A Decidable Variant of Higher Order Matching

Dan Dougherty[1] and ToMasz Wierzbicki[2,*]

[1] Wesleyan University, USA
[2] University of Wrocław, Poland

Abstract. A lambda term is *k-duplicating* if every occurrence of a lambda abstractor binds at most k variable occurrences. We prove that the problem of higher order matching where solutions are required to be k-duplicating (but with no constraints on the problem instance itself) is decidable. We also show that the problem of higher order matching in the affine lambda calculus (where both the problem instance and the solutions are constrained to be 1-duplicating) is in NP, generalizing de Groote's result for the linear lambda calculus [4].

1 Introduction

A lambda term is called *linear* if every bound variable occurs in it exactly once. In [4] de Groote defined *linear higher order matching* as the problem of deciding if there are linear solutions of a matching problem $M =^? N$, where both terms M and N are also linear, and showed that this problem is NP-complete. This relatively low complexity suggests that this is a quite restricted version of matching. Although de Groote's definition is well-justified by applications he is concerned with, nevertheless it seems interesting to consider another version, which also may be called linear, but which consists in deciding if there are linear solutions of an *arbitrary* rather than linear matching problem (incidentally, Levy [5] defined this way an analogous notion of the linear second order unification, by insisting on linearity of solutions, but allowing to use arbitrary terms in problems). This new linearity condition is much less restrictive than that of de Groote, since the new problem inherits the full complexity of β-reduction, and the best known lower bound for general matching also holds for it: the problem is non-elementary (see 1.1).

In the present paper we show decidability of this new version of matching. In fact we show that a rather more general problem is decidable. Say that a lambda term is k-duplicating if every bound variable occurs in it at most k times (Definition 2). We prove that, for any fixed k, a complete set of k-duplicating solutions of an arbitrary matching problem is always finite, and the problem of finding such a set is decidable.

Finally we strengthen slightly de Groote's result in the following way. Say that a term is *affine*, if it is 1-duplicating (an affine function may, as linear one, use its argument once, but it may also forget it). Using tools developed in

* Supported by KBN grant 8T 11C 04319

the paper we are able to replace "linear" with "affine" in de Groote's theorem and prove that the problem of deciding if an affine matching problem has affine solutions is in NP (thus is NP-complete).

All of our results apply to matching under β-equality as well as $\beta\eta$-equality. The k-duplicating matching is especially appealing in the $\lambda\beta$ calculus, since it turns out to be quite general decidable approximation of β-matching, which has recently been shown to be undecidable [6]. On the other hand the decidability of k-duplicating matching in $\beta\eta$-calculus sheds no light on the status of the full problem in this setting, which is still open.

There have been several other attempts to define some restricted variants of matching, for which decidability could be proved. The most successful include: bounding the order of solutions by 3 [3], 4 [9, 10, 1], or (in special case) by 5 [12], and restricting the signature to the first order (atomic) constants [8] (requiring at the same time that both sides of an instance are of base type). From the other hand a similar approach to ours has been applied to the problem of higher order unification [11].

1.1 Lower Bounds

To put our results in perspective it may be useful to recall that all of the variants of matching thus far studied in the full lambda calculus are non-elementary, since a straightforward reduction to the problem of checking β-convertibility [14] can be applied to them as well. We shall briefly recall this argument here. It is well known that the problem "given a closed pure term M of type $o^2 \to o$, is it true that $M \stackrel{*}{\to}_{\beta\eta} \lambda xy.x$?" is non-elementary [13, 7]. Since it is required that both terms M and N in the matching problem $M =^? N$ should be in normal form, we cannot just write the equation $M =^? \lambda xy.x$. However we may easily reduce this problem to matching in the following way: replace any redex $(\lambda x_i.P_i)Q_i$ in M with a term $f_i(\lambda x_i.P_i)Q_i$ obtaining this way a term M', which does not contain redexes, where f_i are "fresh" variables of appropriate types. Write equations $f_i =^? \lambda xy.xy$ together with $M' =^? \lambda xy.x$. If one insists on having a single equation instead of a set $\{M_j =^? N_j\}_{j=1}^n$, it suffices to transform it to $\lambda z.zM_1 \ldots M_n =^? \lambda z.zN_1 \ldots N_n$. The only solution of the set $\{f_i =^? \lambda xy.xy\}_i$ is a substitution $\theta = [f_i/\lambda xy.xy]_i$. Since $M'\theta \stackrel{*}{\to}_\beta M$, the whole problem has a (unique) solution θ if and only if $M \stackrel{*}{\to}_\beta \lambda xy.x$, which turns out to be non-elementary. The solution θ is linear. Of course M is *not* linear. In case of a nonempty signature it is sometimes required that both sides of a matching equation are of base type o. Suppose that the signature contains two constants 0 and 1 of type o. Then a reduction from the problem of checking if $M \stackrel{*}{\to}_{\beta\eta} \lambda xy.x$ to matching proceeds as follows: write a term $M10$, "hide" all β-redexes using fresh variables f_i obtaining a term M', then write the equation $M' =^? 1$ together with equations $f_i(\lambda x.x)0 =^? 0$ and $f_i(\lambda x.x)1 =^? 1$. Again the only solution of the last pair of equations is the substitution $[f_i/\lambda xy.xy]$. Recall, that a matching problem is *atomic* if all right hand sides of equations are single constants of type o. Atomic matching is decidable [8]. Thus pure as well as atomic matching with linear solutions is non-elementary. Hence also pure and atomic k-duplicating, for any $k \geq 1$, as well as general matching is non-elementary.

