Degrees of Parallelism

Mastersthesis
(English version)
of
Bernd Lichtenthäler

January 1996

INFORMATIK-BERICHT Nr. 96-01
Fachgruppe Informatik
D - 57068 Siegen

Contents

1 Introduction

This paper deals with various extensions of the class of sequentially computable Scott-continuous functions with non-sequential ones. It is based on the article [Plo75], where Plotkin has introduced the theoretical functional programming language PCF (Programming language for Computable Functions). This language consists of a special simply typed λ-calculus, including the elementary arithmetic. Plotkin studied the relations between denotational and operational semantics. Each time it turned out that denotationally equivalent PCF-terms, i.e. terms with the same denotation in the considered mathematical model, are also operationally equivalent, i.e. they are evaluated with the same result. However, the inversion fails. The evaluation of PCF-terms has the following property: At each step there is only one subterm to be transformed, i.e. the evaluation goes sequentially. If it exists a step where more than one subterm has to be evaluated simultaneously to get the correct result, the term is called "parallel". By adding the parallel conditional *pcond* to PCF Plotkin achieved the coincidence of operational semantics and the denotational semantics w.r.t. Scott's model of the complete partial orders (cpo-s) and continuous functions. With extending the calculus moreover with the parallel term $\exists$, whose semantics is the best continuous approximation to the existential quantifier, exactly all computable functions can be described. A computable element in Scott's model is one that can be approximated by so-called compact elements, which correspond to basic informations, as exactly as you like in an effective way. There is a difference in "parallelism" of the two terms. Whereas $\exists$ is able to consider an infinite amount of information, *pcond* merely can handle a finite amount. On the other hand it is not possible to describe *pcond* by a PCF-term that makes only use of $\exists$, neither.

In 1976 Sazonov already found different degrees of non-sequentiality (cp. [Saz76]). The used measure for parallelism of a continuous function is defined analogous to the Turing-degree of a partial function in common theory of computability: A function f is called sequentially reducible on a function g, if f can be described by a PCF-term making use of an additional constant whose semantics equals g. Two functions are said to be of equal parallelism, if PCF is able to describe the same set of functions with either. The equivalence classes we get in this way are called degrees of parallelism. These degrees form an upper semilattice, whose structure is not exactly known yet. This paper is intended to give a survey on former results referring to computable functions, to represent methods for analysing the semilattice and to insert further functions in this structure. The main tool for separating the degrees are the sequentiality relationsof Sieber, a special sort of logical relations, the only containing all semantics of PCF-terms. When considering functions of restricted order in a finite model, an algorithm of Stoughton (cp. [Sto94]), determining whether a function f is reducible on a function g or not, is very helpful. The upper subsemilattice represented in [Saz76] is mostly proved in this paper, extended e.g. by the parallel convergence tester, the Gustave-function and the Berry-Plotkin-function, whose degree is greatest among all stable functions. Functions being element of all sequentiality relations are called logically sequential. If the order of such a function is less than two in a finite model, there is a PCF-term with the same semantics. However, in general functions with different degree of parallelism can be element of the same sequentiality relations. In section 10, using the results in [Buc95], ascending and descending sequences of degrees and sequences, where each pair of degrees is incomparable, are constructed.

At last I have to point at the usage and the order of binding concerning the symbols for the logical operators. "$\exists$" is reserved for the already mentioned parallel existential quantifier function. Therefore the logical quantors "$\exists$" and "$\forall$" are exchanged for "$ex.$", resp. "$f.a.$". To avoid too many parentheses we keep to the usual convention that "$\vee$" and "$\wedge$" bind more tightly than "$\Rightarrow$" and "$\Leftrightarrow$" and the latter more tightly than the quantifiers.

2 The Programming Language PCF

The simply typed λ-calculus is a system based on the principles of abstraction and application. The type of a term is decisive for the way the term may be used in the calculus. A very important rule is the β-contraction. It means that applying a function to an argument is equivalent to exchanging the formal parameter in the function body for the argument term.

Definition 2.1 *Let Σ_0 be a finite set of ground types and $\Sigma ::= \Sigma_0 \mid (\Sigma{\rightarrow}\Sigma)$ the set of all types. Further let $C := \dot{\bigcup}_{\sigma\in\Sigma} C_\sigma$ be a countable set of typed constants. The set $\Lambda(C)$ of the* simply typed λ-terms *for C and Σ is defined by:*

1. *For all $\sigma \in \Sigma$ and $c \in C_\sigma$ it is $c \in \Lambda(C)_\sigma$.*
2. *For all $\sigma \in \Sigma$ set $V_\sigma := \{x_i^\sigma \mid i \in \mathbf{N}\} \subseteq \Lambda(C)_\sigma$ (variables of type σ) and $V := \dot{\bigcup}_{\sigma\in\Sigma} V_\sigma$.*
3. *For all $\sigma, \tau \in \Sigma$ and $M \in \Lambda(C)_{\sigma\rightarrow\tau}, N \in \Lambda(C)_\sigma$ it is $(MN) \in \Lambda(C)_\tau$ (application).*
4. *For all $\sigma, \tau \in \Sigma$, $M \in \Lambda(C)_\tau$ and $i \in \mathbf{N}$ it is $\lambda x_i^\sigma.M \in \Lambda(C)_{\sigma\rightarrow\tau}$ (abstraction).*
5. *These terms only.*

The set $\Gamma(C)$ of the simply typed λ-free terms *for C and Σ is defined by the same construction rules, but without rule 4.*

To simplificate the terms we use the following notations:

- LMN stands for $((LM)N)$, i.e. application associates to the left.
- Abstraction and the operator "$\rightarrow$" associate to the right: $\lambda x_1^{\sigma_1}, x_2^{\sigma_2}, \ldots, x_n^{\sigma_n}.M$ or even more abridged $\lambda\vec{x}^{\vec{\sigma}}$ stands for $\lambda x_1^{\sigma_1}.\lambda x_2^{\sigma_2} \ldots \lambda x_n^{\sigma_n}.M$ and $(\sigma_1, \ldots, \sigma_n)$ stands for $\sigma_1{\rightarrow}(\sigma_2{\rightarrow}(\ldots{\rightarrow}(\sigma_{n-1}{\rightarrow}\sigma_n)\ldots))$. Define σ^n for $n \in \mathbf{N}$ and $\sigma \in \Sigma$ by $\sigma^0 := \sigma$ and $\sigma^{n+1} := \sigma{\rightarrow}\sigma^n$.
- Instead of $LM_1 \ldots M_n$ we mostly use the uncurry-notation $L(M_1, \ldots, M_n)$.
- Greek indices in the right upper corner stand for types. They are omitted, if the type of the term is evident or unmistakable.

Obviously, each type can be written in the form $\sigma = (\sigma_1, \ldots, \sigma_n, \sigma_{n+1})$ such that $\sigma_{n+1} \in \Sigma_0$.

Definition 2.2 1. *The set $FV(M)$ of the* free occuring variables *in a term M is defined as follows:*

(a) *For all $x_i^\sigma \in V$ set $FV(x_i^\sigma) := \{x_i^\sigma\}$.*

(b) *For all $c \in C$ set $FV(c) := \emptyset$.*

(c) *For all $M^{\sigma\to\tau}$ and $N^\sigma \in \Lambda(C)$ set $FV(MN) := FV(M) \cup FV(N)$.*

(d) *For all $x_i^\sigma \in V$ and $M^\tau \in \Lambda(C)$ set $FV(\lambda x_i^\sigma.M) := FV(M) \setminus \{x_i^\sigma\}$.*

2. *A term M is called* closed, *if $FV(M) = \emptyset$.*

To evaluate terms correctly a replacing of variables by terms in a consistent way is needful:

Definition 2.3 *Let M, N be terms and x a variable of the same type as M. The* substitution *of x in N by M (abbr. $[M/x](N)$) is defined by:*

1. *For all $x \in V$ set $[M/x](x) := M$.*

2. *For all $y \in V$ set $[M/x](y) := y$ whenever $y \neq x$.*

3. *For all $c \in C$ set $[M/x](c) := c$.*

4. *For all $N_1, N_2 \in \Lambda(C)$ set $[M/x](N_1N_2) := ([M/x](N_1))([M/x](N_2))$.*

5. *For all $N \in \Lambda(C)$ and $y \in V$ set*

$$[M/x](\lambda y.N) := \begin{cases} \lambda x.N & , x = y \\ \lambda y.[M/x](N) & , x \neq y \ \wedge \ (y \notin FV(M) \ \vee \ x \notin FV(N)) \\ \lambda z.[M/x][z/y](N) & , x \neq y \ \wedge \ y \in FV(N) \ \wedge \ x \in FV(N) \\ & \quad \wedge\, z \notin FV(MN). \end{cases}$$

Notice that every free occurence of a variable being not substituted remains free after any substitution.

Terms that are intended to have the same semantics are regarded as syntactically equivalent, i.e. they will be identified from now on:

Definition 2.4 *The equivalence relation $\equiv_\Lambda$ on $\Lambda(C)$ is defined as follows:*

1. *For all $M \in \Lambda(C)$ it is $M \equiv_\Lambda M$.*

2. *For all $M, N \in \Lambda(c)$ we have $M \equiv_\Lambda N \Rightarrow N \equiv_\Lambda M$.*

3. *For all $M, M, P \in \Lambda(C)$ we have $M \equiv_\Lambda N \wedge N \equiv_\Lambda P \Rightarrow M \equiv_\Lambda P$.*

4. *For all* $M_1, M_2, N \in \Lambda(C)$ *we have* $\dfrac{M_1 \equiv_\Lambda M_2}{NM_1 \equiv_\Lambda NM_2}$ *and* $\dfrac{M_1 \equiv_\Lambda M_2}{M_1N \equiv_\Lambda M_2N}$.

5. *For all* $M_1, M_2 \in \Lambda(C)$ *and* $x \in V$ *we have* $\dfrac{M_1 \equiv_\Lambda M_2}{\lambda x.M_1 \equiv_\Lambda \lambda x.M_2}$.

6. *For all* $M \in \Lambda(C)$ *and* $x, y \in V_\sigma$ *it is* $\lambda x.M \equiv_\Lambda \lambda y.[y/x](M)$ *whenever* $y \notin FV(M)$ *(α-conversion).*

7. *For all* $M \in \Lambda(C)$, $N \in \Lambda(C)_\sigma$ *and* $x \in V_\sigma$ *it is* $(\lambda x.M)N \equiv_\Lambda [N/x](M)$ *(β-contraction).*

8. *For all* $M \in \Lambda(C)$ *and* $x \in V$ *it is* $\lambda x.Mx \equiv_\Lambda M$ *whenever* $x \notin FV(M)$ *(η-contraction).*

The functional programming language PCF consists of special λ-terms. For the sake of simplicity we use only one ground type, representing the natural numbers. The type *boolean* can be embedded in the type *integer* with interpreting zero as "true" and all other numbers as "false".

Definition 2.5 *For* $\Sigma_0 := \{\iota\}$, $C_\iota^{PCF} := \{\underline{n} \mid n \in \mathbf{N}\}$, $C_{\iota^1}^{PCF} := \{\dot{+}\underline{1}, \dot{-}\underline{1}\}$, $C_{\iota^3}^{PCF} := \{cond\}$ *and* $C_{\sigma^1 \to \sigma}^{PCF} := \{Y_\sigma\}$ *for all* $\sigma \in \Sigma$ *a λ-term in* $\Lambda(C^{PCF})$ *is called* PCF-term*. Closed PCF-terms of the ground type* ι *are called* programs*.*

In this paper the category $\underline{CPO}$ of the complete partial orders and continuous functions is considered as model for the typed λ-calculus. Before specifying the denotational semantics of PCF we introduce the required expressions and notations, mostly adopted from [Wei80]. The missing proofs can be found in this work. We begin with

Definition 2.6 *A* partial order $(D, \sqsubseteq_D)$ *consists of a set* D *and a relation* $\sqsubseteq_D \subseteq D \times D$ *meeting the following properties:*

- *For all* $x \in D$ *it is* $x \sqsubseteq_D x$ *(reflexivity).*
- *For all* $x, y \in D$ *we have* $\sqsubseteq_D y \wedge y \sqsubseteq_D x \Rightarrow x = y$ *(antisymmetry).*
- *For all* $x, y, z \in D$ *we have* $x \sqsubseteq_D y \wedge y \sqsubseteq_D z \Rightarrow x \sqsubseteq_D z$ *(transitivity).*

If the relation is clear, D stands for $(D, \sqsubseteq_D)$ and the index is omitted. $x \sqsubseteq y$ means "x approximates y" or "x contains less information than y".

Definition 2.7 *Let D be a partial order. A subset $\emptyset \neq M \subseteq D$ is said to be* directed*, if for all $x, y \in M$ there is an element $z \in M$ such that $x \sqsubseteq z$ and $y \sqsubseteq z$. A set $M \subseteq D$ is said to be* bounded *(abbr. $\uparrow M$), if there is an* upper bound *$z \in D$, i.e. for all $x \in M$ it is $x \sqsubseteq z$.*

Instead of $\uparrow\{x, y\}$ we write $x \uparrow y$. If every directed set of informations is surpassed by an "upper information", we get

Definition 2.8 *A triple $(D, \sqsubseteq, \bot_D)$ is called* cpo*, if $(D, \sqsubseteq)$ is a partial order such that every directed subset M has a least upper bound $\bigsqcup M$ and for all $x \in D$ it is $\bot_D \sqsubseteq x$.*

The index of $\bot$ will be dropped, if the cpo is clear. Instead of $\bigsqcup\{x, y\}$ we will write $x \sqcup y$. The domains for the ground types will be flat cpo-s: For any set X let $X_\bot := (X \,\dot\cup\, \{\bot_X\}, \bot_X)$ the assigned flat cpo, whereat $x \sqsubseteq y :\Leftrightarrow x = y \;\vee\; x = \bot_X$ for all $x, y \in X \,\dot\cup\, \{\bot_X\}$, e.g. the natural numbers $\mathbf{N}_\bot$ and the true values $\mathbf{B}_\bot$.

If "directed" is exchanged for "bounded" in the last definition, we get

Definition 2.9 *A partial order is said to be* consistently complete*, if every bounded subset has a least upper bound (lub).*

The morphisms in the category <u>*CPO*</u> are the continuous functions:

Definition 2.10 *Let D and E be cpo-s.*

1. *A function $f : D \to E$ is called* monotone*, if we have $x \sqsubseteq_D y \Rightarrow f(x) \sqsubseteq_E f(y)$ for all $x, y \in D$.*
2. *A monotone function $f : D \to E$ is called* continuous*, if $f(\bigsqcup M) = \bigsqcup f(M)$ for all directed $M \subseteq D$. Let $[D \to E] := \{f : D \to E \mid f \text{ continuous}\}$.*

Continuous functions are closed under composition, and every identity map on a cpo is continuous. Therefore, we have

Lemma 2.11 *Cpo-s and continuous functions form a category (<u>CPO</u>).*

Further closure properties of cpo-s we get by

Theorem 2.12 *If D and E are cpo-s, then so are the following two constructions:*

1. *the product-cpo $(D \times E, \sqsubseteq_{D\times E}, \bot_{D\times E})$ with the coordinatewise order $(d, e) \sqsubseteq_{D\times E} (d', e') :\Leftrightarrow d \sqsubseteq_D d' \;\wedge\; e \sqsubseteq_E e'$ and $\bot_{D\times E} := (\bot_D, \bot_E)$ and*

2. *the cpo of the continuous functions* $([D \to E], \sqsubseteq_{[D\to E]}, \bot_{[D\to E]})$ *with the pointwise order* $f \sqsubseteq_{[D\to E]} g :\Leftrightarrow$ *f.a.* $d \in D: \ f(d) \sqsubseteq_E g(d)$ *and* $\bot_{[D\to E]}(d) := \bot_E$ *for all* $d \in D$.

The projections $\pi_1 : D \times E \to D,\ (d,e) \mapsto d$ *and* $\pi_2 : D \times E \to E,\ (d,e) \mapsto e$ *are continuous.*

The absence of product-types in the chosen λ-calculus is evident because of

Lemma 2.13 *Let* D, E, F *be cpo-s and*

$$curry(f)(d)(e) := f(d,e), uncurry(g)(d,e) := g(d)(e)$$

for all $d \in D,\ e \in E,\ f \in [D \times E \to F]$ *and* $g \in [D \to [E \to F]]$. *Then we have*

1. $curry \in [[D \times E \to F] \to [D \to [E \to F]]]$
2. $uncurry \in [[D \to [E \to F]] \to [D \times E \to F]]$
3. $curry^{-1} = uncurry$.

The following lemma is fundamental for the semantics of programming languages. It explains the meaning of infinite objects. For the proof see e.g. [Gun92].

Lemma 2.14 *Let* D *be a cpo and* $f : D \to D$ *a continuous function. Then* f *has a least fixed point* $d_0 \in D$, *that is* $f(d_0) = d_0$ *and* $d_0 \sqsubseteq d$ *for all* $d \in D$ *with* $f(d) = d$. *Furthermore it is* $d_0 = \bigsqcup_{i \in \mathbf{N}} f^i(\bot_D)$.

After this insertion let us return back to PCF. The identifiers of the constants hint at their semantics: $\dot{+}\underline{1}$ means the successor function, $\dot{-}\underline{1}$ the predecessor-function. *cond* stands for the sequential conditional and Y is the symbol for the fixed point operator introduced above.

Definition 2.15 *The* denotational semantics *of PCF-terms is defined by the following type frame* $\mathcal{D}$:

- $\mathcal{D}[\![\ ]\!] : \Sigma \to \underline{CPO},\ \sigma \mapsto D_\sigma$, *whereat* $D_\iota := \mathbf{N}_\bot$ *and* $D_{\sigma\to\tau} := [D_\sigma \to D_\tau]$.
- *For all* $\sigma, \tau \in \Sigma$ *let* $D^{\sigma,\tau} : \mathcal{D}[\![\sigma\to\tau]\!] \times \mathcal{D}[\![\sigma]\!] \to \mathcal{D}[\![\tau]\!]$ *be the usual application, i.e.* $D^{\sigma,\tau}(f,x) = f(x)$ *for all* $f \in [D_\sigma \to D_\tau]$ *and* $x \in D_\sigma$.

- $[\![\ ]\!] := [\![\ ]\!]_{\mathcal{D}} : \Lambda(C^{PCF}) \to (Env \to \mathcal{D}[\![\Sigma]\!])$, *whereat the set of the environments*

$$Env := \{\rho : V \to \mathcal{A}[\![\Sigma]\!] \mid f.a.\ \sigma \in \Sigma, x_i^\sigma \in V : \rho(x_i^\sigma) \in \mathcal{A}[\![\sigma]\!]\}$$

is satisfying the conditions

1. $[\![\underline{n}]\!] := n \in D_\iota$ *for all* $n \in \mathbf{N}$
2. $[\![\dot{+}\underline{1}]\!] \in D_{\iota^1} : x \mapsto \begin{cases} n+1 & , x = n \\ \perp & , x = \perp \end{cases}$
3. $[\![\dot{-}\underline{1}]\!] \in D_{\iota^1} : x \mapsto \begin{cases} n & , x = n+1 \\ \perp & , x \in \{\perp, 0\} \end{cases}$
4. $[\![cond]\!] \in D_{\iota^3} : (x,y,z) \mapsto \begin{cases} y & , x = 0 \\ z & , x = n+1 \\ \perp & , x = \perp \end{cases}$ *and*
5. $[\![Y_\sigma]\!] \in D_{\sigma^1\to\sigma} : f \mapsto \bigsqcup_{n\geq 0} f^n(\perp_\sigma)$ *for all* $\sigma \in \Sigma$
6. $[\![x_i^\sigma]\!]_{\mathcal{A}}(\rho) = \rho(x_i^\sigma)$ *for all* $\rho \in Env$ *and* $x_i^\sigma \in V$.
7. *For all* $\rho \in Env$ *and* $M, N \in \Lambda(C)$:
$$[\![MN]\!]_{\mathcal{A}}(\rho) = [\![M]\!]_{\mathcal{A}}(\rho)([\![N]\!]_{\mathcal{A}}(\rho)).$$
8. *For all* $\rho \in Env$, $M \in \Lambda(C)$, $\sigma \in \Sigma$, $x^\sigma \in V$ *and* $d \in \mathcal{A}[\![\sigma]\!]$:
$$[\![\lambda x^\sigma.M]\!]_{\mathcal{A}}(\rho)(d) = [\![M]\!]_{\mathcal{A}}(\rho[d/x]),$$
whereat
$$\rho[d/x](y) := \begin{cases} d & , x = y \\ \rho(y) & , \textit{otherwise.} \end{cases}$$

Note that there is no predecessor of 0. $x = n$ and $x = n+1$ is short for $x \in \mathbf{N}_\perp \setminus \{\perp\}$, resp. $x \in \mathbf{N}_\perp \setminus \{\perp, 0\}$.

Definition 2.16 *The* arity *ar and the* order *ord of a type is defined inductively by*

1. *For all* $\sigma \in \Sigma_0$ *let* $ar(\sigma) := 0$ *and for all* $\sigma, \tau \in \Sigma$ *let* $ar(\sigma\to\tau) := ar(\tau) + 1$.
2. *For all* $\sigma \in \Sigma_0$ *let* $ord(\sigma) := 0$ *and for all* $\sigma, \tau \in \Sigma$ *let* $ord(\sigma\to\tau) := \max\{ord(\sigma) + 1, ord(\tau)\}$.

The arity, resp. order of a function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ *or a term* $M \in \Lambda(C)$ *is given by the corresponding type. The order of a set* M *of functions or terms let be* $ord(M) := lub\{ord(x) \mid x \in M\}$.

Obviously, we get

Lemma 2.17 *Let σ be a PCF-type. Then $ord(\sigma) \leq 1$ if, and only if, $\sigma = \iota^n$ for some $n \in \mathbf{N}$.*

Lemma 2.18 *1. For all $M \in \Lambda(C^{PCF})$ and $\rho \in Env$ is $[\![M^\sigma]\!](\rho) \in D_\sigma$.*

2. The denotational semantics of a PCF-term is only dependent on the values of the free occuring variables. $[\![M]\!](\rho_1) = [\![M]\!](\rho_2)$ holds for all $M \in \Lambda(C^{PCF})$ and $\rho_1, \rho_2 \in Env$ whenever $\rho_1(x) = \rho_2(x)$ for all $x \in FV(M)$.

