Higher-order Matching via Intersection Type Inhabitation in Bounded Dimension

Andrej Dudenhefner[1] and Aleksy Schubert[2]

[1] TU Dortmund University, Germany
andrej.dudenhefner@cs.tu-dortmund.de
[2] University of Warsaw, Poland
alx@mimuw.edu.pl

Abstract

Decidability of higher-order matching was established by Stirling via a sophisticated game-semantic argument. We propose an alternative specification of higher-order matching as an intersection type inhabitation problem. Concretely, we use the recently proposed system **R**, a syntax-directed intersection type system in which judgements derive vectors of types and binary relations control information flow between coordinates. Bounding the dimension of type vectors in system **R** derivations makes inhabitation in system **R** decidable. We outline how a solution to a higher-order matching instance corresponds to an inhabitant in system **R** of bounded dimension. The work is in progress; the central challenge is establishing a dimensional bound that depends only on the matching instance.

1 Introduction

Higher-order matching is the following decision problem: given two simply typed λ-terms, can the first term be instantiated so that it is $\beta\eta$-equivalent to the second term? Higher-order matching was formulated by Huet in the 1970s and constitutes a fundamental decision problem in the simply-typed λ-calculus, with applications in theorem proving, program transformation, and program synthesis.

Decidability at low orders was established incrementally: order 2 by Huet, order 3 by Dowek [3], and order 4 by Padovani [6]. Decidability of the general problem was settled by Stirling [8] using a game-semantic characterisation. Stirling's argument is technically demanding and not easily related to standard type-theoretic constructions.

We propose to pursue a different approach: encoding higher-order matching as an inhabitation problem in a decidable fragment of an intersection type system. The encoding aims to reduce higher-order matching to inhabitation in a dimensionally bounded system **R**, a recently introduced intersection type system [4]. This long abstract describes the encoding, identifies the central technical obstacle (a uniform dimensional bound), and discusses possible approaches towards its resolution.

2 System R

We briefly recall system **R** [4]. Intersection types A, B and finite sets of intersection types σ, τ are given by

$$A, B ::= a \mid \sigma \to A \qquad \sigma, \tau ::= \{A_1, \ldots, A_n\}, \quad n \geq 0,$$

where a ranges over type atoms. A vector $\bar{A} = (A_1, \ldots, A_n)$ has dimension $\dim(\bar{A}) = n$, and we write $\bar{A}_{[i]}$ for A_i. The relation $\in$ is extended pointwise to vectors, and denoted $\bar{\in}$. We set

$\bar{\sigma} \Rightarrow \bar{A} := (\sigma_1 \to A_1, \ldots, \sigma_n \to A_n)$. An *environment* Γ of dimension d assigns to each variable in its domain a vector $\bar{\sigma}$ with $\dim(\bar{\sigma}) = d$.

A binary relation $R \subseteq \{1, \ldots, n\} \times \{1, \ldots, m\}$ acts on a type vector $\bar{A}$ of dimension n by $R(\bar{A})_{[i]} = \{A_j \mid j\, R\, i\}$, analogously on vectors of sets, and pointwise on environments.

Definition 1 (System **R** with Dimension Bound k)**.**

$$\frac{\bar{A} \mathrel{\bar{\in}} \bar{\sigma} \qquad \dim(\Gamma) = \dim(\bar{A}) \leq k}{\Gamma, x : \bar{\sigma} \vdash_k x : \bar{A}}\ (\mathrm{Ax}) \qquad \frac{\Gamma, x : \bar{\sigma} \vdash_k t : \bar{A}}{\Gamma \vdash_k \lambda x.t : \bar{\sigma} \Rightarrow \bar{A}}\ (\Rightarrow\mathrm{I})$$

$$\frac{\Gamma \vdash_k t : R(\bar{A}) \Rightarrow \bar{B} \qquad \Delta \vdash_k u : \bar{A} \qquad R \subseteq \{1, \ldots, \dim(\Delta)\} \times \{1, \ldots, \dim(\Gamma)\}}{\Gamma \cup R(\Delta) \vdash_k t\, u : \bar{B}}\ (\Rightarrow\mathrm{E})$$

The dimension bound k limits the length of all type vectors occurring in a derivation. The relation R in rule ($\Rightarrow$E) accommodates intersection introduction in a syntax-directed manner: the argument is typed with a vector $\bar{A}$ whose components are combined via R. In particular, relations are the mechanism by which intersection introduction is managed within a syntax-directed, Curry-style presentation. The following example illustrates the action of different relations in one derivation.

