

Hypergraphs and Degrees of Parallelism: a Completeness Result

Antonio Bucciarelli and Benjamin Leperchey

Preuves, Programmes, Systèmes,
CNRS - Université Paris 7
{buccia,leperche}@pps.jussieu.fr

Abstract. In order to study relative PCF-definability of boolean functions, we associate a hypergraph H_f to any boolean function f (following [3, 5]).

We introduce the notion of timed hypergraph morphism and show that it is:

- Sound: if there exists a timed morphism from H_f to H_g then f is PCF-definable relatively to g .
- Complete for subsequential functions: if f is PCF-definable relatively to g , and g is subsequential, then there exists a timed morphism from H_f to H_g .

We show that the problem of deciding the existence of a timed morphism between two given hypergraphs is NP-complete.

1 Introduction

PCF is a simple, paradigmatic functional programming language, defined by D. Scott in his seminal paper [11], a milestone in the area of denotational semantics.

Following Scott, Plotkin studied in [8] the relationship between operational and denotational semantics of PCF. The main results of [8] may be summarized as follows:

- The Scott model of PCF is adequate with respect to contextual equivalence.
- The model is not complete, due to the presence of non-definable, “parallel” functions.
- All the (algebraic) elements of the model become definable if a *parallel conditional statement* is added to the language.

Since then, a lot of work has been devoted to the search of a satisfactory semantic characterization of the notion of PCF-definable function (see [2] for a survey). We have now a number of different notions of sequentiality, and all of them characterize exactly PCF definability for first order functions.

In this paper, we study the relative definability problem for Finitary PCF (FPCF) with respect to its Scott model. FPCF is the finitary fragment of PCF: it has a single ground type \mathbf{B} , the corresponding constants \perp , \mathbf{tt} , \mathbf{ff} , and just one more constant, the *if – then – else*.

The Scott model of FPCF is the finite type hierarchy where $\llbracket \text{bool} \rrbracket$ is the flat domain of boolean values, and $\llbracket \sigma \rightarrow \tau \rrbracket$ is the set of monotonic functions from $\llbracket \sigma \rrbracket$ to $\llbracket \tau \rrbracket$, ordered pointwise. FPCF-terms are interpreted in the standard way in this model, and in particular, for every closed term $M : \sigma$, $\llbracket M \rrbracket \in \llbracket \sigma \rrbracket$.

An instance of the relative definability problem is a pair $f \in \llbracket \sigma \rrbracket$, $g \in \llbracket \tau \rrbracket$, and a solution is either a term $M : \tau \rightarrow \sigma$ such that $\llbracket M \rrbracket g = f$, or a proof that such a term does not exist (when M does exist, we say that f is *less parallel* than g , and we write $f \leq_{\text{par}} g$).

Conceptually, the relative definability problem for the finitary fragment of PCF is settled: we know that it is undecidable in general [7] and decidable for functions of order 1 or 2 [12].

Nevertheless, decidability results may be not completely satisfactory: from a theoretical point of view, we still lack a characterization of the poset of degrees of parallelism (i.e. equivalence classes of inter-definable functions, noted $[f]$) which, even in the decidable case, is rich and complex [3, 9].

In this paper, we give a complete, geometric characterization of relative definability for “subsequential”, first-order functions; the exact correspondence we establish between geometric objects (a particular kind of hypergraph morphisms) and computational ones (the terms solving relative definability problems), is, we believe, interesting in itself.

Moreover, our analysis of relative definability problems provide a simple way of choosing, among the terms solving a given instance, an “optimal” one (for instance, a term defining f with as few calls of g as possible).

1.1 Related Work

The study of degrees of parallelism was pioneered by Sazonov and Trakhtenbrot [10, 14] who singled out some finite subposets of degrees. Some results on degrees are corollaries of well known facts: for instance Plotkin’s full abstraction result for PCF+por implies that this poset has a top. The bottom of degrees is the set of PCF-definable functions which is fully characterized, for first order functions, by the notion of sequentiality (in any of its formulations). Moreover Sieber’s sequentiality relations [12] provide a characterization of first-order degrees of parallelism and this characterization is effective: given f and g one can decide if $f \leq_{\text{par}} g$. A. Stoughton [13] has implemented an algorithm which solves this decision problem. R. Loader has shown that the problem of deciding if a given continuous function(al) is PCF-definable, is undecidable [7]. As a consequence, the relation \leq_{par} is undecidable in general (at higher-order), since, if g is PCF-definable and f continuous, then f is PCF-definable if and only if $f \leq_{\text{par}} g$.

In [3], the first author investigates the poset of degrees of parallelism using categories of hypergraphs for representing boolean function. The starting point of the investigation was the observation that the trace of a function f (i.e. the subset of the graph of f whose first projection is the set of minimal points on which f is defined) can be turned into a hypergraph H_f , in such a way that hypergraph morphisms from H_f to H_g are “witnesses” of the inequality $f \leq_{\text{par}} g$. In particular, a rich subposet of degrees for which the hypergraph representation

is sound and complete is singled out in [3]. If $[f], [g]$ belong to that subposet $f \leq_{\text{par}} g$ holds if and only if there exists a morphism from H_f to H_g . In [5] P. Malacaria and the first author showed a general result about hypergraphs and degrees: if there exists a morphism from H_f to H_g , then $f \leq_{\text{par}} g$. However, for the notion of hypergraph morphism they used (the standard one, based on the preservation of hyperarcs), no general completeness result seems to hold.