3. For all $M^\sigma, N \in \Lambda(C^{PCF})$, $x_i^\sigma \in V$ and $\rho \in Env$ is

$$[\![[M^\sigma/x_i^\sigma](N)]\!](\rho) = [\![N]\!](\rho[[\![M]\!](\rho)/x_i^\sigma]) = [\![(\lambda x_i^\sigma.N)(M^\sigma)]\!](\rho).$$

4. Λ-equivalent terms have the same semantics, i.e.

$$f.a.\ M, N \in \Lambda(C^{PCF}) :\ M \equiv_\Lambda N \Rightarrow [\![M]\!] = [\![N]\!].$$

Proof:

1. The proof can be carried out by structural induction on M: If $M \in C$ or $M \in V$, look at the definition of the semantics. In case M is an application, use the fact that the continuous functions are closed under composition. In case M is an abstraction compare [Gun92]. There it is shown that the function $\varphi : D_\sigma \to D_\tau : d \mapsto [\![\lambda x^\sigma.M^\tau]\!](\rho[d/x])$ is continuous for all $\sigma, \tau \in \Sigma$, $M \in \Lambda(C^{PCF})_\tau$ and $\rho \in Env$.

2. By structural induction it can be shown that $[\![M]\!](\rho)$ is already well-defined by ρ restricted on $FV(M)$.

3. One shows the equation by induction according to the cases in definition 2.3.

4. According to the cases in definition 2.4 the statement can be proved.

The second part of the previous lemma yields that the semantics of a closed PCF-term M is independent of the current environment. We define $[\![M]\!] := [\![M]\!](\rho)$ for all $\rho \in Env$. The fourth part allows to identify Λ-equivalent terms when considering the semantics. Since a term-rewriting system, describing the way a machine is evaluating the terms, belongs to a "regular" programming language, we introduce now the operational semantics.

Definition 2.19 *The* operational semantics *eval of a PCF-program M is defined by the transition rules as follows:*

1. *For all $n \in \mathbf{N}$ it is* $\dot{+}\underline{1}(\underline{n}) \to \underline{n+1}$ *and* $\dot{-}\underline{1}(\underline{n+1}) \to \underline{n}$.
2. *For all $M, N \in \Lambda(C^{PCF})_\iota$ it is* $cond(\underline{0}, M, N) \to M$ *and* $cond(\underline{n+1}, M, N) \to N$.
3. *For all $\sigma \in \Sigma$ and $M \in \Lambda(C^{PCF})_{\sigma^1}$ it is* $Y_\sigma M \to M(Y_\sigma M)$.
4. *For all $\sigma \in \Sigma$, $M \in \Lambda(C^{PCF})$ and $N \in \Lambda(C^{PCF})_\sigma$ it is* $(\lambda x^\sigma.M)N \to [N/x](M)$.
5. *For all $M_1, M_2, N \in \Lambda(C^{PCF})$ we have* $\dfrac{M_1 \to M_2}{M_1 N \to M_2 N}$.
6. *For all $N_1, N_2 \in \Lambda(C^{PCF})_\iota$ and $M \in \{\dot{+}\underline{1}, \dot{-}\underline{1}, cond\}$ we have:* $\dfrac{N_1 \to N_2}{MN_1 \to MN_2}$.

Let $eval(M) = \underline{n} :\Leftrightarrow M \to^* \underline{n}$ *for all* $M \in \Lambda(C^{PCF})_\iota$ *and* $n \in \mathbf{N}$*, whereat* $\to^*$ *is the reflexive and transitive closure of the "*$\to$*"-transition.*

The evaluation of each term goes from the left to the right. Rule 4 says that the "call-by-name"-principle holds, i.e. when evaluating an application of an abstraction to an argument, one substitutes the argument for the formal parameter in the body of the abstraction without being evaluated itself. It is easy to see that whenever evaluating a term at each step at most one rule can be used. Hence, *eval* is a well-defined partial function.

The following lemma says the consistence of both semantics for PCF.

Lemma 2.20 *For all PCF-programs M and $n \in \mathbf{N}$ it is* $eval(M) = \underline{n} \Leftrightarrow [\![M]\!] = n$.

Proof: "$\Rightarrow$": For all PCF-terms M, N it is

$$M \to N \Rightarrow f.a.\ \rho \in Env : [\![M]\!](\rho) = [\![N]\!](\rho).$$

This can easily be verified by induction on the transition rules.

"$\Leftarrow$": This direction is proved in [Plo77]. □

In order to construct complex functions we introduce symbols for often needed subterms:

Definition 2.21 *Abbreviations for special PCF-terms and their semantics:*

Ω^ι	$:= Y_\iota(\lambda x^\iota.x)$ *and*
$\Omega^{\sigma\to\tau}$	$:= \lambda x^\sigma.\Omega^\tau$ *for all* $\sigma, \tau \in \Sigma$
	mean the $\perp$-elements of the corresponding cpo-s.
Z	$:= \lambda x^\iota.cond(x, \underline{0}, \underline{1})$

is the translation of integer into boolean

$[\![Z]\!] \quad : x \mapsto \begin{cases} 0 & , x = 0 \\ 1 & , x = n+1 \\ \bot & , x = \bot. \end{cases}$

$\neg := not \quad := \lambda x^\iota.cond(x, \underline{1}, \underline{0})$

means the logical negation:

$[\![not]\!] \quad : x \mapsto \begin{cases} 0 & , x = n+1 \\ 1 & , x = 0 \\ \bot & , x = \bot. \end{cases}$

$\underline{\vee} := lor \quad := \lambda x^\iota, y^\iota.cond(x, \underline{0}, Z(y))$

means the logical or-relation, whereat the first argument is evaluated at first. $\underline{\vee}$ *is used as infix operator:*

$[\![lor]\!] \quad : (x, y) \mapsto \begin{cases} 0 & , x = 0 \\ [\![Z]\!](y) & , x = n+1 \\ \bot & , x = \bot. \end{cases}$

$ror \quad := \lambda x^\iota, y^\iota.cond(y, \underline{0}, Z(x))$

means the logical or-relation, whereat the second argument is evaluated at first.

$\underline{\wedge} := land \quad := \lambda x^\iota, y^\iota.cond(x, Z(y), \underline{1})$

means the logical and-relation, whereat the first argument is evaluated at first. $\underline{\wedge}$ *is used as infix operator:*

$[\![land]\!] \quad : (x, y) \mapsto \begin{cases} [\![Z]\!](y) & , x = 0 \\ 1 & , x = n+1 \\ \bot & , x = \bot. \end{cases}$

$rand \quad := \lambda x^\iota, y^\iota.cond(y, Z(x), \underline{1})$

means the logical and-relation, whereat the second argument is evaluated first.

$conv \quad := \lambda x^\iota.cond(x, \underline{0}, \underline{0})$

is true, if the argument is not $\bot$*, i.e. is defined:*

$[\![conv]\!] \quad : x \mapsto \begin{cases} 0 & , x = n \\ \bot & , x = \bot. \end{cases}$

Instead of $\dot{+}\underline{1}(x)$ and $\dot{-}\underline{1}(x)$ we write $x \dot{+} \underline{1}$, resp. $x \dot{-} \underline{1}$ for the sake of more clearly reading.

Definition 2.22 *1. A PCF-term M^σ is said to* define *the function $f \in D_\sigma$ in the environment $\rho \in Env$, if $[\![M]\!](\rho) = f$.*

2. A function $f \in D_\sigma$ is said to be (PCF-)definable, *if $[\![M]\!] = f$ for some closed PCF-term M^σ.*

When looking only into the semantics it is sufficient to consider terms in the form of $\lambda x_1^{\sigma_1}, \ldots, x_n^{\sigma_n}.N^\iota$, because for all terms M^σ with $\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$ a term N^ι exists such that $[\![\lambda x_1^{\sigma_1}, \ldots, x_n^{\sigma_n}.N]\!] = [\![M]\!]$. Define $N := M x_1^{\sigma_1} \ldots x_n^{\sigma_n}$ (cp. η-contraction in Definition 2.4).

Not every element in the model $\mathcal{D}$ is PCF-definable. However, the name of PCF is correct, since all in common meaning computable, i.e. μ-recursive, partial functions have a corresponding term in PCF:

Theorem 2.23 *For every μ-recursive partial function $f : \mathbf{N}^n \rightharpoonup \mathbf{N}$ the function $cu(\overline{f}) := curry(\overline{f})$ is PCF-definable, whereat*

$$\overline{f} : \mathbf{N}_\perp^n \to \mathbf{N}_\perp,\ (x_1, \ldots, x_n) \mapsto \begin{cases} f(x_1, \ldots, x_n) & , \text{ f.a. } 1 \le i \le n:\ x_i \ne \perp \\ & \quad \wedge f(\vec{x}) \text{ is defined} \\ \perp & , \text{ otherwise.} \end{cases}$$

Proof: $\overline{f}$ is the strict expansion of f, i.e. all elements not being at the top in $\mathbf{N}_\perp^n$ were assigned to $\perp$. Therefore $cu(\overline{f})$ is element of D_{ι^n}. Currying is necessary because of the missing of product types in Σ. The mapping $f \mapsto \overline{f}$ is bijective.

Set $F_n(M^\iota) := \lambda x_1^\iota, \ldots, x_n^\iota.cond(conv(x_1) \mathbin{\underline{\wedge}} \ldots \mathbin{\underline{\wedge}} conv(x_n), M^\iota, \Omega^\iota)$. We prove the assertion by induction on the structure of the μ-recursive functions. The used notations are taken from [Spr92].

1. For the successor-function S it is $\overline{S} = [\![\dot{+}\underline{1}]\!]$.

2. For all $1 \le i \le n$ it is $\overline{U_i^n} = [\![F_n(x_i^\iota)]\!]$ (projections onto coordinates).

3. For all $i \in \mathbf{N}$ it is $\overline{C_i^n} = [\![F_n(\underline{i})]\!]$ (constant functions).

4. Closure under simultaneous substitution:

   Let $cu(\overline{g}) \in D_{\iota^n}$ and $cu(\overline{h_i}) \in D_{\iota^m}$ for $1 \le i \le n$ be defined by $\lambda y_1^\iota, \ldots, y_n^\iota.G^\iota$, resp. $\lambda x_1^\iota, \ldots, x_m^\iota.H_i^\iota$. We get $cu(\overline{g(h_1, \ldots, h_n)}) = [\![F_m([H_1/y_1^\iota] \ldots [H_n/y_n^\iota](G))]\!]$. Notice that $h_i(\vec{y})$ is defined for all $1 \le i \le n$, if $g(\vec{h}(\vec{y}))$ is defined.

5. Closure under primitive recursion:

   Let $cu(\overline{g}) \in D_{\iota^n}$ be defined by $\lambda x_1^\iota, \ldots, x_n^\iota.G^\iota$ and $cu(\overline{h}) \in D_{\iota^{n+2}}$ by $\lambda y_1^\iota, \ldots, y_{n+2}^\iota.H^\iota$. With $\overline{f}(\vec{x}, 0) := \overline{g}(\vec{x}), \overline{f}(\vec{x}, m+1) := \overline{h}(\vec{x}, m, \overline{f}(\vec{x}, m))$ and $\overline{f}(\vec{x}, \perp) := \perp$ we have $f = PR(g, h)$. Therefore is

$$\begin{aligned} cu(\overline{f}) = [\![& Y_{\iota^{n+1}} \lambda \varphi^{\iota^{n+1}}, x_1^\iota, \ldots, x_n^\iota, m^\iota. \\ & cond(m, G, [x_1/y_1] \ldots [x_n/y_n][m \mathbin{\dot{-}} \underline{1}/y_{n+1}][\varphi(\vec{x}, m \mathbin{\dot{-}} \underline{1})/y_{n+2}](H))]\!]. \end{aligned}$$

6. Closure under μ-operator:

   Let $cu(\overline{g}) \in D_{\iota^{n+1}}$ be defined by $\lambda x_1^\iota, \ldots, x_{n+1}^\iota . G^\iota$. With

   $\overline{f}(x_1, \ldots, x_n) := min\{k \in \mathbf{N} \mid f.a.\ i \le k : \overline{g}(x_1, \ldots, x_n, i) \neq \perp \wedge\, \overline{g}(x_1, \ldots, x_n, k) = 0\}$,

   whereat $min(\emptyset) = \perp$, one has $f = \mu(g)$. For arbitrary, but fixed terms $K_1^\iota, \ldots, K_n^\iota$ set

   $$\Phi := \lambda \varphi^{\iota^1}, y^\iota . cond([K_1/x_1] \ldots [K_n/x_n][y/x_{n+1}](G), y, \varphi(y \dot{+} \underline{1})).$$

   We obtain

   $$[\![\Phi]\!](\perp_{\iota^1}) \in D_{\iota^1} :\ y \mapsto \begin{cases} y & , \overline{g}([\![K_1]\!], \ldots, [\![K_n]\!], y) = 0 \\ \perp & , \text{otherwise,} \end{cases}$$

   $$[\![\Phi]\!]^2(\perp_{\iota^1}) \in D_{\iota^1} :\ y \mapsto \begin{cases} y & , \overline{g}([\![K_1]\!], \ldots, [\![K_n]\!], y) = 0 \\ y+1 & , \overline{g}([\![K_1]\!], \ldots, [\![K_n]\!], y) \ge 1 \\ & \wedge\, \overline{g}([\![K_1]\!], \ldots, [\![K_n]\!], y+1) = 0 \\ \perp & , \text{otherwise,} \end{cases}$$

   $$\vdots$$

   $$[\![Y_{\iota^1}\Phi]\!] \in D_{\iota^1} :\ y \mapsto y + min\{k \in \mathbf{N} \mid f.a.\ i \le k : \overline{g}([\![K_1]\!], \ldots, [\![K_n]\!], y+i) \neq \perp \\ \wedge\, \overline{g}([\![K_1]\!], \ldots, [\![K_n]\!], y+k) = 0\}.$$

   Thus, we have

   $$cu(\overline{f}) = [\![\lambda x_1^\iota, \ldots, x_n^\iota . (Y_{\iota^1} \lambda \varphi^{\iota^1}, y^\iota . cond([y/x_{n+1}](G), y, \varphi(y \dot{+} \underline{1})))\underline{0}]\!].$$

$\square$

By this theorem the construction of many terms is made easier. For example the arithmetic comparison operators and functions like addition, multiplication etc. are μ-recursive. In the following they are defined by $\doteq, \dot{\ge}, \dot{\le}, \dot{+}, \dot{-}$ etc. For the inversion of the theorem see section 5.

3 Sequentiality Relations

The following characterization of PCF-definable elements with special relations bases on [Sie92]. It will be the main tool in this paper for analyzing the degrees of parallelism. We begin with relations corresponding in natural way to the structure of the types in the λ-calculus.

Definition 3.1 *Let I be any set of indices and $\mathcal{A}$ a type frame for the simply-typed λ-calculus for C and Σ. A family of relations $R := (R^\sigma)_{\sigma\in\Sigma}$ is called an I-indexed* logical relation *on $\mathcal{A}$, if the following conditions hold:*

- *For all* $\sigma \in \Sigma_0$ *is* $R^\sigma \subseteq A^I_\sigma$.
- *For all* $\sigma, \tau \in \Sigma$ *and* $(f_i)_{i\in I} \in A^I_{\sigma\to\tau}$ *is*

$$(f_i)_{i\in I} \in R^{\sigma\to\tau} \Leftrightarrow \textit{ f.a. } (x_i)_{i\in I} \in R^\sigma : (f_i(x_i))_{i\in I} \in R^\tau.$$

- *For all* $c \in C$ *is* $[\![c]\!]_{\mathcal{A}}$ *invariant under* R*, whereat* $x \in A_\sigma$ *is said to be* invariant *under* R*, if* $(x)_{i\in I} \in R^\sigma$*. The abridged notation is* $x \in R$*.*

The whole relation R is uniquely determined by its definition on the ground types. The invariance of the constants is transmitted to all λ-terms.

Lemma 3.2 *Let* R *be a logical relation on a type frame* $\mathcal{A}$*. Then* $[\![M]\!]$ *is invariant under* R *for every closed* $M \in \Lambda(C)$*.*

Proof: It is sufficient and moreover easy to show that we have for all sets of indices I, I-indexed logical relations R, $\sigma \in \Sigma$, $M^\sigma \in \Lambda(C)$ and $(\rho_i)_{i\in I} \in Env^I$:

$$(\textit{f.a. } \tau \in \Sigma, x^\tau \in FV(M): \ (\rho_i)_{i\in I} \in R^\tau) \Rightarrow ([\![M]\!](\rho_i))_{i\in I} \in R^\sigma.$$

For closed terms M the left side trivially is true. Therefore is $([\![M]\!])_{i\in I} \in R^\sigma$ (cp. [Plo80]).

Definition 3.3 1. *For all* $n \in \mathbf{N}$ *and* $A, B \subseteq \{1, \ldots, n\}$ *with* $A \subseteq B$ *let* $S^n_{A,B} \subseteq \mathbf{N}^n_\perp$ *be the relation defined as follows:*

$$(d_1, \ldots, d_n) \in S^n_{A,B} :\Leftrightarrow \ (\textit{ex. } i \in A: \ d_i = \perp) \ \vee \ (\textit{f.a. } i, j \in B: \ d_i = d_j).$$

2. *An* n*-ary logical relation* R *is said to be a* sequentiality relation*, if* R^ι *is an intersection of relations of the form* $S^n_{A,B}$*. The set of all sequentiality relations is marked as* SR*.*
3. *An element* $d \in D_\sigma$ *is said to be* logically sequential*, if* d *is invariant under every sequentiality relation.*

The often considered sequentiality relations $S^n_{\{1,\ldots,n\},\{1,\ldots,n\}}$ and $S^n_{\{1,\ldots,m\},\{1,\ldots,n\}}$ will be written in abridged form as $\tilde{n}$, resp. as $(1, \ldots, m \mid m+1, \ldots, n)$.

Remark: For all logical relations R and S and functions f with $ord(f) \leq 1$ one has $f \in R \ \wedge \ f \in S \Rightarrow f \in R \cap S$. Yet for $ord(f) \geq 2$ is this generally not true. An example will be given later.

The sequentiality relations are the logical relations characterizing the behaviour of PCF-definable functions:

Theorem 3.4 *The sequentiality relations are the only finitely indexed logical relations which the semantics of all PCF-constants and thus of all PCF-definable functions is invariant under.*

Proof: A proof in detail can be found in [Sie92]. There is shown that on the one hand every PCF-constant is logically sequential, and on the other hand every n-ary logical relation which is invariant under $[\![cond]\!]$, $[\![\dot{+}\underline{1}]\!]$, $[\![\dot{-}\underline{1}]\!]$ and all $x \in \mathbf{N}_\perp$, is a sequentiality relation. The invariance under the fixed-point operators is not required here.

With these characterization it is easy to prove that the continuous parallel or-function

$$por: \ \mathbf{N}_\perp^2 \to \mathbf{N}_\perp, \ (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \vee \ y = 0 \\ 1 & , x = n+1 \ \wedge \ y = m+1 \\ \perp & , \text{otherwise} \end{cases}$$

is not PCF-definable:

$(\perp, 0, 1), (0, \perp, 1) \in \tilde{3}$, but $por((\perp, 0, 1), (0, \perp, 1)) = (por(\perp, 0), por(0, \perp), por(1, 1)) = (0, 0, 1) \notin \tilde{3}$, i.e. we have $por \notin \tilde{3}$.

A second example for a non-definable function is

$$\mathcal{O} \in D_{\iota^2}: \ (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \vee \ y = 0 \\ \perp & , \text{otherwise.} \end{cases}$$

It is not invariant under $(1, 2 \mid 3)$, because $(0, \perp, \perp), (\perp, 0, \perp) \in (1, 2 \mid 3)$, but $\mathcal{O}((0, \perp, \perp), (\perp, 0, \perp)) = (0, 0, \perp) \notin (1, 2 \mid 3)$.

4 Degrees of Parallelism

What about differences between non-definability of continuous functions? How are the various non-definabilities comparable with each other? For that we introduce a classification for the sets $\Lambda(C)$ whose set of constants C is an extension of C^{PCF} with regard to their power.

In the common theory of computation the Turing reducibility of partial functions $f, g : \mathbf{N} \rightharpoonup \mathbf{N}$ is defined by $f \leq_T g :\Leftrightarrow f \in F_\mu(g)$. In analogous way we introduce:

Definition 4.1 *For all types $\sigma, \tau \in \Sigma$ and functions $f \in D_\sigma, g \in D_\tau$ we call f* sequentially- *or* PCF-reducible *on g, if there is a closed $\Lambda(C^{PCF} \cup \{\underline{g}\})$-term M^σ such that $[\![M]\!] = f$, whereat $\underline{g}$ is a constant of type τ with semantics $[\![\underline{g}]\!] = g$. The abridged notation is $f \leq_p g$.*

The meaning of $f \leq_p g$ is that f can be described, if PCF is extended by a constant $\underline{g}$ standing for g. The occurences of $\underline{g}$ in an appropriate term can be interpreted as questions to an oracle representing the function g. A further interpretation of $f \leq_p g$ is:

Lemma 4.2 *Let $\sigma, \tau \in \Sigma$ and $f \in D_\sigma, g \in D_\tau$. Then f is sequentially reducible on g if, and only if, $f = \llbracket G\rrbracket(g)$ for some closed PCF-Term $G^{\tau\to\sigma}$.*

Proof: "$\Leftarrow$": Defining $G' := \lambda x^\tau.G$ we have $f = \llbracket G\rrbracket(g) = \llbracket [\underline{g}/x^\tau](G')\rrbracket$.

"$\Rightarrow$": Let $f = \llbracket M\rrbracket$. Replace in M for each occurence of $\underline{g}$ the same variable $z^\tau \notin FV(M) \cup GV(M)$. Calling the obtained term M' one gets $\llbracket \lambda z^\tau.M'\rrbracket(g) = f$. □

$\leq_p$ forms a preorder on the continuous functions:

Lemma 4.3 *$\leq_p \subseteq \bigcup_{\sigma\in\Sigma} D_\sigma \times \bigcup_{\sigma\in\Sigma} D_\sigma$ is a reflexive and transitive relation.*

Proof: Reflexivity: It is $\llbracket \lambda x^\sigma.x\rrbracket(f) = f$ for all $\sigma \in \Sigma$ and $f \in D_\sigma$.

