrow}$. For example, a has β-normal form a' iff a^D has β-normal form a'^D. Type lifting is associative: $(A^B)^C = A^{(B^C)}$.

We now encode the elements of the model $\mathcal{M}$ using closed terms of $\lambda^{\rightarrow}$, using type lifting. The construction we use is given by Statman [13]. We use the Church-style encoding of a finite set to encode $\mathcal{M}_0$, and extend to function types using a logical relations condition.

The closed normal forms of type 7 are just the *projection terms* $\pi_i = \lambda \mathsf{z}_1 \ldots \mathsf{z}_7 \colon 0.\, \mathsf{z}_i$ for $1 \leqslant i \leqslant 7$, and these are used to encode the seven elements of $\mathcal{M}_0$. This will lead to elements of $\mathcal{M}_A$ being encoded by closed terms of type A^7. Every element of $\mathcal{M}$ is encoded by some term (lemma 2.3 below), however the encoding is not a function; there are many terms representing each element. In particular, each 7-lifted closed term t^7 is the encoding of the element $[\![t]\!]$ of $\mathcal{M}$ (theorem 2.5 below).

We define the *encoding relation* $\sim$ between closed terms of type A^7 and elements of $\mathcal{M}_A$ as follows, using recursion on the type.

- Take $\sim$ to be any one of the 7! bijections between the projections $\pi_1 \ldots \pi_7$ and $\mathcal{M}_0$.
- Extend this to all closed terms of type 7 by putting $a \sim \alpha \in \mathcal{M}_0$ if and only if $\mathsf{nf}(a) \sim \alpha$ where $\mathsf{nf}(a)$ is the normal form of a.
- For $f : A^7 \Rightarrow B^7$ and $\phi \in \mathcal{M}_{A \Rightarrow B}$, we set $f \sim \phi$ if and only if $f(a) \sim \phi(\alpha)$ for all $\alpha \in \mathcal{M}_A$ and $a \sim \alpha$.

The next three lemmas confirm that this definition is well behaved.

Lemma 2.2 If $a \equiv_{\beta\eta} a' \sim \alpha$ then $a \sim \alpha$.

Proof. For closed $a, a' : 7$, if $a \equiv_{\beta\eta} a'$ then $a \equiv_{\beta} a'$ and the result follows from the definition of $\sim$. The result extends to higher types by induction on the type. ∎

Lemma 2.3 For every $\alpha \in \mathcal{M}_A$, there is a closed term $p : A^7$ such that $p \sim \alpha$.

Proof. Let $\mathsf{tt} = \lambda \mathsf{x}, \mathsf{y} \colon 0.\, \mathsf{x}$ and $\mathsf{ff} = \lambda \mathsf{x}, \mathsf{y} \colon 0.\, \mathsf{y}$, so that tt and ff are the only closed normal forms of type $2 = [0\,0]$. For each type $A = [A_1 \ldots A_m]$, let

$$\mathsf{if}_A = \lambda \mathsf{c} \colon 2\,.\, \lambda \mathsf{r}, \mathsf{r}' \colon A\,.\, \lambda \mathsf{x}_1 \colon A_1 \ldots \lambda \mathsf{x}_m \colon A_m\,.\, \mathsf{c}(\mathsf{r}\,\mathsf{x}_1 \ldots \mathsf{x}_m)(\mathsf{r}'\,\mathsf{x}_1 \ldots \mathsf{x}_m)$$

so that $\mathsf{if}_A\, \mathsf{tt}\, r\, r' \equiv_{\beta\eta} r$ and $\mathsf{if}_A\, \mathsf{ff}\, r\, r' \equiv_{\beta\eta} r'$.

We show by induction on types A that for each $\alpha \in \mathcal{M}_A$ there is $p_\alpha \sim \alpha$, and also closed $E_\alpha : A^7 \Rightarrow 2$ such that whenever $r \sim \rho \in \mathcal{M}_A$, we have $E_\alpha\, r \equiv_\beta \mathsf{tt}$ iff $\alpha = \rho$.

For $\alpha \in \mathcal{M}_0$, we can take $p_\alpha = \pi_k \sim \alpha$ for some k, as in the definition of $\sim$. Let E be

$$\lambda \mathsf{u}, \mathsf{v} \colon 7\,.\, \lambda \mathsf{x}, \mathsf{y} \colon 0\,.\, \mathsf{u}(\mathsf{v}\, \delta_{11} \ldots \delta_{17}) \ldots (\mathsf{v}\, \delta_{71} \ldots \delta_{77})$$

where $\delta_{ii} = \mathsf{x}$ and $\delta_{ij} = \mathsf{y}$ for $i \neq j$. Clearly $E\, \pi_i\, \pi_i \equiv_\beta \mathsf{tt}$ for each i and $E\, \pi_i\, \pi_j \equiv_\beta \mathsf{ff}$ for $i \neq j$, so we may take $E_\alpha = E\, p_\alpha$.

For a function type $A \Rightarrow B$, let $\alpha_1 \ldots \alpha_m$ be an enumeration of the finite set $\mathcal{M}_A$. Then for $\phi \in \mathcal{M}_{A \Rightarrow B}$, define p_ϕ to be

$$\lambda \mathsf{x} \colon A^7\,.\, \mathsf{if}_{B^7}(E_{\alpha_1}\, \mathsf{x})(p_{\phi\alpha_1})\big(\ldots\big(\mathsf{if}_{B^7}(E_{\alpha_{m-1}}\, \mathsf{x})(p_{\phi\alpha_{m-1}})\big(p_{\phi\alpha_m}\big)\big)\ldots\big).$$

Define E_ϕ to be $\lambda \mathsf{g}\colon A^7 \Rightarrow B^7.\, \lambda \mathsf{x}, \mathsf{y}\colon 0.\, e_m$ where $e_0 = \mathsf{x}$ and

$$e_i = E_{\phi\alpha_i}(\mathsf{g}\, p_{\alpha_i})(e_{i-1})(\mathsf{y})$$

for $1 \leqslant i \leqslant m$. The verification that these are correct is left to the reader. ∎

Lemma 2.4 If $a \sim \alpha$ and $a \sim \alpha'$, then $\alpha = \alpha'$.

PROOF. Taking E_α as in the proof of the previous lemma, if $a \sim \alpha$ and $a \sim \alpha'$, then $E_\alpha\, a \equiv_\beta \mathtt{tt}$ and so $\alpha' = \alpha$. ∎

Apart from the lifting, the following theorem is a typical logical relations lemma. It contains all we need to know about $[\![\cdot]\!]$.

Theorem 2.5 Let t be a closed term. Then $t^7 \sim [\![t]\!]$.

PROOF. We show that if $\mathsf{x}_1\colon A_1 \ldots \mathsf{x}_m\colon A_m \vdash s : B$, and v is a valuation with $v(\mathsf{x}_i) \in \mathcal{M}_{A_i}$ for each i, and $a_i \sim v(\mathsf{x}_i)$ for each i, then

$$s^7[\mathsf{x}_i/a_i]_{i=1}^m \sim [\![s]\!]_v. \tag{2.1}$$

Use induction on s. If $s = \mathsf{x}_i$, then (2.1) is just $a_i \sim v(\mathsf{x}_i)$. If $s = f\, r$ then

$$s^7[\mathsf{x}_i/a_i]_{i=1}^m = f^7[\mathsf{x}_i/a_i]_{i=1}^m\big(r^7[\mathsf{x}_i/a_i]_{i=1}^m\big) \sim [\![f]\!]_v\big([\![r]\!]_v\big) = [\![s]\!]_v.$$

If $s = \lambda \mathsf{x}_0\colon A_0.\, s'$, then for any a_0 and α_0 with $a_0 \sim \alpha_0 \in \mathcal{M}_{A_0}$, let v' be the valuation that extends v by mapping x_0 to α_0. Then

$$s^7[\mathsf{x}_i/a_i]_{i=1}^m(a_0) \equiv_\beta s'^7[\mathsf{x}_i/a_i]_{i=0}^m \sim [\![s']\!]_{v'} = [\![s]\!]_v(\alpha_0).$$

so that lemma 2.2 and the definition of $\sim$ gives (2.1). ∎

The theorem below uses $\sim$ to tie together definability problems and matching problems. A term t is said to be *encoding* if $t \sim \tau$ for some τ.

Theorem 2.6 Let $\alpha \in \mathcal{M}_A$. There is a matching equation P such that for all closed encoding terms $t : A^7$, we have t solves P iff $t \sim \alpha$. This P can be given as a computable function of α.

PROOF. As allowed by theorem 1.1, we construct finite sets rather than single equations.