Example 2 ([4, Example 13])**.**
For the following definitions

- $\Gamma_1 := \{x : (\{\{b, c\} \to a\})\}$
- $\Gamma_2 := \{y : (\{\{a\} \to b\}, \{\{a\} \to c\})\}$
- $\Gamma_3 := \{x : (\{\{b, c\} \to a\}), y : (\{\{a\} \to b, \{a\} \to c\}), z : (\{a\})\}$
- $R_1 := \{(1, 1), (2, 1)\}$ *with* $R_1((b, c)) = (\{b, c\})$
- $R_2 := \{(1, 1), (1, 2)\}$ *with* $R_2((a)) = (\{a\}, \{a\})$

we have the following derivation:

$$\frac{\Gamma_1 \vdash_2 x : R_1((b, c)) \Rightarrow (a) \qquad \Gamma_2 \cup R_2(\{z : (\{a\})\}) \vdash_2 y\, z : (b, c)}{\Gamma_3 \vdash_2 x\,(y\, z) : (a)}\ (\Rightarrow\mathrm{E})$$

for which the judgement $\Gamma_2 \cup R_2(\{z : (\{a\})\}) \vdash_2 y\, z : (b, c)$ *is derived in dimension* 2 *as follows:*

$$\frac{\Gamma_2 \vdash_2 y : R_2((a)) \Rightarrow (b, c) \qquad \{z : (\{a\})\} \vdash_2 z : (a)}{\Gamma_2 \cup R_2(\{z : (\{a\})\}) \vdash_2 y\, z : (b, c)}\ (\Rightarrow\mathrm{E})$$

In the above rule applications, the role of R_1 *is to combine types in coordinates into an intersection, and the role of* R_2 *is to copy a type to two coordinates.*

System **R** satisfies subject reduction with respect to β-reduction in each bounded dimension and corresponds to the standard intersection type system of Barendregt, Coppo, and Dezani-Ciancaglini [1] up to top-level subtyping [4]. The inhabitation problem

> Inh$_{\mathbf{R}}$: *Given* Γ, $\bar{A}$, *and* k, *does* $\Gamma \vdash_k t : \bar{A}$ *hold for some term* t*?*

is the central decision problem of system **R**.

Theorem 3 (Decidability of Inhabitation [4])**.** Inh$_{\boldsymbol{R}}$ *is decidable in 2-EXPTIME.*

The decision procedure relies on a subformula property: there exists a derivation in β-normal form in which all coordinates of all type vectors are subformulae of types appearing in the input Γ or $\bar{A}$ (or the empty set). Combined with the dimensional bound k, this yields polynomially many distinct vectors per coordinate, which an alternating procedure traverses in exponential space.

3 Higher-order Matching via Inhabitation

A higher-order matching instance can be presented as a system of equations of shape either

$$X\, u_i^1\, \dots\, u_i^n =_{\beta\eta} r_i \qquad (i \in \{1, \dots, m\}) \tag{1}$$

or

$$X\, u_i^1\, \dots\, u_i^n \neq_{\beta\eta} r_i \qquad (i \in \{1, \dots, m\}) \tag{2}$$

where the metavariable X has simple type $\varphi_1 \to \cdots \to \varphi_n \to \varphi$, each u_i^j has simple type φ_j, each r_i is a simply-typed $\beta\eta$-normal form of type φ, and free variables are taken from a fixed simply-typed context Φ [5]. A *solution* is a λ-term $\lambda x_1 \dots x_n.t$ of type $\varphi_1 \to \cdots \to \varphi_n \to \varphi$ such that, after substituting it for the metavariable, every equation and disequation holds.

The encoding combines two ingredients. First, *characteristic typings* capture $\beta\eta$-equality at type φ via intersection-type derivability. Second, the collection of these typings into a single vector judgement then *combines* into one inhabitation problem.