2 Preliminaries

Let $\mathbb{T}$ be the set of *simple types* built over one base type o and given by the following grammar: $\mathbb{T} ::= o \mid \mathbb{T} \to \mathbb{T}$. We assume that $\to$ associates to the right, and omit parentheses whenever possible. Small Greek letters σ, τ, ρ, etc., denote arbitrary types. The notions of *size* and *order* of types are defined as usual:

$$\begin{aligned} |o| &= 1 \\ |\sigma \to \tau| &= 1 + |\sigma| + |\tau| \\ \mathrm{order}(o) &= 1 \\ \mathrm{order}(\sigma \to \tau) &= \max(\mathrm{order}(\sigma) + 1, \mathrm{order}(\tau)) \end{aligned}$$

Let $\langle \mathcal{X}, \mathrm{type}(\cdot) \rangle$ be a set of *variables* together with a function $\mathrm{type} : \mathcal{X} \to \mathbb{T}$, which assigns a unique type to each variable $x \in \mathcal{X}$. We assume that there are infinitely many variables of each type. Similarly let $\langle \Sigma, \mathrm{type}(\cdot) \rangle$ be a collection of *constants*. The set Σ is sometimes called *a signature*. This set may be empty, finite or infinite (we say, that the lambda calculus is *pure*, if $\Sigma = \emptyset$, and *atomic*, if $\mathrm{type}(c) = o$ for all $c \in \Sigma$). We use small Latin letters x, y, z, etc., to denote arbitrary variables, and a, c, f, etc., to denote arbitrary constants.

Let $\mathcal{X}_\sigma = \{x \in \mathcal{X} \mid \mathrm{type}(x) = \sigma\}$ and $\Sigma_\sigma = \{c \in \Sigma \mid \mathrm{type}(c) = \sigma\}$ be sets of respectively variables and constants of type σ. For all types $\sigma \in \mathbb{T}$ we simultaneously define sets $\Lambda^{\to}_\sigma$ of *lambda terms of type* σ to be the smallest sets satisfying the following conditions:

1. $\mathcal{X}_\sigma \subseteq \Lambda^{\to}_\sigma$,
2. $\Sigma_\sigma \subseteq \Lambda^{\to}_\sigma$,
3. if $x \in \mathcal{X}_\sigma$ and $M \in \Lambda^{\to}_\tau$, then $\lambda x.M \in \Lambda^{\to}_{\sigma\to\tau}$,
4. if $M \in \Lambda^{\to}_{\sigma\to\tau}$ and $N \in \Lambda^{\to}_\sigma$, then $MN \in \Lambda^{\to}_\tau$.

We use capital letters M, N, P, Q, etc., to denote arbitrary terms. If $M \in \Lambda^{\to}_\sigma$, we sometimes write $\mathrm{type}(M) = \sigma$ or $M : \sigma$. The *size* of a term is defined inductively:

$$\begin{aligned} |x| &= 1 \\ |c| &= 1 \\ |MN| &= 1 + |M| + |N| \\ |\lambda x.M| &= 1 + |M| \end{aligned}$$

An *address* is a sequence $p \in \{0,1\}^*$. Empty sequence is denoted by ϵ. A *subterm* $M\restriction_p$ *of a term* M *at address* p is defined as follows:

$$\begin{aligned} M\restriction_\epsilon &= M \\ (\lambda x.M)\restriction_{0p} &= M\restriction_p \\ (M_1 M_2)\restriction_{0p} &= M_1\restriction_p \\ (M_1 M_2)\restriction_{1p} &= M_2\restriction_p \end{aligned}$$

We say that an address p is *valid* in a term M, if $M\restriction_p$ is well defined. Note that $(M\restriction_p)\restriction_q = M\restriction_{pq}$, where pq is the concatenation of sequences p and q. We say

that a term N is a *subterm* of M, and write $N \subseteq M$, if there exists an address p, such that $M\restriction_p = N$. The result of the *substitution of a term N at address p in a term M*, providing that p is valid in M and $\mathrm{type}(M\restriction_p) = \mathrm{type}(N)$, is a term $M[p\restriction N]$ obtained by replacing a subterm at address p in M with N. Similarly we define a *simultaneous* substitution $M[p_i\restriction N_i]_{i=1}^k$, providing there are no two addresses p_i and p_j, such that p_i is a prefix of p_j. The set $\mathrm{Occ}(x, M)$ of *free occurrences* of a variable x in a term M is the set of all addresses in M at which x occurs as a free variable. The set $\mathrm{FV}(M)$ of *free variables* of a term M is the set of those variables, which have at least one free occurrence in M. A term M is *closed*, if $\mathrm{FV}(M) = \emptyset$. The *substitution $M[x/N]$ of a term N for variable x in a term M*, providing that $\mathrm{type}(x) = \mathrm{type}(N)$, is the result of substitution of a term N at all free occurrences of variable x in M. Similarly we define the simultaneous substitution $M[x_i/N_i]_{i=1}^n$.