For each element β of the model $\mathcal{M}$, take $p_\beta \sim \beta$, as given by lemma 2.3. A is in the form $[A_1 \ldots A_m]$. Let P be the set of all equations in the form

$$\mathsf{X}(p_{\alpha_1}) \ldots (p_{\alpha_m}) = p_{\alpha(\alpha_1)\ldots(\alpha_m)}$$

where each α_i ranges over $\mathcal{M}_{A_i}$. If t solves P and t is encoding, say $t \sim \tau$, then for all $\alpha_1 \ldots \alpha_m$, we have

$$p_{\alpha(\alpha_1)\ldots(\alpha_m)} \equiv_\beta t\, p_{\alpha_1} \ldots p_{\alpha_m} \sim \tau\, \alpha_1 \ldots \alpha_m$$

so that $\alpha\, \alpha_1 \ldots \alpha_m = \tau\, \alpha_1 \ldots \alpha_m$. By extensionality, $\alpha = \tau$ and $t \sim \alpha$.

Conversely, if $t \sim \alpha$, then for all $\alpha_1 \dots \alpha_m$,

$$t\, p_{\alpha_1} \dots p_{\alpha_m} \sim \alpha\, \alpha_1 \dots \alpha_m.$$

By the definition of $\sim$ at type 0, this is equivalent to

$$t\, p_{\alpha_1} \dots p_{\alpha_m} \equiv_\beta p_{\alpha\, \alpha_1 \dots \alpha_m}$$

so that t solves P. ∎

The restriction that t be encoding can be removed. This strengthening can be shown using the results of [8].

3 Myopic terms and the myopic order

We now develop some specialised machinery concerning 7-lifted types and terms.

The 'myopic pre-order' will be defined on a certain set of terms, and will give us subtle machinery for exposing structural properties of terms via matching equations. The pre-order has greatest lower bounds, and also a logical relations lemma for 7-lifted terms.

This will allow us to write matching equations that precisely control where certain variables occur in their solutions (specifically, type 7 variables must be bound by the innermost λ-abstraction). This control will be needed in the proof of theorem 4.4, the crucial argument of this paper.

The definition below is very similar to that of the familiar notion of a logical relation. The types involved are 7-lifted, but also the clause for function types is modified. The modification simplifies the basic properties of the relation (part 1 of lemma 3.1 below), while not disturbing the usual properties of logical relations.

In this section and the next, we must carry out some detailed calculations involving the myopic pre-order in the lead up to theorem 4.4. The reader may wish to skip the proofs of these on a first reading.

We define the *myopic pre-order* $\leqslant^{\mathsf{m}}$ on closed $\bot$-terms with 7-lifted type as follows, using recursion on the type, and then extend to other $\bot$-terms.

- At type 7, the closed normal $\bot$-terms are just $K_\bot = \lambda \mathsf{z}_1 \dots \mathsf{z}_7.\, \bot$ and the projections $\pi_1 \dots \pi_7$. We put $K_\bot \leqslant^{\mathsf{m}} K_\bot \leqslant^{\mathsf{m}} \pi_i \leqslant^{\mathsf{m}} \pi_i$ for each i, and compare other closed type 7 $\bot$-terms by comparing normal forms.
- For closed $\bot$-terms $f, g : A^7 \Rightarrow B^7$, we put $f \leqslant^{\mathsf{m}} g$ if, for all closed $\bot$-terms $a, b : A^7$ with $a \leqslant^{\mathsf{m}} b$, we have $f\, a \leqslant^{\mathsf{m}} f\, b$ and $g\, a \leqslant^{\mathsf{m}} g\, b$ and $f\, a \leqslant^{\mathsf{m}} g\, b$.
- Given $\bot$-terms s_1 and s_2 with $\mathsf{x}_1 \colon A_1^7 \dots \mathsf{x}_m \colon A_m^7 \vdash s_k : B^7$ for $k = 1, 2$, we define $s_1 \leqslant^{\mathsf{m}} s_2$ iff

$$\lambda \mathsf{x}_1 \colon A_1^7 \dots \lambda \mathsf{x}_m \colon A_m^7 \,.\, s_1 \leqslant^{\mathsf{m}} \lambda \mathsf{x}_1 \colon A_1^7 \dots \lambda \mathsf{x}_m \colon A_m^7 \,.\, s_2.$$

A $\bot$-term t is called *myopic* if $t \leqslant^{\mathsf{m}} t$, and *myopic equivalence* is defined by $r \equiv^{\mathsf{m}} s$ iff $r \leqslant^{\mathsf{m}} s \leqslant^{\mathsf{m}} r$.

Of course, merely calling $\leqslant^{\mathsf{m}}$ a pre-order does not make it so. Lemma 3.1 below verifies that $\leqslant^{\mathsf{m}}$ is a pre-order on the myopic $\bot$-terms, along with some other basic facts.

Our extension of $\leqslant^{\mathsf{m}}$ beyond closed $\perp$-terms is slightly misleading. The notation $s_1 \leqslant^{\mathsf{m}} s_2$ ignores the context and type appearing in the text of the definition. In reality, $\leqslant^{\mathsf{m}}$ is a relation on the triple (context, term, type), not just the term, and we mention the context explicitly when necessary.

Lemma 3.1 1. If $a \leqslant^{\mathsf{m}} b$ then a and b are myopic. 2. The myopic pre-order is transitive. 3. There are only finitely many $\equiv^{\mathsf{m}}$-equivalence classes of closed $\perp$-terms at each type. 4. If $a \equiv_{\beta\eta} a' \leqslant^{\mathsf{m}} b \equiv_{\beta\eta} b'$ then $a \leqslant^{\mathsf{m}} b'$.

Proof. We consider only closed $\perp$-terms, as this implies the general case.

1. At type 7, all closed $\perp$-terms are myopic. At higher types, 1. is immediate from the definition of $\leqslant^{\mathsf{m}}$.

2. Use induction on types. At type 7 this is immediate from the definition. Suppose that $f, g, h : A^7 \Rightarrow B^7$ are closed $\perp$-terms and $f \leqslant^{\mathsf{m}} g \leqslant^{\mathsf{m}} h$. Take any closed $\perp$-terms $a, b : A^7$ with $a \leqslant^{\mathsf{m}} b$. We have $f\,a \leqslant^{\mathsf{m}} f\,b$ (because $f \leqslant^{\mathsf{m}} g$) and $h\,a \leqslant^{\mathsf{m}} h\,b$ (because $g \leqslant^{\mathsf{m}} h$). By 1., $a \leqslant^{\mathsf{m}} a$, so that $f\,a \leqslant^{\mathsf{m}} g\,a \leqslant^{\mathsf{m}} h\,b$ and thus $f\,a \leqslant^{\mathsf{m}} h\,b$ by the induction hypothesis. This verifies that $f \leqslant^{\mathsf{m}} h$.

3. First note that 1. and 2. imply that for closed myopic $\perp$-terms $f, g : A^7 \Rightarrow B^7$, we have $f \leqslant^{\mathsf{m}} g$ iff $f\,a \leqslant^{\mathsf{m}} g\,a$ for all closed myopic $\perp$-terms $a : A^7$. Therefore, the equivalence class of any closed myopic $\perp$-term $f : A^7 \Rightarrow B^7$ is determined by the function it induces from equivalence classes at type A^7 to equivalence classes at type B^7.

Now use induction on types. At type 7 there are eight equivalence classes. If there are m and n equivalence classes at type A^7 and B^7 respectively, then by the paragraph above, there are at most n^m equivalence classes at type $A^7 \Rightarrow B^7$.

4. For type 7, the implication follows immediately from the definition of $\leqslant^{\mathsf{m}}$. An induction extends this to higher types. ∎

Note that what lemma 3.1 does not say is nearly as important as what it does. $\leqslant^{\mathsf{m}}$ and $\equiv^{\mathsf{m}}$ are not reflexive on the set of appropriately typed $\perp$-terms. They give a pre-order and equivalence only on the myopic $\perp$-terms. This restriction is necessary, as the following discussion indicates.[2] Let $I : 7 \Rightarrow 7 \Rightarrow 7$ be

$$\lambda \mathsf{x}, \mathsf{y} \colon 7\,.\, \lambda \mathsf{z}_1 \ldots \mathsf{z}_7 \colon 0\,.\, \mathsf{x}\,(\mathsf{y}\perp \ldots \perp) \ldots (\mathsf{y}\perp \ldots \perp)$$

Then $I\,a\,b \equiv_\beta K_\perp$ for any closed $\perp$-terms $a, b : 7$. Therefore, I is myopic, and setting $J = \lambda \mathsf{x}, \mathsf{y} \colon 7.\, I\,\mathsf{y}\,\mathsf{x}$, we have $I \equiv^{\mathsf{m}} J \equiv^{\mathsf{m}} \lambda \mathsf{x}, \mathsf{y} \colon 7.\, K_\perp$.