1.2 Plan of the Paper

In this paper we introduce a weaker notion of hypergraph morphism (the timed morphisms) and we show that it is sound in general, and complete for subsequential functions (i.e. for functions which have a sequential upper bound). The proof of soundness presented in [5] goes through for the framework of timed morphisms with some very minor changes. The proof of completeness is an application of Sieber’s sequentiality relations.

In Section 2 we introduce the notions of hypergraphs representing boolean functions and of h -morphisms between them (h -morphisms were called “weak” in [5]; since timed morphisms are weaker, we change the terminology here). In Section 3 the “timed” hypergraph morphisms are defined, and we show by some examples how they behave as boolean function transformers. In Section 4, we recall some useful properties of subsequential functions. Sections 5, 6 and 7 are devoted to the proof of soundness and completeness of timed morphisms w.r.t. the relation \leq_{par} . Finally, section 8 sketches a few complexity considerations.

2 Hypergraphs and h -Morphisms

We denote by \mathbf{B} the flat domain of boolean values $\{\perp, \mathbf{tt}, \mathbf{ff}\}$. Tuples of boolean values are ordered in the product order. Given a monotone function $f : \mathbf{B}^n \rightarrow \mathbf{B}$, the *trace* of f is defined by

$$\text{tr}(f) = \{(v, b) \mid f(v) = b \neq \perp \text{ and } v \text{ minimal}\}$$

We note the first and second projection π_1 and π_2 . In particular, $\pi_1(\text{tr}(f))$ is the set of minimal points where f is defined.

A subset $A = \{v_1, \dots, v_k\}$ of \mathbf{B}^n is *linearly coherent* (or simply *coherent*) if for all $1 \leq i \leq n$ either $\exists 1 \leq j \leq k, v_j^i = \perp$, or $\forall 1 \leq j, j' \leq k, v_j^i = v_{j'}^i$. The set of coherent subsets of \mathbf{B}^n is denoted $\mathcal{C}(\mathbf{B}^n)$. The coherence is related to sequentiality: if f is a n -ary boolean function, and $\pi_1(\text{tr}(f))$ is coherent, then f has no sequentiality index and it is not PCF-definable. Actually f is definable if and only if no subset of $\pi_1(\text{tr}(f))$ is coherent.

The following easy property of the coherence will be useful:

Fact 1 *If $A \in \mathcal{C}(\mathbf{B}^n)$ and B is an Egli-Milner lower bound of A (that is if $\forall x \in A \exists y \in B, y \leq x$ and $\forall y \in B \exists x \in A, y \leq x$) then $B \in \mathcal{C}(\mathbf{B}^n)$.*

Definition 1. A colored hypergraph $H = (V_H, A_H, C_H)$ is given by:

- a finite set V_H of vertices,
- a set $A_H \subseteq \{A \subseteq V_H \mid \#A \geq 2\}$ of (hyper)arcs,
- a coloring function $C_H : V_H \rightarrow \{\mathbf{ff}, \mathbf{tt}\}$.

Definition 2. Let $f : \mathcal{B}^n \rightarrow \mathcal{B}$ be the n -ary function defined by $\mathbf{tr}(f) = \{(v_1, b_1), \dots, (v_k, b_k)\}$. The hypergraph H_f is defined by

- $V_{H_f} = \pi_1(\mathbf{tr}(f))$,
- A_{H_f} contains the coherent subsets of $\pi_1(\mathbf{tr}(f))$ with at least two elements,
- $C_{H_f}(v_i) = b_i$

One can check that the hypergraphs associated to monotone functions by the definition above (*functional hypergraph*) verify the following conditions:

H1 : If $\{x, y\} \in A_H$ then $C_H(x) = C_H(y)$.

H2 : If X_1, X_2 are hyperarcs and $X_1 \cap X_2 \neq \emptyset$ then $X_1 \cup X_2$ is a hyperarc.

Definition 3. A h -morphism from a hypergraph H to a hypergraph K is a function $m : V_H \rightarrow V_K$ such that:

- For all $A \subseteq V_H$, if $A \in A_H$ then $m(A) \in A_K$.
- for all $X \in A_H$, if $x, x' \in X$ and $C_H(x) \neq C_H(x')$ then $C_K(m(x)) \neq C_K(m(x'))$.

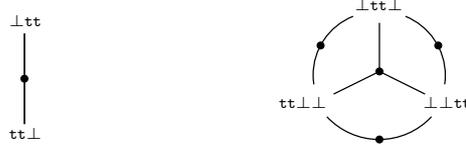
Colored hypergraphs and h -morphisms form a category, \mathcal{H} . In [5], it has been proved that, if there exists a h -morphism from H_f to H_g , then $f \leq_{\text{par}} g$. The problem of finding a weaker notion of hypergraph morphism, for which some sort of completeness result would hold, was left open.