3.1 Characteristic Typings

For each β-normal η-long term r with $\Phi \vdash_{\mathrm{STLC}} r : \varphi$, we associate a refining environment Γ and intersection type A, written $\Gamma \prec \Phi$ and $A \prec \varphi$, such that derivability of a typing for $\Gamma \vdash t : A$ uniquely identifies r up to β-equivalence among η-long candidates.

The construction lifts to vectors: a vector $\bar{A}$ characterises a finite set of right-hand sides $r_1, \dots, r_m$ by stacking, with $\bar{A}_{[i]} = A_i$ and the corresponding characteristic environments stacked coordinate-wise into Γ with $\dim(\Gamma) = m$.

Lemma 4 (Equality Characterization)**.** *For each β-normal η-long r with $\Phi \vdash_{\mathrm{STLC}} r : \varphi$, there exist $\Gamma \prec \Phi$ and $A \prec \varphi$ such that:*

1. *$\Gamma \vdash r : A$ in system $\boldsymbol{R}$*
2. *for any η-long t with $\Phi \vdash_{\mathrm{STLC}} t : \varphi$, if $\Gamma \vdash t : A$ then $t \twoheadrightarrow_\beta r$.*

A complementary disequality characterisation is also possible by exhibiting, for each alternative head shape, an environment ruling it out.

Lemma 5 (Disequality Characterization)**.** *For each β-normal η-long r with $\Phi \vdash_{\mathrm{STLC}} r : \varphi$, there exist $\Gamma \prec \Phi$ and $A \prec \varphi$ such that for any η-long t with $\Phi \vdash_{\mathrm{STLC}} t : \varphi$ and $\Gamma \vdash t : A$ we have $t \not\twoheadrightarrow_\beta r$.*

3.2 Reduction to Inhabitation

For simplicity, let us focus only on $\beta\eta$-equations, since inequations can be dealt with in an analogous manner.

Suppose $s = \lambda x_1 \ldots x_n.t$ is a solution to (1). Then $s\, u_i^1 \, \ldots \, u_i^n \twoheadrightarrow_\beta r_i$ for every i. By Lemma 4 together with subject reduction [4], this is equivalent to the existence of some dimension bound k such that

$$\Gamma_{[i]} \vdash_k (\lambda x_1 \ldots x_n.t)\, u_i^1 \, \ldots \, u_i^n : \bar{A}_{[i]} \qquad (i \in \{1, \ldots, m\}). \tag{3}$$

By inversion of rule ($\Rightarrow$E), there are some relations R_i^j and vectors $\bar{B}_i^j$ with

$$\Gamma_{[i]} \vdash_k \lambda x_1 \ldots x_n.t : R_i^1(\bar{B}_i^1) \Rightarrow \cdots \Rightarrow R_i^n(\bar{B}_i^n) \Rightarrow \bar{A}_{[i]}, \qquad \Gamma_{[i]} \vdash_k u_i^j : \bar{B}_i^j.$$

Stacking the m derivations into vectors of dimension at most mk and combining the relations yields, for some $R^1, \ldots, R^n$ and $\bar{B}^1, \ldots, \bar{B}^n$:

$$\Gamma, x_1 : \bar{B}^1, \ldots, x_n : \bar{B}^n \vdash_{mk} t : R^1(\bar{B}^1) \Rightarrow \cdots \Rightarrow R^n(\bar{B}^n) \Rightarrow \bar{A}.$$

The term t is an inhabitant in system $\mathbf{R}$ of dimension mk, with environment and target type computable from $r_1, \ldots, r_m$ and the arguments u_i^j.

Due to the characterisation theorem, the higher-order matching instance (1) has a solution iff the type $R^1(\bar{B}^1) \Rightarrow \cdots \Rightarrow R^n(\bar{B}^n) \Rightarrow \bar{A}$ is inhabited in the environment $\Gamma, x_1 : \bar{B}^1, \ldots, x_n : \bar{B}^n$ for some dimensional bound.

3.3 The Dimension Bound

The remaining ingredient is to compute a sufficient dimensional bound based on the given matching instance. Without such a bound, the inhabitation problem is undecidable.