As usual we consider β- and η-reduction rules:

$$(\lambda x.M)N \rhd_\beta M[x/N] \qquad \lambda y.My \rhd_\eta M$$

for any variables x, y and terms M, N, such that $y \notin \mathrm{FV}(M)$. A binary relation R on $\Lambda_\sigma^\rightarrow$ is *monotonic*, if for every every terms M, N, P, Q, and variable x, such that MRN and PRQ, there is $(MP)R(NQ)$ and $(\lambda x.P)R(\lambda x.Q)$. The one step β-reduction $\rightarrow_\beta$ is the monotonic closure of $\rhd_\beta$, the one step η-reduction $\rightarrow_\eta$ is the monotonic closure of $\rhd_\eta$, the (many step) β-reduction $\stackrel{*}{\rightarrow}_\beta$ is the monotonic, reflexive and transitive closure of $\rhd_\beta$, and the (many step) $\beta\eta$-reduction $\stackrel{*}{\rightarrow}_{\beta\eta}$ is the monotonic, reflexive and transitive closure of $\rhd_\beta \cup \rhd_\eta$. The two relations $\stackrel{*}{\rightarrow}_\beta$ and $\stackrel{*}{\rightarrow}_{\beta\eta}$ lead to two slightly different lambda calculi.

A term M is a *β-redex* if it is of the form $(\lambda x.M_1)M_2$; a term is in *β-normal form* if it contains no β-redex. A term M is in $\bar{\eta}$-normal form (called also η-long form), if for every address p in M, such that $M\restriction_p$ is not an abstraction, and $\mathrm{type}(M\restriction_p) \neq o$, there is $p = q0$, and $M\restriction_q$ is an application. $\beta\bar{\eta}$-normal forms are unique modulo $\beta\eta$-conversion and are closed under substitutions and β-reduction.

A β-reduction sequence is a sequence of terms $(M_n)_{i=0}^n$, such that $M_i \rightarrow_\beta M_{i+1}$ for $i = 0, \ldots, n-1$. We say that this reduction sequence *starts* at M_0, *leads* to M_n, and *has length* n.

We say that a term M is an *instance* of N, and write $M \geq N$ (or say, that N is a *generalization* of M, and write $N \leq M$), if there exists a substitution θ, such that $M = N\theta$. The relation $\leq$ is a partial preorder on terms (i.e., is reflexive and transitive). Even though $\leq$ is not an order (is not weakly asymmetric), the notions of the smallest and greatest elements, lower and upper bounds, and suprema and infima of sets of terms with respect to $\leq$ are well defined. Unlike for order relations, these elements are usually not unique (but they are unique modulo renaming of free variables).

Lemma 1 (basic properties of the instance relation).

1. *Every nonempty set $\mathcal{P} \subseteq \Lambda_\sigma^\rightarrow$ of terms has an infimum,* $\inf \mathcal{P}$. *In particular, the set of the smallest elements in $\Lambda_\sigma^\rightarrow$ coincides with the set of variables $\mathcal{X}_\sigma$.*

2. *If a nonempty set $\mathcal{P}$ possesses an upper bound, then it has a supremum,* $\sup \mathcal{P}$.
3. *If $M_1 \leq M_2$ and p is a valid address in M_1, then p is a valid address in M_2, and $M_1\restriction_p \leq M_2\restriction_p$.*
4. *If there exists a substitution θ, such that $M_1\theta = M_2$, and $N_1\theta = N_2$, and p is a valid address in M_1, then p is a valid address in M_2, and $M_1[p\restriction N_1] \leq M_2[p\restriction N_2]$.*
5. *If there exists a substitution θ, such that $M_1\theta = M_2$, $N_1\theta = N_2$, and $x \notin \mathrm{Dom}(\theta)$, then $M_1N_1 \leq M_2N_2$, and $\lambda x.M_1 \leq \lambda x.M_2$.*
6. *If $M_1 \leq M_2$ (with additional proviso that there exists substitution θ, such that $M_1\theta = M_2$ and every $x \in \mathrm{Dom}(\theta)$ occurs at most once in M_1), and p is a valid address in M_2 but not in M_1, then $M_1 \leq M_2[p\restriction N]$ for any term N.*
7. *If $\mathrm{FV}(M) = \emptyset$ and $M \leq N$, then $M = N$.*
8. *If $M \leq N$, and $M\restriction_p$ is a β-redex, then $N\restriction_p$ is a β-redex.*
9. *If $M \leq N$, then $|M| \leq |N|$.*
10. *If $M = \sup\{N_i\}_{i=1}^k$, where for each i there exists a substitution θ_i such that $N_i\theta_i = M$ and every $x \in \mathrm{Dom}(\theta_i)$ occurs at most once in N_i, then $|M| \leq 1 - k + \sum_{i=1}^k |N_i|$.*

We define a similar instance relation for substitutions: we say that a substitution θ_1 is *at least as general as* θ_2, and write $\theta_1 \leq \theta_2$, if there exists a substitution ρ, such that $\theta_2 = \theta_1\rho$ (i.e., if $x\theta_1 \leq x\theta_2$ for every variable $x \in \mathcal{X}$).