However, I and J are distinguished by the term $D : (7 \Rightarrow 7 \Rightarrow 7) \Rightarrow 7$ defined to be

$$\lambda \mathsf{f} \colon 7 \Rightarrow 7 \Rightarrow 7\,.\, \lambda \mathsf{z}_1 \ldots \mathsf{z}_7 \colon 0\,.\, \mathsf{f}(\lambda \mathsf{v}_1 \ldots \mathsf{v}_7 .\, \mathsf{z}_1)(\lambda \mathsf{v}_1 \ldots \mathsf{v}_7 .\, \mathsf{z}_2)\, \mathsf{z}_1 \ldots \mathsf{z}_7$$

as we have $D\,I \equiv_\beta \pi_1$ and $D\,J \equiv_\beta \pi_2$. This shows that D is not myopic. For any closed $\perp$-terms $s, t : A$, a similar construction gives $D_{st} : (7 \Rightarrow 7 \Rightarrow 7) \Rightarrow A$ such that $D_{st} I \equiv_{\beta\eta} s$ and $D_{st} J \equiv_{\beta\eta} t$.

Furthermore, any reflexive extension $\equiv$ of $\equiv^{\mathsf{m}}$ that is compatible with both $\beta\eta$-conversion and application, is trivial: $s \equiv_{\beta\eta} D_{st}\, I \equiv D_{st}\, J \equiv_{\beta\eta} t$, and hence $s \equiv t$, for

[2] The example is based on one suggested by Vincent Padovani.

all s and t of the same 7-lifted type. This shows that the failure of $\equiv^{\mathsf{m}}$ to be reflexive is necessary.

Despite the failure of reflexivity in general, the next two lemmas show that it holds for a useful set of terms.

Lemma 3.2 1. Every closed $\perp$-term with type in the form $7 \Rightarrow \ldots \Rightarrow 7 \Rightarrow 7$ is myopic. 2. If s is a $\perp$-term with $\mathsf{x}_1 : 7 \ldots \mathsf{x}_m : 7 \vdash s : 7$ then s is myopic.

Proof. For $\perp$-terms a and b, define $a \leqslant^{\perp} b$ iff a can be obtained by replacing zero or more type 0 sub-terms within b by $\perp$.

We show that if $a \leqslant^{\perp} b$ and $b \rightarrow_{\beta} b'$, then there is $a' \leqslant^{\perp} b'$ with $a \rightarrow_{\beta} a'$. First consider the reduction of a redex: $b = (\lambda \mathsf{x}.\, c)(d)$ and $b' = c[\mathsf{x}/d]$. If $a = \perp$, then we can take $a' = \perp$ also. Otherwise, a is in the form $(\lambda \mathsf{x}.\, c_1)(d_1)$ where $c_1 \leqslant^{\perp} c$ and $d_1 \leqslant^{\perp} d$, and we may take $a' = c_1[\mathsf{x}/d_1]$. $\rightarrow_{\beta}$ is the least partial-order containing the reduction of redexes and closed under term formation. Noting that $\leqslant^{\perp}$ is also closed under term formation, an induction gives the construction of a' for general $b \rightarrow_{\beta} b'$ and $a \leqslant^{\perp} b$.

When the b' of the previous paragraph is the normal form of b, then also a' must be the normal form of a. It follows that for closed $\perp$-terms $a, b : 7$, (a) if $a \leqslant^{\perp} b$, then $a \leqslant^{\mathsf{m}} b$, and (b) if $a \leqslant^{\mathsf{m}} b$ and a and b are normal, then $a \leqslant^{\perp} b$.

For any natural number n, let n^7 be the type $7 \Rightarrow \ldots \Rightarrow 7 \Rightarrow 7$ with 7 repeated $n+1$ times. We show by induction on n that, if $f, g : n^7$ are closed $\perp$-terms with $f \leqslant^{\perp} g$, then $f \leqslant^{\mathsf{m}} g$.

For $n = 0$, the induction predicate is just (a) above.

Suppose that $f, g : (n+1)^7$ are closed $\perp$-terms with $f \leqslant^{\perp} g$. Take closed $\perp$-terms $a, b : 7$ with $a \leqslant^{\mathsf{m}} b$. Let a' and b' be the normal forms of a and b, so that $a' \leqslant^{\perp} b'$ by (b) above. Then $f\, a' \leqslant^{\perp} f\, b'$ and $g\, a' \leqslant^{\perp} g\, b'$ and $f\, a' \leqslant^{\perp} g\, b'$, so that by the induction hypothesis, $f\, a \leqslant^{\mathsf{m}} f\, b$ and $g\, a \leqslant^{\mathsf{m}} g\, b$ and $f\, a \leqslant^{\mathsf{m}} g\, b$. Thus $f \leqslant^{\mathsf{m}} g$ which completes the induction.

If $f : n^7$ is a closed $\perp$-term, then obviously $f \leqslant^{\perp} f$ and so $f \leqslant^{\mathsf{m}} f$. This shows 1., and applying this to $\lambda \mathsf{x}_1 \ldots \mathsf{x}_m : 7.\, s$ gives 2. ∎

Below is a logical relations lemma for $\leqslant^{\mathsf{m}}$.

Lemma 3.3 Every 7-lifted term is myopic. More generally, if

$$\mathsf{x}_1 : A_1 \ldots \mathsf{x}_m : A_m \vdash s : B$$

and $a_i, b_i : A_i^7$ are $\perp$-terms with $a_i \leqslant^{\mathsf{m}} b_i$ (in some fixed 7-lifted context Δ) for each $i = 1 \ldots m$, then

$$s^7[\mathsf{x}_i/a_i]_{i=1}^m \leqslant^{\mathsf{m}} s^7[\mathsf{x}_i/b_i]_{i=1}^m \tag{3.1}$$

(in the context Δ).

Proof. We first show that the generalised statement holds when the context Δ is empty (so that the a_i and b_i are closed), using induction on the term s.

If $s = \mathsf{x}_i$, then (3.1) is just $a_i \leqslant^{\mathsf{m}} b_i$.

If $s = f\, r$ then the induction hypothesis gives $f^7[\mathsf{x}_i/a_i]_{i=1}^m \leqslant^{\mathsf{m}} f^7[\mathsf{x}_i/b_i]_{i=1}^m$ and $r^7[\mathsf{x}_i/a_i]_{i=1}^m \leqslant^{\mathsf{m}} r^7[\mathsf{x}_i/b_i]_{i=1}^m$ and then the definition of $\leqslant^{\mathsf{m}}$ gives (3.1).

If $s = \lambda \mathsf{x}_0 \colon A_0 .\, s'$, then take any closed $\perp$-terms $a_0, b_0 : A_0^7$ with $a_0 \leqslant^{\mathsf{m}} b_0$. Using the induction hypothesis,

$$s^7[\mathsf{x}_i/a_i]_{i=1}^m(a_0) \equiv_\beta s'^7[\mathsf{x}_i/a_i]_{i=0}^m \leqslant^{\mathsf{m}} s'^7[\mathsf{x}_i/b_i]_{i=0}^m \equiv_\beta s^7[\mathsf{x}_i/b_i]_{i=1}^m(b_0).$$

Since $a_i \leqslant^{\mathsf{m}} a_i$ and $b_i \leqslant^{\mathsf{m}} b_i$ for each i, similar reasoning gives

$$s^7[\mathsf{x}_i/a_i]_{i=1}^m(a_0) \leqslant^{\mathsf{m}} s^7[\mathsf{x}_i/a_i]_{i=1}^m(b_0)$$

and

$$s^7[\mathsf{x}_i/b_i]_{i=1}^m(a_0) \leqslant^{\mathsf{m}} s^7[\mathsf{x}_i/b_i]_{i=1}^m(b_0).$$

Applying the definition of $\leqslant^{\mathsf{m}}$ gives (3.1), completing the induction.

This implies that all 7-lifted terms are myopic. We now derive from this the general statement, for arbitrary Δ. The context Δ is in the form $\mathsf{z}_1 \colon C_1^7 \ldots \mathsf{z}_p \colon C_p^7$. Let $A_i' = (C_1 \Rightarrow \ldots \Rightarrow C_p \Rightarrow A_i)$ for each i, and let

$$F = \lambda \mathsf{X}_1 \colon A_1' \ldots \lambda \mathsf{X}_m \colon A_m' \,.\, \lambda \mathsf{z}_1 \colon C_1 \ldots \lambda \mathsf{z}_p \colon C_p \,.\, s[\mathsf{x}_i/\mathsf{X}_i\, \mathsf{z}_1 \ldots \mathsf{z}_p]_{i=1}^m$$

and let $a_i' = \lambda \mathsf{z}_1 \colon C_1^7 \ldots \lambda \mathsf{z}_p \colon C_p^7 .\, a_i$ and $b_i' = \lambda \mathsf{z}_1 \colon C_1^7 \ldots \lambda \mathsf{z}_p \colon C_p^7 .\, b_i$ for each i. Then F^7 is closed, and myopic as it is 7-lifted, and $a_i' \leqslant^{\mathsf{m}} b_i'$ for each i, as the a_i' and b_i' are the closures of the a_i and b_i. Hence

$$F^7\, a_1' \ldots a_m' \leqslant^{\mathsf{m}} F^7\, b_1' \ldots b_m'.$$

β-reducing both sides of this gives

$$\lambda \mathsf{z}_1 \colon C_1^7 \ldots \lambda \mathsf{z}_p \colon C_p^7 \,.\, s^7[\mathsf{x}_i/a_i]_{i=1}^m \leqslant^{\mathsf{m}} \lambda \mathsf{z}_1 \colon C_1^7 \ldots \lambda \mathsf{z}_p \colon C_p^7 \,.\, s^7[\mathsf{x}_i/b_i]_{i=1}^m$$

which implies (3.1). ∎

As noted after the definition of $\leqslant^{\mathsf{m}}$, the notation is ambiguous outside of closed terms. The lemma below shows that $\leqslant^{\mathsf{m}}$ is sufficiently well-behaved to make this ambiguity harmless.