Transitivity: Let $\sigma_1, \sigma_2, \sigma_3 \in \Sigma$ and $f_i \in D_{\sigma_i}$ for $1 \leq i \leq 3$ be any functions such that $f_1 \leq_p f_2$ and $f_2 \leq_p f_3$. Then there are PCF-terms $G_1^{\sigma_1}$ and $G_2^{\sigma_2}$ with $FV(G_1) \subseteq \{x^{\sigma_2}\}$ and $FV(G_2) \subseteq \{x^{\sigma_3}\}$, so that for all $\rho \in Env$ the equations $\llbracket \lambda x^{\sigma_2}.G_1\rrbracket(\rho)(f_2) = \llbracket G_1\rrbracket(\rho[f_2/x^{\sigma_2}]) = f_1$ and $\llbracket \lambda x^{\sigma_3}.G_2\rrbracket(\rho)(f3) = f_2$ hold. We get for all $\rho \in Env$:

$$\begin{aligned}
\llbracket \lambda x^{\sigma_3}.[G_2/x^{\sigma_2}](G_1)\rrbracket(\rho)(f_3) &= \llbracket [G_2/x^{\sigma_2}](G_1)\rrbracket(\rho[f_3/x^{\sigma_3}]) \\
&= \llbracket G_1\rrbracket\Big(\rho[f_3/x^{\sigma_3}][\llbracket G_2\rrbracket(\rho[f_3/x^{\sigma_3}])/x^{\sigma_2}]\Big) \\
&= \llbracket G_1\rrbracket(\rho[f_3/x^{\sigma_3}][f_2/x^{\sigma_2}]) \qquad = f_1.
\end{aligned}$$

$\lambda x^{\sigma_3}.[G_2/x^{\sigma_2}](G_1)$ is closed, so we have $f_1 \leq_p f_3$. □

Definition 4.4 *For any function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ set $\langle f\rangle_p := \{g \in \bigcup_{\sigma\in\Sigma} D_\sigma \mid g \leq_p f\}$.*

$\langle f\rangle_p$ includes exactly the functions definable from the extension of PCF with $\underline{f}$. Obviously, we have $f \leq_p g \Leftrightarrow \langle f\rangle_p \subseteq \langle g\rangle_p$. Therefore, we make the

Definition 4.5 *For all $f, g \in \bigcup_{\sigma\in\Sigma} D_\sigma$ set*

- $f <_p g :\Leftrightarrow \langle f\rangle_p \subset \langle g\rangle_p$,
- $f \,|_p\, g :\Leftrightarrow \langle f\rangle_p \not\subseteq \langle g\rangle_p \;\wedge\; \langle g\rangle_p \not\subseteq \langle f\rangle_p$ *(read: f and g are incomparable) and*
- $f \equiv_p g :\Leftrightarrow \langle f\rangle_p = \langle g\rangle_p$.

One obtains directly:

Lemma and definition 4.6 *$\equiv_p \subseteq \bigcup_{\sigma\in\Sigma} D_\sigma \times \bigcup_{\sigma\in\Sigma} D_\sigma$ is an equivalence relation. The equivalence class of $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$, that is $[f]_p := \{g \in \bigcup_{\sigma\in\Sigma} D_\sigma \mid f \equiv_p g\}$, is called the* degree of parallelism *(dop) of f.* □

Two functions f, g have the same *dop*, if the set of functions being reducible on f and the set of functions being reducible on g are identical, i.e. both extensions of PCF with $\underline{f}$, resp. with $\underline{g}$ reach the same functions.

Lemma 4.7 *$f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ is PCF-definable if, and only if, it is*

$$\langle f\rangle_p = \{g \in \bigcup_{\sigma\in\Sigma} D_\sigma \mid g \text{ PCF-definable}\}.$$

Proof: "$\Leftarrow$": Clear because of the reflexivity of $\leq_p$.

"$\Rightarrow$":

"$\subseteq$": Let $f \in D_\sigma$ be defined by the PCF-term M and $g \in D_\tau$ a function with $g \leq_p f$. Then there is a PCF-term G^τ such that $FV(G) \subseteq \{x^\sigma\}$ and for all $\rho \in Env$

$$g = [\![\lambda x^\sigma.G]\!](\rho)(f) = [\![G]\!](\rho[f/x^\sigma]) = [\![[M/x^\sigma](G)]\!](\rho).$$

Thus, g is definable.

"$\supseteq$": Let g be defined by N. Then we have $g = [\![\lambda x^\sigma.N]\!](f)$ and therefore $g \leq_p f$. □

The sequential reducibility can be lifted onto the equivalence classes:

Definition 4.8 *1. Let $[\,]_p$ be the notation for the dop of the PCF-definable functions.*

2. For all $f, g \in \bigcup_{\sigma\in\Sigma} D_\sigma$ set $[f]_p \leq [g]_p :\Leftrightarrow f \leq_p g$.

The first item is well-defined by the previous lemma. The second item is well-defined, because for any two representatives f_1, f_2 of a class $[f]_p$ the sets $\langle f_1\rangle_p$ and $\langle f_2\rangle_p$ coincide with $\langle f\rangle_p$.

Theorem 4.9 *The dops form an upper semilattice with regard to $\leq$ in the previous definition, i.e. they form a partial order with a least element, namely $[\,]_p$, such that each two degrees $[f]_p$ and $[g]_p$ have a lub, that is $[f \oplus g]_p$.*

Proof: Lemma 4.3 implies directly reflexivity and transitivity.

$\leq$ is antisymmetric: Let $[f]_p$ and $[g]_p$ be *dops* such that $[f]_p \leq [g]_p$ and $[g]_p \leq [f]_p$. Then there are $f_0 \in [f]_p$ and $g_0 \in [g]_p$ with $f_0 \equiv_p g_0$. Thus, we have $[f]_p = [g]_p$.

$[\,]_p$ is the least degree: Let φ be defined by a PCF-term N and $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ an arbitrary function. We get $\varphi = [\![\lambda x^\sigma.N]\!](f)$, thus $\varphi \leq_p f$, i.e. $[\,]_p \leq [f]_p$.

For any $\sigma = (\sigma_1, \ldots, \sigma_n, \iota), \tau = (\tau_1, \ldots, \tau_m, \iota) \in \Sigma$ and $f \in D_{(\sigma_1,\ldots,\sigma_n,\iota)}, g \in D_{(\tau_1,\ldots,\tau_m,\iota)}$ we define

$$f \oplus g := [\![\lambda x^\iota, y_1^{\sigma_1}, \ldots, y_n^{\sigma_n}, z_1^{\tau_1}, \ldots, z_m^{\tau_m}.\; cond(x, \underline{f}(y_1, \ldots, y_n), \underline{g}(z_1, \ldots, z_m))]\!] \in D_{\iota,\sigma_1,\ldots,\sigma_n,\tau_1,\ldots,\tau_m,\iota)}.$$

Hence, $f = [\![\lambda \vec{y}^\sigma.\underline{f \oplus g}(\underline{0}, \vec{y}, \Omega^{\tau_1}, \ldots, \Omega^{\tau_m})]\!]$ and $g = [\![\lambda \vec{z}^\tau.\underline{f \oplus g}(\underline{1}, \Omega^{\sigma_1}, \ldots, \Omega^{\sigma_n}, \vec{z})]\!]$ hold. This implies $f \leq_p f \oplus g$ and $g \leq_p f \oplus g$. Let $h \in D_{(\upsilon_1,\ldots,\upsilon_k,\iota)}$ be such that $f \leq_p h$ and $g \leq_p h$. Then there are PCF-terms M_1^σ, M_2^τ with $[\![\lambda \vec{w}^\upsilon.M_1]\!](h) = f$ and $[\![\lambda \vec{w}^\upsilon.M_2]\!](h) = g$. So it is

$$[\![\lambda \vec{w}^\upsilon, x^\iota, \vec{y}^\sigma, \vec{z}^\tau.cond(x, M_1(\vec{y}), M_2(\vec{z}))]\!](h) = f \oplus g,$$

i.e. $f \oplus g \leq_p h$. With that it is shown $[f \oplus g]_p = [f]_p \sqcup_p [g]_p =: [f]_p \oplus [g]_p$. □

Obviously, the $\oplus$-operator is commutative and associative on the *dops*, i.e. one has $f \oplus g \equiv_p g \oplus f$ and $(f \oplus g) \oplus h \equiv_p f \oplus (g \oplus h)$ for all $f, g, h \in \bigcup_{\sigma \in \Sigma} D_\sigma$. Therefore, we can define the *dop* of a finite set of functions:

$$[\{f_i \mid 1 \leq i \leq n\}]_p := \oplus\{[f_i]_p \mid 1 \leq i \leq n\} := [f_1]_p \oplus \cdots \oplus [f_n].$$

In order to prove $f \leq_p g$, one has to construct a closed PCF-term G such that $[\![G]\!](g) = f$. But what about verifying $f \not\leq_p g$?

Lemma 4.10 *Let S be any sequentiality relation. If $f, g \in \bigcup_{\sigma \in \Sigma} D_\sigma$ are functions such that g is invariant under S, but f is not, then we have $f \not\leq_p g$.*

Proof: Assuming $f \leq_p g$ there is a closed term M in PCF extended by $\underline{g}$ with $[\![M]\!] = f$. Then lemma 3.2 implies the invariance of M under S. This is a contradiction to the supposition. □

The inversion of this lemma does not hold. An example can be found in section 9.

Corollary 4.11 *For any $S \in SR$ and $f, g \in \bigcup_{\sigma \in \Sigma} D_\sigma$ we have*

1. $f \leq_p g \Rightarrow (g \in S \Rightarrow f \in S)$
2. $f \equiv_p g \Rightarrow (f \in S \Leftrightarrow g \in S)$.

Proof: The first item is the contraposition of the previous lemma; the second item is implied by the first one. □

Thus, we obtain: Invariance under a certain sequentiality relation is "hereditary" downwards in the semilattice, and proving invariance of a *dop* requires only checking an arbitrary representative.

5 Computability

In the following sections we deal with *dops* of computable functions, i.e. effectively representable constructions like it is usual in information theory. "computable" is here used with the meaning as for effective cpo-s. With this restriction one yields that there is a maximal *dop*, the *dop* of $\exists{\oplus}pcond$. Every computable function is sequentially reducible on this function. Theorem 5.8 shows a connection between common computability and PCF-definability. Before, we give the definitions for the needed expressions from cpo-theory and computability.

We presume that the meaning of "recursive" and "recursive enumerable" is known. $F_\mu^{(n)}$ stands for the set of the computable (μ-recursive) partial functions from $\mathbf{N}^n$ to $\mathbf{N}$. Here the symbol "$\rightharpoonup$" is used for a partial function. $R^{(n)}$ marks the total n-ary computable functions, $F_{prim}^{(n)}$ the n-ary primitive recursive functions.

$\langle\ ,\rangle : \bigcup_{n\in\mathbf{N}} \mathbf{N}^n \to \mathbf{N}$ is the primitive recursive standard encoding, i.e. $\langle y_1, \ldots, y_n\rangle = p_1^{y_1+1} \cdot \ldots \cdot p_n^{y_n+1}$, whereat p_i is the i-th prime number ordered by ascending size. $(\ )_i$ stands for the i-th component of the decoding. Let $l(m)$ for any $m \in \mathbf{N}$ be the number of the largest prime factor of m. Then we have $m = \langle (m)_1, \ldots, (m)_{l(m)}\rangle$.

Let G be the standard numbering of the finite subsets of $\mathbf{N}$, i.e. $G_m \in \mathcal{P}_{fin}(\mathbf{N})$ is the unique set with $m = \sum_{m_i\in G_m} 2^{m_i}$, e.g. $G_8 = \{3\}$, $G_3 = \{0,1\}$ and $G_0 = \emptyset$.

W stands for the standard numbering of the recursive enumerable (r.e.) subsets of $\mathbf{N}$.

Not every element of a cpo can be described in a finite way. But there is only interest in those which can be approximated sufficiently precise in an effective and finite way.

Definition 5.1 *Let D be a cpo.*

1. *The* strongly smaller-*relation $\ll\ \subseteq D \times D$ is defined as follows:*

$$x \ll y :\Leftrightarrow \textit{ for all directed } M \subseteq D:\ y \sqsubseteq \bigsqcup M \Rightarrow \textit{ ex. } m \in M:\ x \sqsubseteq m.$$

2. *$x \in D$ is said to be* compact *whenever $x \ll x$. Set $\hat{D} := \{x \in D \mid x \ll x\}$.*

3. *$B \subseteq D$ is said to be a* basis *of D, if the set $B_x := \{b \in B \mid b \ll x\}$ is directed and $x = \bigsqcup B_x$ for all $x \in D$.*

$x \ll y$ can be interpreted as "x is necessary information for y". Hence, the compact elements represent the basic informations. Cpo-s which every element is obtainable as lub of these elements in are desired.

Definition 5.2 *A cpo D is said to be* algebraic, *if $\hat{D}$ is a basis of D. It is said to be* ω-algebraic, *if $\hat{D}$ is also countable.*

The basic informations in function spaces are composed of so-called step functions.

Theorem and definition 5.3 *If D, E are consistently complete, algebraic cpo-s, then so is $[D \to E]$. An element in $[D \to E]$ is compact if, and only if, it is the lub of a finite set of step functions. A* step function *in $[D \to E]$ has the form*

$$(d{\to}e)(x) := \begin{cases} e & , d \sqsubseteq x \\ \perp & , \text{otherwise.} \end{cases}$$

for any $d \in \hat{D}$ and $e \in \hat{E}$.

The **proof** can be found in [Plo77].

Definition 5.4
1. *A cpo D is called* effective, *if there is a basis B with a (surjective) numbering $\beta : \mathrm{N} \to B$ such that $\{\langle i, j\rangle \mid \beta(i) \ll \beta(j)\}$ is r.e.*
2. *Let D be an effective cpo. An element $x \in D$ is called a* (cpo-)computable *element, if $\{i \in \mathrm{N} \mid \beta(i) \ll x\}$ is r.e.*
3. *Let D, E be effective cpo-s with basis numberings β^D, resp. β^E. A continuous function $f : D \to E$ is called* (cpo-)computable, *if $\{\langle i, j\rangle \mid \beta^E(j) \ll f(\beta^D(i))\}$ is r.e.*

Useful equivalences for checking the computabiliy of an element are:

Theorem 5.5 *If D is an effective cpo, the following items are equivalent:*

1. *x is computable.*
2. *There is a r.e. set $A \subseteq \mathrm{N}$ such that $\beta(A)$ is directed and $x = \bigsqcup \beta(A)$.*
3. *There is a function $f \in R^{(1)}$ such that $\beta(f(i)) \ll \beta(f(i+1))$ and $x = \bigsqcup_{i \in \mathrm{N}} \beta(f(i))$ for all $i \in \mathrm{N}$.*

This theorem and also the following statements are proved in [Wei80]. If the considered cpo-s are algebraic, in the definitions of effective and computable "$\ll$" can be replaced by "$\sqsubseteq$" because of

Lemma 5.6 *Let D be an algebraic cpo with basis B. Then for all $b \in B$ and $x \in D$ we have*

$$b \ll x \Leftrightarrow b \sqsubseteq x \quad \textit{und} \quad x \ll b \Leftrightarrow x \sqsubseteq b.$$

Requiring additional properties for effective cpo-s we get the basis of a function space $[D \to E]$ in effective way with the help of the bases of D and E:

Definition 5.7 *A cpo D is called* recursive, *if D is an ω-algebraic, consistently complete cpo with basis B and basis numbering β such that the following conditions hold:*

1. $\{\langle i,j\rangle \mid \beta(i) \sqsubseteq \beta(j)\}$ *is recursive.*
2. $L := \{k \mid \beta(G_k)$ *bounded*$\}$ *is recursive.*
3. *There is a function $q \in R^{(1)}$ such that $\beta(q(k)) = \bigsqcup \beta(G_k)$ for all $k \in L$.*

Theorem 5.8 *If D and E are recursive cpo-s w.r.t. the basis numberings β^D and β^E, the set*

$$M := \{k \mid \{\beta^D(i) {\rightarrow} \beta^E(j) \mid \langle i,j\rangle \in G_k\} \text{ bounded}\}$$

is recursive and also is $[D \to E]$ a recursive cpo w.r.t the following numbering:

$$\beta^{[D\to E]}(k) := \begin{cases} \bigsqcup\{\beta^D(i){\rightarrow}\beta^E(j) \mid \langle i,j\rangle \in G_k\} & , \ k \in M \\ \bot & , \ \textit{otherwise.} \end{cases}$$

In recursive cpo-s "computable function" and "computable element" have the same meaning:

Corollary 5.9 *For recursive cpo-s D, E a function $f \in [D \to E]$ is computable if, and only if, it is a computable element of $f \in [D \to E]$.*

Every domain in our model forms a recursive cpo:

Theorem 5.10 $\mathbf{N}_\bot$ *is a recursive cpo, in which every element is computable.*

Proof: $\{\bot\}$ and $\{\bot, n\}$ for any $n \in \mathbf{N}$ are the only directed, resp. bounded subsets of $\mathbf{N}_\bot$. Therefore, $\mathbf{N}_\bot$ is consistently complete. One checks readily that $x \ll y \Leftrightarrow x \sqsubseteq y$. So we have $\hat{\mathbf{N}}_\bot = \mathbf{N}_\bot$. $\mathbf{N}_\bot$ is a basis of $\mathbf{N}_\bot$, since $\bigsqcup\{x \in \mathbf{N}_\bot \mid x \sqsubseteq \bot\} = \bigsqcup\{\bot\} = \bot$ and $\bigsqcup\{x \in \mathbf{N}_\bot \mid x \sqsubseteq n\} = \bigsqcup\{\bot, n\} = n$ hold. This implies the ω-algebraicity of $\mathbf{N}_\bot$.

With

$$\beta : \mathbf{N} \to \mathbf{N}_\bot, \ x \mapsto \begin{cases} \bot & , \ x = 0 \\ n & , \ x = n+1 \end{cases}$$

we have $\{\langle i,j\rangle \mid \beta(i) \sqsubseteq \beta(j)\} = \{\langle 0,n\rangle \mid n \in \mathbf{N}\} \cup \{\langle n,n\rangle \mid n \geq 1\}$ recursive. Also are $\{i \in \mathbf{N} \mid \beta(i) \sqsubseteq \bot\} = \{0\}$ and $\{i \in \mathbf{N} \mid \beta(i) \sqsubseteq n\} = \{0, n+1\}$ recursive for all $n \in \mathbf{N}$. This shows that every element is computable.

The items 2 and 3 in definition 5.7 follow from

$\beta(G_k)$ is bounded
$\Leftrightarrow \sharp G_k = 1 \ \vee\ (\sharp G_k = 2 \ \wedge\ 0 \in G_k)$
$\Leftrightarrow\ (ex.\ i \in \mathbf{N}:\ k = 2^i) \ \vee\ (ex.\ i \geq 1:\ k = 2^i + 1).$

and defining $q:\ k \mapsto \max\{j \in \mathbf{N} \mid j \in G_k\} \ \in R^{(1)}$. For any set M we write $\sharp M$ for its cardinality. □

This proof is nearly the same for each finite flat cpo. What are the compact elements in our special function spaces like?

Theorem 5.11 *D_σ is a consistently complete, algebraic cpo for all $\sigma \in \Sigma$. For $\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$ an element is compact if, and only if, it is lub of a finite set F containing elements of the form $d_1 \to (\ldots \to (d_n \to e) \ldots)$, whereat*

1. *$d_i \in \hat{D}_{\sigma_i}$ for all $1 \leq i \leq n$ and $e \neq \bot_\iota$, and*
2. *for all $d_1 \to (\ldots \to (d_n \to e) \ldots), d'_1 \to (\ldots \to (d'_n \to e') \ldots) \in F$:*

$$(f.a.\ 1 \leq i \leq n:\ d_i \uparrow d'_i) \Rightarrow e = e'.$$

Proof: It suffices to verify the special form of the compact elements.

It is

$$\bigsqcup\{d \to e_i \mid 1 \leq i \leq n\}(x) = \bigsqcup\{e_i \mid x \sqsupseteq d, 1 \leq i \leq n\} = (d \to \bigsqcup\{e_i \mid 1 \leq i \leq n\})(x) \quad [*]$$

and

$$\begin{aligned}
f \in & \hat{D}_{(\sigma_1, \ldots, \sigma_n, \iota)} \\
& \Leftrightarrow f = \bigsqcup\{d_{i_1} \to e_{i_1} \mid 1 \leq i_1 \leq n_1(f), d_{i_1} \in \hat{D}_{\sigma_1}, e_{i_1} \in \hat{D}_{(\sigma_2, \ldots, \iota)}\} \\
& \quad = \bigsqcup\{d_{i_1} \to (\bigsqcup\{d_{i_1,i_2} \to e_{i_1,i_2} \mid 1 \leq i_2 \leq n_2(e_{i_1}), d_{i_1,i_2} \in \hat{D}_{\sigma_2}, e_{i_1,i_2} \in \hat{D}_{(\sigma_3, \ldots, \iota)}\}) \\
& \qquad\qquad \mid 1 \leq i_1 \leq n_1(f), d_{i_1} \in \hat{D}_{\sigma_1}\} \\
& \quad = \ldots \\
& \quad = \bigsqcup_{i_1}\{d_{i_1} \to \bigsqcup_{i_2}(d_{i_1,i_2} \to \ldots \bigsqcup_{i_n}(d_{i_1,\ldots,i_n} \to e_{i_n,\ldots,i_n}) \ldots)) \\
& \qquad\qquad \mid d_{i_1} \in \hat{D}_{\sigma_1}, d_{i_1,i_2} \in \hat{D}_{\sigma_2}, \ldots, e_{i_1,\ldots,i_n} \in D_\iota\} \\
& \quad = \bigsqcup_{i_1,\ldots,i_n}\{d_{i_1} \to d_{i_1,i_2} \to \ldots \to d_{i_1,\ldots,i_n} \to e_{i_1,\ldots,i_n}\} \text{ (because of [*])} \\
& \quad =: \bigsqcup_i\{d_i^{(1)} \to \ldots \to d_i^{(n)} \to e_i\}.
\end{aligned}$$

ad 1: For $e_i = \bot_\iota$ it is $d_i^{(1)} \to \ldots \to e_i = \bot_\sigma$ and $\bigsqcup F = \bigsqcup(F \setminus \{d_i^{(1)} \to \ldots \to e_i\})$, and by definition $\bot_\sigma = \bigsqcup \emptyset$.

ad 2: $e \sqcup e'$ exists, because we have $(d_1 \to \ldots \to e)(d_1 \sqcup d'_1, \ldots, d_n \sqcup d'_n) = e$ and $(d'_1 \to \ldots \to e')(d_1 \sqcup d'_1, \ldots, d_n \sqcup d'_n) = e'$. Item 1 implies $e = e'$. □

We encode now the compact elements of D_σ according to theorem 5.8 to get recursive function spaces:

Definition 5.12 *Encoding of D_σ:*

$\sigma = \iota$: *Define* $\beta : \mathbf{N} \to \mathbf{N}_\perp,\ 0 \mapsto \perp,\ n+1 \mapsto n$ *and* $\beta^\iota_x := \beta(x)$.

$\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$: *If* $F'_m := \{\beta^{\sigma_1}_{(m_i)_1} \to \ldots \to \beta^{\sigma_n}_{(m_i)_n} \to \beta^\iota_{(m_i)_{n+1}} \mid m_i \in G_m\}$ *meets the conditions in previous characterization, we define* $F_m := F'_m$*, otherwise* $F_m := \emptyset$*. We set* $\beta^\sigma_m := \bigsqcup F_m$.