2.1 Higher-order matching

Higher-order $\beta\eta$- (resp. β-) matching is the following problem: "Given two terms M and N of the same type, in $\beta\bar{\eta}$- (resp. β-) normal form, where N is closed, usually written in the form of equation $M =^? N$. Is there any substitution θ for free variables in M, where all terms $x\theta$ for $x \in \mathrm{FV}(M)$ are in $\beta\bar{\eta}$- (resp. β-) normal form, such that $M\theta \stackrel{*}{\to}_\beta N$?" We say, that θ is a *solution* of the instance $M =^? N$. There is a subtle difference between those two variants of matching [6]. In particular it is known that β-matching is undecidable, while decidability of $\beta\eta$-matching is still open. The results of the present paper apply to both variants. A set $\mathcal{S}$ of substitutions is a *complete set of solutions* of a matching problem $M =^? N$, if for every $\theta \in \mathcal{S}$ and every $\theta' \geq \theta$, the substitution θ' is a solution of $M =^? N$, and for every solution θ' of $M =^? N$ there exists a substitution $\theta \in \mathcal{S}$, such that $\theta \leq \theta'$.

It is not the case that if a matching problem $M =^? N$ has a solution then it has a closed solution, since, depending on the signature Σ, the types of some of the free variables of M may not be inhabited. This motivates the following definition and lemma.

Definition 1. *A term is* almost-closed *if its free variables (if any) are all of base type. A substitution θ is almost-closed if each $\theta(x)$ is an almost-closed term.*

Lemma 2. *If a matching problem has a solution then it has an almost-closed solution.*

Proof. If θ is a solution to $M =^? N$ and $\theta(x)$ has a variable $v : \tau$ free then we may choose some base-type variable z and replace v by the term $\lambda y_1 \ldots y_n.z : \tau$; since v does not occur in N this is obviously still a solution.

3 Decidability of k-Duplicating Matching

Definition 2. *A lambda term is* k-duplicating *if for every subterm* $N \subseteq M$, *such that* $N = \lambda x.P$, *there are at most* k *free occurrences of* x *in* P, *i.e.,* $|\operatorname{Occ}(x, P)| \leq k$.

In this section we show the decidability of k-duplicating matching, that is, matching with no restriction on the problem instance but with the constraint that solutions must be k-duplicating. It will suffice to show that we can, given a matching problem, effectively generate a finite set of terms which includes all the potential k-duplicating solutions to the problem.

The essential insight is that for each k there are in fact only finitely many *pure* k-duplicating normal forms at each type (independently of the matching problem). When the given matching problem contains constants there is a slight complication since we must allow for the fact that these constants may occur in solutions. But Lemma 3 below shows that we can bound the number of occurrences of constants in potential solutions by inspecting the problem.

Definition 3. *Let* N *be a normal-form term over the signature* Σ *and let* $c \in \Sigma$. *Then* $\#(c, N)$ *is the number of occurrences of* c *in* N.

Lemma 3. *Let* $M =^? N$ *be a matching problem and let* c *be some constant. For both* β- *and* $\beta\eta$-*matching: if the matching problem has a solution it has an almost-closed solution in which each solution term* X *satisfies* $\#(c, X) \leq \#(c, N)$.

Proof. For simplicity of notation suppose there is only one free variable x in M. Let X be a solution, so that $M[x/X] = N$. By Lemma 2 we may assume that X is almost closed.

Let us define the notion of a c-occurrence in X being *relevant* to the solution, as follows. For each occurrence of c inside of X choose some fresh constant of the same type as c: let us call these new ones $c^1, \ldots, c^m$. Build X' by replacing the given occurrences of c by the c^i in X, and let N' be the normal form of $M[x/X']$. Obviously N' will differ from N just in that some of the c occurrences in N will now be certain c^i. The original c occurrences in X whose corresponding c^i appear in N' are declared to be the relevant ones.

Now: obviously no more than $\#(c, N)$ such c^i will occur in N', so there are at most this many relevant occurrences in X. If in X we replace the non-relevant c-occurrences by an arbitrary term of the right type then we still have a solution. If we replace each non-relevant occurrence by an almost closed term, say $\lambda \boldsymbol{y}.z$ where z is some base-type variable, the lemma follows.

Definition 4. *Let* $\sigma \prec \tau$ *mean that* $\operatorname{order}(\sigma)$ *is less than* $\operatorname{order}(\tau)$ *Let* $\prec^*$ *be the multiset extension of* $\prec$.

Definition 5. *Given*

$$M = \lambda x_1..\lambda x_n.B,$$

let $\|M\|$ *be the multiset of types formed by including type* τ *with multiplicity* m *if* τ *is* not *a base type and in* B *there are* m *occurrences of free variables or constants of type* τ.

Then write $M \prec^* N$ *if* $|M| \prec^* |N|$.

Note that in computing $\|M\|$ we consider, in B, constants and the x_i as well as variables which are not among the x_i. There should be no ambiguity in using $\prec^*$ to refer to both the relation on type-multisets and the relation on terms. The relations $\prec^*$ are well-founded [2].

Lemma 4. *Let* $M \equiv \lambda x_1 \ldots \lambda x_n.hM_1 \cdots M_p$ *(where the atom* h *is either a constant, one of the* x_i*, or a free variable). Then* $M_j \prec^* M$ *for all* j.

Proof. Let the type M be $\sigma_1 \to \cdots \to \sigma_n \to \sigma$; let the type of M_i be μ_i, for $1 \leq i \leq p$; so that the type of h is $\mu = \mu_1 \to \cdots \to \mu_m \to \sigma$.