We give here the motivating example for the definition of timed morphisms. Let $\text{por}_2 : \mathcal{B}^2 \rightarrow \mathcal{B}$ and $\text{por}_3 : \mathcal{B}^3 \rightarrow \mathcal{B}$ be defined by

$$\text{por}_2(x, y) = \begin{cases} \mathbf{tt} & \text{if one of } x, y \text{ is } \mathbf{tt} \\ \perp & \text{otherwise} \end{cases}$$

$$\text{por}_3(x, y, z) = \begin{cases} \mathbf{tt} & \text{if one of } x, y, z \text{ is } \mathbf{tt} \\ \perp & \text{otherwise} \end{cases}$$

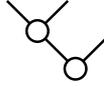
The associated hypergraphs are:



It is easy to see that there exists no h -morphism $m : H_3 \rightarrow H_2$. Nevertheless $\text{por}_3 \leq_{\text{par}} \text{por}_2$, since for instance $\text{por}_3 = \llbracket M \rrbracket \text{por}_2$ where

$$M = \lambda f \lambda x_1 x_2 x_3 \text{ if } f(f(x_1, x_2), x_3) \text{ then } \mathbf{tt} \text{ else } \perp$$

The tree of nested calls to f in M (the *nesting tree* of M), where the nodes are the occurrences of f , and the links are the arguments of f , is:

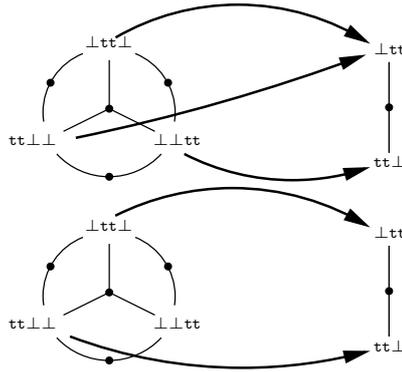


Actually, the nesting of calls to f in the term which defines por_3 with respect to por_2 is necessary. By looking at the way M “maps” the minimal points of por_3 onto the ones of por_2 , we realize that at the outermost level $(\text{tt}, \perp, \perp)$ and $(\perp, \text{tt}, \perp)$ are both mapped on (tt, \perp) , while $(\perp, \perp, \text{tt})$ is mapped on (\perp, tt) . The internal call of f maps $(\text{tt}, \perp, \perp)$ on (tt, \perp) and $(\perp, \text{tt}, \perp)$ on (\perp, tt) .

3 Timed Morphisms

The idea is the following: morphisms should be able to “collapse” a hyperarc on a singleton, provided that we have another morphism mapping this hyperarc on a hyperarc. More precisely, we want a finite sequence of morphisms $m_1 \dots m_l$ with domains $D_i \in A_H$, such that if m_i collapses an hyperarc B , there exists m_{i+k} with domain B . In the proof of soundness, each step in the sequence will appear as a nesting in the term.

For our example, the sequence corresponding to M is :



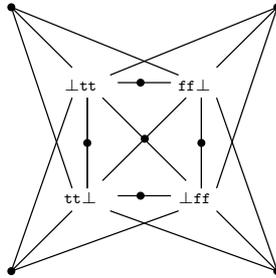
In general, by looking at the morphism from H_f to H_g , one can easily see the nesting of calls to the defining function g (and then build a term quite easily). First, we spot the vertices of H_g corresponding to each argument of g^1 : tt, \perp for the first argument, \perp, tt for the second. Then, we know how to organize the nested calls to g : if we collapse an hyperarc X on the vertex corresponding to the argument i , we put a call to g at argument i , which will be defined by the morphism with domain X .

It should be noticed here that, in the general case, one cannot associate vertices of H_g to arguments of g . Nevertheless, as shown in the proof of soundness, the existence of a timed morphism from H_f to H_g allows us to construct a term

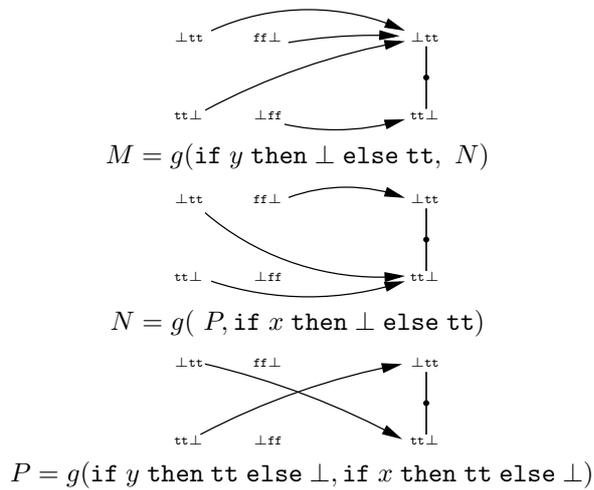
¹ In the general case, one cannot associate vertices to an argument. In our example por_2 , this is obvious. For more details, see the proof of soundness.

g -defining f , in general, even if the construction is more complicated than the one sketched above.

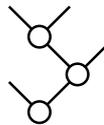
Another example: let $f(x, y)$ be \mathbf{tt} whenever x or y is defined, and \perp elsewhere. H_f is



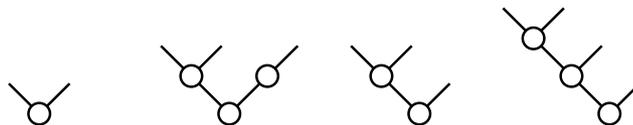
The only subsets that are not coherent are $\{\mathbf{tt}\perp, \mathbf{ff}\perp\}$ and $\{\perp\mathbf{tt}, \perp\mathbf{ff}\}$. In the following, we will not put the hyperarcs again. Here is a timed morphism from H_f to H_2 , and the corresponding term $\lambda g \lambda x \lambda y M$ defining f with \mathbf{por}_2 :



The corresponding tree is:



but one can also easily find morphisms (and terms) for these nesting trees (and for some others, too):



The leftmost nesting tree corresponds to the “natural” solution to this relative definability problem, namely:

$$\lambda g \lambda x \lambda y g(\text{if } x \text{ then tt else tt, if } y \text{ then tt else tt})$$

Timed morphisms are sequences. For a given problem, shorter sequences correspond to terms with smaller depth, w.r.t. the nesting of calls of g . Timed morphisms provide a handy tool for constructing these optimal solution.