Lemma 3.4 Suppose that both $\Delta^7 \vdash s_k : B^7$ and $\Delta'^7 \vdash s_k : B^7$ for $k = 1, 2$, and that the contexts Δ and Δ' give the same types for any variables that appear in both. If $s_1 \leqslant^{\mathsf{m}} s_2$ (in context Δ^7), then $s_1 \leqslant^{\mathsf{m}} s_2$ (in context Δ'^7).

PROOF. The contexts Δ and Δ' can be respectively written as $\mathsf{x}_1 \colon A_1 \ldots \mathsf{x}_m \colon A_m$ and $\mathsf{x}_1' \colon A_1' \ldots \mathsf{x}_{m'}' \colon A_{m'}'$. By the definition of $\leqslant^{\mathsf{m}}$, it suffices to give a closed myopic $\perp$-term

$$F : \left(A_1^7 \Rightarrow \ldots \Rightarrow A_m^7 \Rightarrow B^7\right) \Rightarrow \left(A_1'^7 \Rightarrow \ldots \Rightarrow A_{m'}'^7 \Rightarrow B^7\right)$$

such that

$$F\left(\lambda \mathsf{x}_1 \colon A_1^7 \ldots \lambda \mathsf{x}_m \colon A_m^7 \,.\, s_k\right) \equiv_\beta \lambda \mathsf{x}_1' \colon A_1'^7 \ldots \lambda \mathsf{x}_{m'}' \colon A_{m'}'^7 \,.\, s_k \tag{3.2}$$

for $k = 1, 2$.

Firstly, for any type C, there is some myopic closed term $K_C : C^7$. By lemma 3.3, we may take $K_C = \lambda \mathsf{z}_1 \colon C_1^7 \ldots \lambda \mathsf{z}_p \colon C_p^7 .\, \pi_1$ where $C = [C_1 \ldots C_p]$. Now define F to be

$$\lambda \mathsf{g} \colon A_1^7 \Rightarrow \ldots \Rightarrow A_m^7 \Rightarrow B^7 \,.\, \lambda \mathsf{x}_1' \colon A_1'^7 \ldots \lambda \mathsf{x}_{m'}' \colon A_{m'}'^7 \,.\, \mathsf{g}\, a_1 \ldots a_m$$

where $a_i = \mathsf{x}_i$ if x_i occurs in Δ' and $a_i = K_{A_i}$ otherwise. The closed term F is myopic by lemma 3.3. Noting that if x_i does not occur in Δ' then it cannot occur in either s_1 or s_2, equation (3.2) is derived by β-reducing its LHS. ∎

The next theorem states a fact that we will use to define $\perp$-terms by relations involving $\leqslant^{\mathsf{m}}$.

Theorem 3.5 There is a $\perp$-term $\wedge : 7 \Rightarrow 7 \Rightarrow 7$ that gives greatest lower bounds for $\leqslant^{\mathsf{m}}$ (considering $\wedge$ as a binary operator on the type 7 closed $\perp$-terms). This extends point-wise to higher types: if $f, g : [A_1 \ldots A_m]^7$ are myopic, then

$$\lambda \mathsf{x}_1 \colon A_1^7 \ldots \lambda \mathsf{x}_m \colon A_m^7 \,.\, \wedge (f\, \mathsf{x}_1 \ldots \mathsf{x}_7)(g\, \mathsf{x}_1 \ldots \mathsf{x}_7)$$

is a $\leqslant^{\mathsf{m}}$ g.l.b. of f and g.

Proof. Let $\wedge$ be the $\perp$-term

$$\lambda \mathsf{x} \colon 7 \,.\, \lambda \mathsf{y} \colon 7 \,.\, \lambda \mathsf{z}_1 \ldots \mathsf{z}_7 \colon 0 \,.\, \mathsf{x}\,(\mathsf{y}\, u_{11} \ldots u_{17}) \ldots (\mathsf{y}\, u_{71} \ldots u_{77})$$

where $u_{ii} = \mathsf{z}_i$ and $u_{ij} = \perp$ for $i \neq j$. Calculation shows that it is the required term: $\wedge\, \pi_i\, \pi_i \equiv_\beta \pi_i$, while for $i \neq j$, all of $\wedge\, \pi_i\, \pi_j$, $\wedge\, \pi_i\, K_\perp$ and $\wedge\, K_\perp\, \pi_j$ are $\equiv_\beta K_\perp$. The extension to higher types follows by an easy computation. ∎

4 Characterising lifts

Theorems 2.5 and 2.6 above would suffice to show our undecidability result if, for each type A, we could find another matching equation Q such that a term $a : A^7$ solves Q iff a is 7-lifted. Unfortunately, this appears not to be possible. However, we will be able to do this—or close enough—for all A that are themselves 1-lifted (where $1 = 0 \Rightarrow 0$). This will suffice to establish our result.

As an example of what is to come, consider the type $(7 \Rightarrow 7) \Rightarrow 7 \Rightarrow 7$ matching equation

$$\mathsf{X}\,(\lambda \mathsf{z} \colon 7 \,.\, \mathsf{z}) = \lambda \mathsf{z} \colon 7 \,.\, \mathsf{z}. \tag{4.1}$$

Any 7-lifted closed term with type $(7 \Rightarrow 7) \Rightarrow 7 \Rightarrow 7$ must solve this equation, as $\lambda \mathsf{z} \colon 7 .\, \mathsf{z}$ is the only normal 7-lifted closed term of type $7 \Rightarrow 7$. We show that a converse also holds: all normal solutions are 7-lifted. Take any normal closed $F : (7 \Rightarrow 7) \Rightarrow 7 \Rightarrow 7$ that is not 7-lifted. Then F must be in the form

$$\lambda \mathsf{g} \colon 7 \Rightarrow 7 \,.\, \lambda \mathsf{z} \colon 7 \,.\, \mathsf{g}\big(\ldots \big(\mathsf{g}\big(\lambda \mathsf{x} \colon 0 \,.\, a\big)\big) \ldots\big),$$

possibly with zero occurrences of g. We have

$$F(\lambda \mathsf{z} \colon 7 \,.\, \mathsf{z}) \equiv_\beta \lambda \mathsf{z} \colon 7 \,.\, \lambda \mathsf{x} \colon 0 \,.\, a[\mathsf{g}/\lambda \mathsf{z} \colon 7 \,.\, \mathsf{z}]$$

so that F cannot solve (4.1). Theorem 4.4 will give a generalisation of this converse.

The 'composer' terms defined below allow us to generalise (4.1) to all 1^7-lifted types. Unfortunately, not all normal solutions to these generalised equations are 7-lifted. We will need to add further equations to obtain that property. These additional equations involve the 'checker' terms that are defined subsequently.

For each type A, we define two *composer* terms $\Gamma_A : A^1$ and $\Gamma'_A : (A^1) \Rightarrow 0 \Rightarrow 0$. Given $A = [A_1 \ldots A_m]$, these are given recursively by:

$$\Gamma_A = \lambda \mathsf{x}_1 \colon A_1^1 \ldots \lambda \mathsf{x}_m \colon A_m^1 \,.\, \lambda \mathsf{z} \colon 0 \,.\, \Gamma'_{A_1} \mathsf{x}_1 \big(\ldots \big(\Gamma'_{A_m} \mathsf{x}_m\, \mathsf{z}\big) \ldots\big)$$

and

$$\Gamma'_A = \lambda \mathsf{f}\colon A^1 \,.\, \mathsf{f}\,\Gamma_{A_1}\ldots\Gamma_{A_m}.$$

In what follows, it is actually the lifts Γ_A^7 and $\Gamma_A'^7$ that are used.

The definition of the 'checker' $\perp$-terms is similar, but uses the @ operator that we now define.

Let $@ : (7 \Rightarrow 7) \Rightarrow 7 \Rightarrow 7$ be a $\leqslant^{\mathsf{m}}$-minimum closed $\perp$-term such that $@(\lambda \mathsf{z}\colon 7.\, \mathsf{z}) \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$. This equation is satisfied by $\lambda \mathsf{x}\colon 7 \Rightarrow 7.\, \mathsf{x}$ and preserved by the g.l.b. given by lemma 3.5, so a minimum exists using lemma 3.1. The @ operator can be characterised equationally:

Lemma 4.1 Let $f : 7 \Rightarrow 7$ be a closed $\perp$-term. Then $@\,f \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$ if $f \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$, and otherwise $@\,f \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, K_\perp$.