Of course if h has base type then $p = 0$ and the lemma is vacuously true. To verify the claim in the non-trivial case: let μ_j be $\delta_1 \to \cdots \to \delta_d \to o$.

In comparing $\|M_j\|$ with $\|M\|$ we have lost one occurrence of μ, the type of h, and possibly added occurrences of the types $\delta_1, \ldots, \delta_d$, due to the fact that M_j may have bound variables of these types (free in the matrix of M_j but not free in M). But each δ_k is of lower order than μ. This establishes the lemma.

The preceding is a general fact about all terms, k-duplicating or not. But the hypothesis of k-duplication and bounds on the occurrences of constants permit us to bound $\|M\|$.

Proposition 1. *Let* w *be a function assigning to each constant* c *in* Σ *a non-negative integer* $w(c)$*. For each* σ *and* k *we can effectively write down finitely many terms* M *comprising all of the almost-closed* β*-normal forms of type* σ *which are* k*-duplicating and satisfy* $\#(c, M) \leq w(c)$.

Proof. Let $M : \sigma_1 \to \cdots \to \sigma_n \to o$ satisfy the hypotheses. Recall that base-type variables do not figure in computing $\|M\|$. Since M is almost-closed we may observe that $\|M\|$ is bounded by the multiset consisting of

- k copies of each σ_i, and
- for each constant c of non-base type τ, $\#(c, M)$ copies of τ.

Together with the fact that $\prec^*$ is well-founded this means that we may effectively bound the depth of all the M satisfying the hypotheses. This establishes the proposition.

So we can solve the general k-duplicating matching problem by an exhaustive search:

Theorem 1. *The problems of* k*-duplicating* β*-matching and* $\beta\eta$*-matching are each decidable.*

Proof. Let problem $M =^? N$ be given. For each free variable $x : \sigma$ of M we may, by Lemma 1, compute the set of almost-closed k-duplicating β-normal forms X of type σ satisfying $\#(c, X) \leq \#(c, N)$ for each c. If we are considering $\beta\eta$ matching we filter out those terms which are not in $\bar{\eta}$-normal form. By Lemma 3 this is a complete search procedure for candidates for a solution.

4 Complexity of Matching in the Affine Lambda Calculus

Definition 6. *A lambda term is* linear *(resp.* affine*, a* λI-term*), if for every subterm* $N \subseteq M$*, such that* $N = \lambda x.P$*, there is exactly one (resp. at most one, at least one) free occurrence of* x *in* P*, i.e.,* $|\operatorname{Occ}(x, P)| = 1$ *(resp.* $|\operatorname{Occ}(x, P)| \leq 1$*,* $|\operatorname{Occ}(x, P)| \geq 1$*).*

Note, that the notion of affinity coincides with 1-linearity from our previous investigations, and that a term is linear exactly when it is an affine λI-term. As opposed to λI-terms, arbitrary terms are sometimes called λK-terms. The sets of linear, affine and λI-terms are closed under β-reduction (as opposed to k-duplicating terms, for $k \geq 2$, which are not). Thus it is possible to investigate theories of lambda conversion for such terms, and to consider the ordinary problem of higher order matching in these calculi, instead of modifying the formulation of the problem itself. Such variants of matching are less general than the variant considered above, since we put restrictions not only on the form of solutions, but also on the instances of the problem. One might expect, that the complexity of such problems should be lower than that of the k-duplicating matching. Indeed, de Groote [4] showed, that matching in the linear lambda calculus is NP-complete. In this section we generalize his proof to the case of affine lambda calculus.

Consider the following *potential function*:

$$\Phi(M) = \sum_{\substack{N \subseteq M \\ N = (\lambda x.N_1)N_2}} |\operatorname{type}(\lambda x.N_1)|.$$

Lemma 5 (de Groote). *If* M *is affine, and* $M \to_\beta N$*, then* $\Phi(M) > \Phi(N)$*.*

Proof. Suppose that M is reduced to N at address p, that is $M\restriction_p = (\lambda x.P)Q$ and $N = M[p \restriction P[x/Q]]$. In a reduction step $(\lambda x.P)Q \rhd_\beta P[x/Q]$ all redexes occurring in P and Q are left intact, and since x occurs in P at most once, there may be created at most one new redex inside P as a result of substitution. This new redex will be created if x occurs at active position (as an operator in an application) in a subterm of the form xP' and Q is a lambda abstraction. Moreover the reduct $P[x/Q]$ may also occur in active position and if it turns out to be a lambda abstraction, it will be a part of another new redex. Let $\operatorname{type}(\lambda x.P) = \sigma \to \tau$. Closing the redex $(\lambda x.P)Q$ decreases the potential by $|\operatorname{type}(\lambda x.P)|$, and adds to it at most $|\operatorname{type}(x)| = |\sigma|$ plus $|\operatorname{type}(P[x/Q])| = |\tau|$. Thus $\Phi(N) \leq \Phi(M) - |\operatorname{type}(\lambda x.P)| + |\operatorname{type}(x)| + |\operatorname{type}(P[x/Q]) = \Phi(M) - |\sigma \to \tau| + |\sigma| + |\tau| = \Phi(M) - 1$.