Actually, we give a more abstract, equivalent definition of timed morphisms. We will argue that the two notions coincide after the following couple of definitions.

Definition 4. Let $H = (V_H, A_H, C_H)$ be a (functional) hypergraph.

- The timed image of H , \overline{H} is defined by: $V_{\overline{H}} = V_H$, $C_{\overline{H}} = C_H$ and $A_{\overline{H}} = A_H \cup \{\{v\} \mid v \in V_H\}$.
- Let $B \subseteq V_H$. $H|_B$ is the sub-hypergraph of H defined by:

$$\begin{cases} V_{H|_B} = B \\ A_{H|_B} = \{X \in A_H \mid X \subseteq B\} \\ C_{H|_B} = (C_H)|_B \end{cases}$$

Given two functional hypergraphs H, K , we say that a morphism $\alpha \in \mathcal{H}(H, \overline{K})$ is non-trivial if $\#\alpha(V_H) > 1$.

Definition 5. Let H, K be functional hypergraphs. A timed morphism $\alpha \in \mathcal{TH}(H, K)$ is a collection

$$\{\alpha_X \in \mathcal{H}(H|_X, \overline{K})\}_{X \in A_H}$$

where all the α_A 's are non-trivial, and non-redundant in the following sense:

$$\forall X \subseteq Y \in A_H \quad \alpha_Y|_X \text{ is non trivial} \Rightarrow \alpha_X = \alpha_Y|_X$$

The intuitive description of timed morphisms in terms of sequences, given in the examples of this section coincides with the definition above. Given a sequence $m = m^1, \dots, m^k$ of h -morphisms from H to \overline{K} , and a hyperarc $X \in A_H$, define $\alpha_X^m = m^j|_X$, where j is the smallest index such that $m^j|_X$ is non trivial. Conversely, given $\{\alpha_X\}_{X \in A_H}$ we have to construct a sequence of morphisms m^1, \dots, m^k from (restrictions of) H to \overline{K} , such that if m^i collapses an hyperarc B , there exists m^{i+k} non-trivial of domain B . Let $\{A_i\}_{i \in I}$ be the set of maximal elements of V_H (note that these are disjoint, H being functional); m^1 is obtained by “gluing” all the α_{A_i} , $i \in I$. Now, letting $\{A_i\}_{i \in J}$ $J = \{j^1, \dots, j^l\}$ be the set of maximal elements of V_H which are “collapsed” by m^1 , we define $m^2 = \alpha_{A_{j^1}}, \dots, m^{l+1} = \alpha_{A_{j^l}}$, and we proceed by considering the hyperarcs collapsed by m^2, \dots, m^{l+1} . By finiteness of H , iterating this construction we obtain a sequence m^1, \dots, m^k obeying the definition of timed morphism in terms of sequences.

Timed morphisms compose componentwise (i.e. $(\alpha \circ \beta)_A = \alpha_{\beta(A)} \circ \beta_A$). To any h -morphism $m : H \rightarrow K$ corresponds canonically the timed morphism defined by $\alpha_A = m|_A$.

4 Subsequential Functions

A monotone function $f : \mathcal{B}^n \rightarrow \mathcal{B}$ is *subsequential* if it is extensionally upper bounded by a sequential (i.e. PCF-definable) function. As shown in proposition 6 subsequential functions correspond to hypergraphs with monochromatic hyperarcs and to functions preserving linear coherence. Such a class of functions admits hence a natural characterization in order theoretic, graph theoretic and algebraic terms.

Proofs of the statements of this section can be found in [5].

Proposition 6. *Let $f : \mathcal{B}^n \rightarrow \mathcal{B}$ be a monotone function. The following are equivalent:*

1. f is subsequential.
2. For all $A \in \mathcal{C}(\mathcal{B}^n)$, $f(A) \in \mathcal{C}(\mathcal{B})$. (i.e. f preserves the linear coherence of \mathcal{B}^n .)
3. If $X \in A_{H_f}$ then for all $x, y \in X$ $C_{H_f}(x) = C_{H_f}(y)$ (i.e. X is monochromatic).

Given a set $A = \{v_1, \dots, v_k\} \subseteq \mathcal{B}^n$, there exist in general a number of functions whose minimal points are exactly the elements of A . For instance, if the v_i are pairwise unbounded, there exist 2^k such functions. The following lemma states that, among these functions, the subsequential ones are those whose degree of parallelism is minimal.