Proof. The case for $f \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$ is immediate from the definition.

For each $i = 1 \ldots 7$, define $\delta_i : (7 \Rightarrow 7) \Rightarrow 7 \Rightarrow 7$ by

$$\delta_i = \lambda \mathsf{f}\colon 7 \Rightarrow 7 \,.\, \lambda \mathsf{z}\colon 7 \,.\, \lambda \mathsf{x}_1 \ldots \mathsf{x}_7\colon 0 \,.\, \mathsf{f}\,\pi_i\, a_{i1} \ldots a_{i7}$$

where $a_{ii} = \mathsf{z}\,\mathsf{x}_1 \ldots \mathsf{x}_7$ and $a_{ij} = \perp$ for $j \neq i$. We can rewrite this as

$$\delta_i \equiv_\beta \lambda \mathsf{f}\colon 7 \Rightarrow 7 \,.\, \epsilon_i(\mathsf{f}\,\pi_i)$$

where $\epsilon_i = \lambda \mathsf{u}, \mathsf{z}\colon 7.\, \lambda \mathsf{x}_1 \ldots \mathsf{x}_7\colon 0.\, \mathsf{u}\, a_{i1} \ldots a_{i7} : 7 \Rightarrow 7 \Rightarrow 7$, so that δ_i is myopic by lemmas 3.2 and 3.3. We have

$$\begin{aligned} \delta_i(\lambda \mathsf{z}\colon 7 \,.\, \mathsf{z}) &\equiv_\beta \lambda \mathsf{z}\colon 7 \,.\, \lambda \mathsf{x}_1 \ldots \mathsf{x}_7\colon 0 \,.\, \pi_i\, a_{i1} \ldots a_{i7} \\ &\equiv_\beta \lambda \mathsf{z}\colon 7 \,.\, \lambda \mathsf{x}_1 \ldots \mathsf{x}_7\colon 0 \,.\, a_{ii} \quad \equiv_{\beta\eta} \quad \lambda \mathsf{z}\colon 7 \,.\, \mathsf{z}, \end{aligned}$$

so that $@ \leqslant^{\mathsf{m}} \delta_i$.

Suppose that $f : 7 \Rightarrow 7$ is a closed $\perp$-term and $f \not\equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$. We have one of $f(K_\perp) \equiv^{\mathsf{m}} \pi_j$, $f(\pi_i) \equiv^{\mathsf{m}} K_\perp$ or $f(\pi_i) \equiv^{\mathsf{m}} \pi_j$ for suitable i and j with $i \neq j$. In any case, there are i and j with $i \neq j$ and $f(\pi_i) \leqslant^{\mathsf{m}} \pi_j$. Then

$$\begin{aligned} @\,f \leqslant^{\mathsf{m}} \delta_i\, f \quad &\leqslant^{\mathsf{m}} \lambda \mathsf{z}\colon 7 \,.\, \lambda \mathsf{x}_1 \ldots \mathsf{x}_7\colon 0 \,.\, \pi_j\, a_{i1} \ldots a_{i7} \\ &\equiv_\beta \lambda \mathsf{z}\colon 7 \,.\, \lambda \mathsf{x}_1 \ldots \mathsf{x}_7\colon 0 \,.\, a_{ij} \; = \; \lambda \mathsf{z}\colon 7 \,.\, K_\perp. \end{aligned}$$

as required. ∎

The *checker* $\perp$-terms $\Sigma_A : (A^1)^7$ and $\Sigma'_A : (A^1)^7 \Rightarrow 7 \Rightarrow 7$ are now defined similarly to the composers, but 7-lifting and inserting @:

$$\Sigma_A = \lambda \mathsf{x}_1\colon A_1^{17} \ldots \lambda \mathsf{x}_m\colon A_m^{17} \,.\, \lambda \mathsf{z}\colon 7 \,.\, \Sigma'_{A_1} \mathsf{x}_1 \big(\ldots \big(\Sigma'_{A_m} \mathsf{x}_m\, \mathsf{z}\big) \ldots\big)$$

and

$$\Sigma'_A = \lambda \mathsf{f}\colon A^{17} \,.\, @(\mathsf{f}\,\Sigma_{A_1} \ldots \Sigma_{A_m}).$$

The following lemma gives us a convenient way of simplifying expressions involving the checker terms.

Lemma 4.2 For any type A, we have $\Sigma_A \leqslant^{\mathsf{m}} \Gamma_A^7$ and $\Sigma'_A \leqslant^{\mathsf{m}} \Gamma_A'^7$.

PROOF. We use induction on the type $A = [A_1 \dots A_m]$. We have

$$\begin{array}{rcl} \Sigma_A & = & \lambda \mathsf{x}_1 \colon A_1^{17} \dots \lambda \mathsf{x}_m \colon A_m^{17} \,.\, \lambda \mathsf{z} \colon 7 \,.\, \Sigma'_{A_1} \mathsf{x}_1 \big(\dots \big(\Sigma'_{A_m} \mathsf{x}_m\, \mathsf{z} \big) \dots \big) \\ & \leqslant^{\mathsf{m}} & \lambda \mathsf{x}_1 \colon A_1^{17} \dots \lambda \mathsf{x}_m \colon A_m^{17} \,.\, \lambda \mathsf{z} \colon 7 \,.\, \Gamma'^{7}_{A_1} \mathsf{x}_1 \big(\dots \big(\Gamma'^{7}_{A_m} \mathsf{x}_m\, \mathsf{z} \big) \dots \big) \quad = \quad \Gamma^7_A \end{array}$$

using lemma 3.3 and the induction hypothesis for each of the A_i. Also,

$$\begin{array}{rcl} \Sigma'_A & = & \lambda \mathsf{f} \colon A^{17} \,.\, @ \big(\mathsf{f}\, \Sigma_{A_1} \dots \Sigma_{A_m} \big) \\ & \leqslant^{\mathsf{m}} & \lambda \mathsf{f} \colon A^{17} \,.\, \big(\lambda \mathsf{x} \colon 7 \Rightarrow 7 \,.\, \mathsf{x} \big) \big(\mathsf{f}\, \Gamma^7_{A_1} \dots \Gamma^7_{A_m} \big) \\ & \equiv_\beta & \lambda \mathsf{f} \colon A^{17} \,.\, \mathsf{f}\, \Gamma^7_{A_1} \dots \Gamma^7_{A_m} \quad = \quad \Gamma'^{7}_A \end{array}$$

using lemma 3.3, the induction hypothesis for each of the A_i, and the fact that $@ \leqslant^{\mathsf{m}} \lambda \mathsf{x} \colon 7 \Rightarrow 7 .\, \mathsf{x}$. ∎

The next lemma gives a calculation used in the proof of theorem 4.4. Recall that terms (as opposed to $\perp$-terms) do not contain the constant $\perp$.

Lemma 4.3 Suppose that t_j are terms with

$$\mathsf{x}_1 \colon A_1^1 \dots \mathsf{x}_m \colon A_m^1 \vdash t_j : C_j^1$$

for $1 \leqslant j \leqslant p$. Then

$$\Gamma'^{7}_{C_1}\, t_1^7 \big(\dots \big(\Gamma'^{7}_{C_p}\, t_p^7\, (e) \big) \dots \big) \big[\mathsf{x}_i / \Gamma^7_{A_i} \big]_{i=1}^m \;\equiv_\beta\; e \big[\mathsf{x}_i / \Gamma^7_{A_i} \big]_{i=1}^m$$

and

$$\Sigma'_{C_1}\, t_1^7 \big(\dots \big(\Sigma'_{C_p}\, t_p^7\, (e) \big) \dots \big) \big[\mathsf{x}_i / \Sigma_{A_i} \big]_{i=1}^m \;\leqslant^{\mathsf{m}}\; e \big[\mathsf{x}_i / \Sigma_{A_i} \big]_{i=1}^m$$

for any $\perp$-term $e : 7$ such that $e[\mathsf{x}_i / \Sigma_{A_i}]_{i=1}^m$ is myopic.