Definition 7. *We say that a β-reduction sequence $(M_i)_{i=0}^m$ is* variable-only-forgetting, *if for every $i = 0, \ldots, m-1$ and $p \in \{0,1\}^*$, if $M_i\restriction_p = (\lambda x.P)Q \rhd_\beta M_{i+1}\restriction_p$, and $x \notin \mathrm{FV}(P)$, then Q is a variable.*

Since if $x \notin \mathrm{FV}(P)$, then $|(\lambda x.P)y| = 3 + |P| = 3 + |P[x/y]|$, and if $x \in \mathrm{FV}(P)$ and x occurs $n > 0$ times in P, then $|P[x/Q]| = |P| + n(|Q| - 1) \geq |P| + |Q| - 1$, so $|(\lambda x.P)Q| = |P| + |Q| + 2 \leq |P[x/Q]| + 3$, we have:

Lemma 6. *If $(M_i)_{i=0}^m$ is variable-only-forgetting, then $|M_m| - |M_0| \leq 3m$.*

The next two lemmas state that every β-reduction sequence containing only affine terms can be replaced with an equivalent (in some sense) variable-only-forgetting sequence. The key idea is to replace any subterm that is forgotten during reduction with a fresh free variable (which is allowed to be forgotten).

Lemma 7. *For any β-reduction sequence $(M_i)_{i=0}^m$ containing only affine terms and any term $N' \leq M_m$ (with additional proviso that there exists substitution θ, such that $N'\theta = M_m$ and every variable from $\mathrm{Dom}(\theta)$ occurs at most once in N'), there exists a variable-only-forgetting reduction sequence $(N_j)_{j=0}^n$ leading to N', where $N_0 \leq M_0$, and $n \leq m$.*

Proof. In the whole proof when stating that $M \leq N$ we will tacitly assume, that there exists a substitution θ, such that $M\theta = N$ and every variable from $\mathrm{Dom}(\theta)$ occurs in M at most once. The proof is by induction on the length m of the reduction sequence $(M_i)_{i=0}^m$. For $m = 0$ the conclusion is obvious. Suppose that such a sequence $(N_j)_{j=1}^n$ exists for an arbitrary sequence $(M_i)_{i=1}^m$ of length m (for convenience we shifted indexes i and j by one), and consider a sequence $(M_i)_{i=0}^m$ of length $m+1$. Since $M_0 \to_\beta M_1$, suppose that M_0 is reduced to M_1 at address p, that is $M_0\restriction_p = (\lambda x.P)Q \rhd_\beta P[x/Q] = M_1\restriction_p$. By induction assumption $N_1 \leq M_1$. If p is not a valid address in N_1, then $N_1 \leq M_0$, by Lemma 1.6. Thus $(N_j)_{j=1}^n$ is the desired reduction sequence. Otherwise we need to find a new term N_0 as shown in the following diagram:

$$\begin{array}{ccccc} M_0 & \to_\beta & M_1 & \overset{*}{\to}_\beta & M_m \\ \mid\vee & & \mid\vee & & \mid\vee \\ N_0 & \to_\beta & N_1 & \overset{*}{\to}_\beta & N_n \end{array}$$

in such a way, that the reduction step $N_0 \to_\beta N_1$ is variable-only-forgetting. Fix a fresh free variable y (which does not occur anywhere else in considered terms). Since $N_1 \leq M_1$, then by Lemma 1.3 we have

$$N_1\restriction_p \leq M_1\restriction_p = P[x/Q]. \tag{1}$$

The term M_1 is affine, so $|\mathrm{Occ}(x,P)| \leq 1$. If $|\mathrm{Occ}(x,P)| = 0$, i.e., $x \notin \mathrm{FV}(P)$, then $P[x/Q] = P$, thus $N_1\restriction_p = P$. Hence, by Lemma 1.5 we have $P' = (\lambda x.(N_1\restriction_p))y \leq (\lambda x.P)Q$, where y is fresh. Let $N_0 = N_1[p \backslash P']$. By Lemma 1.4 we get $N_0 \leq M_1[p \backslash (\lambda x.P)Q] = M_0$. Since $N_1\restriction_p$ does not contain free occurrences of x, then $P' \rhd_\beta N_1\restriction_p$, thus $N_0 \to_\beta N_1$. Moreover $(N_j)_{j=0}^n$ is a desired variable-only-forgetting sequence leading to M_m and not longer than $(M_i)_{i=0}^m$. Now suppose that $|\mathrm{Occ}(x,P)| = 1$, and let $\mathrm{Occ}(x,P) = \{q\}$. If q is not a valid address

in $N_1|_p$, then, by Lemma 1.6, we have $N_1|_p \leq (M_1|_p)[q|x] = P$. From (1) and Lemma 1.5 we have $P' = (\lambda x.(N_1|_p))y \leq (\lambda x.P)Q$, where y is fresh. So let, as before, $N_0 = N_1[p|P']$. By Lemma 1.4 we get $N_0 \leq M_1[p|(\lambda x.P)Q] = M_0$. Since $N_1|_p$ does not contain free occurrences of x, then $P' \rhd_\beta N_1|_p$, thus $N_0 \to_\beta N_1$. Moreover $(N_j)_{j=0}^n$ is a desired variable-only-forgetting sequence leading to M_m and not longer than $(M_i)_{i=0}^m$. The last case to be considered is when q is a valid address in $N_1|_p$. From (1) and Lemma 1.3 we have $N_1|_{pq} \leq (P[x/Q])|_q = Q$. By Lemma 1.4 we get $(N_1|_p)[q|x] \leq (P[x/Q])[q|x] = P$, so by Lemma 1.5 we have $P' = (\lambda x.((N_1|_p)[q|x]))(N_1|_{pq}) \leq (\lambda x.P)Q$. Let $N_0 = N_1[p|P']$. Since $N_1 \leq M_1$, then by Lemma 1.4 we have $N_0 \leq M_1[p|(\lambda x.P)Q] = M_0$. Moreover $P' \rhd_\beta (N_1|_p)[q|N_1|_{pq}] = N_1|_p$, thus $N_0 \to_\beta N_1$. The reduction sequence $(N_j)_{j=0}^n$ is not longer than $(M_i)_{i=0}^m$, and, since x occurs in $(N_1|_p)[q|x]$, it is also variable-only-forgetting.