Lemma 7. *Let $f, g : \mathcal{B}^n \rightarrow \mathcal{B}$ be such that g is subsequential and $\pi_1(\text{tr}(f)) = \pi_1(\text{tr}(g))$. Then $g \leq_{\text{par}} f$.*

In section 5, we prove that if there exists a timed morphism $\alpha : H_f \rightarrow H_g$, then $f \leq_{\text{par}} g$. The following lemma introduces a key notion toward that result, namely that of *slice function*. The idea is the following: in order to reduce $f : \mathcal{B}^m \rightarrow \mathcal{B}$ to $g : \mathcal{B}^n \rightarrow \mathcal{B}$ we start by transforming the minimal points of f into the ones of g . This amounts to defining a function from \mathcal{B}^m to \mathcal{B}^n , that we describe as a set of functions $f_1, \dots, f_n : \mathcal{B}^m \rightarrow \mathcal{B}$. If these functions are g -definable, then we can already g -define a function which is defined (that is, not equal to \perp) if and only if f is defined, namely

$$h = \lambda \mathbf{x}. g(f_1 \mathbf{x}) \dots (f_n \mathbf{x})$$

and we are left with the problem of forcing h to agree with f whenever it converges.

For the time being we show that, if the f_i 's are defined via a timed morphism $\alpha : H_f \rightarrow H_g$, then they are subsequential, hence "relatively simple".

Lemma 8. *Let $f : \mathcal{B}^m \rightarrow \mathcal{B}$, $g : \mathcal{B}^n \rightarrow \mathcal{B}$ be monotone functions and $\alpha : H_f \rightarrow H_g$ be a timed morphism. For $B \in A_{H_f}$, $1 \leq i \leq n$ let $f_i^B : \mathcal{B}^m \rightarrow \mathcal{B}$ be the function defined by*

$$\text{tr}(f_i^B) = \{(v, \alpha_B(v)_i) \mid v \in B, \alpha_B(v)_i \neq \perp\}$$

Then f_i^B is subsequential. We will call f_i^B the i th-slice of α_B .

5 Soundness

Timed morphisms are sound with respect to \leq_{par} , in the sense expressed by the following theorem:

Theorem 9. *Let $f : \mathcal{B}^l \rightarrow \mathcal{B}$, $g : \mathcal{B}^m \rightarrow \mathcal{B}$ be monotone functions such that $\mathcal{TH}(H_f, H_g) \neq \emptyset$. Then $f \leq_{\text{par}} g$.*

The proof is essentially the same as in [5]. The key point lies in the restriction of morphisms to a hypergraph. In [5], the hypothesis was too strong: we only need a morphism from this hyperarc to H_g , we do not need it to be a part of the initial morphism from H_f to H_g . This generalization allows us to prove a completeness result.

6 Sequentiality Relations

Definition 10 (Sieber). *For each $n \geq 0$ and each pair of sets $A \subseteq B \subseteq \{1, \dots, n\}$ let $S_{A,B}^n \subseteq \mathbf{B}^n$ be defined by*

$$S_{A,B}^n(b_1, \dots, b_n) \Leftrightarrow (\exists i \in A \ b_i = \perp) \vee (\forall i, j \in B \ b_i = b_j)$$

An n -ary logical relation R is called a sequentiality relation if it is an intersection of relations of the form $S_{A,B}^n$.

We define $S_{n,n+1} = S_{\{1,\dots,n\},\{1,\dots,n+1\}}^{n+1}$.

We write

$$\begin{pmatrix} x_{11} & \dots & x_{1n} \\ \vdots & & \vdots \\ x_{m1} & \dots & x_{mn} \end{pmatrix} \in R$$

meaning that each row is in R . A function $f : \mathbf{B}^m \rightarrow \mathbf{B}$ is invariant under the logical relation R of arity n whenever the matrix $(x_{ij})_{1 \leq i \leq m, 1 \leq j \leq n}$ is in R :

$$(f(x_{11}, \dots, x_{m1}), \dots, f(x_{1n}, \dots, x_{mn})) \in R$$

Proposition 11. *For any $f : \mathbf{B}^n \rightarrow \mathbf{B}$ and $g : \mathbf{B}^m \rightarrow \mathbf{B}$ continuous functions, $f \leq_{\text{par}} g$ if and only if for any sequentiality relation R , if g is invariant under R then f is invariant too.*

Actually this is a relativized version of the main theorem of [12]: a continuous function of first or second order is PCF-definable if and only if it is invariant under all sequentiality relations.

Coherence is tightly related to sequentiality relations:

Lemma 12. *Let $A = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \subseteq \mathbf{B}^m$, and B be a subset of $\{1, \dots, n\}$. $\{\mathbf{x}_i\}_{i \in B}$ is coherent iff $(x_{ij}) \in S_{B,B}^n$. Moreover, A is coherent iff:*

$$\begin{pmatrix} x_{11} & \dots & x_{1n} & \bigwedge_{1 \leq i \leq n} x_{1i} \\ \vdots & & \vdots & \vdots \\ x_{m1} & \dots & x_{mn} & \bigwedge_{1 \leq i \leq n} x_{mi} \end{pmatrix} \in S_{n,n+1}$$

These sequentiality relations are closely related to strong stability at first order (see [4] for an overview on strong stability): f is strongly stable if it preserves linear coherence (that is, f is invariant by the relations $S_{B,B}^n$), and f is conditionally multiplicative: if A is coherent $f(\bigwedge A) = \bigwedge_{a \in A} f(a)$, (that is, f is invariant for the relation $S_{n,n+1}$).