For fourth-order matching, an explicit bound is plausible, and the following three avenues appear promising.

Bounded subject expansion. Starting from the $\beta\eta$-normal r_i and expanding to a candidate solution $(\lambda x_1 \ldots x_n.t)\, u_i^1 \, \ldots \, u_i^n$, the dimension grows in a controlled fashion governed by the structure of the r_i and the arity profile of the metavariable. For order 4, the expansion shape is sufficiently regular that the dimension is bounded by the product of the number of right-hand sides and a quantity depending on the simple types of the arguments.

Padovani-style saturation. Padovani's decision procedure for fourth-order matching [6] bounds the number of necessary type assumptions at intermediate variables. This induces a bound on the number of applications of intersection introduction in derivations. Given a bounded number of sufficient subformulae in a type derivation, this may limit the required dimension.

Krivine-machine states. Binary relations in system $\mathbf{R}$ bear a resemblance to relational denotational semantics [2]. The states reachable in a Krivine-machine evaluation of a candidate solution provide a candidate combinatorial bound on the dimension of any associated derivation.

We conjecture that a similar analysis extends to fifth-order matching along the lines suggested by the type structure of fifth-order metavariables, where intersection introduction is required only at applications headed by a main variable.

A complementary line of research, orthogonal to the search for a dimensional bound, is to reinterpret Stirling's game-semantic argument [8] in the language of intersection type derivations. A faithful translation of game positions into system $\mathbf{R}$ judgements could provide an independent type-theoretic justification of Stirling's argument.

3.4 Stirling's Example

We present here an example that illustrates that using type information in system **R** we can substantiate a term transformation that is used in Stirling's construction. Stirling in his unpublished paper [7] illustrates his technique by means of the following example.

$$
\begin{aligned}
a_1 &= \lambda y_1^{o}.\, \lambda y_2^{o\to o\to o}.\, y_2\, y_1\, y_1 \\
a_2 &= \lambda y_3^{o\to(o\to o\to o)\to o}.\, y_3\, \Big(y_3\, a\, (\lambda w_1^{o}.\, \lambda w_2^{o}.\, w_1)\Big)\, (\lambda v_1^{o}.\, \lambda v_2^{o}.\, g\, v_2) \\
s &= \lambda x_1^{o\to(o\to o\to o)\to o}.\, \lambda x_2^{(o\to(o\to o\to o)\to o)\to o}. \\
&\qquad x_2\, \Big(\lambda z_1^{o}.\, \lambda z_2^{o\to o\to o}.\, x_1\, z_1\, \big(\lambda s_1^{o}.\, \lambda s_2^{o}.\, x_1\, b\, (\lambda u_1^{o}.\, \lambda u_2^{o}.\, z_2\, s_1\, s_2)\big)\Big)
\end{aligned}
$$

Here sa_1a_2 β-reduces to ga. The unfolding transformation defined by Stirling replaces the variable s_1 with term $x_1z_1(\lambda s_1s_2.s_1)$ so that resulting term s' is

$$
s' = \lambda x_1.\, \lambda x_2.\, x_2\, \Big(\lambda z_1.\, \lambda z_2.\, x_1\, z_1\, \big(\lambda s_1.\, \lambda s_2.\, x_1\, b\, (\lambda u_1.\, \lambda u_2.\, z_2\, \underline{(x_1 z_1(\lambda s_1 s_2.s_1))}\, s_2)\big)\Big)
$$

A typing of this term in system **R** along the lines of Lemma 4 can be succinctly described as follows.

$$
\begin{aligned}
a_1 = {}& \lambda y_1{:}(\{a\},\quad \emptyset,\quad \{a\},\quad \emptyset\quad). \\
& \lambda y_2{:}(\{\{a\} \to \emptyset \to a\},\quad \{\emptyset \to \emptyset \to a\},\quad \{\emptyset \to a \to ga\},\quad \{\emptyset \to \emptyset \to ga\}\quad). \\
& \quad y_2\, y_1\, y_1 : (a, a, ga, ga)
\end{aligned}
$$