Lemma 8. *If $M \stackrel{*}{\to}_\beta M'$ and $\mathrm{FV}(M') = \emptyset$, then $N \stackrel{*}{\to}_\beta M'$ for every $N \geq M$.*

Proof. It is sufficient to show that if $(M_i)_{i=0}^m$ is a reduction sequence such that $\mathrm{FV}(M_m) = \emptyset$, then for any N_0 there exists a reduction sequence $(N_i)_{i=0}^m$, such that $N_m = M_m$. Proof is by induction on m. For $m = 0$ we have $N_0 \geq M_0$, so by Lemma 1.7 we get $N_0 = M_0$, and $(N_i)_{i=0}^0$ is the desired sequence. Now suppose that for any sequence $(M_i)_{i=1}^m$ and any $N_1 \geq M_1$ such a sequence $(N_i)_{i=1}^m$ exists (for convenience, as in previous lemma, we shifted the index i by one), and consider a sequence $(M_i)_{i=0}^m$ and a term $N_0 \geq M_0$. We need to find a term N_1, as shown in the following diagram:

$$\begin{array}{ccccc} M_0 & \to_\beta & M_1 & \stackrel{*}{\to}_\beta & M_m \\ |\wedge & & |\wedge & \nearrow^{*} & \\ N_0 & \to_\beta & N_1 & & \end{array}$$

Since $M_0 \to_\beta M_1$, there exists an address p, such that $M_0|_p \rhd_\beta M_1|_p$. Since $N_0 \geq M_0$, then by Lemma 1.8 the subterm $N_0|_p$ is a β-redex. Suppose $N_0|_p = (\lambda x.P)Q$. Let N_1 be the term obtained by contracting a redex at address p, i.e., $N_1 = N_0[p|P[x/Q]]$. By Lemma 1.3 we have $(\lambda x.P)Q \geq M_0|_p$. Thus $P[x/Q] \geq M_1|_p$, so $N_1 \geq M_1$. By induction hypothesis there exists a reduction sequence $(N_i)_{i=1}^m$ leading to M_m. Then $(N_i)_{i=0}^m$ is the desired reduction sequence leading to M_m and starting from N_0.

Using Lemmas 7 and 8 we are able to prove the following:

Proposition 2. *For any matching problem $M =^? N$ in the affine lambda calculus there always exists a finite complete set of solutions $\mathcal{S}$, such that if*

$$[x/N_x]_{x \in \mathrm{FV}(M)} \in \mathcal{S},$$

then

$$|N_x| \leq |N| + 3 \sum_{x \in \mathrm{FV}(M)} |\operatorname{type}(x)| \cdot |\operatorname{Occ}(x, M)|, \tag{2}$$

for any $x \in \mathrm{FV}(M)$. (In particular $\mathcal{S}$ may be empty, if $M =^? N$ has no solution.)

Proof. Let θ be an arbitrary solution of $M =^? N$. It may be too big to satisfy (2), since it may contain huge terms that are forgotten during reduction of $M\theta$ to N. Suppose $(M_i)_{i=0}^m$ is a reduction sequence starting from $M\theta$ and leading to N. By Lemma 7 there exists a variable-only-forgetting sequence $(N_j)_{j=0}^n$, such that $N_0 \leq M_0$, $N_n = M_m = N$, and $n \leq m$. Now N_0 contains only necessary fragments of $M\theta$, so let

$$N_x = \sup(\{N_0\restriction_p \mid p \in \mathrm{Occ}(x, M) \text{ and } p \text{ valid in } N_0\} \cup \{x\}).$$

Since $N_0 \leq M\theta$, then $N_0\restriction_p \leq (M\theta)\restriction_p = x\theta$ by Lemma 1.3, for any $p \in \mathrm{Occ}(x, M)$. Thus the above set possesses an upper bound $x\theta$, and by Lemma 1.2 there exists its supremum, so N_x is well-defined. Let $\theta' = [x/N_x]_{x \in \mathrm{FV}(M)}$. We have immediately $\theta' \leq \theta$. Moreover for any substitution $\theta'' \geq \theta'$, since $M\theta'' \geq M\theta' \geq N_0$ and N is closed, then by Lemma 8 there exists a reduction sequence starting from $M\theta''$ and leading to N. Thus θ'' is a solution of $M =^? N$. From the other hand, by Lemma 6 we have $|N_0| - |N| \leq 3n$, while by Lemma 5 we have $n \leq \Phi(N_0) - \Phi(N)$. Note that $\Phi(N) = 0$, since N is in normal form. Comparing the last two inequalities we get

$$|N_0| \leq |N| + 3n \leq |N| + 3\Phi(N_0).$$

By Lemmas 1.9 and 1.10 we have $|N_x| \leq |N_0|$. From the other hand it is easy to see, that since $N_0 \leq M\theta$, then $\Phi(N_0) \leq \Phi(M\theta)$. Moreover, since M and N_x are in normal form, then all redexes in $M\theta$ are those of the form $N_x P'$, created as a result of substitution of terms N_x for variables from $\mathrm{FV}(M)$. Since $|\,\mathrm{type}(N_x)| = |\,\mathrm{type}(x)|$, we get (2).