7 Completeness

Theorem 13. *Let $f : \mathcal{B}^n \rightarrow \mathcal{B}$ and $g : \mathcal{B}^m \rightarrow \mathcal{B}$ be subsequential functions, such that $\mathcal{TH}(H_f, H_g) = \emptyset$. Then $f \not\leq_{\text{par}} g$.*

Proof. The first remark is that $\mathcal{TH}(H_f, H_g) = \emptyset$ if and only if there exists $A \in A_{H_f}$ such that there is no non-trivial morphism from $H_{f|_A}$ to $\overline{H_g}$. Throughout this proof, we restrict our attention to $H_{f|_A}$, for such an $A = \{\mathbf{v}_1, \dots, \mathbf{v}_k\}$. Let A_1, \dots, A_l be the arcs of $H_{f|_A}$, and, for $1 \leq j \leq l$, let B_j be the corresponding set of the indices: $A_j = \{\mathbf{v}_i\}_{i \in B_j}$.

We consider the $(k+1)$ -ary sequential logical relation

$$S_A = (\bigcap_{1 \leq j \leq l} S_{B_j, B_j}^{k+1}) \cap S_{k, k+1}$$

If we prove that g is invariant with respect to S_A and f is not, we are done. Let us start by proving that f is not invariant.

Let $V = (\mathbf{v}_1, \dots, \mathbf{v}_k, \bigwedge_{1 \leq j \leq k} \mathbf{v}_j)$: by lemma 12, for $1 \leq j \leq l$, $V \in S_{B_j, B_j}^{k+1}$ and $V \in S_{k, k+1}$, i.e. $V \in S_A$. On the contrary:

$$\left(f(\mathbf{v}_1), f(\mathbf{v}_2), \dots, f(\mathbf{v}_k), f\left(\bigwedge_{1 \leq j \leq k} \mathbf{v}_j\right) \right) \notin S_{k, k+1}$$

since the first k components of this vector are defined (the \mathbf{v}_j are in the trace of f), and the last is \perp ($\bigwedge \mathbf{v}_i$ can't be *above* a \mathbf{v}_j). Therefore, this tuple does not belongs to S_A .

It remains to show that that $g \in S_A$. Let us suppose by *reductio ad absurdum* that there exists a matrix $W = (\mathbf{w}_1, \dots, \mathbf{w}_{k+1}) \in \mathcal{B}^{m \times (k+1)}$ such that:

$$W \in S_A \text{ and } g(W) = (g(\mathbf{w}_1), \dots, g(\mathbf{w}_{k+1})) \notin S_A$$

First, we note that, since $W \in S_A$, for all $1 \leq j \leq l$, $W \in S_{B_j, B_j}^{k+1}$, that is $\{\mathbf{w}_i\}_{i \in B_j}$ is coherent, so $\{g(\mathbf{w}_i)\}_{i \in B_j}$ is coherent, and g is subsequential, which entails, by proposition 6 and lemma 12, that $g(W)$ is invariant by S_{B_j, B_j}^{k+1} . Therefore, $g(W) \notin S_A$ means that $g(W) \notin S_{k, k+1}$, that is, $\forall j \leq k, g(\mathbf{w}_j) \neq \perp$ and $\exists j, j' \leq k+1, g(\mathbf{w}_j) \neq g(\mathbf{w}_{j'})$. Since g is subsequential and $\{\mathbf{w}_1, \dots, \mathbf{w}_k\}$ is coherent (lemma 12), $\forall j, j' \leq k, g(\mathbf{w}_j) = g(\mathbf{w}_{j'})$: there exists $b \in \{\mathbf{tt}, \mathbf{ff}\}$ such that

$$\forall j \leq k, g(\mathbf{w}_j) = b \text{ and } g(\mathbf{w}_{k+1}) = \perp$$

Hence any w_j , for $1 \leq j \leq k$, has at least a lower bound in $\pi_1(\text{tr}(g))$, which we denote by z_j . We have:

- the set $\{z_1, \dots, z_k\}$ is not a singleton, otherwise $g(w_{k+1}) = b$, being $w_{k+1} \geq \bigwedge_{1 \leq j \leq k} w_j$, by definition of $S_{k,k+1}$.
- for all $1 \leq j \leq l$ the set $\{z_i\}_{i \in B_j}$ is coherent, being an Egli-Milner lower bound of the coherent set $\{w_i\}_{i \in B_j}$ (see fact 1).
- Last, by proposition 6, f being subsequential, C_{H_f} is constant on A .

Hence the function $\alpha : A \rightarrow H_g$ defined by $\alpha_A(v_i) = z_i$ is in $\mathcal{H}(H_f|_A, \overline{H_g})$, and it is not trivial, a contradiction.

Remark that, if g is subsequential and f is not, then $f \not\leq_{\text{par}} g$, hence the hypothesis of Theorem 13 could be weakened.