$\beta^\sigma : \mathbf{N} \to \hat{D}_\sigma$ is a surjective map, because every finite subset of the step functions in D_σ is represented by some G_m. However, the code is not injective, e.g. it is not only $\beta^\sigma_0 = \bigsqcup \emptyset = \perp_\sigma$, but also $\beta^\sigma_m = \perp_\sigma$ for all $m \in \mathbf{N}$ with $l(m) \le n$.

Example: Let $\sigma = (\iota, \iota)$ and $F = \{d_1 \to e_1, \ldots, d_r \to e_r\}$. Then we have

$$\beta^\iota_{(m_0)_1} = d_1, \beta^\iota_{(m_0)_2} = e_1, \ldots, \beta^\iota_{(m_{r-1})_1} = d_r, \beta^\iota_{(m_{r-1})_2} = e_r, \text{ therefore is}$$

$$m_0 = \langle d_1, e_1 \rangle, \ldots, m_{r-1} = \langle d_r, e_r \rangle \text{ and } m = \sum_{i=1}^{r} 2^{\langle d_i, e_i \rangle}.$$

For $\beta^{\iota^1}_x = 0 \to 0$ it is $x = 2^{\langle 1,1 \rangle} = 2^{2^2 \cdot 3^2}$. If it is $\beta^{\iota^1}_{10} = y$, we get $y = \perp_{\iota^1}$ because of $10 = 2^3 + 2^1$ and $1 \notin \langle \mathbf{N}, \mathbf{N} \rangle$.

Remark: The maps

$$\psi_\iota := \mathbf{N} \to \mathcal{P}_{\mathit{fin}}(\mathbf{N}),\ m \mapsto \{m\}$$

and for $\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$:

$$\psi_\sigma : \mathbf{N} \to \mathcal{P}_{\mathit{fin}}(\underbrace{\mathbf{N} \to \ldots \to \mathbf{N}}_{(n+1)-\text{times}}),\ m \mapsto [F_m] \text{ with } \bigsqcup F_m = \beta^\sigma_m,$$

whereat $[F_m]$ are the symbols out of $\{\to, 0, 1, 2, \ldots\}$ standing for F_m, are total computable, i.e. there is an algorithm computing the correct function value at each input from $\mathbf{N}$. The compact elements are represented by their code number:

- Input: $m \in \mathbf{N}$.
- Compute G_m.
- Compute $(m_i)_1, \ldots, (m_i)_{n+1}$ for all $m_i \in G_m$.
- Remove m_i with $(m_i)_{n+1} = 0$.
- Check, if there are $m_i, m_j \in G_m$ such that $(m_i)_{n+1} \ne (m_j)_{n+1}$ and for all $1 \le k \le n$ the following holds: It is $(m_i)_k = (m_j)_k$ or $\psi_{\sigma_k}((m_i)_k) \cup \psi_{\sigma_k}((m_j)_k)$ is bounded, i.e. this set is mapped to $d \ne 0$ by φ_{σ_k} (see below).
- If the result is positive, output $F_m := \emptyset$, otherwise output the set containing the elements $(m_i)_1 \to \ldots \to (m_i)_{n+1}$ for all remaining $m_i \in G_m$.

Also are the maps

$$\varphi_\iota : \mathcal{P}_{fin}(\mathbf{N}) \to \mathbf{N}, F \mapsto \begin{cases} n & , F = \{n\} \ \vee \ F = \{0, n\} \\ 0 & , \text{otherwise} \end{cases}$$

and for $\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$:

$$\varphi_\sigma : \mathcal{P}_{fin}(\underbrace{\mathbf{N}\to \ldots \to\mathbf{N}}_{(n+1)-\text{times}}) \to \mathbf{N}, \ [F'] \mapsto m \text{ with } \bigsqcup F_m = \beta^\sigma_m$$

total computable:

- Input: $[F'] \in \mathcal{P}_{fin}(\underbrace{\mathbf{N}\to \ldots \to\mathbf{N}}_{(n+1)-\text{times}})$.
- Remove all $\vec{x} \in [F']$ with $x_{n+1} = 0$.
- Check, if for all $x_1, \ldots, x_{n+1}, y_1, \ldots, y_{n+1} \in [F']$ the following holds: Whenever the set $\psi_{\sigma_k}(x_k) \cup \psi_{\sigma_k}(y_k)$ is bounded for all $1 \leq k \leq n$, it is $x_{n+1} = y_{n+1}$.
- If the result is negative, then stop and output 0, otherwise compute $(m_i)_1, \ldots, (m_i)_n$ by means of $\psi_{\sigma_1}, \ldots, \psi_{\sigma_n}$ and from that m.

With the help of the two following functions exactly all in the sense of effective cpo-s computable objects in the chosen model can be described.

Definition 5.13 *The* parallel conditional *is defined by*

$$pcond \in D_{\iota^3}, \ (x, y, z) \mapsto \begin{cases} y & , x = 0 \\ z & , x = n + 1 \ \vee \ y = z \\ \perp & , \textit{otherwise.} \end{cases}$$

A continuous approximation of the existential quantor *is defined by*

$$\exists \in D_{\iota^1 \to \iota}, \ f \mapsto \begin{cases} 0 & , \textit{ex. } x \in \mathbf{N} : \ f(x) = 0 \\ 1 & , f(\perp) = 1 \\ \perp & , \textit{otherwise.} \end{cases}$$

Obviously, both functions are not sequentially computable. *pcond* performs two tests "at the same time": the evaluation of the first argument and the comparison of the two last arguments; $\exists$ is "infinitely parallel": it tests the argument "at the same time" on all elements of $\mathbf{N}_\perp$.

With a special extension of the set of constants C the elements definable by λ-terms on $C^{PCF} \cup \{\underline{pcond}, \underline{\exists}\}$ can be defined by λ-free terms. The omitting of abstractions makes proving of computability easier.

Lemma 5.14 *For all* $M \in \Lambda(C^{PCF} \cup \{\underline{pcond}, \underline{\exists}\})$ *there is an* $N \in \Gamma((C^{PCF} \setminus \{cond\}) \cup \{\underline{pcond}, \underline{\exists}\} \cup \{\underline{K}_{\sigma_1,\sigma_2} \mid \sigma_1, \sigma_2 \in \Sigma\} \cup \{\underline{S}_{\sigma_1,\sigma_2,\sigma_3} \mid \sigma_1, \sigma_2, \sigma_3 \in \Sigma\})$ *such that* $[\![M]\!] = [\![N]\!]$, *whereat* $[\![\underline{K}_{\sigma_1,\sigma_2}]\!] := [\![\lambda x_1^{\sigma_1}, x_2^{\sigma_2}.x_1]\!]$ *and* $[\![\underline{S}_{\sigma_1,\sigma_2,\sigma_3}]\!] := [\![\lambda x_1^{(\sigma_1,\sigma_2,\sigma_3)}, x_2^{(\sigma_1,\sigma_2)}, x_3^{\sigma_1}.((x_1x_3)(x_2x_3))]\!]$ *for all* $\sigma_1, \sigma_2, \sigma_3 \in \Sigma$.

Proof: We show the assertion by structural induction:

If M is a variable, there is nothing to be shown; if M is a constant, only the case $M = cond$ has to be considered. The sequential conditional can be defined by the parallel one:

$$[\![cond]\!] = [\![\lambda x^\iota, y^\iota, z^\iota.\underline{pcond}(x, y, \underline{pcond}(x, \Omega^\iota, z))]\!].$$

In case of an application use the inductive hypothesis.

The abstraction is definable from the combinators S and K:

- If $x \notin FV(M_1)$, we have $[\![\lambda x^\sigma.M_1^\tau]\!] = [\![\underline{K}_{\tau,\sigma} M_1^\tau]\!]$.
- For $M = \lambda x^\sigma.M_1$ and $x \in FV(M_1)$ we distinguish $[\![\lambda x^\sigma.x]\!] = [\![\underline{S}_{\sigma,\sigma^1,\sigma}\underline{K}_{\sigma,\sigma^1}\underline{K}_{\sigma,\sigma}]\!]$ and $[\![\lambda x^{\sigma_1}.M_1^{\sigma_2\to\sigma_3}M_2^{\sigma_2}]\!] = [\![\underline{S}_{\sigma_1,\sigma_2,\sigma_3}(\lambda x^{\sigma_1}.M_1)(\lambda x^{\sigma_1}.M_2)]\!]$.

The elimination of abstractions in a term is done successively: substitute always an "innermost" abstraction, i.e. some whose body is a λ-free term. □

These representation is helpful for the proof of the following theorem. The proof is shown in part only, the omitted cases are left to the reader.

Theorem 5.15 *Every closed term in* $\Lambda(C^{PCF} \cup \{\underline{pcond}, \underline{\exists}\})$ *defines a computable element.*

Proof: We have to verify that all constants in $C^{PCF} \setminus \{cond\}) \cup \{\underline{pcond}, \underline{\exists}\} \cup \{\underline{K}_{\sigma_1,\sigma_2} \mid \sigma_1, \sigma_2 \in \Sigma\} \cup \{\underline{S}_{\sigma_1,\sigma_2,\sigma_3} \mid \sigma_1, \sigma_2, \sigma_3 \in \Sigma\}$ are computable and the application is closed under computability.

- $\underline{m}$ $(m \in \mathbf{N})$: $\{n \in \mathbf{N} \mid \beta_n^\iota \sqsubseteq m\} = \{0, m+1\}$ is r.e.
- $\underline{\dot{+}1}$: Let $\bigsqcup F = \beta_n^{\iota^1} \sqsubseteq [\![\underline{\dot{+}1}]\!]$. If $(\bot\to r) \in F$, it is $(\bot\to r)(r) = r \sqsubseteq [\![\underline{\dot{+}1}]\!](r)$, thus $r = \bot$. Hence, this case need not to be considered. We have $F = \{s_i\to r_i \mid 0 \le i \le k\}$ with $k \in \mathbf{N}$ and s_i pairwise different. Since $r_i = (s_i\to r_i)(s_i) \sqsubseteq [\![\underline{\dot{+}1}]\!](s_i) = s_i+1$, we get $r_i = s_i+1$, i.e. $F = \{s_i\to(s_i+1) \mid 0 \le i \le k\}$. This implies $n = \sum_{i=0}^k 2^{\langle s_i+1, s_i+2\rangle}$, and

  $$\{n \in \mathbf{N} \mid \beta_n^{\iota^1} \sqsubseteq [\![\underline{\dot{+}1}]\!]\} = \{\sum_{i=0}^k 2^{\langle s_i+1,s_i+2\rangle} \mid k \in \mathbf{N}, s_i \in \mathbf{N}, 1 \le i \le k\}$$

  is r.e.

- $\underline{pcond}$: Let $\bigsqcup F = \beta^{\iota^3} \sqsubseteq [\![\underline{pcond}]\!]$. We look for candidates for F:

  $(a{\to}b{\to}c{\to}d)(0, x, y) = \left\{ \begin{array}{ll} d & , a = 0 \ \wedge \ x \sqsupseteq b \ \wedge \ y \sqsupseteq c \\ \perp & , \text{otherwise} \end{array} \right\} \sqsubseteq x.$

  This implies $a = 0$ and $b = d$.

  $(a{\to}b{\to}c{\to}d)(n+1, x, y) = \left\{ \begin{array}{ll} d & , a = n+1 \ \wedge \ x \sqsupseteq b \ \wedge \ y \sqsupseteq c \\ \perp & , \text{otherwise} \end{array} \right\} \sqsubseteq y.$

  This implies $a = n+1$ and $c = d$.

  $(a{\to}b{\to}c{\to}d)(\perp, x, x) = \left\{ \begin{array}{ll} d & , x \sqsupseteq b \ \wedge \ x \sqsupseteq c \\ \perp & , \text{otherwise} \end{array} \right\} \sqsubseteq x.$

  This implies $b = c = d$.

  It follows for all $n \in \mathbf{N}$:

  $$\begin{aligned} &\beta_n \sqsubseteq pcond \Leftrightarrow \\ &\quad F_n \in \mathcal{P}_{fin}(\{0{\to}b{\to}c{\to}b \mid b \in \mathbf{N}, c \in \mathbf{N}_\perp\} \cup \{(n+1){\to}c{\to}b{\to}b \mid b, n \in \mathbf{N}, c \in \mathbf{N}_\perp\} \\ &\qquad\qquad \cup \{\perp{\to}b{\to}b{\to}b \mid b \in \mathbf{N}\}). \end{aligned}$$

  The test, whether F_n has the convenient form, is decidable, i.e. $\{n \in \mathbf{N} \mid \beta_n \sqsubseteq pcond\}$ is r.e.

- $\underline{\exists}$: Set $\exists_n := \bigsqcup\{(\perp{\to}1){\to}1, (0{\to}0){\to}0, \ldots, (n{\to}0){\to}0\} \in \hat{D}_{(\iota^1,\iota)}$ for all $n \in \mathbf{N}$. Then $\exists_n \sqsubseteq \exists_{n+1}$ for all $n \in \mathbf{N}$ and $\exists = \bigsqcup_{n \in \mathbf{N}} \exists_n$. As we have $\beta_{\iota^1}(\perp{\to}1) = 2^{\langle 0,2\rangle}$ and $\beta_{\iota^1}(n{\to}0) = 2^{\langle n+1,1\rangle}$, for $\beta_{k(n)}^{(\iota^1,\iota)} := \exists_n$ the function

  $$k : \mathbf{N} \to \mathbf{N},\ n \mapsto 2^{\langle 2^{\langle 0,2\rangle},2\rangle} + \sum_{i=0}^{n} 2^{\langle 2^{\langle i+1,1\rangle},1\rangle}$$

  is μ-recursive, i.e. $\exists = \bigsqcup\{\beta_{k(n)}^{(\iota^1,\iota)} \mid n \in \mathbf{N}\}$ is computable.

- Y_σ:

  $[\![Y_\sigma]\!] = \bigsqcup\{\Big((\beta_0^\sigma{\to}\beta_{n[1]}^\sigma) \sqcup \ldots \sqcup (\beta_{n[k]}^\sigma{\to}\beta_{n[k+1]}^\sigma)\Big){\to}\beta_{n[k+1]}^\sigma \mid k \geq 1, n[1], \ldots, n[k+1] \in \mathbf{N}\}.$

  Proof: Set $[\![Y_\sigma^{(m)}]\!] \in D_{(\sigma^1,\sigma)} :\ f \mapsto f^m(\perp_\sigma)$ for all $m \in \mathbf{N}$. Then we have $[\![Y_\sigma]\!] = \bigsqcup_m [\![Y_\sigma^{(m)}]\!]$ and it remains to show that

  $$[\![Y_\sigma^{(m)}]\!] = \bigsqcup\{\Big((\beta_0^\sigma{\to}\beta_{n[1]}^\sigma) \sqcup \ldots \sqcup (\beta_{n[m-1]}^\sigma{\to}\beta_{n[m]}^\sigma)\Big){\to}\beta_{n[m]}^\sigma \mid n[1], \ldots, n[m] \in \mathbf{N}\}.$$

  "$\sqsupseteq$": $\Big((\beta_0^\sigma{\to}\beta_{n[1]}^\sigma) \sqcup \ldots \sqcup (\beta_{n[m-1]}^\sigma{\to}\beta_{n[m]}^\sigma)\Big)(f)$

  $$= \left\{ \begin{array}{ll} \beta_{n[m]}^\sigma & , f(\beta_0^\sigma) \sqsupseteq \beta_{n[1]}^\sigma \ \wedge \ \ldots \ \wedge \ f(\beta_{n[m-1]}^\sigma) \sqsupseteq \beta_{n[m]}^\sigma \quad [*] \\ \perp & , \text{otherwise.} \end{array} \right.$$

  In the case [*] we have $[\![Y_\sigma^{(m)}]\!](f) = f^m(\beta_0^\sigma) \sqsupseteq \beta_{n[m]}^\sigma$.

"$\sqsubseteq$":

$$\begin{aligned}
f^m(\beta_0^\sigma) = \; & \bigsqcup\{f(x_{m-1}) \mid x_1,\ldots,x_{m-1}\in\hat{D}_\sigma, \\
& \qquad\qquad x_1\sqsubseteq f(\beta_0^\sigma), x_2\sqsubseteq f(x_1),\ldots,x_{m-1}\sqsubseteq f(x_{m-2})\} \\
& (D_\sigma \text{ algebraic and } f \text{ continuous}) \\
= \; & \bigsqcup\{\Big((\beta_0^\sigma\to x_1)\sqcup\ldots\sqcup(x_{m-1}\to x_m)\Big)\to x_m \\
& \qquad \mid x_1,\ldots,x_m\in\hat{D}_\sigma, x_1\sqsubseteq f(\beta_0^\sigma),\ldots,x_m\sqsubseteq f(x_{m-1})\}(f) \\
\sqsubseteq \; & \bigsqcup\{\Big((\beta_0^\sigma\to\beta_{n[1]}^\sigma)\sqcup\ldots\sqcup(\beta_{n[m-1]}^\sigma\to\beta_{n[m]}^\sigma)\Big)\to\beta_{n[m]}^\sigma \mid n[1],\ldots,n[m]\in\mathbf{N}\}.
\end{aligned}$$

For all bounded $B\subseteq(\hat{D}_{\sigma_1}\to\ldots\to\hat{D}_{\sigma_n})$ we have

$$\bigsqcup B = \bigsqcup\{\bigsqcup F \mid F\in\mathcal{P}_{fin}(B)\} =: \bigsqcup\tilde{B},$$

whereat $\tilde{B}$ is directed. Here we define:

$$\begin{aligned}
\tilde{B} := \beta(\{x \mid (x_i)_1 = 2\cdot 3^{n[1]+1} + \textstyle\sum_{r=1}^{k} 2^{2^{n[r]+1}\cdot 3^{n[r+1]+1}}, (x_i)_2 = n[k]+1, \\
k, n[1],\ldots,n[k+1]\in\mathbf{N}, x_i\in G_x\}) =: \beta(A)
\end{aligned}$$

with A r.e.

- Application: We show that the computability of $f\in D_{\sigma\to\tau}$ and $x\in D_\sigma$ implies the computability of $f(x)\in D_\tau$. By assumption $A := \{n\in\mathbf{N} \mid \beta_n^\sigma\sqsubseteq x\}$ and $B := \{m\in\mathbf{N} \mid \beta_m^{\sigma\to\tau}\sqsubseteq f\}$ are r.e. We have $f(x) = \bigsqcup\{e\in\hat{D}_\tau \mid e\sqsubseteq f(x)\}$ and $\bigsqcup\{e\in\hat{D}_\tau \mid e\sqsubseteq f(x)\} = \{e\in\hat{D}_\tau \mid (d\to e)\sqsubseteq f, d\sqsubseteq x, d\in\hat{D}_\sigma\}$, so we get

$$\{k\in\mathbf{N} \mid \beta_k^\tau\sqsubseteq f(x)\} = \{k\in\mathbf{N} \mid ex.\ m\in B, m_i\in G_m:\ k=(m_i)_2 \ \wedge\ (m_i)_1\in A\}.$$

This set is r.e. □

For strict functions of order 1, i.e. such that map tuples of arguments containing $\perp$ to $\perp$, common computability and PCF-definability coincide:

Theorem 5.16 *For all $n\geq 1$ a function $f:\ \mathbf{N}^n \rightharpoonup \mathbf{N}$ is μ-recursive if, and only if, $cu(\overline{f})$ is PCF-definable.*

Here $\overline{f}$ is the strict expansion of f as defined in theorem 2.23. With the help of [Wei80] for a function $f\in\mathbf{N}\rightharpoonup\mathbf{N}$ one has the equivalence:

$f\in F_\mu^{(1)}$ $\Leftrightarrow \overline{f}\in[\mathbf{N}_\perp\to\mathbf{N}_\perp]$ is a cpo-computable function.
$\Leftrightarrow \overline{f}$ is PCF-definable.

Using Cantor's numbering the general case is obtained by:

Let

$$\varphi_2:\mathbf{N}^2\to\mathbf{N},\ (x,y)\mapsto\sum\{i \mid i\leq x+y\}+y = \frac{1}{2}(x^2+2xy+y^2+x+3y)$$

and for all $n \geq 3$

$$\varphi_n : \mathbf{N}^n \to \mathbf{N},\ (x_1, \ldots, x_n) \mapsto \varphi_2(\varphi_{n-1}(x_1, \ldots, x_{n-1}), x_n).$$

Hence, $\varphi_n \in F_\mu^{(n)}$ is bijective for all $n \geq 2$ and the coordinate functions of φ_n^{-1} are μ-recursive. Defining $f^* := f \circ \varphi_n^{-1}$ we get the commutative diagram 1.

$$\begin{tikzcd}
\mathbf{N}_\perp^n \arrow[rrrr, "\overline{\varphi_n}"] \arrow[rrddd, "\overline{f}"'] & & & & \mathbf{N}_\perp \arrow[llddd, "\overline{f^*}"] \\
& \mathbf{N}^n \arrow[rr, "\varphi_n"] \arrow[dr, "f"'] & \arrow[d, phantom, "///"] & \mathbf{N} \arrow[dl, "f^*"] & \\
& & \mathbf{N} & & \\
& & \mathbf{N}_\perp & &
\end{tikzcd}$$

diagram 1

For any $f : \mathbf{N}^n \rightharpoonup \mathbf{N}$ it can be shown:

$f \in F_\mu^{(n)} \Leftrightarrow cu(\overline{f}) \in [D_\iota \to D_{\iota^{n-1}}]$ is a cpo-computable function.
$\Leftrightarrow cu(\overline{f}) \in D_{\iota^n}$ is definable.

The following theorem is the main result in this section.

Theorem 5.17 *Exactly all cpo-computable functions in $\bigcup_{\sigma \in \Sigma} D_\sigma$ are definable from PCF extended with $\underline{\exists}$ and $\underline{pcond}$.*

By theorem 5.15 it remains to verify that every cpo-computable function is definable from the extended PCF. The very technical proof can be found in [Plo77]. The term constructions make use of the fact that for all $\sigma \in \Sigma$ the predicate

$$\not\uparrow^\sigma \subseteq D_\sigma \times \hat{D}_\sigma \text{ with } x \not\uparrow^\sigma y :\Leftrightarrow \{x, y\} \text{ has no upper bound}$$

is expressible in that very extension of PCF. To be more precise, for all $\sigma \in \Sigma$ there are closed terms $\underline{\not\uparrow^\sigma}$ of type (ι, σ, ι), so that for all $n \in \mathbf{N}$

$$[\![\underline{\not\uparrow^\sigma}(\underline{n})]\!] = \not\uparrow_n^\sigma :\ D_\sigma \to D_\iota,\ x \mapsto \begin{cases} 0 & , x \not\uparrow^\sigma \beta_n^\sigma \\ 1 & , x \sqsupseteq \beta_n^\sigma \\ \perp & ,\text{sonst.} \end{cases}$$

holds. In case of $\sigma = \iota$ this term uses only PCF-constructions.