We have shown that for any solution θ of $M =^? N$ there exists a substitution $\theta' \leq \theta$ satisfying (2), such that any $\theta'' \geq \theta'$ is a solution of $M =^? N$. Let $\mathcal{S}$ be the set of such substitutions θ' for all solutions θ of $M =^? N$. The set $\mathcal{S}$ is finite, since the size of θ' is bounded.

Theorem 2. *The problem of checking if an instance of a matching problem in the affine lambda calculus has a solution is NP-complete.*

Proof. The inequality (2) gives an upper bound on the size of solution, which depends only on terms M and N, and is a polynomial function of the length of any reasonable encoding of the problem $M =^? N$ as a word over a finite alphabet. Thus it suffices to guess terms N_x and check if $\theta = [x/N_x]_{x \in \mathrm{FV}(M)}$ is a solution. Since a normal form of a linear term may be found in a polynomial number of steps, the problem of checking if an instance $M =^? N$ has a solution is in NP.

From the other hand the simplest proof of NP-hardness of the second order matching [1] works also in the affine case. Also de Groote's proof of NP-hardness of matching in the linear lambda calculus can be applied here.

Again, as in the proof of Theorem 1, if the $\beta\eta$-matching is considered, then, when guessing a solution we omit terms, which are not in $\bar{\eta}$-normal form, so the theorem is valid for β- as well as $\beta\eta$-lambda calculus.

References

1. Hubert Comon, Yan Jurski, Higher-Order Matching and Tree Automata, *Proc. 11th Int'l Workshop Computer Science Logic*, CSL'97, Mogens Nielsen, Wolfgang Thomas, *eds.*, Aarhus, Denmark, August 1997, LNCS **1414**, Springer-Verlag, 1998, 157–176.
2. Nachum Dershowitz, Zohar Manna, Proving termination with multiset orderings, *Comm. Assoc. for Computing Machinery*, **6** (22), 465–476, 1979.
3. Giles Dowek, Third order matching is decidable, *Proc. 7th IEEE Symp. Logic in Computer Science*, LICS'92, IEEE Press, 1992, 2–10, also in *Annals of Pure and Applied Logic*, **69**, 1994, 135–155.
4. Philippe de Groote, Linear Higher-Order Matching Is NP-Complete, *Proc. 11th Int'l Conf. Rewriting Techniques and Applications*, RTA 2000, Leo Bachmair, *ed.*, Norwich, UK, July 10–12, 2000, LNCS **1833**, Springer-Verlag, 2000, 127–140.
5. Jordi Levy, Linear Second Order Unification, *Proc. 7th Int'l Conf. Rewriting Techniques and Applications*, RTA'96, H. Ganzinger, *ed.*, New Brunswick, NJ, 1996, LNCS **1103**, Springer-Verlag, 1996, 332–346.
6. Ralph Loader, *Higher Order β Matching is Undecidable*, October 2001, manuscript.
7. Harry G. Mairson, A Simple Proof of a Theorem of Statman, *Theoretical Computer Science*, **103**, 1992, 213–226.
8. Vincent Padovani, Decidability of All Minimal Models, *Proc. 3rd Int'l Workshop Types for Proofs and Programs*, TYPES'95, Stefano Berardi, Mario Coppo, *eds.*, Torino, Italy, 1995, LNCS **1158**, Springer-Verlag, 1996, 201–215.
9. Vincent Padovani, On equivalence classes of interpolation equations, *Proc.Int'l Conf. Typed Lambda Calculi and Applications*, TLCA'95, M. Dezani-Ciancaglini, G. Plotkin, *eds.*, LNCS **902**, Springer-Verlag, 1995, 335–349.
10. Vincent Padovani, Decidability of fourth-order matching, *Mathematical Structures in Computer Science*, **3** (10), 2000, 361–372.
11. Manfred Schmidt-Schauß, Klaus U. Schulz, *Decidability of bounded higher order unification*, technical report Frank-report-15, Institut für Informatik, J. W. Goethe-Universität, Frankfurt am Main, 2001.
12. Aleksy Schubert, Linear interpolation for the higher order matching problem, *Proc. 7th Int'l Joint Conf. Theory and Practice of Software Development*, TAPSOFT'97, M. Bidoit, M. Dauchet, *eds.*, LNCS **1214**, Springer-Verlag, 1997.
13. Richard Statman, The Typed λ-Calculus is Not Elementary Recursive, *Theoretical Computer Science*, **15**, 1981, 73–81.
14. Sergei Vorobyov, The"Hardest" Natural Decidable Theory, *Proc. 12th Annual IEEE Symp. Logic in Computer Science*, LICS'97, IEEE Press, 1997, 294–305.