In order to see that completeness of timed morphisms fails in general, let us consider the following monotone functions:

$$\begin{cases} f(\perp, \mathbf{tt}, \mathbf{tt}, \mathbf{ff}) = \mathbf{tt} \\ f(\mathbf{ff}, \perp, \mathbf{tt}, \mathbf{tt}) = \mathbf{tt} \\ f(\mathbf{tt}, \mathbf{ff}, \perp, \mathbf{tt}) = \mathbf{tt} \\ f(\mathbf{tt}, \mathbf{tt}, \mathbf{ff}, \perp) = \mathbf{tt} \end{cases}$$

$$\begin{cases} g(\perp, \mathbf{tt}, \mathbf{ff}) = \mathbf{tt} \\ g(\mathbf{ff}, \perp, \mathbf{tt}) = \mathbf{tt} \\ g(\mathbf{tt}, \mathbf{ff}, \perp) = \mathbf{ff} \end{cases}$$

Since all subsets of H_f with at least three elements are hyperarcs, and H_g is composed by a single ternary hyperarc, it is easy to see that there is no non trivial h -morphism from the maximal hyperarc of H_f to $\overline{H_g}$, and hence no timed morphism from H_f to H_g . On the other hand $f \leq_{\text{par}} g$, since the degree of g (the “B-K function”) is the top of stable degrees ([6], p. 334), and f is stable.

8 On the Complexity of Weak and Timed Morphisms

Saturated hypergraphs are particularly simple functional hypergraphs, namely those whose sets of arcs are closed by union.

We will reduce the set-splitting problem to the problem of the existence of a h -morphism between two saturated hypergraphs; then we will reduce this latter problem to that of the existence of a timed morphism between two saturated hypergraphs.

Hence, we show that deciding the existence of both h -morphisms and timed morphism between saturated hypergraphs are NP-hard problems.

On the other hand, deciding the existence of h -morphisms and timed morphisms, in the general case, are NP problems, since checking that a given function preserves hyperarcs is a polynomial task, and the guess of such a function is also polynomial in the size of the hypergraphs.

Summing up, we show that deciding the existence of both h -morphisms and timed morphism are NP-complete problems.

Let us start by defining the set splitting problem (see for instance [1], problem number 37):

Instance: a set U and a family A_1, \dots, A_k of subsets of U .

Question: is there a (non-trivial) partition $U = U_1 \cup U_2$ such that $A_j \cap U_1 \neq \emptyset$ and $A_j \cap U_2 \neq \emptyset$, for all j ?

Given an instance U, A_1, \dots, A_k of the set splitting problem, we may suppose that for all i, j $A_i \not\subset A_j$ (otherwise we drop A_j , without altering the instance).

We define the saturated hypergraph H_1 , whose set of vertices is U and whose minimal arcs are A_1, \dots, A_k , and the hypergraph $\mathbf{2}$ whose set of vertices is a pair, say $\{0, 1\}$, and whose unique arc is $\{0, 1\}$.

Lemma 14. *The given instance of the set splitting problem has a solution if and only if there exists a h -morphism from H_1 to $\mathbf{2}$.*

Proof. If we have a h -morphism $f : H_1 \rightarrow \mathbf{2}$, we set $U_1 = f^{-1}(0)$ and $U_2 = f^{-1}(1)$. By construction, $U_1 \cup U_2 = U$, and for all i , $f(A_i) = \{0, 1\}$ because A_i is a hyperarc of H_1 , so $U_1 \cap A_i \neq \emptyset$ and $U_2 \cap A_i \neq \emptyset$.

Conversely, if U_1, U_2 is a solution, we define $f : x \mapsto j$ s.t. $x \in U_j$. f is a h -morphism from H_1 to $\mathbf{2}$: f is well-defined because $U_1 \cap U_2 = \emptyset$, and if A is a hyperarc of H_1 , then $A = A_i$ for some i and since $A_i \cap U_j \neq \emptyset$ ($j = 1, 2$), $f(A) = \{0, 1\}$ which is a hyperarc of $\mathbf{2}$.

We are left with the problem of reducing the existence of an h -morphism between a saturated hypergraph and $\mathbf{2}$ to the existence of some timed morphism.

Given a saturated hypergraph H and $* \notin V_H$, let H^* be the (saturated) hypergraph defined by $V_{H^*} = V_H \cup \{*\}$ and $A_{H^*} = \{A \cup \{*\} \mid A \in A_H\}$. Let $\mathbf{3}$ be the hypergraph with three vertices, say $\{0, 1, 2\}$ whose unique arc is the set of vertices itself.

Lemma 15. *Let H be a saturated hypergraph. There exists a h -morphism from H to $\mathbf{2}$ if and only if there exists a timed morphism from H^* to $\mathbf{3}$.*

Proof. Let H be a saturated hypergraph.

First, if f is an h -morphism from H to $\mathbf{2}$, define, for $B \in A_{H^*}$,

$$\alpha_B(x) = \begin{cases} f(x) & \text{if } x \neq * \\ 2 & \text{if } x = * \end{cases}$$

$\{\alpha_B\}$ is clearly a timed morphism from H^* to $\mathbf{3}$.

Now, for the converse, let $\{\alpha_B\}$ be a timed morphism from H^* to $\mathbf{3}$. We have to define a h -morphism from H to $\mathbf{2}$ which “splits” every arc of H .

The idea is the following: given two arcs $A \subset B$ of H^* , we know that either $\alpha_B(A) = \{\alpha_B(*)\}$ or $\alpha_B(A) = \{0, 1, 2\}$. In the latter case we can easily construct a (partial) morphism from H to $\mathbf{2}$ which splits $A \setminus \{*\}$, in the former, there must exist an arc $C \subset \alpha_B^{-1}\{*\}$ such that $\alpha_C(A) = \{0, 1, 2\}$. In any case any arc will eventually be split. Gluing together all the (partial) splitting morphisms obtained in this way gives us the result.