PROOF. As $\lambda \mathsf{z} \colon 0.\, \Gamma'_{C_1}\, t_1 \,(\dots (\Gamma'_{C_p}\, t_p\, (\mathsf{z})) \dots)[\mathsf{x}_i / \Gamma_{A_i}]_{i=1}^m$ is a closed term with type $0 \Rightarrow 0$, its normal form can only be $\lambda \mathsf{z} \colon 0.\, \mathsf{z}$. Lifting by 7, we get

$$\lambda \mathsf{z} \colon 7 \,.\, \Gamma'^{7}_{C_1}\, t_1^7 \big(\dots \big(\Gamma'^{7}_{C_p}\, t_p^7\, (\mathsf{z}) \big) \dots \big) \big[\mathsf{x}_i / \Gamma^7_{A_i} \big]_{i=1}^m \;\equiv_\beta\; \lambda \mathsf{z} \colon 7 \,.\, \mathsf{z},$$

and lemmas 3.3 and 4.2 give also

$$\lambda \mathsf{z} \colon 7 \,.\, \Sigma'_{C_1}\, t_1^7 \big(\dots \big(\Sigma'_{C_p}\, t_p^7\, (\mathsf{z}) \big) \dots \big) \big[\mathsf{x}_i / \Sigma_{A_i} \big]_{i=1}^m \;\leqslant^{\mathsf{m}}\; \lambda \mathsf{z} \colon 7 \,.\, \mathsf{z}.$$

Applying both sides of the equations above to $e[\mathsf{x}_i / \Gamma^7_{A_i}]_{i=1}^m$ and $e[\mathsf{x}_i / \Sigma_{A_i}]_{i=1}^m$ respectively, gives the result. ∎

The next theorem is the technical heart of this paper. The theorem shows that some subset of the 7-lifted terms is characterisable via matching equations. Later, we will show that this subset is large enough to derive our undecidability theorem.

Theorem 4.4 Let the term $s : (B^1)^7$ be closed and normal. Suppose that

$$\Gamma'^{7}_B(s) \equiv_\beta \lambda \mathsf{z} \colon 7 \,.\, \mathsf{z} \tag{4.2}$$

and

$$\Sigma'_B(s) \equiv^{\mathsf{m}} \lambda \mathsf{z} \colon 7 \,.\, \mathsf{z}. \tag{4.3}$$

Then s is the 7-lift of a type B^1 term.

PROOF. We prove a slightly stronger statement:

Claim: Suppose s is a normal term with

$$\mathsf{x}_1\colon A_1^{17}\ \ldots\ \mathsf{x}_m\colon A_m^{17}\quad \mathsf{z}_1\colon 7\ \ldots\ \mathsf{z}_h\colon 7\quad\vdash\quad s : B^{17}.$$

Then either

I. the term s is a 7-lift, and none of $\mathsf{z}_1 \ldots \mathsf{z}_h$ occur free in s,

II. or there is α such that

$$\Gamma_B'^7\, s\left[\mathsf{x}_i/\Gamma_{A_i}^7\right]_{i=1}^m \equiv_\beta \lambda\mathsf{z}\colon 7\,.\,\lambda\mathsf{z}'\colon 0\,.\,\alpha,$$

III. or

$$\Sigma_B'\, s\left[\mathsf{x}_i/\Sigma_{A_i}\right]_{i=1}^m \equiv^{\mathsf{m}} \lambda\mathsf{z}\colon 7\,.\,K_\perp.$$

We prove the claim by induction on s. The induction step splits into cases depending on the form of s. The type B is in the form $[B_1 \ldots B_n]$.

Case 1. Suppose that s starts with at most n λ-abstractions, so that it is in the form

$$\lambda\mathsf{y}_1\colon B_1^{17}\ldots\lambda\mathsf{y}_{n'}\colon B_{n'}^{17}\,.\,\mathsf{u}\,t_1\ldots t_p$$

where $0 \leqslant n' \leqslant n$ and $p \geqslant 0$. The type of $\mathsf{u}\,t_1 \ldots t_p$ must be $[B_{n'+1}\ldots B_n]^{17}$, so that the variable u cannot have type 7, and must be one of either the x_i or the y_j. Each t_k has type in the form C_k^{17}, so that the type of u is $C^{17} = [C_1 \ldots C_p, B_{n'+1}\ldots B_n]^{17}$. Expanding $\Gamma_B'^7$ and s gives

$$\begin{aligned}\Gamma_B'^7\, s\big[\mathsf{x}_i/\Gamma_{A_i}^7\big]_{i=1}^m \quad &\equiv_\beta \quad s\big[\mathsf{x}_i/\Gamma_{A_i}^7\big]_{i=1}^m\,\Gamma_{B_1}^7\ldots\Gamma_{B_n}^7\\ &\equiv_\beta \quad \Gamma_C^7\, t_1\ldots t_p\,\Gamma_{B_{n'+1}}^7\ldots\Gamma_{B_n}^7\,\left[\Phi\right]\end{aligned}$$

where

$$\left[\Phi\right] = \left[\mathsf{x}_i/\Gamma_{A_i}^7\right]_{i=1}^m\left[\mathsf{y}_j/\Gamma_{B_j}^7\right]_{j=1}^{n'}.$$

Then expanding Γ_C^7 gives

$$\Gamma_B'^7\, s\left[\mathsf{x}_i/\Gamma_{A_i}^7\right]_{i=1}^m \ \equiv_\beta\ \lambda\mathsf{z}\colon 7\,.\,\Gamma_{C_1}'^7\, t_1\left(\ldots\left(\Gamma_{C_p}'^7\, t_p\left(\phi\right)\right)\ldots\right)\left[\Phi\right] \tag{4.4}$$

where $\phi = \Gamma_{B_{n'+1}}'^7\Gamma_{B_{n'+1}}^7(\ldots(\Gamma_{B_n}'^7\Gamma_{B_n}^7(\mathsf{z}))\ldots)$. Expanding the checkers similarly gives

$$\Sigma_B'\, s\left[\mathsf{x}_i/\Sigma_{A_i}\right]_{i=1}^m \ \equiv_\beta\ @\big(\lambda\mathsf{z}\colon 7\,.\,\Sigma_{C_1}'\, t_1\left(\ldots\left(\Sigma_{C_p}'\, t_p\left(\psi\right)\right)\ldots\right)\big)\left[\Psi\right] \tag{4.5}$$

for some ψ, where

$$\left[\Psi\right] = \left[\mathsf{x}_i/\Sigma_{A_i}\right]_{i=1}^m\left[\mathsf{y}_j/\Sigma_{B_j}\right]_{j=1}^{n'}.$$

Now apply the induction hypothesis to each of the t_k, including the y_j and z in the context. If clause I holds for all the t_k, then it holds for s also. On the other hand, suppose that either clause II or III applies to some t_k. Taking k to be as small as possible, clause I holds for $t_1 \ldots t_{k-1}$, so that lemma 4.3 can be applied (with $p = k-1$) to simplify (4.4) and (4.5) to

$$\Gamma_B'^7\, s\left[\mathsf{x}_i/\Gamma_{A_i}^7\right]_{i=1}^m \ \equiv_\beta\ \lambda\mathsf{z}\colon 7\,.\,\Gamma_{C_k}'^7\, t_k\,\phi'\left[\Phi\right]$$

and

$$\Sigma'_B\, s\left[\mathsf{x}_i/\Sigma_{A_i}\right]_{i=1}^{m} \;\leqslant^{\mathsf{m}}\; @\big(\lambda\mathsf{z}\colon 7\,.\,\Sigma'_{C_k}\, t_k\, \psi'\left[\Psi\right]\big) \;\leqslant^{\mathsf{m}}\; \lambda\mathsf{z}\colon 7\,.\,\Sigma'_{C_k}\, t_k\, \psi'\left[\Psi\right].$$

for some ϕ' and ψ'. Then clause II holds for s if it holds for t_k, and clause III holds for s if it holds for t_k.

Case 2. Suppose that s starts with at least $n{+}1$ λ-abstractions, so that s is in the form

$$\lambda\mathsf{y}_1\colon B_1^{17}\ldots\lambda\mathsf{y}_n\colon B_n^{17}\,.\,\lambda\mathsf{z}\colon 7\,.\,d.$$

If the sub-term $d:7$ is an application, then it must be in the form $\mathsf{u}\,t_1\ldots t_p\,d'$. Then the variable u must be one of either the x_i or the y_j, and has 1^7-lifted type, so that $d':7$ also. Recursively expanding d', we get a general form of s for case 2:

$$\lambda\mathsf{y}_1\colon B_1^{17}\ldots\lambda\mathsf{y}_n\colon B_n^{17}\,.\,\lambda\mathsf{z}\colon 7\,.\,\mathsf{u}_1\, t_{11}\ldots t_{1p_1}\big(\ldots\big(\mathsf{u}_q\, t_{q1}\ldots t_{qp_q}\,(e)\big)\ldots\big)$$

where $q\geqslant 0$, each u_k is one of the x_i or the y_j, each t_{kl} has a 1^7-lifted type C_{kl}^{17}, and $e:7$ is not an application, and hence must be either z, or one of the $\mathsf{z}_{h'}$, or a λ-abstraction $\lambda\mathsf{z}'\colon 0.\,a$.