To define elements of order 1 the constant $\underline{\exists}$ is not required, the parallel conditional is sufficient already. Therefore, we get

Corollary 5.18 *A function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ of order 1 is cpo-computable if, and only if, f is definable from the extension of PCF with $\underline{pcond}$.*

Expressing theorem 5.17 and corollary 5.18 by means of degrees one has

Corollary 5.19 *1. $f \leq_p pcond \oplus \exists$. for all cpo-computable functions $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$*

2. $f \leq_p pcond$ for all cpo-computable functions $f \in \bigcup_{n\in\mathbf{N}} D_{\iota^n}$.

□

6 The Algorithm of Stoughton

We have shown how to prove $f \leq_p g$, resp. $f \not\leq_p g$, when two functions $f, g \in \bigcup_{\sigma\in\Sigma} D_\sigma$ are given. Is this decidable in general? We will now introduce a decision algorithm, which supposes that the basic domain is finite, $ord(f) \leq 2$ and $ord(g) \leq 1$. The following notation will be used below: For any index set I the application of $x \in D^I_{\sigma\to\tau}$ to $y \in D^I_\sigma$ is the I-tuple $x(y) := (x_i(y_i))_{i\in I} \in D^I_\tau$. An element $x \in D_\sigma$ is identified with the constant I-tuple $(x)_{i\in I} \in D^I_\sigma$. We get a measure for the construction of λ-free terms by

Definition 6.1 *The* depth *dp of a λ-free term $M \in \Gamma(C)$ is given by:*

- *For all $c \in C$ and $x^\sigma_i \in V$ set $dp(c) := dp(x^\sigma_i) := 0$.*
- *For all $M, N \in \Gamma(C)$ set $dp(MN) := \max\{dp(M), dp(N)+1\}$.*

Example: Let $d \in C_{(\sigma_1,\ldots,\sigma_n,\iota)} \cup V_{(\sigma_1,\ldots,\sigma_n,\iota)}$ and $M_i \in \Gamma(C)_{\sigma_i}$ for all $1 \leq i \leq n$. Then we have

$$\begin{aligned} dp(d(M_1,\ldots,M_n)) &= \max\{dp(d), dp(M_1)+1,\ldots,dp(M_n)+1\} \\ &= \max\{dp(M_1),\ldots,dp(M_n)\}+1. \end{aligned}$$

For arbitrary functions f, g with restricted order as above the following algorithm specifies a proof for $f \leq_p g$, resp. $f \not\leq_p g$:

Input:	D_ι:	finite flat cpo as basic domain
	C:	finite set of constants such that $ord(C) \leq 1$ and $g \in C$
	$[\![C]\!]$:	semantics of C
	$m \in \mathbf{N}$:	arity of f
	$p \in \mathbf{N}$:	number of assignments characterizing the function f

$\sigma_1, \ldots, \sigma_m$: types of the arguments of f, i.e. $f \in D_{(\sigma_1,\ldots,\sigma_m,\iota)}$,
whereat $ord(\sigma_i) \leq 1$ for all $1 \leq i \leq m$.

test characterizing f:
$\Delta = (\Delta_i \in D^p_{\sigma_i} \mid 1 \leq i \leq m)$: argument tuples of the assignments
$X \in D^p_\iota$: value tuples of the assignments,
i.e. $f(\Delta) = f(\Delta_1, \ldots, \Delta_m) = (f(\Delta_{1j}, \ldots, \Delta_{mj}))_{1 \leq j \leq p} = X$

begin
$k := 0$
$S := \{([\![c]\!], c) \mid c \in C_\iota\} \cup \{(\Delta_i, x_i^{\sigma_i}) \mid f.a.\ 1 \leq i \leq m:\ \sigma_i = \iota\} \subseteq D^p_\iota \times \Gamma(C)_\iota$
pos.stop:=false
neg.stop:=false
while (not pos.stop) and (not neg.stop)
  { **if** $(X, M) \in S$
    then pos.stop:=true
    else { $Z := \{([\![c]\!](y_1, \ldots, y_n), c(M_1, \ldots, M_n)) \mid$
      $c \in \bigcup_{n \geq 1} C_{\iota^n}, f.a.\ 1 \leq j \leq n : (y_j, M_j) \in S\}$
      $\cup \{(\Delta_i(y_1, \ldots, y_{ar(\sigma_i)}), x_i(M_1, \ldots, M_{ar(\sigma_i)}) \mid$
      $f.a.\ 1 \leq i \leq m : ar(\sigma_i) \geq 1, f.a.\ 1 \leq j \leq n : (y_j, M_j) \in S\}$
      $S' := Z|_{\pi_1(Z) \setminus \pi_1(S)}$
      if $S' = \emptyset$
        then neg.stop:=true
        else { $k := k + 1$
          $S := S \cup S'$
  }   }   }
if pos.stop **then** output:= k and $\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.M$
if neg.stop **then** output:= k and $\pi_1(S)$
end.

Remark: S is the set of all pairs in $D^p_\iota \times \Gamma(C)_\iota$ with depth $\leq k$, being candidates for meeting the test, Z is the set of the appropriate pairs with depth $k+1$. S' is the set of all pairs in Z having a value tuple which has not appeared in S yet. If S' is empty, then as well after passing through the **while**-loop once more. In this case the algorithm stops with negative result. A stop with positive result happens, if a term M is found such that $(X, M) \in S$. To be more precise, we have

Theorem 6.2 *The previous algorithm terminates always.*

If the output consists of the term $\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.M$, the conditions below are satisfied:

- $FV(M) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$, $M \in \Gamma(C)_\iota$ *and* $dp(M) = k$

- *Let $\mathcal{E}$ be a type frame with $E_\iota = D_\iota$, $[\![c]\!]_{\mathcal{E}} = [\![c]\!]_{\mathcal{D}}$ for all $c \in C$ and $\Delta_i \in E^p_{\sigma_i}$ for all $1 \leq i \leq m$. Then we have $[\![\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X$.*
- *For all $N \in \Gamma(C)_\iota$ with $[\![\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.N]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X$ and $FV(N) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$ we have $dp(M) \leq dp(N)$.*

If the output is $\pi_1(S) \subseteq D^p_\iota$, one has:

- *$X \notin \pi_1(S)$*
- *Let $\mathcal{E}$ be a type frame with $E_\iota = D_\iota$, $[\![c]\!]_{\mathcal{E}} = [\![c]\!]_{\mathcal{D}}$ for all $c \in C$ and $\Delta_i \in E^p_{\sigma_i}$ for all $1 \leq i \leq m$. Then there is no function $e \in E_{(\sigma_1, \ldots, \sigma_m, \iota)}$ definable in $\Lambda(C)$ such that $e(\Delta_1, \ldots, \Delta_m) = X$.*

The previous theorem says that the output is a closed term in $\Lambda(C)_\iota$, whose meaning matches the test conditions in a suitable type frame if, and only if, there is existing some. Moreover, this term has minimal depth. The proof is given in several steps.

Lemma 3.2 implies directly

Lemma 6.3 *Let $m, p \in \mathbf{N}$, $\mathcal{E}$ a type frame for $\Lambda(C)$ with $\Delta_{\sigma_i} \in E^p_{\sigma_i}$ for $1 \leq i \leq m$, $X \in E^p_\iota$ and R a p-ary logical relation on $\mathcal{E}$. If $\Delta_i \in R^{\sigma_i}$ for all $1 \leq i \leq m$, but $X \notin R^\iota$, then it does not exist a closed PCF-term $M^{(\sigma_1, \ldots, \sigma_m, \iota)}$ such that $[\![M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X$.* $\square$

In the basic domain the logical relation $R(\Delta)$ below includes exactly the value tuples which are definable by $\Lambda(C)$-terms of convenient type using the tuples in Δ as arguments.

Definition 6.4 *Let $\mathcal{E}$ and $\Delta = (\Delta_i \mid 1 \leq i \leq m)$ be as in lemma 6.3, $ord(C) \leq 1$ and $ord(\sigma_i) \leq 1$ for all $1 \leq i \leq m$. Then $R(\Delta)$ is the p-ary logical relation on $\mathcal{E}$ such that $X \in R(\Delta)^\iota$ if, and only if, $f(\Delta_1, \ldots, \Delta_m) = X$ fore some function $f \in E_{(\sigma_1, \ldots, \sigma_m, \iota)}$ definable in $\Lambda(C)$.*

$R(\Delta)$ is well-defined by

Lemma 6.5 *$R(\Delta)$ is a logical relation on $\mathcal{E}$ satisfying $\Delta_i \in R(\Delta)^{\sigma_i}$ for all $1 \leq i \leq m$.*

Proof: For arbitrary $c \in C_{\iota^n}$ and $y_1, \ldots, y_n \in R(\Delta)^\iota$ there are closed $\Lambda(C)$-terms $M_1, \ldots, M_n$ of type $(\sigma_1, \ldots, \sigma_m, \iota)$ such that $[\![M_j]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = y_j$ for all $1 \leq j \leq n$. Defining

$$M := \lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.c(M_1(x_1, \ldots, x_m), \ldots, M_n(x_1, \ldots, x_m))$$

we get $[\![M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = [\![c]\!]_{\mathcal{E}}(y_1, \ldots, y_n)$, i.e. $[\![c]\!]_{\mathcal{E}}(y_1, \ldots, y_n) \in R(\Delta)^\iota$ and $[\![c]\!]_{\mathcal{E}} \in R(\Delta)^{\iota^n}$. Thus, $R(\Delta)$ is a logical relation on $\mathcal{E}$.

For arbitrary $\sigma_i = \iota^{n_i}$ and $y_i, \ldots, y_{n_i} \in R(\Delta)^\iota$ there are closed $\Lambda(C)$-terms $M_1, \ldots, M_{n_i}$ of type $(\sigma_1, \ldots, \sigma_m, \iota)$ such that $[\![M_j]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = y_j$ for all $1 \leq j \leq n_i$. Defining

$$N_i := \lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.x_i(M_1(x_1, \ldots, x_m), \ldots, M_{n_i}(x_1, \ldots, x_m))$$

one has $[\![N_i]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = \Delta_i(y_1, \ldots, y_{n_i}) \in R(\Delta)^\iota$ and therefore $\Delta_i \in R(\Delta)^{\sigma_i}$. □

We obtain the fact that there is a function meeting all test conditions if, and only if, the invariance of the argument tuples in Δ under a logical relation always implies the invariance of the value tuple X:

Theorem 6.6 *Giving the assumptions in definition 6.4 and $X \in E_\iota^p$ the following conditions are equivalent:*

1. *There is a $\Lambda(C)$-definable function $f \in E_{(\sigma_1, \ldots, \sigma_m, \iota)}$ such that $f(\Delta_1, \ldots, \Delta_m) = X$.*
2. *For all p-ary logical relations R on $\mathcal{E}$ with $\Delta_i \in R^{\sigma_i}$ for all $1 \leq i \leq m$ it is $X \in R^\iota$.*

Proof: "$\Rightarrow$": Contraposition of lemma 6.3. "$\Leftarrow$": Lemma 6.5 yields $\Delta_i \in R(\Delta)^{\sigma_i}$ for all $1 \leq i \leq m$, thus, we have $X \in R(\Delta)^\iota$. The definition of $R(\Delta)$ implies the assertion. □

We introduce now another logical relation, which is proved to be a further characterizing of $R(\Delta)$.

Definition 6.7 *Let the assumptions in definition 6.4 be satisfied. Then $L(\Delta)$ is the p-ary logical relation on $\mathcal{E}$ such that $L(\Delta)^\iota$ is the least p-ary relation on D_ι closed under $[\![c]\!]_{\mathcal{E}}$ for all $c \in C$ and under Δ_i for all $1 \leq i \leq m$, i.e.*

- *For all $c \in \bigcup_{n \in \mathbf{N}} C_{\iota^n}$ we have*

$$(f.a.\ 1 \leq j \leq n:\ (X_{l,j})_{1 \leq l \leq p} \in L(\Delta)^\iota) \Rightarrow ([\![c]\!]_{\mathcal{E}}(X_{l,1}, \ldots, X_{l,n}))_{1 \leq l \leq p} \in L(\Delta)^\iota.$$

- *For all $1 \leq i \leq m$ and $\sigma_i = \iota^{n_i}$ we have*

$$(f.a.\ 1 \leq j \leq n_i:\ (X_{l,j})_{1 \leq l \leq p} \in L(\Delta)^\iota) \Rightarrow (\Delta_i(X_{l,1}, \ldots, X_{l,n_i}))_{1 \leq l \leq p} \in L(\Delta)^\iota.$$

Lemma 6.8 *Let the assumptions be as in definition 6.4 and a set of variables $\{x_i \in V_{\sigma_i} \mid 1 \leq i \leq m\}$ given, whereat $\Delta_i \neq \Delta_j \Rightarrow x_i^{\sigma_i} \neq x_j^{\sigma_j}$.*

1. *Let $c \in C_{\iota^n}$, $y_1, \ldots, y_n \in D_\iota^p$ and $M_1, \ldots, M_n \in \Gamma(C)_\iota$ be such that $FV(M_j) \subseteq \{x_1, \ldots, x_m\}$ and $[\![\lambda x_1, \ldots, x_m.M_j]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = y_j$ for all $1 \leq j \leq n$. Then it is $M := c(M_1, \ldots, M_n) \in \Gamma(C)_\iota$ with $FV(M) \subseteq \{x_1, \ldots, x_m\}$ and $[\![\lambda x_1, \ldots, x_m.M]\!]_{\mathcal{E}}$ $(\Delta_1, \ldots, \Delta_m) = [\![c]\!]_{\mathcal{E}}(y_1, \ldots, y_n)$.*

2. *Let* $y_1, \ldots, y_{ar(\sigma_i)} \in D_\iota^p$ *and* $M_1, \ldots, M_{ar(\sigma_i)} \in \Gamma(C)_\iota$ *for some* $1 \le i \le m$ *such that* $FV(M_j) \subseteq \{x_1, \ldots, x_m\}$ *and* $[\![\lambda x_1, \ldots, x_m.M_j]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = y_j$ *for all* $1 \le j \le ar(\sigma_i)$*. Then it is* $M := x_i(M_1, \ldots, M_{ar(\sigma_i)}) \in \Gamma(C)_\iota$ *with* $FV(M) \subseteq \{x_1, \ldots, x_m\}$ *and* $[\![\lambda x_1, \ldots, x_m.M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = \Delta_i(y_1, \ldots, y_{ar(\sigma_i)})$*.*

3. *For all* $X \in L(\Delta)^\iota$ *it exists a term* $M \in \Gamma(C)_\iota$ *such that* $FV(M) \subseteq \{x_1, \ldots, x_m\}$ *and* $[\![\lambda x_1, \ldots, x_m.M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X$*.*

Proof:

1. For all $\rho \in Env$ we have

$$[\![\lambda \vec{x}.M]\!]_{\mathcal{E}}(\rho)(\Delta) = [\![c]\!]_{\mathcal{E}}\Big([\![M_1]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}]), \ldots, [\![M_n]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}])\Big) = [\![c]\!]_{\mathcal{E}}(y_1, \ldots, y_n).$$

2. For all $\rho \in Env$ we have also

$$\begin{aligned}[\![\lambda \vec{x}.M]\!]_{\mathcal{E}}(\rho)(\Delta) &= [\![x_i]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}])\Big([\![M_1]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}]), \ldots, [\![M_{ar(\sigma_i)}]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}])\Big)\\ &= \Delta_i(y_1, \ldots, y_{ar(\sigma_i)}).\end{aligned}$$

3. Structural induction on $L(\Delta)^\iota$:

   - Let $c \in \bigcup_{n \in \mathbb{N}} C_{\iota^n}$, $X = ([\![c]\!]_{\mathcal{E}}(X_{l,1}, \ldots, X_{l,n}))_{1 \le l \le p}$ and $(X_{l,j})_{1 \le l \le p} \in L(\Delta)^\iota$ for all $1 \le j \le n$. By inductive hypothesis one has $M_j \in \Gamma(C)_\iota$for all $1 \le j \le n$ such that $FV(M_j) \subseteq \{x_1, \ldots, x_m\}$ and $[\![\lambda x_1, \ldots, x_m.M_j]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X_j$. Define $M := c(M_1, \ldots, M_n)$. This implies for all $\rho \in Env$:

   $$\begin{aligned}[\![\lambda \vec{x}.M]\!]_{\mathcal{E}}(\rho)(\Delta) &= [\![c]\!]_{\mathcal{E}}([\![M_1]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}]), \ldots, [\![M_n]\!]_{\mathcal{E}}(\rho[\Delta/\vec{x}]))\\ &= [\![c]\!]_{\mathcal{E}}(X_1, \ldots, X_n) = X.\end{aligned}$$

   - Let $X = (\Delta_i(X_{l,1}, \ldots, X_{l,ar(\sigma_i)}))_{1 \le l \le p}$ for some $1 \le i \le m$ with $(X_{l,j})_{1 \le l \le p} \in L(\Delta)^\iota$ for all $1 \le j \le ar(\sigma_i)$. By inductive hypothesis there is a term $M_j \in \Gamma(C)_\iota$ such that $FV(M_j) \subseteq \{x_1, \ldots, x_m\}$ and $[\![\lambda x_1, \ldots, x_m.M_j]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X_j$ for all $1 \le j \le ar(\sigma_i)$. Analogous to the previous case we define

   $$M := x_i(M_1, \ldots, M_{ar(\sigma_i)}). \qquad \Box$$

Lemma 6.9 *With the assumptions as in definition 6.4 the equation* $L(\Delta) = R(\Delta)$ *holds.*

Proof: With $L(\Delta)$ and $R(\Delta)$ being logical relations, we have to show $L(\Delta)^\iota = R(\Delta)^\iota$, only.

"$\subseteq$": This direction is directly concluded from lemma 6.8 item 3.

"$\supseteq$": Definition 6.7 implies $\Delta_i \in L(\Delta)^{\sigma_i}$ for all $1 \le i \le m$. For any $X \in E_\iota^p$ with $X \notin L(\Delta)^\iota$ we obtain $X \notin R(\Delta)^\iota$ by lemma 6.3 and definition 6.4. $\Box$

Proof of Theorem 6.2:

For any $r \in \mathbf{N}$ let S_r be the set S, when the algorithm is in the state $k = r$. Let $\mathcal{E}$ be a type frame satisfying the conditions in 6.2. We will first verify the following statements:

1. If S_r is defined, then $S_r \subseteq D_\iota^p \times \Gamma(C)_\iota$ is functional.

2. If S_{r+1} is defined, then it is $S_r \subset S_{r+1}$.

3. If S_r is defined, $(Y, M) \in S_r$ and either $r = 0$ or $Y \notin \pi_1(S_{r-1})$, then $dp(M) = r$.

4. If S_r is defined, then it is $\pi_1(S_r) \subseteq L(\Delta)^\iota$.

5. If S_r is defined and $(Y, M) \in S_r$, then it is $FV(M) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$ and
$$[\![\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = Y.$$

6. For all $M \in \Gamma(C)_\iota$ such that $FV(M) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$ and $dp(M) = r$ one has
$$[\![\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) \in \pi_1(S_r).$$

ad 1: $S_r \subseteq D_\iota^p \times \Gamma(C)_\iota$ is obviously by construction. W.l.o.g. there are no two constants with the same semantics. Furthermore, no p-tuple Δ_i is a constant tuple $[\![c]\!]_{\mathcal{E}}$ for some $c \in C$, because in that case the correspondent $x_i^{\sigma_i}$ in the considered terms can be replaced by c. If the tuples Δ_i, Δ_j are the same, one can also assign the same variable. Hence, S_0 can be assumed functional. If S_r is functional, then also is S_{r+1} because of $\pi_1(S_r) \cap \pi_1(S'_r) = \emptyset$.

ad 2: Obviously, because S_{r+1} is constructed, only if $S'_r \neq \emptyset$.

ad 3: The case $r = 0$ is clear. Let $r \geq 1$ and $Y \notin \pi_1(S_{r-1})$. Then we have $Y \in \pi_1(S'_{r-1})$. We just consider the case $M = c(M_1, \ldots, M_n)$ with $(Y_j, M_j) \in S_{r-1}$ for all $1 \leq j \leq n$. Item 2 implies
$$f.a.\ 1 \leq j \leq n:\ ex.\ p_j < r:\ Y_j \in \pi_1(S_{p_j})\ \wedge\ (p_j = 0\ \vee\ Y_j \notin \pi_1(S_{p_j - 1})).$$
By inductive hypothesis we get $dp(M_j) = p_j$ for all $1 \leq j \leq n$. Therefore, it is $dp(M) = 1 + \max\{dp(M_j) \mid 1 \leq j \leq n\} < r + 1$. There is a j with $p_j = r - 1$, otherwise it is $Y \in \pi_1(S_{r-1})$. We obtain $dp(M) = r$. The case $M = x_i(M_1, \ldots, M_{ar(\sigma_i)})$ is proved analogous.

ad 4: Evident from construction of $L(\Delta)^\iota$.

ad 5: Clearly we have $FV(M) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$. For $r = 0$ the equations $[\![\lambda\vec{x}.c]\!]_{\mathcal{E}}(\Delta) = [\![c]\!]_{\mathcal{E}}$ and $[\![\lambda\vec{x}.x_i]\!]_{\mathcal{E}}(\Delta) = \Delta_i$ hold. For the step $r \to r+1$ apply lemma 6.8 item 1, resp. item 2 to $(Y, M) \in S'_r$.

ad 6: For $r = 0$ it is $M \in \{x_i^{\sigma_i} \mid 1 \leq i \leq n\} \cup C_\iota$, hence $[\![\lambda\vec{x}.M]\!]_{\mathcal{E}}(\Delta) \in S_0$. We assume that the statement is shown for r and we have $dp(M) = r + 1$. $M \in \Gamma(C)_\iota$ and $FV(M) \subseteq$

$\{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$ implies $M = d(M_1, \ldots, M_n)$ with $d \in C \cup \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$. Because of $r+1 = 1 + \max\{dp(M_j) \mid 1 \leq j \leq n\}$ the inductive hypothesis yields $[\![\lambda\vec{x}.M_j]\!]_{\mathcal{E}}(\Delta) \in \pi_1(S_r)$. With the help of lemma 6.8 items 1 and 2 we get the last statement.