More formally, we define a decreasing sequence of arcs of H^* , as follows:

- $A_0 = \bigcup \{A \in A_{H^*}\}$
- $A_{n+1} = \bigcup \{A \in A_{H^*} \mid A \subseteq \alpha_{A_n}^{-1}\{*\}\}$

For all n , such that $A_n \neq \emptyset$, $A_{n+1} \subset A_n$, since α_{A_n} is non-trivial. Let l_0 be the smallest index such that $A_{l_0} = \emptyset$ (remark that $l_0 \leq |A_{H^*}|$, since at least one arc is split at each stage).

Let

$$\begin{aligned} i_n &= \alpha_{A_n}(*), \\ j_n &= (i_n + 1) \bmod 3 \\ k_n &= (i_n + 2) \bmod 3 \end{aligned}$$

We define two disjoint subsets of V_H :

$$U_1 = \bigcup_{s=0}^{l_0-1} \alpha_{A_s}^{-1}\{j_s\}, \quad U_2 = \bigcup_{s=0}^{l_0-1} \alpha_{A_s}^{-1}\{k_s\}$$

It is not hard to check that U_1, U_2 split all the arcs of H ; let us define:

$$f(x) = \begin{cases} 0 & \text{if } x \in U_1 \\ 1 & \text{otherwise} \end{cases}$$

We check that f is an h -morphism from H to $\mathbf{2}$, *i.e.* that given $A \in A_H$, $f(A) = \{0, 1\}$.

First, note that, for some n , $A \subseteq A_{n-1}$ and $A \not\subseteq A_n$, since A_n is a decreasing sequence. Moreover, $0 < n < l_0$, since A_0 is the union of all the arcs of H^* , and $A_{l_0} = \emptyset$.

This means that $\alpha_{A_{n-1}}(A \cup \{*\}) = \{0, 1, 2\}$, hence $A \cap U_1 \neq \emptyset$ and $A \cap U_2 \neq \emptyset$. Finally, $f(a) = \{0, 1\}$ and we are done.

9 Conclusion

For a wide class of boolean functions (the subsequential ones) we are able to solve relative definability problems in a geometric way, using a suitable representation of functions as hypergraphs and PCF-terms as hypergraphs morphisms.

We can also list all the (sensible) terms solving a given problem f, g , by enumerating the timed morphisms from H_f to H_g , and choose, for instance, the one which uses as few calls of g as possible (but other notions of optimality could be considered).

A natural question is whether this approach can be extended to non subsequential boolean functions and/or to higher-order functions. We do not know at present, but probably a combination of more complex representations of functions as hypergraphs and of more involved notions of morphisms is required.

In section 8 we have seen that the problem of relative definability is intractable in general; on the other hand, the algorithm of [13] can solve first order definability problems in reasonable time in some cases; it could be worth to compare the performances of the two approaches.

References

1. An annotated list of selected NP-complete problems.
http://www.csc.liv.ac.uk/~ped/teachadmin/COMP202/annotated_np.html.
2. R. M. Amadio and P.-L. Curien. *Domains and Lambda-Calculi*, volume 46 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, Cambridge, 1998.
3. A. Bucciarelli. Degrees of parallelism in the continuous type hierarchy. *Theoretical Computer Science*, 177(1):59–71, Apr. 1997.
4. A. Bucciarelli and T. Ehrhard. Sequentiality in an extensional framework. *Information and Computation*, 110(2):265–296, 1 May 1994.
5. A. Bucciarelli and P. Malacaria. Relative definability of boolean functions via hypergraphs. *Theoretical Computer Science*, 278:91–110, 2002.
6. P.-L. Curien. *Categorical Combinators, Sequential Algorithms and Functional Programming*. Birkhäuser, revised edition, 1993.
7. R. Loader. Finitary PCF is not decidable. *Theoretical Computer Science*, 266(1–2):341–364, Sept. 2001.
8. G. D. Plotkin. LCF considered as a programming language. *Theoretical Computer Science*, 5(3):223–255, Dec. 1977.
9. Pucella and Panangaden. On the expressive power of first-order boolean functions in PCF. *TCS: Theoretical Computer Science*, 266, 2001.
10. V. Y. Sazonov. Degrees of parallelism in computations. In *Proc. Symp. on Mathematical Foundations of Computer Science*, number 45 in *Lecture Notes in Computer Science*, pages 517–523, 1976.
11. D. S. Scott. A type-theoretical alternative to ISWIM, CUCH, OWHY. *Theoretical Computer Science*, 121(1–2):411–440, 6 Dec. 1993.
12. K. Sieber. Reasoning about sequential functions via logical relations. In M. P. Fourman, P. T. Johnstone, and A. M. Pitts, editors, *Proc. LMS Symposium on Applications of Categories in Computer Science, Durham 1991*, LMS Lecture Note Series 177, pages 258–269. Cambridge University Press, 1992.
13. A. Stoughton. Mechanizing logical relations. *Lecture Notes in Computer Science*, 802:359–377, 1994.
14. M. B. Trackhtenbrot. On representation of sequential and parallel functions. In *Proc. Fourth Symp. on Mathematical Foundations of Computer Science*, number 32 in *Lecture Notes in Computer Science*, 1975.