The argument now proceeds similarly to case 1. In equations (4.4) and (4.5) the sequence $t_1\ldots t_p$ must be replaced with $t_{11}\ldots t_{1p_1}\ldots t_{q1}\ldots t_{qp_q}$, giving

$$\Gamma'^{7}_B\, s\left[\mathsf{x}_i/\Gamma^7_{A_i}\right]_{i=1}^{m} \;\equiv_\beta\; \lambda\mathsf{z}\colon 7\,.\,\Gamma'^{7}_{C_{11}}\, t_{11}\,\big(\ldots\big(\Gamma'^{7}_{C_{qp_q}}\, t_{qp_q}\,(e)\big)\ldots\big)\left[\Phi\right] \tag{4.6}$$

and

$$\Sigma'_B\, s\left[\mathsf{x}_i/\Sigma_{A_i}\right]_{i=1}^{m} \;\equiv_\beta\; @\big(\lambda\mathsf{z}\colon 7\,.\,\Sigma'_{C_{11}} t_{11}\big(\ldots\big(\Sigma'_{C_{qp_q}} t_{qp_q}(e)\big)\ldots\big)\big)\left[\Psi\right]. \tag{4.7}$$

where $[\Phi]$ and $[\Psi]$ are as in case 1, but setting $n'=n$. If clause II or III holds for some of the t_{kl}, then clause II or III holds for s, just as in case 1 above. If clause I holds for all the t_{kl} then we cannot immediately infer that clause I holds for s; we must consider the possible values of e.

If $e=\mathsf{z}$, then clause I does hold for s.

If $e=\lambda\mathsf{z}'\colon 0.\,a$ for some a, then applying lemma 4.3, we can simplify equation (4.6) to obtain

$$\Gamma'^{7}_B\, s\left[\mathsf{x}_i/\Gamma^7_{A_i}\right]_{i=1}^{m} \;\equiv_\beta\; \lambda\mathsf{z}\colon 7\,.\,\lambda\mathsf{z}'\colon 0\,.\,a\left[\Phi\right]$$

and clause II holds for s.

If $e=\mathsf{z}_{h'}$ with $1\leqslant h'\leqslant h$, then applying lemma 4.3, we can simplify equation (4.7) to obtain

$$\Sigma'_B\, s\left[\mathsf{x}_i/\Sigma_{A_i}\right]_{i=1}^{m} \;\leqslant^{\mathsf{m}}\; @(\lambda\mathsf{z}\colon 7\,.\,\mathsf{z}_{h'}).$$

Lemma 4.1 shows that $@(\lambda\mathsf{z}\colon 7.\,b)\equiv^{\mathsf{m}}\lambda\mathsf{z}\colon 7.\,K_\perp$ for each closed $\perp$-term $b:7$, so that $\lambda\mathsf{z}_{h'}\colon 7.\,@(\lambda\mathsf{z}\colon 7.\,\mathsf{z}_{h'})\equiv^{\mathsf{m}}\lambda\mathsf{z}_{h'}\colon 7.\,\lambda\mathsf{z}\colon 7.\,K_\perp$ and hence $@(\lambda\mathsf{z}\colon 7.\,\mathsf{z}_{h'})\equiv^{\mathsf{m}}\lambda\mathsf{z}\colon 7.\,K_\perp$ and clause III holds for s. ∎

5 Retraction giving all 7-lifts

Theorem 4.4 gives an interesting set of lifted terms, but does not characterise the property of being 7-lifted. This leaves us with two problems to deal with.

The first is that it does not apply at all types in the form B^7—only those that are in the form $(B^1)^7$. This will be overcome by giving a term of type $(B^1)^7 \Rightarrow B^7$ that has every 7-lifted term in its range (up to $\equiv_{\beta\eta}$).

The second is that it is not an "if and only if". While we could give an exact characterisation of the solutions to (4.2) and (4.3), we just give a sufficiently large set of terms for which they hold.

We extend the lift operations $(\cdot)^1$ and $(\cdot)^7$ to $\perp$-terms in an ad hoc manner by setting $\perp^1 = \lambda \mathsf{z}\colon 0.\, \mathsf{z}$ and $\perp^7 \sim [\![\perp]\!] \in \mathcal{M}_0$. These map $\perp$-terms to terms and are chosen to make the rest of this section work.

Theorem 5.1 Suppose that $t : B$ is a closed $\perp$-term and let $s = (t^1)^7$. Then equations (4.2) and (4.3) hold.

Proof. The closed term $\Gamma_B'^7 s$ is 7-lifted and so (4.2) must hold as $\lambda \mathsf{z}\colon 7.\, \mathsf{z}$ is the only closed normal 7-lifted term of type $7 \Rightarrow 7$. For (4.3), we verify that if t is a $\perp$-term with

$$\mathsf{x}_1\colon A_1 \ldots \mathsf{x}_m\colon A_m \vdash t : B$$

then

$$\Sigma_B'\, t^{17}[\mathsf{x}_i/\Sigma_{A_i}]_{i=1}^m \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7\, .\, \mathsf{z}.$$

Since $\equiv^{\mathsf{m}}$ is invariant under $\beta\eta$-conversion, it suffices to consider t that are β-normal and η-expanded. We use induction on t. B is in the form $[B_1 \ldots B_n]$. If $t = \lambda \mathsf{y}_1\colon B_1 \ldots \lambda \mathsf{y}_n\colon B_n.\perp$ then

$$\Sigma_B'\, t^{17}[\mathsf{x}_i/\Sigma_{A_i}]_{i=1}^m \;\equiv_\beta\; @(\perp^{17}) \;=\; @(\lambda \mathsf{z}\colon 7\, .\, \mathsf{z}) \;\equiv^{\mathsf{m}}\; \lambda \mathsf{z}\colon 7\, .\, \mathsf{z}.$$

If $t = \lambda \mathsf{y}_1\colon B_1 \ldots \lambda \mathsf{y}_n\colon B_n.\, \mathsf{v}\, u_1 \ldots u_p$, where $\mathsf{v} : C = [C_1 \ldots C_p]$ is one of either the x_i or the y_j, then expanding Σ_B' and Σ_C we get

$$\Sigma_B'\, t^{17}[\mathsf{x}_i/\Sigma_{A_i}]_{i=1}^m \;\equiv_\beta\; @\big(\lambda \mathsf{z}\colon 7\, .\, \Sigma_{C_1}' u_1^{17}\big(\ldots\big(\Sigma_{C_p}' u_p^{17}(\mathsf{z})\big)\ldots\big)\big)\,[\Psi]$$

where $[\Psi] = [\mathsf{x}_i/\Sigma_{A_i}]_{i=1}^m[\mathsf{y}_j/\Sigma_{B_j}]_{j=1}^n$. Applying the induction hypothesis to each u_k we have $\Sigma_{C_k}' u_k^{17}[\Psi] \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$ and so the RHS above is $\equiv^{\mathsf{m}} @(\lambda \mathsf{z}\colon 7.\, \mathsf{z}) \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7.\, \mathsf{z}$. ∎

We shall show that any type A is the image of A^1 by a $\perp$-definable retraction. This will enable us to relate $\perp$-definability at type A to definability at type A^1, and to matching problems at type $(A^1)^7$.

Our retraction is given at each type A by a pair of closed $\perp$-terms $\mathsf{inj}_A : A \Rightarrow A^1$ and $\mathsf{proj}_A : A^1 \Rightarrow A$. These are defined at type 0 by:

$$\mathsf{proj}_0 = \lambda \mathsf{x}\colon 1\, .\, \mathsf{x}\perp, \qquad \mathsf{inj}_0 = \lambda \mathsf{x}\colon 0\, .\, \lambda \mathsf{z}\colon 0\, .\, \mathsf{x},$$

and extended to higher types by composition:

$$\mathsf{proj}_{A\Rightarrow B} = \lambda \mathsf{f}\colon A^1 \Rightarrow B^1\, .\, \lambda \mathsf{x}\colon A\, .\, \mathsf{proj}_B(\mathsf{f}(\mathsf{inj}_A\, \mathsf{x}))$$

and

$$\mathsf{inj}_{A\Rightarrow B} = \lambda \mathsf{f}\colon A \Rightarrow B\, .\, \lambda \mathsf{x}\colon A^1\, .\, \mathsf{inj}_B(\mathsf{f}(\mathsf{proj}_A\, \mathsf{x})).$$

Lemma 5.2 1. $\mathsf{proj}_A(\mathsf{inj}_A\ x) \equiv_{\beta\eta} x$ for any $x : A$. 2. $\mathsf{proj}_B\, s^1 \equiv_{\beta\eta} s$ for any closed $\perp$-term $s : B$.