The algorithm terminates because of $\sharp D_\iota^p < \infty$ and the items 1 and 2, i.e. $\tilde{r} := \max\{r \in \mathbf{N} \mid ex.\ S_r\}$ is well-defined. If the output consists of the term $\lambda\vec{x}.M$, then there is a unique $(X, M) \in S_{\tilde{r}}$. Item 1 implies $M \in \Gamma(C)_\iota$, item 3 implies $dp(M) = \tilde{r} = k$ and with item 5 we get $FV(M) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$ and $[\![\lambda x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}.M]\!]_{\mathcal{E}}(\Delta_1, \ldots, \Delta_m) = X$. Let $N \in \Gamma(C)_\iota$ such that $FV(N) \subseteq \{x_1^{\sigma_1}, \ldots, x_m^{\sigma_m}\}$ and $[\![\lambda\vec{x}.N]\!]_{\mathcal{E}}(\Delta) = X$. We obtain $dp(N) \geq dp(M)$, since otherwise item 6 implies $X \in \pi_1(S_r)$ with $r < \tilde{r}$, i.e. the procedure stops before constructing $S_{\tilde{r}}$. If the output is $\pi_1(S_{\tilde{r}}) \subseteq D_\iota^p$, one has $X \notin \pi_1(S_{\tilde{r}})$ and item 4 implies $\pi_1(S_{\tilde{r}}) \subseteq L(\Delta)^\iota$. As $\tilde{r}$ is maximal, the definition of $L(\Delta)^\iota$ yields $\pi_1(S_{\tilde{r}}) = L(\Delta)^\iota = R(\Delta)^\iota$, i.e. $e(\Delta) = X$ for no function e definable in $\Lambda(C)$. □

Considering PCF with a finite basic domain (finitary PCF,FPCF), we obtain herewith a helpful procedure for analyzing the *dops*. Define $C := \{\underline{0}, \ldots, \underline{n}, \Omega, cond\}$. The constants $\dot{+}\underline{1}$ and $\dot{-}\underline{1}$ can be omitted. The fixed point operators Y_σ too are unnecessary because of

$$\begin{aligned}[\![Y_\sigma(M^{\sigma^1})]\!] &= \textstyle\bigsqcup_{n\in\mathbf{N}}[\![M]\!]^n(\bot_\sigma)\\ &= [\![M]\!]^{n_0}(\bot_\sigma)\\ &\quad (n_0 \in \mathbf{N} \text{ chosen conveniently})\\ &= [\![\underbrace{M(M\ldots(M}_{n_0-\text{times}}\,\Omega^\sigma)\ldots)]\!].\end{aligned}$$

The element $\bot_\iota$ is represented explicitly as the constant Ω.

Example 6.10 $D_\iota := \mathbf{B}_\bot, C := \{\underline{0}, \underline{1}, \Omega, cond\}, m := 2, p := 3, \sigma_1 := \sigma_2 := \sigma_3 := \iota$

1. $\Delta_1 := \begin{pmatrix} 0 \\ \bot \\ 1 \end{pmatrix}, \Delta_2 := \begin{pmatrix} \bot \\ 0 \\ 1 \end{pmatrix}, X := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$

$k := 0$

$S_0 := \{(\bot, \Omega), (0, \underline{0}), (1, \underline{1}), ((0, \bot, 1), x_1^\iota), ((\bot, 0, 1), x_2^\iota)\} \subseteq \mathbf{B}_\bot^3 \times \Gamma(C)_\iota$

$X \notin \pi_1(S_0)$

$$\begin{aligned}Z_0 := \{\ldots,\ &([\![cond]\!]((\bot,0,1),(\bot,0,1),(\bot,0,1)), cond(x_1^\iota, x_1^\iota, x_1^\iota)),\\ &([\![cond]\!]((0,\bot,1),(\bot,0,1),\bot), cond(x_1^\iota, x_2^\iota, \Omega)), \ldots\}\end{aligned}$$

$$\begin{aligned}\pi_1(Z_0) = \{\ &(\bot,\bot,\bot), (0,0,0), (1,1,1), (0,\bot,1), (\bot,0,1),\\ &(\bot,\bot,0), (\bot,\bot,1), (\bot,0,\bot), (\bot,0,0), (\bot,1,\bot),\\ &(\bot,1,0), (\bot,1,1), (0,\bot,\bot), (0,\bot,0), (1,\bot,\bot),\\ &(1,\bot,0), (1,\bot,1)\}\end{aligned}$$

$S'_0 := Z_0 \mid_{\pi_1(Z_0)\setminus\pi_1(S_0)} \neq \emptyset$
$k := 1$
$S_1 := S_0 \cup S'_0$
$X \notin \pi_1(S_1)$
$Z_1 := \{\ldots\} = Z_0$
$S'_1 := Z_1 \mid_{\pi_1(Z_1)\setminus\pi_1(S_1)} = \emptyset$
neg.stop:=true
output:=1 and $\pi_1(S_1) = \pi_1(Z_0) = (1,2 \mid 3)$

2. $\Delta_1 := \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \Delta_2 := \begin{pmatrix} \perp \\ 0 \\ 1 \end{pmatrix}, X := \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$
$k := 0$
$S_0 := \{(\perp, \Omega), (0, \underline{0}), (1, \underline{1}), ((0,1,1), x_1^\iota), ((\perp,0,1), x_2^\iota)\} \subseteq \mathbf{B}_\perp^3 \times \Gamma(C)_\iota$
$X \notin \pi_1(S_0)$

$$\begin{aligned} Z_0 := \{\ldots, &(\llbracket cond \rrbracket((\perp,0,1),(\perp,0,1),(\perp,0,1)), cond(x_1^\iota, x_1^\iota, x_1^\iota)),\\ &(\llbracket cond \rrbracket((0,1,1), 0, (\perp,0,1), cond(x_1^\iota, \underline{0}, x_2^\iota)), \ldots\} \end{aligned}$$

$S'_0 := Z_0 \mid_{\pi_1(Z_0)\setminus\pi_1(S_0)} \neq \emptyset$
$k := 1$
$S_1 := S_0 \cup S'_0$
$X \in \pi_1(S_1)$
pos.stop:=true
output:=1 and $\lambda x_1^\iota, x_2^\iota . cond(x_1, \underline{0}, x_2)$.

The first example says that there is no function $f \in D_{\iota^2}$ definable in FPCF such that $f(0,\perp) = 0, f(\perp,0) = 0$ and $f(1,1) = 1$. The test conditions determine the matching continuous function uniquely. They specify the trace of *por*:

Definition 6.11 *For any $n \in \mathbf{N}$ the trace of a function $f \in [\mathbf{B}_\perp^n \to \mathbf{B}_\perp]$ is defined as follows:*

$$tr(f) := \{(v,b) \mid v \in \mathbf{B}_\perp^n, b \in \mathbf{B}_\perp \setminus \{\perp\}, f(v) = b, f.a.\ v' \sqsubseteq v : f(v') = \perp\}.$$

Every continuous function is given uniquely by its trace.

In the second example there exists more than one function matching the conditions. However, the trace of the function *lor* represented by the output term equals the test assignments. This makes clear a problem which appears often when carrying out the algorithm: Which test characterizes f uniquely without growing the number of the assignments too much? The reader should notice that the running time of the procedure depends on this number exponentially. A further problem is that the algorithm can be used for finite basic domains, only. Functions which utilize the infinity of the $\mathbf{N}$ hardly can be analysed with the help of finite approximations. Just as hindering is the restriction of the order of the functions in the premises of the algorithm.

7 The Degree of the Parallel Conditional

This section is dedicated to the functions which have the same *dop* as *pcond*. First we will verify that all computable functions are definable with the help of *pcond*, if the basic domain is $\mathbf{B}_\perp$. However, if we take $\mathbf{N}_\perp$, the existential quantifier function $\exists$ is necessary.

Theorem 7.1 *Every compact function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ is sequentially reducible on pcond.*

Proof: In [Plo77] is shown by induction on the type the more generally fact that for all $\sigma \in \Sigma$ and $i, j \in \mathbf{N}$ the functions β_i^σ, $\beta_i^\sigma{\to}0$ and, if it exists, $(\beta_i^\sigma{\to}0) \sqcup (\beta_j^\sigma{\to}1)$ are sequentially reducible on *pcond*. □

Thus, we obtain that lubs of finitely many step functions are definable when using the parallel conditional. If the basic domain is finite, all semantic domains are finite, i.e. all elements in the model are compact. So we have

Corollary 7.2 *If $D_\iota = \mathbf{B}_\perp$, every function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ is sequentially reducible on pcond.* □

If $D_\iota = \mathbf{N}_\perp$, the inversion of theorem 7.1 does not hold, e.g. in this case *pcond* is not compact. Notice that arbitrary directed lubs of compact elements are not definable with *pcond*:

Theorem 7.3 *The existential quantifier function $\exists$ is not sequentially reducible on pcond, i.e. we have*

$$pcond <_p pcond \oplus \exists.$$

A detailed proof can be looked up in [Plo77]. In this paper we will only give an outline.

Define additional transition rules for the operational semantics of the language PPCF, i.e. PCF extended by *pcond*:

- For all $n \in \mathbf{N}$ and PPCF-terms M^ι set $\underline{pcond}(M, \underline{n}, \underline{n}) \to \underline{n}$.
- For all PPCF-terms M^ι, N^ι set $\underline{pcond}(\underline{0}, M, N) \to M$.
- For all $n \in \mathbf{N}$ and PPCF-terms M^ι, N^ι set $\underline{pcond}(\underline{n+1}, M, N) \to N$.

- For all PPCF-terms $M, M_1, M_2, N, N_1, N_2, L_1, L_2$ of type ι set

$$\frac{M_1 \to M_2}{\underline{pcond}(M_1) \to \underline{pcond}(M_2)} \quad \frac{N_1 \to N_2}{\underline{pcond}(M, N_1) \to \underline{pcond}(M, N_2)}$$

$$\frac{L_1 \to L_2}{\underline{pcond}(M, N, L_1) \to \underline{pcond}(M, N, L_2)}.$$

With this extension the evaluation of the terms becomes nondeterministic. The function *eval* stays well-defined by

Lemma 7.4 *The evaluation of PPCF-terms has the Church-Rosser property, i.e. for all PPCF-terms M_1, M_2, M_3 we have*

$$M_1 \to M_2 \wedge M_1 \to M_3 \;\Rightarrow\; ex.\ M_4 : (M_2 \to M_4 \vee M_2 = M_4) \wedge (M_3 \to M_4 \vee M_3 = M_4).$$

Operational and denotational semantics of PPCF are consistent:

Theorem 7.5 *For all PPCF-programs M and $n \in \mathbf{N}$ it is*

$$[\![M]\!] = n \Leftrightarrow eval(M) = \underline{n}.$$

We obtain the non-definability of $\exists$ in PPCF from the following more general result:

Lemma 7.6 *Let E be a PPCF-program such that $FV(E) \subseteq \{x^{\iota^1}\}$ and $\perp_\iota \neq [\![[F/x](E)]\!] \neq [\![[T_n/x](E)]\!] \neq \perp_\iota$ for infinitely many $n \in \mathbf{N}$, whereat F is standing for $\lambda x^\iota.\underline{1}$ and T_n for $\lambda x^\iota.cond(x \doteq \underline{n}, \underline{0}, \Omega^\iota)$, i.e. $[\![F]\!] = \perp \to 1$ and $[\![T_n]\!] = n \to 0$. Then we have $[F/x](E) \to^{k'} \underline{n} \Rightarrow k' > k$ for all $k \in \mathbf{N}$, i.e. $eval([F/x](E))$ is not defined.*

Proof: Induction on k and the structure of E. □

Supposing that $\exists$ is definable by a PPCF-term $\lambda x^{\iota^1}.M$ we get $[\![[F/x](M)]\!] = 1$ and $[\![[T_m/x](M)]\!] = 0$ for all $m \in \mathbf{N}$. Then the assumptions in the previous lemma are satisfied. Therefore, $eval([F/x](M))$ is not defined. This contradicts theorem 7.5, and the proof of theorem 7.3 has completed. □

Plotkin added *pcond* to PCF, in order to make all compact elements definable. One obtains the same result by extending PCF with *por*:

Lemma 7.7 *The functions pcond and por have the same dop.*

Proof: One checks readily that $por = [\![\lambda \dot{x}^\iota, \dot{y}^\iota.\underline{pcond}(\dot{x}, \underline{0}, Z(\dot{y}))]\!]$ holds. It remains to show $pcond \leq_p por$.

In case of $D_\iota = \mathbf{B}_\perp$ the algorithm of Stoughton yields the term

$$M := \lambda \dot{x}^\iota, \dot{y}^\iota, \dot{z}^\iota.\underline{por}(\ \neg(\underline{por}(\dot{x}, cond(\dot{z}, \dot{x}, \underline{0}))),\\ \neg(\underline{por}(\neg(\underline{por}(\dot{x}, \dot{z})), \neg\dot{y}))),$$

whose semantics approximates the function *pcond* w.r.t. the embedding of $\mathbf{B}_\perp$ in $\mathbf{N}_\perp$. We have

$$pcond_{\mathbf{B}_\perp} := [\![M]\!](x, y, z) = \begin{cases} 0 & , (x,y,z) \in \{(n,0,0) \mid n \in \mathbf{N}_\perp\} \cup \{(0, 0.n) \mid n \in \mathbf{N}_\perp\} \\ & \qquad \cup \{(n+1, m, 0) \mid n \in \mathbf{N}, m \in \mathbf{N}_\perp\} \\ 1 & , (x,y,z) \in \{(n, m+1, l+1) \mid n \in \mathbf{N}_\perp, m, l \in \mathbf{N}\} \\ & \qquad \cup \{(0, n+1, m) \mid n \in \mathbf{N}, m \in \mathbf{N}_\perp\} \\ & \qquad \cup \{(n+1, m, l+1) \mid n, l \in \mathbf{N}, m \in \mathbf{N}_\perp\} \\ \perp & , \text{otherwise.} \end{cases}$$

With the help of the fixed point operator we succeed the "jump" to $\mathbf{N}_\perp$:

$$M' := Y\Phi := Y_{\iota^3}\lambda f^{\iota^3}, \dot{x}^\iota, \dot{y}^\iota, \dot{z}^\iota.cond(\underline{pcond}_{\mathbf{B}_\perp}(\dot{x}, \dot{y}, \dot{z}), \underline{0}, f((\dot{x}, \dot{y} \dot{-} \underline{1}, \dot{z} \dot{-} \underline{1}) \dot{+} \underline{1})).$$

We have

$$[\![\Phi]\!](\perp_{\iota^3}) : (x,y,z) \mapsto \begin{cases} 0 & , y = z = 0 \ \vee \ x = y = 0 \ \vee \ (x = n+1 \ \wedge \ z = 0) \\ \perp & , \text{otherwise} \end{cases}$$

$$\vdots$$

$$[\![\Phi]\!]^{k+1}(\perp_{\iota^3}) : (x,y,z) \mapsto \begin{cases} y & , y = z \in \{0, \ldots, k\} \ \vee \ (x = 0 \ \wedge \ y \in \{0, \ldots, k\}) \\ z & , x = n+1 \ \wedge \ z \in \{0, \ldots, k\} \\ \perp & , \text{otherwise.} \end{cases}$$

At each step $k \to k+1$ one additional number can be distinguished. We obtain $pcond = \bigsqcup_{k \in \mathbf{N}} [\![\Phi]\!]^k(\perp_{\iota^3}) = [\![M']\!]$. □

In analogous way to the parallel Or we define a parallel And:

$$pand \in D_{\iota^2} : (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \wedge \ y = 0 \\ 1 & , x = n+1 \ \vee \ y = n+1 \\ \perp & , \text{otherwise.} \end{cases}$$

Like in logical calculus *por* and *pand* are mutually definable with the help of the negation:

$$por = [\![\lambda x^\iota, y^\iota.\neg(\underline{pand}(\neg x, \neg y))]\!], \ pand = [\![\lambda x^\iota, y^\iota.\neg(\underline{por}(\neg x, \neg y))]\!].$$

The generalized functions

$por_2 := por$,
$por_{n+3} := [\![\lambda x_1^\iota, \ldots, x_{n+3}^\iota.\underline{por}(x_1, \underline{por}_{n+2}(x_2, \ldots, x_{n+3}))]\!] \in D_{\iota^{n+3}}$ and
$pand_{n+2} \in D_{\iota^{n+2}}$ defined in analogous way

have the same *dop* as *por*:

$$por = [\![\lambda x_1^\iota, x_2^\iota.\underline{por}_{n+2}(\underbrace{1,\ldots,1}_{n-\text{times}}, x, y)]\!],$$
$$pand = [\![\lambda x_1^\iota, x_2^\iota.\underline{pand}_{n+2}(\underbrace{0,\ldots,0}_{n-\text{times}}, x, y)]\!].$$

Also the same *dop* have the following finite approximations of *por* and *pand* for $s, t \geq 1$:

$$por[s,t] \in D_{\iota^2} : (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \vee \ y = 0 \\ 1 & , 1 \leq x \leq s \ \wedge \ 1 \leq y \leq t \\ \perp & , \text{otherwise}, \end{cases}$$

$$por[\infty,t] \in D_{\iota^2} : (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \vee \ y = 0 \\ 1 & , x \geq 1 \ \wedge \ 1 \leq y \leq t \quad \text{and} \\ \perp & , \text{otherwise} \end{cases}$$

$$por[s,\infty] \in D_{\iota^2} : (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \vee \ y = 0 \\ 1 & , 1 \leq x \leq s \ \wedge \ y \geq 1 \\ \perp & , \text{otherwise}. \end{cases}$$

Obviously, $por = por[\infty,\infty]$ holds. Set $s < \infty$ for all $s \in \mathbf{N}$. Then we have:
$s_1 \leq s_2 \ \wedge \ t_1 \leq t_2 \ \Rightarrow \ por[s_1,t_1] \leq por[s_2,t_2]$ because of

$$\begin{aligned} por[s_1,t_1] = [\![\lambda x^\iota, y^\iota.cond(&\underline{por[s_2,t_2]}(x,y),\\ &\underline{0},\\ &cond(\left\{\begin{array}{cl} x \dot{\leq} s_1 & , s_1 < \infty \\ \underline{0} & , s_1 = \infty \end{array}\right\} \underline{\triangle} \left\{\begin{array}{cl} y \dot{\leq} t_1 & , t_1 < \infty \\ \underline{0} & , t_1 = \infty \end{array}\right\},\\ &\quad \underline{1},\\ &\quad \Omega^\iota))]\!]. \end{aligned}$$

Moreover, define for $s, t \geq 1$:

$$pand[s,t] \in D_{\iota^2} : (x,y) \mapsto \begin{cases} 0 & , x = 0 \ \wedge \ y = 0 \\ 1 & , 1 \leq x \leq s \ \vee \ 1 \leq y \leq t \\ \perp & , \text{otherwise}. \end{cases}$$

We get for all $s, t \in \mathbf{N} \cup \{\infty\}$:

$$pand = [\![\lambda x^\iota, y^\iota.\neg(\underline{por[s,t]}(\neg x, \neg y))]\!],$$
$$por = [\![\lambda x^\iota, y^\iota.\neg(\underline{pand[s,t]}(\neg x, \neg y))]\!],$$
$$pand[s,t] = [\![\lambda x^\iota, y^\iota.cond(\underline{pand}(x,y), \underline{0}, cond(\underline{por}(x \dot{\leq} s, y \dot{\leq} t), \underline{1}, \Omega^\iota))]\!].$$

So we have for all $s_1, s_2, t_1, t_2 \in \mathbf{N} \cup \{\infty\}$:

$$pand \leq_p por[s_1,t_1] \leq_p por \leq_p pand[s_2,t_2] \leq_p pand$$

and hence for all $s, t \in \mathbf{N} \cup \{\infty\}$:

$$por[s,t] \equiv_p por \equiv_p pand[s,t].$$

Trachtenbrot introduced the following function:

Definition 7.8 *$V \in D_{\iota^3}$ is defined by*

$$(x,y,z) \mapsto \begin{cases} x & , \; x = y \neq \bot \\ y & , \; y = z \neq \bot \\ z & , \; x = z \neq \bot \\ \bot & , \; \textit{otherwise.} \end{cases}$$

V checks, whether two of the three arguments coincide, or to use Trachtenbrot's parlance, the "voting"-function V outputs the "absolute majority" of the arguments. Again we have:

Lemma 7.9 *The functions V and pcond have the same dop.*

Proof: It is easy to see that $por = [\![\lambda x^\iota, y^\iota . \underline{V}(Z(x), Z(y), \underline{0})]\!]$ holds, i.e. $pcond \leq_p V$. For the inversion we construct an approximation to V on $\mathbf{B}_\bot$:

$$V_{\mathbf{B}_\bot} := [\![\lambda \dot{x}^\iota, \dot{y}^\iota, \dot{z}^\iota . \underline{pcond}(\dot{x}, \underline{por}(\dot{y}, \dot{z}), \underline{pand}(\dot{y}, \dot{z}))]\!]:$$

$$(x,y,z) \mapsto \begin{cases} 0 & , \; (x,y,z) \in \{(n,0,0) \mid n \in \mathbf{N}_\bot\} \cup \{(0,n,0) \mid n \in \mathbf{N}_\bot\} \\ & \qquad\qquad \cup \{(0,0,n) \mid n \in \mathbf{N}_\bot\} \\ 1 & , \; (x,y,z) \in \{(n, m+1, l+1) \mid n \in \mathbf{N}_\bot, m,l \in \mathbf{N}\} \\ & \qquad\qquad \cup \{(n+1, m, l+1) \mid n,l \in \mathbf{N}, m \in \mathbf{N}_\bot\} \\ & \qquad\qquad \cup \{(n+1, m+1, l) \mid n,m \in \mathbf{N}, l \in \mathbf{N}_\bot\} \\ \bot & , \; \text{otherwise.} \end{cases}$$

Set $Y\Phi := Y_{\iota^3} \lambda f^{\iota^3}, \dot{x}^\iota, \dot{y}^\iota, \dot{z}^\iota . cond(\underline{V}_{\mathbf{B}_\bot}(\dot{x}, \dot{y}, \dot{z}), \underline{0}, f(\dot{x} \mathbin{\dot{-}} \underline{1}, \dot{y} \mathbin{\dot{-}} \underline{1}, \dot{z} \mathbin{\dot{-}} \underline{1}) \mathbin{\dot{+}} \underline{1})$. So we have

$$[\![\Phi]\!](\bot_{\iota^3}) : (x,y,z) \mapsto \begin{cases} 0 & , \; x = y = 0 \;\vee\; x = z = 0 \;\vee\; y = z = 0 \\ \bot & , \; \text{otherwise} \end{cases}$$

$$\vdots$$

$$[\![\Phi]\!]^{k+1}(\bot_{\iota^3}) : (x,y,z) \mapsto \begin{cases} r & , \; r \in \{0, \ldots, k\} \;\wedge\; (x = y = r \;\vee\; x = z = r \;\vee\; y = z = r) \\ \bot & , \; \text{otherwise.} \end{cases}$$

Therefore, $V = [\![Y\Phi]\!] = \bigsqcup_{k \in \mathbf{N}} [\![\Phi]\!]^k(\bot_{\iota^3})$ holds, i.e. $pcond \leq_p V$. □

Without altering the *dop* the approximations $por[s,t]$ can be generalized as follows:

Definition 7.10 *For any $r \geq 2$ and recursive sets $\emptyset \neq N_1, \ldots, N_r \subset \mathbf{N}$ set*

$$f_{N_1,\ldots,N_r} \in D_{\iota^r} : (x_1, \ldots, x_r) \mapsto \begin{cases} 0 & , \; \textit{ex.}\; 1 \leq i \leq r : \; x_i \in N_i \\ 1 & , \; \textit{f.a.}\; 1 \leq i \leq r : \; x_i \notin N_i \\ \bot & , \; \textit{otherwise.} \end{cases}$$

Lemma 7.11 *For all $r \geq 2$ and recursive sets $\emptyset \neq N_1, \ldots, N_r \subset \mathbf{N}$ we have*

$$f_{N_1,\ldots,N_r} \equiv_p pcond.$$

Proof: Let $\leq_m$ denote the many-one-reducibility in the common theory of computability, i.e. for any $N, M \subseteq \mathbf{N}$ the relation $N \leq_m M$ means that there is some function $g \in F_\mu^{(1)}$ such that $x \in N \Leftrightarrow g(x) \in M$ for all $x \in \mathbf{N}$. As it is well-known, all recursive sets $\emptyset \neq N \subset \mathbf{N}$ are $\leq_m$-equivalent. Especially there are $\psi_i, \varphi_i \in F_\mu^{(1)}$ such that $\{0\} \leq_m N_i$ via ψ_i and $N_i \leq_m \{0\}$ via φ_i for all $1 \leq i \leq r$. Hence, we get

$$por_r = [\![\lambda x_1^\iota, \ldots, x_r^\iota . \underline{f}_{N_1,\ldots,N_r}(\underline{\psi}_1(x_1), \ldots, \underline{\psi}_r(x_r))]\!]$$

and

$$f_{N_1,\ldots,N_r} = [\![\lambda x_1^\iota, \ldots, x_r^\iota . \underline{por}_r(\underline{\varphi}_1(x_1), \ldots, \underline{\varphi}_r(x_r))]\!],$$

and we conclude $f_{N_1,\ldots,N_r} \equiv_p por_r \equiv_p pcond$. □

Let us summarize: The extensions of PCF with the parallel Or, with the parallel conditional, resp. with Trachtenbrot's V-function have the same power, i.e. these languages define the same set of functions. For *por* increasing the number of arguments or restricting the domain of definition do not affect its *dop*. Lemma 7.11 shows that even any simultaneous finite test for recursive sets is of the same degree.