$$
\begin{aligned}
a_2 = {}& \lambda y_3{:}(\{\{a\} \to \{\{a\} \to \emptyset \to a\} \to a\},\quad \{a\} \to \{\emptyset \to \{a\} \to ga\} \to ga\}). \\
& \quad y_3\, \Big(y_3\, a\, (\lambda w_1{:}\{a\}.\, \lambda w_2{:}\emptyset.\, w_1)\Big)\, (\lambda v_1{:}\emptyset.\, \lambda v_2{:}\{a\}.\, g\, v_2) : (ga)
\end{aligned}
$$

$$
\begin{aligned}
s \; = {}& \lambda x_1{:}(\{\{a\} \to \{\{a\} \to \emptyset \to a\} \to a, \\
&\quad \emptyset \to \{\emptyset \to \emptyset \to a\} \to a, \\
&\quad \{a\} \to \{\emptyset \to \{a\} \to ga\} \to ga, \\
&\quad \emptyset \to \{\emptyset \to \emptyset \to ga\} \to ga\}). \\
& \lambda x_2{:}(\{\{\{a\} \to \{\{a\} \to \emptyset \to a\} \to a,\ \{a\} \to \{\emptyset \to \{a\} \to ga\} \to ga\} \to ga\}). \\
&\quad x_2\, \Big(\lambda z_1{:}(\{a\}, \{a\}).\, \lambda z_2{:}(\{\{a\} \to \emptyset \to a\}, \{\emptyset \to \{a\} \to ga\}). \\
&\qquad x_1\, z_1\, \big(\lambda s_1{:}(\{a\}, \emptyset).\, \lambda s_2{:}(\emptyset, \{a\}). \\
&\qquad\quad x_1\, b\, (\lambda u_1{:}(\emptyset, \emptyset).\, \lambda u_2{:}(\emptyset, \emptyset).\, z_2\, s_1\, s_2)\big)\Big) : (ga)
\end{aligned}
$$

We can see from the typing that x_1 can serve faithfully as a transformer of type $\{a\} \to \{\{a\} \to \emptyset \to a\} \to a$. At the same time z_1 in all contexts is of type $\{a\}$. Types of s_1, s_2 are in some typing contexts $\{a\}, \emptyset$ and in some other $\emptyset, \{a\}$ respectively. In the former typing contexts, $x_1z_1(\lambda s_1s_2.s_1)$ has type $\{a\}$. Note that when z_2 has the type that reflects that it uses its argument s_1 it must be the former context. Consequently, we can replace s_1 with $x_1z_1(\lambda s_1s_2.s_1)$ and the result of the reduction will not change. As it can be seen, the necessary description that makes it possible to apply the transformation is expressible in terms of typings in system **R**.

References

[1] Henk Barendregt, Mario Coppo, and Mariangiola Dezani-Ciancaglini. A filter lambda model and the completeness of type assignment. *J. Symb. Log.*, 48(4):931–940, 1983.

[2] Daniel de Carvalho. Execution time of λ-terms via denotational semantics and intersection types. *Math. Struct. Comput. Sci.*, 28(7):1169–1203, 2018.

[3] Gilles Dowek. Third order matching is decidable. *Annals of Pure and Applied Logic*, 69(2-3):135–155, 1994.

[4] Andrej Dudenhefner, Aleksy Schubert, and Jakob Rehof. A bounded parallel intersection type system. In *11th International Conference on Formal Structures for Computation and Deduction (FSCD 2026)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2026 (in publication).

[5] Vincent Padovani. On equivalence classes of interpolation equations. In Mariangiola Dezani-Ciancaglini and Gordon D. Plotkin, editors, *Typed Lambda Calculi and Applications, Second International Conference on Typed Lambda Calculi and Applications, TLCA '95, Edinburgh, UK, April 10-12, 1995, Proceedings*, Lecture Notes in Computer Science, pages 335–349. Springer, 1995.

[6] Vincent Padovani. Decidability of fourth-order matching. *Mathematical Structures in Computer Science*, 10(3):361–372, 2000.

[7] Colin Stirling. An introduction to decidability of higher-order matching. Downloaded from the webpage at University of Edinburgh.

[8] Colin Stirling. Decidability of higher-order matching. *Logical Methods in Computer Science*, 5, 2009.