PROOF. 1. is easily verified for $A = 0$ and extends by induction to higher types using the equation

$$\mathsf{proj}_{A\Rightarrow B}(\mathsf{inj}_{A\Rightarrow B}\ f) \equiv_\beta \lambda \mathsf{x}\colon A\ .\ \mathsf{proj}_B(\mathsf{inj}_B(f(\mathsf{proj}_A(\mathsf{inj}_A\ \mathsf{x})))).$$

For 2., we show that if

$$\mathsf{x}_1\colon A_1 \ldots \mathsf{x}_m\colon A_m \vdash s : B$$

where s is a normal $\perp$-term, then

$$\mathsf{proj}_B\ s^1[\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=1}^m \equiv_{\beta\eta} s \tag{5.1}$$

and if s is a variable or application term, also

$$s^1[\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=1}^m \equiv_{\beta\eta} \mathsf{inj}_B\ s. \tag{5.2}$$

Note that (5.2) implies (5.1) by 1. We use induction on s. If $s = \perp$ then (5.1) is immediate from the definitions of proj and $\perp^1$. If s is a variable, then (5.2) is trivial. If $s = \lambda \mathsf{x}_0\colon A_0 .\ s'$ with $B = A_0 \Rightarrow B'$, then using the definition of proj_B and the induction hypothesis,

$$\begin{aligned} \mathsf{proj}_B\ s^1[\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=1}^m &\equiv_\beta \lambda \mathsf{x}_0\colon A_0\ .\ \mathsf{proj}_{B'}\ s'^1[\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=0}^m \\ &\equiv_{\beta\eta} \lambda \mathsf{x}_0\colon A_0\ .\ s'. \end{aligned}$$

If s is an application $f\,c$ where $f : C \Rightarrow B$ and $c : C$, then as s is normal, f must be a variable or an application term. Using (5.2) for f, the definition of $\mathsf{inj}_{C\Rightarrow B}$ and then (5.1) for c, we have

$$\begin{aligned} f^1\, c^1\, [\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=1}^m &\equiv_{\beta\eta} \mathsf{inj}_{C\Rightarrow B}\, f\big(c^1\, [\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=1}^m\big) \\ &\equiv_\beta \mathsf{inj}_B\big(f\big(\mathsf{proj}_C\ c^1\, [\mathsf{x}_i/\ \mathsf{inj}_{A_i}\ \mathsf{x}_i]_{i=1}^m\big)\big) \\ &\equiv_{\beta\eta} \mathsf{inj}_B(f\,c) \end{aligned}$$

completing the induction. ∎

The next lemma extends theorem 2.5.

Lemma 5.3 For any closed $\perp$-term s, we have $s^7 \sim [\![s]\!]$.

PROOF. The $\perp$-term s is β-equivalent to $s'\perp$ for some closed term s'. Then theorem 2.5 and the definition of $\perp^7$ give

$$s^7 \equiv_\beta s'^7 \perp^7 \sim [\![s']\!]([\![\perp]\!]) = [\![s]\!]$$

as required. ∎

We are now ready to give a characterisation of definability:

Theorem 5.4 An element $\alpha \in \mathcal{M}_A$ is $\perp$-definable iff there is a closed term $s : (A^1)^7$ such that

$$\Gamma_A'^7\, s \equiv_\beta \lambda \mathsf{z}\colon 7\ .\ \mathsf{z} \tag{5.3}$$

and

$$\Sigma_A'\, s \equiv^{\mathsf{m}} \lambda \mathsf{z}\colon 7\ .\ \mathsf{z} \tag{5.4}$$

and

$$\mathsf{proj}_A^7\, s \sim \alpha. \tag{5.5}$$

PROOF. If $\alpha = [\![a]\!]$ for some closed normal $\perp$-term $a : A$, then let $s = a^{17}$. Then (5.3) and (5.4) are given by theorem 5.1, while $\mathsf{proj}^7_A\, s = \mathsf{proj}^7_A\, a^{17} \equiv_{\beta\eta} a^7 \sim [\![a]\!] = \alpha$ by the two lemmas above.

Conversely, suppose that the three equations hold, WLOG taking s to be normal. By theorem 4.4, $s = a^7$ for some $a : A^1$. Then $\mathsf{proj}^7_A\, a^7 \sim [\![\mathsf{proj}_A\, a]\!]$ and $\mathsf{proj}^7_A\, a^7 = \mathsf{proj}^7_A\, s \sim \alpha$, so that $\alpha = [\![\mathsf{proj}_A\, a]\!]$ by lemma 2.4 and is $\perp$-definable. ∎

The relations giving the characterisation above can be expressed as matching equations. This gives our main result.

Theorem 5.5 Higher order β matching is undecidable.

PROOF. Using theorems 1.1 and 2.1, it suffices to rewrite the three equations of theorem 5.4 as higher order matching problems, to be solved for s.

Equation (5.3) is already such an equation. Using the fact that $\equiv^{\mathsf{m}}$ coincides with $\equiv_\beta$ for closed $\perp$-terms with type 7, equation (5.4) can be replaced by the eight equations

$$\Sigma'_A\, s\, z \equiv_\beta z$$

for each $z \in \{K_\perp, \pi_1 \ldots \pi_7\}$. This is not in the required form, due to the use of $\perp$-terms.

However, if f and b are $\perp$-terms, then we have $f \equiv_\beta f'\perp$ and $b \equiv_\beta b'\perp$ for some terms f' and b', and for any closed term s, we have $f\, s \equiv_\beta b$ iff $\lambda \mathsf{x}\colon 0.\, f'\, \mathsf{x}\, s \equiv_\beta \lambda \mathsf{x}\colon 0.\, b'\, \mathsf{x}$. This shows that the eight equations can in turn be replaced by matching equations.

Equation (5.5) can be replaced by a matching equation using theorem 2.6. Noting that any solution to equations (5.3) and (5.4) is 7-lifted and hence encoding, the restriction to encoding terms in theorem 2.6 is of no consequence. Finally, it is easily verified that all these constructions are computable. ∎

The *order* of a type is defined by order(0) = 1, order($A \Rightarrow B$) is the maximum of $1 + \text{order}(A)$ and order(B). The undecidability of definability occurs at order 4, and the encoding in β-matching increases the order by two (one for the 7-lift and one for the 1-lift) to show undecidable at order 6.

This is one order higher than might have reasonably been expected. While the 7-lift is clearly fundamental to our argument, it is not obvious that it is absolutely necessary to introduce the extra 1-lift to characterise 7-lifted terms. Additionally, Padovani [9] shows decidability at order 4, so one might guess that undecidability starts at order 5.

In both [6] and [7], we derive significant structural constraints on terms that satisfy appropriate equations, without having to increase the order of the terms we consider. This also suggests that it may be possible to do away with the 1-lift.

There is nothing special about the use of the number 7 in our argument. A smaller number could be used, e.g., by encoding the result of [6].

As noted in the introduction, the arguments here do not apply to $\beta\eta$-matching. The absence of η is crucial to our proof. It is not plausible that a direct replacement of (4.2) could be found that would work in the presence of η-conversions. The author refrains from making any prediction as to the decidability of $\beta\eta$-matching on the basis of the current result.

Acknowledgements

Gerard Huet, Stefan Kahrs, Vincent Padovani and the referees are all thanked for providing helpful comments on earlier versions of this paper.

References

[1] Gilles Dowek. Third order matching is decidable. In *Proc. of the Seventh Annual IEEE Symposium on Logic in Computer Science*, pages 2–10, Santa Cruz, CA, 1992.

[2] Warren D. Goldfarb. The undecidability of the second-order unification problem. *Theoretical Computer Science*, 13(2):225–230, February 1981.

[3] G. P. Huet. The undecidability of unification in third order logic. *Information and control*, 22(3):257–267, April 1973.

[4] G. P. Huet. A unification algorithm for typed lambda-calculus. *Journal of Theoretical Computer Science*, 1(1):27–57, June 1975.

[5] G. P. Huet and B. Lang. Proving and applying program transformations expressed with second order patterns. *Acta Informatica*, 11:31–55, 1978.

[6] Ralph C. Loader. Finitary PCF is not decidable. *Theoretical Computer Science*, 266:341–364, 2001.

[7] Ralph C. Loader. The undecidability of λ-definability. In Anthony C. Anderson and Michael Zelëny, editors, *Logic, Meaning and Computation*, 2001.

[8] Vincent Padovani. Decidability of all minimal models. In M. Coppo and S. Beradi, editors, *Types for Proofs and Programs*, volume 1158 of *Lecture Notes in Computer Science*, Torino, June 1996. Springer.

[9] Vincent Padovani. Decidability of fourth order matching. *Mathematical Structures in Computer Science*, 3(1):361–372, June 2000.

[10] Aleksy Schubert. Linear interpolation for the higher-order matching problem. In *TAPSOFT*, pages 441–452, 1997.

[11] Aleksy Schubert. Algebraic higher-order matching. Technical Report TR 02-02 (267), Institute of Informatics, Warsaw University, 2002.

[12] R. Statman. Completeness, invariance and λ-definability. *Journal of Symbolic Logic*, 47(1):17–26, 1982.

[13] R. Statman. λ-definable functionals and $\beta\eta$-conversion. *Archiv fur Math. Logik und Grund.*, 23:21–26, 1983.

Received February 2002, revised November 2002.