8 More Degrees

The functions *por, pcond*, the parallel test for zero $\mathcal{O}$, V and $\exists$ have been introduced already. Now we cite further functions in $\bigcup_{\sigma\in\Sigma} D_\sigma$ and insert them into the semilattice of the *dops*. We start with

Definition 8.1 *For basic domain $D_\iota = \mathbf{B}_\perp$ or $D_\iota = \mathbf{N}_\perp$ the* parallel test for convergence *is defined by*

$$pconv \in D_{\iota^2} : (x, y) \mapsto \begin{cases} 0 & , \ x \neq \perp \ \vee \ y \neq \perp \\ \perp & , \ otherwise, \end{cases}$$

the Gustave- *or* Kahn-Plotkin-function *by*

$$gu \in D_{\iota^3} : (x, y, z) \mapsto \begin{cases} 0 & , \ (x = 0 \ \wedge \ y = n+1) \ \vee \ (y = 0 \ \wedge \ z = n+1) \\ & \quad \vee (z = 0 \ \wedge \ x = n+1) \\ \perp & , \ otherwise, \end{cases}$$

the Berry-Plotkin-function *by*

$$BP \in D_{\iota^3} : (x, y, z) \mapsto \begin{cases} 0 & , \ y = 0 \ \wedge \ z = n+1 \\ 1 & , \ (x = 0 \ \wedge \ y = n+1) \ \vee \ (z = 0 \ \wedge \ x = n+1) \\ \perp & , \ otherwise, \end{cases}$$

whereat $n+1$ is standing for 1 in case of $D_\iota = \mathbf{B}_\perp$, and for any function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ the functional T_f by

$$T_f \in D_{((\sigma,\iota),\iota)} : G \mapsto \begin{cases} 0 & , G(f) = 0 \\ \perp & , \textit{otherwise.} \end{cases}$$

For $n \in \mathbf{N}$ in case of $D_\iota = \mathbf{N}_\perp$, resp. for $n \in \{0,1\}$ in case of $D_\iota = \mathbf{B}_\perp$ the bounded existential quantifier function *is defined by*

$$\exists_n \in D_{(\iota^1,\iota)} : f \mapsto \begin{cases} 0 & , ex.\ 0 \le i \le n :\ f(i) = 0 \\ 1 & , f(\perp) = 1 \\ \perp & , \textit{otherwise.} \end{cases}$$

Only in case of $D_\iota = \mathbf{N}_\perp$ a special form of the V-function is defined, whereat $a \equiv_b c$ means $a \bmod b = c \bmod b$:

$$V' \in D_{\iota^3} : (x,y,z) \mapsto \begin{cases} x & , x = y \equiv_3 0 \\ y & , y = z \equiv_3 2 \\ z & , x = z \equiv_3 1 \\ \perp & , \textit{otherwise} \end{cases}$$

and also the test, whether the definition domain of a function is not empty:

$$some \in D_{(\iota^1,\iota)} : f \mapsto \begin{cases} 0 & , ex.\ i \in \mathbf{N} :\ f(i) \ne \perp \\ \perp & , \textit{otherwise.} \end{cases}$$

The functions $T_f, \exists_n$ and V' base on Sazonov (cp. [Saz76]).

The main theorem in this section is

Theorem 8.2 *The diagrams 2 and 3 are representing parts of the semilattice of all dops in case of $D_\iota = \mathbf{B}_\perp$, resp. of $D_\iota = \mathbf{N}_\perp$. The thick lined arrows between two functions stand for the $<_p$-relation, the thin lined arrows for the $\le_p$-relation. Moreover, we have*

- $\exists_0$ *is PCF-definable.*
- $T_{\mathcal{O}} \equiv_p \mathcal{O} \equiv_p pconv \equiv_p some$
- $BP \equiv_p V'$ *in case of $D_\iota = \mathbf{N}_\perp$.*

The proof follows from the next three lemmata.

At first we examine the T-functionals.

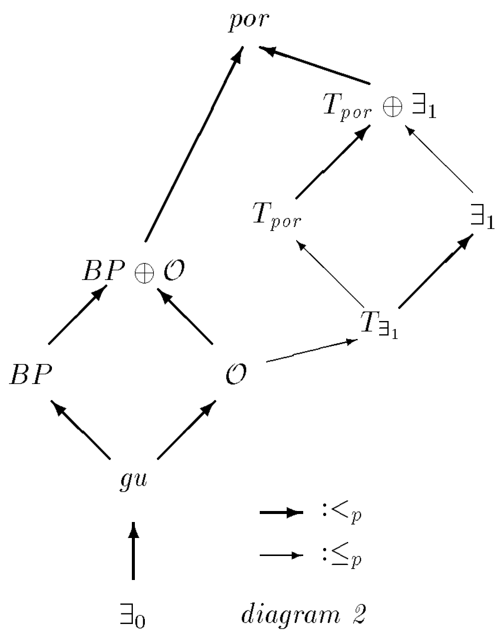

diagram 2

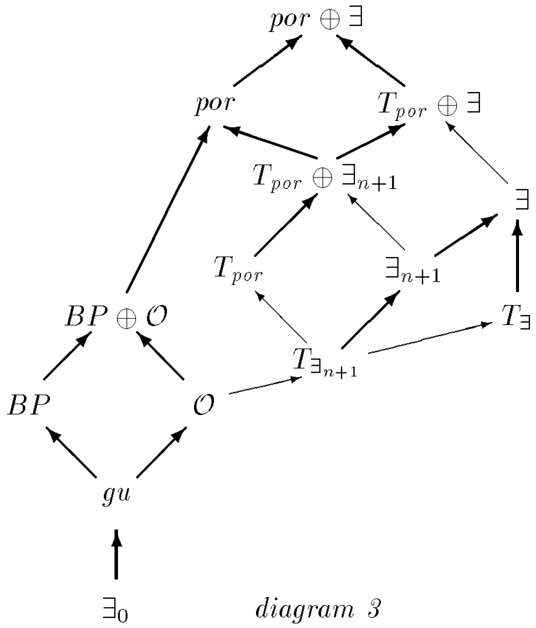

diagram 3

Lemma 8.3 *For all functions $f, g \in \bigcup_{\sigma\in\Sigma} D_\sigma$ we have*

1. $T_f \leq_p f$
2. $f \leq_p g \Rightarrow T_f \leq_p T_g$
3. $f \in D_{(\sigma_1,\ldots,\sigma_n,\iota)} \ \wedge\ \sharp f(D_{\sigma_1})\ldots(D_{\sigma_n}) \leq 2 \ \Rightarrow\ T_f \equiv_p f$
4. $T_{T_f} \equiv_p T_f$.

Proof:

1. For $f \in D_\sigma$ it is $T_f = [\![\lambda G^{(\sigma,\iota)}.cond(G(\underline{f}), \underline{0}, \Omega^\iota)]\!]$.

2. Let $f \in D_\sigma$ and $g \in D_\tau$ with $f \leq_p g$. Then there is a closed PCF-term $M^{\tau\to\sigma}$ such that $[\![M]\!](g) = f$. We obtain for all $\rho \in Env$ and $G \in V_{(\sigma,\iota)}$ with $G \neq y^\tau$:

$$
\begin{aligned}
&[\![\underline{T}_g(\lambda y^\tau.G(M(y)))]\!](\rho) = 0\\
&\quad\Leftrightarrow 0 = [\![\lambda y^\tau.G(M(y))]\!](\rho)(g)\\
&\qquad\quad = [\![G]\!](\rho[g/y])([\![M]\!](\rho[g/y])(g))\\
&\qquad\quad = [\![G]\!](\rho)([\![M]\!](g)) \qquad\qquad (G \neq y \text{ and } M \text{ closed})\\
&\qquad\quad = [\![G]\!](\rho)(f)\\
&\quad\Leftrightarrow T_f([\![G]\!](\rho)) = 0.
\end{aligned}
$$

Hence, $T_f = [\![\lambda G^{(\sigma,\iota)}.\underline{T}_g(\lambda y^\tau.G(M(y)))]\!]$ holds.

3. Let $\sigma := (\sigma_1, \ldots, \sigma_n, \iota)$ and $f \in D_\sigma$ with $f(D_{\sigma_1})\ldots(D_{\sigma_n}) \subseteq \{\perp, 0\}$. So we have $f = [\![\lambda \dot{x}_1^{\sigma_1}, \ldots, \dot{x}_n^{\sigma_n}.\underline{T}_f(\lambda\varphi^\sigma.\varphi(\dot{x}_1, \ldots, \dot{x}_n))]\!]$ by
$$[\![\lambda \dot{x}_1^{\sigma_1}, \ldots, \dot{x}_n^{\sigma_n}.\underline{T}_f(\lambda\varphi^\sigma.\varphi(\dot{x}_1, \ldots, \dot{x}_n))]\!](x_1, \ldots . x_n) = 0 \Leftrightarrow f(x_1, \ldots, x_n) = 0$$
for all $x_1 \in D_{\sigma_1}, \ldots, x_n \in D_{\sigma_n}$. This implies $f \leq_p T_f$ and with the help of item 1 we get $f \equiv_p T_f$. In the case $0 \notin f(D_{\sigma_1})\ldots(D_{\sigma_n})$ apply the construction to $not(f) := \lambda\dot{x}_1^{\sigma_1}, \ldots, \dot{x}_n^{\sigma_n}.\neg(f(\dot{x}_1, \ldots, \dot{x}_n))$, since $f \equiv_p not(f)$ for $\sharp f(D_{\sigma_1})\ldots(D_{\sigma_n}) \leq 2$.

4. Item 3 implies this statement, because for every $f \in D_\sigma$ one has $\sharp T_f(D_{((\sigma,\iota),\iota)}) \leq 2$.□

Notice that in general $f <_p g \Rightarrow T_f <_p T_g$ does not hold. We will give a counterexample later.

If a function assumes at most two different values, testing on invariance under sequentiality relations gets easier:

Lemma 8.4 *For any $\sigma = (\sigma_1, \ldots, \sigma_n, \iota) \in \Sigma$ every function $f \in D_\sigma$ with $\sharp f(D_{\sigma_1})\ldots(D_{\sigma_n}) \leq 2$ is invariant under each relation of the form $S^m_{A,A}$ with $m \in \mathbf{N}$, $m \geq 2$ and $A \subseteq \{1, \ldots, m\}$.*

Proof: If f assumes two values, one of them must be $\perp$ because of the monotony. Therefore, we have for all $(x_1^{(1)}, \ldots, x_m^{(1)}) \in D^m_{\sigma_1}, \ \ldots, \ (x_1^{(n)}, \ldots, x_m^{(n)}) \in D^m_{\sigma_n}$:
$$(f.a.\ i \in A:\ f(x_i^{(1)}, \ldots, x_i^{(n)}) \neq \perp) \Rightarrow (f.a.\ i, j \in A:\ f(x_i^{(1)}, \ldots, x_i^{(n)}) = f(x_j^{(1)}, \ldots, x_j^{(n)})).$$
□

Lemma and definition 8.5 *Let $\tilde{R}$ stand for the sequentiality relation $S^3_{\{1,2\},\{1,2\}} \cap \tilde{3}$, i.e.*
$$(x_1, x_2, x_3) \in \tilde{3} \ \Leftrightarrow \ x_1 = \perp \ \vee \ x_2 = \perp \ \vee \ (x_3 = \perp \ \wedge \ x_1 = x_2) \ \vee \ x_1 = x_2 = x_3.$$
Then T_{por} and BP are invariant, por and $\exists_1$ are not invariant under $\tilde{R}$.

Proof:

- Every function $f \in D_{(\sigma_1, \ldots, \sigma_n, \iota)}$ with $\sharp f(D_{\sigma_1})\ldots(D_{\sigma_n}) \leq 2$ is invariant under $\tilde{R}$, because we have for all $(x_1, y_1, z_1) \in D^3_{\sigma_1}, \ldots, (x_n, y_n, z_n) \in D^3_{\sigma_n}$:
$$f(\vec{x}) \neq \perp \ \wedge \ f(\vec{y}) \neq \perp \ \wedge \ (f(\vec{z}) \neq \perp \vee \ f(\vec{x}) \neq f(\vec{y})) \ \Rightarrow \ f(\vec{x}) = f(\vec{y}) = f(\vec{z}).$$
- $BP \in \vec{R}$: Let $\vec{x}, \vec{y}, \vec{z} \in \tilde{R}$ and $\vec{e} \notin \tilde{R}$ with $e_i := BP(x_i, y_i, z_i)$ for $1 \leq i \leq 3$. We distinguish

- $e_1 = 0 \wedge e_2 = 1 \Rightarrow y_1 = 0 \wedge z_1 = n{+}1 \wedge (y_2 = m{+}1 \vee z_2 = 0) \Rightarrow \vec{y} \notin \vec{R} \vee \vec{z} \notin \vec{R}$
- $e_1 = 1 \ \wedge \ e_2 = 0$ analogous to the previous case
- $e_1 = e_2 = 0 \ \wedge \ e_3 = 1 \Rightarrow y_1 = y_2 = 0 \ \wedge \ z_1 = z_2 = n+1 \ \wedge \ (y_3 = m+1 \ \vee \ z_3 = 0) \Rightarrow \vec{y} \notin \vec{R} \ \vee \ \vec{z} \notin \vec{R}$
- $e_1 = e_2 = 1 \ \wedge \ e_3 = 0 \Rightarrow (y_1 = y_2 = n+1 \ \wedge \ y_3 = 0) \ \vee \ (z_1 = z_2 = 0 \ \wedge \ z_3 = n+1) \Rightarrow \vec{y} \notin \vec{R} \ \vee \ \vec{z} \notin \vec{R}$.

All cases contradict the assumption.

- $por \notin \vec{R}$:

x	y	$por(x,y)$
0	$\bot$	0
$\bot$	0	0
1	1	1

- $\exists_1 \notin \vec{R}$: Define $f_1 := 0{\to}0$, $f_2 := 1{\to}0$ and $f_3 := \bot{\to}1$. As $f_1(x) = \bot \ \vee \ f_2(x) = \bot$ holds for all $x \in D_\iota$, we have $(f_1, f_2, f_3) \in \vec{R}$. However, it is $(\exists_1(f_1), \exists_1(f_2), \exists_1(f_3)) = (0,0,1) \notin \vec{R}$. $\square$

Proof of theorem 8.2:

We begin with term-constructions:

$\triangleright$ $\exists_0$ is PCF-definable:
$\exists_0 = [\![\lambda f^{\iota^1}.cond(f(\underline{0}), \underline{0}, cond(f(\Omega^\iota) \doteq \underline{1}, \underline{1}, \Omega^\iota))]\!]$.

$\triangleright$ $\mathcal{O} \equiv_p pconv$:
$\mathcal{O} = [\![\lambda x_\iota, y_\iota.\underline{pconv}(cond(x, \underline{0}, \Omega^\iota, cond(y, \underline{0}, \Omega^\iota)))]\!]$
$pconv = [\![\lambda x^\iota, y^\iota.\underline{\mathcal{O}}(conv(x), conv(y))]\!]$.

$\triangleright$ $gu \leq_p \mathcal{O}$:
$gu = [\![\lambda x^\iota, y^\iota, z^\iota.\underline{\mathcal{O}}(x \,\underline{\wedge}\, \neg y, y \,\underline{\wedge}\, \neg z, z \,\underline{\wedge}\, \neg x)]\!]$
$\mathcal{O}(x_1, \ldots, x_{n+3})$ is abridged for $\mathcal{O}(x_1, \mathcal{O}(x_2, \mathcal{O}(\ldots, x_{n+3})))$.

$\triangleright$ $gu \leq_p BP$:
$gu = [\![\lambda x^\iota, y^\iota, z^\iota.conv(\underline{BP}(x,y,z))]\!]$.

$\triangleright$ $\mathcal{O} \leq_p \exists_1$:
For all $\rho \in Env$ we have

$$[\![\underline{\exists}_1(\lambda z^\iota.cond(z^\iota, x^\iota, y^\iota))]\!](\rho[x_0/x][y_0/y]) = \begin{cases} 0 & , \ x_0 = 0 \ \vee \ y_0 = 0 \\ \bot & , \ \text{otherwise,} \end{cases}$$

i.e. $\mathcal{O} = [\![\lambda x^\iota, y^\iota.\underline{\exists}_1(\lambda z^\iota.cond(z,x,y))]\!]$.

$\triangleright$ $\exists_1 \leq_p por$:
$\exists_1 = [\![\lambda f^{\iota^1}.cond(\underline{por}(f(\underline{0}), f(\underline{1})), \underline{0}, cond(f(\Omega^\iota) \doteq \underline{1}, \underline{1}, \Omega^\iota))]\!]$.

$\triangleright$ $\exists_1 \leq_p \exists$:
$\exists_1 = [\![\lambda f^{\iota^1}.cond(\underline{\exists}(f) \doteq \underline{1}, \underline{1}, \underline{\exists}(\lambda x^\iota.cond(x \dot{\geq} \underline{2}, \underline{1}, f(x))))]\!]$.

$\triangleright$ $BP \leq_p por$:
Set $M := \lambda \dot{x}^\iota, \dot{y}^\iota, \dot{z}^\iota.\underline{pcond}(\dot{y}, cond(\neg\dot{z}, \underline{0}, cond(\neg\dot{x}, \underline{1}, \Omega^\iota)), cond(\dot{x}, \underline{1}, cond(\dot{z}, \underline{1}, \Omega^\iota)))$.
$BP \sqsubseteq [\![M]\!]$:
$[\![M]\!](\perp, 0, n+1) = pcond(0, 0\perp) = 0$,
$[\![M]\!](n+1, \perp, 0) = pcond(\perp, 1, 1) = 1$,
$[\![M]\!](0, n+1, \perp) = pcond(n+1, \perp, 1) = 1$.
$[\![M]\!] \sqsubseteq BP$:
$[\![M]\!](x, y, z) = 0 \Rightarrow y = 0 \ \wedge \ z = n+1$,
$[\![M]\!](x, \perp, z) = 1 \Rightarrow x = n+1 \ \wedge \ z = 0$,
$[\![M]\!](x, 0, z) = 1 \Rightarrow x = n+1 \ \wedge \ z = 0$,
$[\![M]\!](x, n+1, z) = 1 \Rightarrow x = 0 \ \vee \ (x = n+1 \ \wedge \ z = 0)$.

$\triangleright$ $pconv \equiv_p some$:
$pconv = [\![\lambda x^\iota, y^\iota.\underline{some}(\lambda z^\iota.cond(z, x, y))]\!]$
Set $Y\Phi := Y_{(\iota^1,\iota)}\lambda\varphi^{(\iota^1,\iota)}, f^{\iota^1}.\underline{pconv}(f(x), \varphi(\lambda z^\iota.f(z \dot{+} \underline{1})))$.

$$[\![\Phi]\!](\perp_{(\iota^1,\iota)}) : f \mapsto \begin{cases} 0 & , f([\![x]\!]) \neq \perp \\ \perp & , \text{otherwise,} \end{cases}$$

$$\vdots$$

$$[\![\Phi]\!]^{k+1}(\perp_{(\iota^1,\iota)}) : f \mapsto \begin{cases} 0 & , f([\![x]\!]) \neq \perp \ \vee \ f([\![x]\!]+1) \neq \perp \ \vee \ \ldots \ \vee \ f([\![x]\!]+k) \neq \perp \\ \perp & , \text{otherwise.} \end{cases}$$

Hence, $some = [\![(\lambda x^\iota.Y\Phi)(\underline{0})]\!]$ holds.

$\triangleright$ For all $n, m \in \mathbf{N}$ we get $\exists_{n+1} \equiv_p \exists_{m+1}$ by

1. For all $n \in \mathbf{N}$ it is

$$\exists_n = [\![\lambda f^{\iota^1}.cond(\underline{\exists}_{n+1}(f) \doteq \underline{1}, \underline{1}, \underline{\exists}_{n+1}(\lambda x^\iota.cond(x \doteq \underline{n+1}, \underline{1}, f(x))))]\!].$$

2. For all $k \geq 1$ it is $\exists_{2k} = [\![\lambda f^{\iota^1}.\underline{\exists}_k(\lambda n^\iota.\underline{\exists}_k(\lambda m^\iota.f(n \dot{+} m)))]\!]$, since we have

$$\begin{array}{ll}
f(\perp) = 1 & \Rightarrow \exists_k(\perp \mapsto \exists_k(\perp \mapsto f(\perp + \perp) = 1) = 1) = 1, \\
f(0) = 0 & \Rightarrow \exists_k(0 \mapsto \exists_k(0 \mapsto f(0+0) = 0) = 0) = 0, \\
f(1) = 0 & \Rightarrow \exists_k(0 \mapsto \exists_k(1 \mapsto f(0+1) = 0) = 0) = 0, \\
\vdots & \quad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \\
f(k) = 0 & \Rightarrow \exists_k(0 \mapsto \exists_k(k \mapsto f(0+k) = 0) = 0) = 0, \\
f(k+1) = 0 & \Rightarrow \exists_k(1 \mapsto \exists_k(k \mapsto f(1+k) = 0) = 0) = 0, \\
\vdots & \quad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \\
f(2k) = 0 & \Rightarrow \exists_k(k \mapsto \exists_k(k \mapsto f(k+k) = 0) = 0) = 0,
\end{array}$$

$$\begin{aligned}
&(f(0)\neq 0\wedge\ldots\wedge f(2k)\neq 0\wedge f(\bot)\neq\bot)\Rightarrow\\
&\exists_k(\bot\mapsto\exists_k(\bot\mapsto f(\bot)=\bot,0\mapsto f(\bot)=\bot,\ldots,k\mapsto f(\bot)=\bot)=\bot,\\
&\quad 0\mapsto\exists_k(\bot\mapsto f(\bot)=\bot,0\mapsto f(0)\neq 0,\ldots,k\mapsto f(k)\neq 0)=\bot,\\
&\quad\vdots\qquad\qquad\qquad\qquad\vdots\qquad\qquad\qquad\vdots\\
&\quad k\mapsto\exists_k(\bot\mapsto f(\bot)=\bot,0\mapsto f(k)\neq 0,\ldots,k\mapsto f(2k)\neq 0)=\bot)=\bot.
\end{aligned}$$

$\triangleright$ With these results lemma 8.3 implies $\mathcal{O}\equiv_p T_{\mathcal{O}}\leq_p T_{\exists_{n+1}}\leq_p T_{por}$.

$\triangleright$ $BP\equiv_p V'$:

- $BP\leq_p V'$:

  Set $M:=\lambda x^\iota,y^\iota,z^\iota.\underline{V'}(Z(x),cond(y,\underline{2},\underline{0}),cond(z,\underline{1},\underline{2}))$.

  Hence, we have $[\![M]\!](\bot,0,n+1)=V'(\bot,2,2)=2$, $[\![M]\!](n+1,\bot,0)=V'(1,\bot,1)=1$ and $[\![M]\!](0,n+1,\bot)=V'(0,0,\bot)=0$, and only in these cases $[\![M]\!]\neq\bot$ holds. So we get

  $$BP=[\![\lambda x^\iota,y^\iota,z^\iota.cond(\underline{V'}(Z(x),cond(y,\underline{2},\underline{0}),cond(z,\underline{1},\underline{2}))\doteq\underline{2},\underline{0},\underline{1})]\!].$$

- $V'\leq_p BP$: Here a slight modification of BP is more convenient. Set

  $$BP'\in D_{\iota^3}:(x,y,z)\mapsto\begin{cases}0 & ,\ y=m+1\ \wedge\ z=n+1\\ 1 & ,\ (x=0\ \wedge\ y=0)\ \vee\ (z=0\ \wedge\ x=n+1)\\ \bot & ,\ \text{otherwise.}\end{cases}$$

  Because of $BP'=[\![\lambda x^\iota,y^\iota,z^\iota.\underline{BP}(x,\neg y,z)]\!]$ and $BP=[\![\lambda x^\iota,y^\iota,z^\iota.\underline{BP'}(x,\neg y,z)]\!]$, obviously, $BP\equiv_p BP'$ holds. Let $P\subseteq\mathbf{N}^2$ such that $(x,y)\in P\Leftrightarrow x$ divides y. So the characteristic function c_P is primitive recursive and also the function

  $$\varphi:\mathbf{N}\to\mathbf{N},\ x\mapsto\mu(z<3).c_P(3,x+z)=x\bmod 3.$$

  By theorem 2.23 $\overline{\varphi}$ is PCF-definable by a term $\underline{\overline{\varphi}}$. We write short $\overline{x}$ instead of $\underline{\overline{\varphi}}(x)$. Defining $M:=\lambda x^\iota,y^\iota,z^\iota.\underline{BP}(\overline{x},\overline{y},cond(\overline{z}\doteq\underline{1},\underline{0},\overline{y}\vartriangle\overline{z}))$ one has

  $$\begin{aligned}&[\![M]\!](3n,3n,\bot)=BP(0,0,\bot)=1,\\&[\![M]\!](3n+1,\bot,3n+1)=BP(1,\bot,0)=1,\\&[\![M]\!](\bot,3n+2,3n+2)=BP(\bot,2,1)=0.\end{aligned}$$

  We get

  $$\begin{aligned}V'=[\![\lambda x^\iota,y^\iota,z^\iota.cond(&\underline{BP}(\overline{x},\overline{y},cond(\overline{z}\doteq\underline{1},\underline{0},\overline{y}\vartriangle\overline{z})),\\&cond(y\doteq z\vartriangle\overline{y}=\underline{2},y,\Omega^\iota),\\&cond(\overline{x},cond(x\doteq y,x,\Omega^\iota),\\&\qquad cond(x\doteq z\vartriangle\overline{z}=\underline{1},x,\Omega^\iota)))]\!].\end{aligned}$$

So far we have just proven $\leq_p$-relations. In the following we distinguish *dops* by convenient sequentiality relations.

$\triangleright$ $\exists_0 <_p gu$:

$gu \notin (1,2,3 \mid 4)$:

x	y	z	$gu(x,y,z)$
0	1	$\perp$	0
$\perp$	0	1	0
1	$\perp$	0	0
1	1	1	$\perp$

$\triangleright$ $BP \,|_p\, \mathcal{O}$:

- $\mathcal{O} \not\leq_p BP$:
  - $\mathcal{O} \notin (1,\ldots,k \mid \ldots m)$ for all $2 \leq k < m$:

$$\begin{array}{rcc|c}
 & x & y & \mathcal{O}(x,y) \\
\hline
 & 0 & \perp & 0 \\
 & \perp & 0 & 0 \\
\multirow{3}{*}{$(k-2)-\text{times}\left\{\vphantom{\begin{array}{c}0\\ \vdots\\ 0\end{array}}\right.$} & 0 & \perp & 0 \\
 & \vdots & \vdots & \vdots \\
 & 0 & \perp & 0 \\
\multirow{3}{*}{$(m-k)-\text{times}\left\{\vphantom{\begin{array}{c}0\\ \vdots\\ 0\end{array}}\right.$} & \perp & \perp & \perp \\
 & \vdots & \vdots & \vdots \\
 & \perp & \perp & \perp
\end{array}$$

  - $BP \in (1,2 \mid 3)$: Let $\vec{x},\vec{y},\vec{z} \in (1,2 \mid 3)^\iota$ and $e_i := BP(x_i,y_i,z_i)$ for all $1 \leq i \leq 3$ such that $e_1 \neq \perp$ and $e_2 \neq \perp$. Then it remains to show $e_1 = e_2 = e_3$. We distinguish
    1. $e_1 = e_2 = 0$
$\Rightarrow y_1 = y_2 = 0 \;\wedge\; z_1 = n+1 \;\wedge\; z_2 = m+1$
$\Rightarrow y_3 = 0 \;\wedge\; z_1 = z_2 = z_3 = n+1$
$\Rightarrow e_3 = 0$
    2. $e_1 = 0 \;\wedge\; e_2 = 1$
$\Rightarrow y_1 = 0 \;\wedge\; z_1 = n+1 \;\wedge\; (y_2 = m+1 \;\vee\; z_2 = 0)$
$\Rightarrow \vec{y} \notin (1,2 \mid 3) \;\vee\; \vec{z} \notin (1,2 \mid 3)$,
i.e. this case does not occur.
    3. $e_1 = 1 \;\wedge\; e_2 = 0$ analogous to the previous case.

4. $e_1 = e_2 = 1$
 $\Rightarrow ((x_1, y_1) = (0, n_1+1) \ \vee \ (x_1, z_1) = (m_1+1, 0))$
 $\wedge ((x_2, y_2) = (0, n_2+1) \ \vee \ (x_2, z_2) = (m_2+1, 0))$
 $\Rightarrow (x_1 = x_2 = 0 \ \wedge \ y_1 = y_2 = n+1) \ \vee \ (x_1 = x_2 = m+1 \ \wedge \ z_1 = z_2 = 0)$
 $\Rightarrow e_3 = 1.$

- $BP \not\leq_p \mathcal{O}$:
  - $BP \notin \tilde{3}$:

x	y	z	$BP(x,y,z)$
$\perp$	0	1	0
0	1	$\perp$	1
1	$\perp$	0	1

  - $\mathcal{O} \in \tilde{3}$ according to lemma 8.4.

▷ Consequences:

- $gu <_p \mathcal{O}$ because of $gu \in (1,2 \mid 3)$ and $\mathcal{O} \notin (1,2 \mid 3)$.
- For all $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ we obtain $T_f <_p T_f \oplus por$ and $T_f <_p T_f \oplus BP$, as we have $T_f \in \tilde{3}$, but $BP \notin \tilde{3}$ and therefore $por \notin \tilde{3}$. Especially $T_{por} <_p por$ and $gu \equiv_p T_{gu} \leq_p T_{BP} <_p BP$ holds. So we get a counterexample for the remark after lemma 8.3: $T_{T_{por}} \equiv_p T_{por} <_p por$.

$por \in \tilde{2}$:
Let $(x_1, x_2), (y_1, y_2) \in \tilde{2}$ and $e_i := por(x_i, y_i) \neq \perp$ for $i \in \{1, 2\}$. Furthermore, let $e_1 \neq e_2$ and w.l.o.g. (because of the symmetric definition) $e_1 = 0$ and $e_2 = 1$. Then we have $\vec{x} = (0,1) \notin \tilde{2}$ or $\vec{y} = (0,1) \notin \tilde{2}$.

$\exists \in \tilde{n}$ for all $n \geq 2$:
Assume $f_i \in D_{\iota^1}$ for $1 \leq i \leq n$ such that $(f_1, \ldots, f_n) \in \tilde{n}$ and $(\exists(f_1), \ldots, \exists(f_n)) \notin \tilde{n}$. Permuting the indices in a convenient way there is a $1 \leq k \leq n$ such that

$$\exists(f_i) = \begin{cases} 1 & , \ 1 \leq i \leq k \\ 0 & , \ k+1 \leq i \leq n. \end{cases}$$

Therefore, for all $k+1 \leq i \leq n$ there is a $j_i \neq \perp$ with $f_i(j_i) = 0$. Define

$$d_i := \begin{cases} \perp & , \ 1 \leq i \leq k \\ j_i & , \ k+1 \leq i \leq n. \end{cases}$$

Then we have $\vec{d} \in \tilde{n}$, but $(f_1(d_1), \ldots, f_n(d_n)) = (\underbrace{1, \ldots, 1}_{k-\text{times}}, \underbrace{0, \ldots, 0}_{(n-k)-\text{times}}) \notin \tilde{n}$. This is a contradiction to the assumption $(f_1, \ldots, f_n) \in \tilde{n}$.

▷ Consequences:

- Every computable function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ is invariant under $\tilde{2}$.
- $\exists \,|_p\, por$, because $por \notin \tilde{3}$ and $\exists \in \tilde{3}$ implies $por \not\leq_p \exists$.
- $\exists_{n+1} <_p por$ because of $\exists_{n+1} \in \tilde{3}$.
- $\exists \oplus T_{por} <_p \exists \oplus por$ because of $\exists \oplus T_{por} \in \tilde{3}$.
- $\exists_{n+1} \oplus T_{por} <_p por$ because of $\exists_{n+1} \oplus T_{por} \in \tilde{3}$.
- $\exists_{n+1} \oplus T_{por} <_p \exists \oplus T_{por}$, otherwise we have $\exists \leq_p \exists \oplus T_{por} \equiv_p \exists_{n+1} \oplus T_{por} <_p por$.
- $\exists_{n+1} <_p \exists$ because of the previous item.

Remark: $\exists_1$ is invariant under $S^3_{\{1,2\},\{1,2\}}$ and under $\vec{3}$, but it is not under the intersection of these relations. In section 3 we already indicated that the invariance under logical relations is transferred to the intersection of the relations just for functions of order 1.

▷ More consequences of lemma 8.5:

- $BP \oplus \mathcal{O} <_p por$, because $BP \in \tilde{R}$ and $\mathcal{O} \in \tilde{R}$, but $por \notin \tilde{R}$.
- $T_{\exists_{n+1}} <_p \exists_{n+1}$ and $T_\exists <_p \exists$, because $T_f \in \tilde{R}$ for all $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ and $\exists_{n+1} \notin \tilde{R}$.
- $T_{por} <_p T_{por} \oplus \exists_{n+1}$ because of $\exists_{n+1} \notin \tilde{R}$ and $T_{por} \in \tilde{R}$.

Hence, the proof of theorem 8.2 is done. □

Most of these relations between the *dops* can be lifted by extension with BP, resp. V':

Lemma 8.6 *With the notations introduced in theorem 8.2 one has the dops represented in the diagrams 5 and 6.*

Proof: Since we have $V' \equiv_p BP$, both functions have the same behaviour under all sequentiality relations. So we get

- $V' \notin \tilde{3}$ implies
  1. $T_\varphi <_p T_\varphi \oplus V'$ for all $\varphi \in \bigcup_{\sigma\in\Sigma} D_\sigma$, especially:
  $$T_{por} <_p T_{por} \oplus V', T_{\exists_{n+1}} <_p T_{\exists_{n+1}} \oplus V', T_\exists <_p T_\exists \oplus V' \text{ and } T_{V'} <_p V'.$$
  2. $\mathcal{O} <_p \mathcal{O} \oplus V', \exists_{n+1} <_p \exists_{n+1} \oplus V'$ and $\exists <_p \exists \oplus V'$
- $gu \equiv_p T_{gu} \leq_p T_{V'}$.

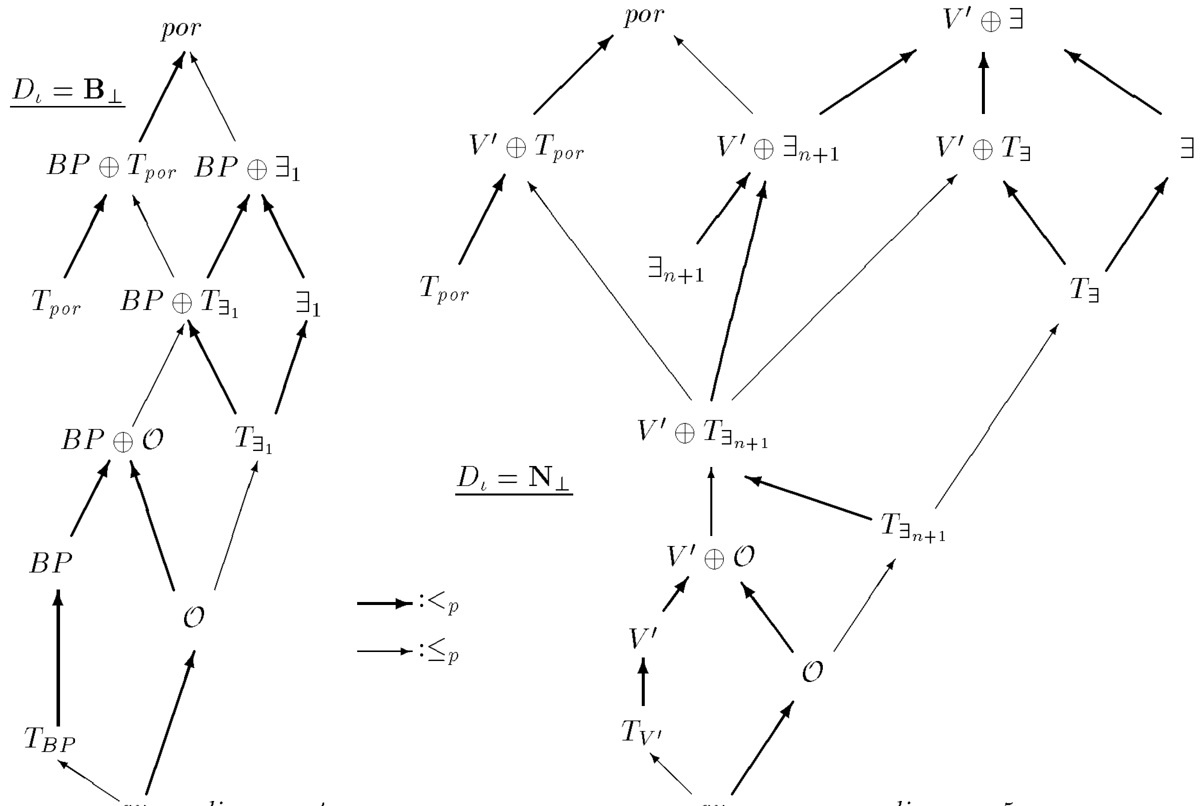

diagram 4 *diagram 5*

- $V' \in (1,2 \mid 3)$ implies $V' <_p \mathcal{O} \oplus V'$.
- $V' \in \tilde{R}$ implies $V' \oplus T_{por} <_p por$, $V' \oplus T_{\exists_{n+1}} <_p V' \oplus \exists_{n+1}$ and $V' \oplus T_\exists <_p V' \oplus \exists$. Moreover, $V' \oplus \exists_{n+1} <_p V' \oplus \exists$, otherwise $\exists \leq_p V' \oplus \exists \equiv_p V' \oplus \exists_{n+1} \leq_p por$ holds.

In the case of $\mathbf{B}_\perp$ we have the corresponding relations for BP. □

Remark: All relations between the *dops* that are proved by sequentiality relations are being preserved, if we extend the PCF-constants with constants standing for the logically sequential functions. For instance, lifting by D_{ι^1} is possible, because every function $f \in D_{\iota^1}$ is strict, i.e. one has $f(\perp) = \perp$, or f is constant, so that f is invariant under each relation of the form $S^n_{A,B}$. Because of $ord(\iota^1) = 1$ the function f is element of each intersection of such relations. One should notice that here the restriction to computability is transgressed: by theorem 5.16 for every function $f \in \mathbf{N} \rightharpoonup \mathbf{N}$ with $f \notin F_\mu$ the function $\overline{f} \in D_{\iota^1}$ is not computable.

We end this section with a computable generalization of $\mathcal{O}$ analogous to lemma 7.11.

Lemma 8.7 *Let $\emptyset \neq N_1, \ldots, N_r \subseteq \mathbf{N}$ be r.e. sets, whereat $r \geq 2$. Then*

$$g_{N_1,\ldots,N_r} \in D_{\iota^r} : (x_1, \ldots, x_r) \mapsto \begin{cases} 0 & , \ ex.\ 1 \leq i \leq r : \ x_i \in N_i \\ \perp & , \ otherwise \end{cases}$$

is computable, and if $\emptyset \neq N_1, \ldots, N_r \subset \mathbf{N}$ *are recursive, we get* $g_{N_1,\ldots,N_r} \equiv_p \mathcal{O}$.

Proof: Choose for all $1 \leq i \leq r$ functions $\varphi_i \in R^{(1)}$ such that $\varphi(\mathbf{N}) = N_i$. Define for all $n \in \mathbf{N}$:

$$g_n \in D_{\iota^r} : (x_1, \ldots, x_r) \mapsto \begin{cases} 0 & , \ ex.\ 1 \leq i \leq r: \ x_i \in \varphi_i(\{0, \ldots, n\}) \\ \perp & , \ \text{otherwise.} \end{cases}$$

Hence, we have

$$g_n = \bigsqcup(\{x_1 \to \perp \to \ldots \to \perp \to 0 \mid x_1 \in \varphi_1(\{0, \ldots, n\})\} \cup \ldots \\ \cup \{\perp \to \ldots \to \perp \to x_r \to 0 \mid x_r \in \varphi_r(\{0, \ldots, n\})\}) \in \hat{D}_{\iota^r}.$$

Evidently, $g_n \sqsubseteq g_{n+1}$ for all $n \in \mathbf{N}$ and $g_{N_1,\ldots,N_r} = \bigsqcup_{n\in\mathbf{N}} g_n$ holds. By $\beta^{\iota^r}_{\Phi(n)} := g_n$ we define a function $\Phi \in R^{(1)}$. Theorem 5.5 implies the computability of $g_{N_1,\ldots,N_r}$. In analogous way to lemma 7.11 we obtain: Let $\emptyset \neq N_1, \ldots, N_r, M_1, \ldots, M_r \subseteq \mathbf{N}$ be r.e. and $N_i \leq_m M_i$ via functions $\psi_i \in R^{(1)}$ for all $1 \leq i \leq r$. Thus, one has $g_{N_1,\ldots,N_r}(x_1, \ldots, x_r) = g_{M_1,\ldots,M_r}(\psi_1(x_1), \ldots, \psi_r(x_r))$. As all recursive sets $\emptyset \neq M \subset \mathbf{N}$ are $\leq_m$-equivalent, for recursive sets $\emptyset \neq N_1, \ldots, N_r \subset \mathbf{N}$ we get

$$\mathcal{O} \equiv_p \mathcal{O}_r = g_{\{0\},\ldots,\{0\}} \equiv_p g_{N_1,\ldots,N_r}. \qquad \Box$$

In contrast to the generalization of *por* here recursivity is not necessary for $N_1, \ldots, N_r$, as one has to recognize only the arguments' membership to these sets.

9 Logically Sequential Functions

By definition it is not possible to study *dops* of logically sequential functions by means of sequentiality relations. A question suggesting itself is: Do the sequentiality relations characterize the PCF-definability in sufficient way, i.e. is every logically sequential function PCF-definable or formulated more generally: Do functions being invariant under the same sequentiality relations also have the same *dop*? In the case $D_\iota = \mathbf{N}_\perp$ this does not hold:

Lemma 9.1 *For all* $R \in SR$ *we have*

$$\exists \in R \Leftrightarrow f.a\ n \in \mathbf{N} : \exists_{n+1} \in R.$$

Proof: Lemma 4.11 implies "$\Rightarrow$". To show "$\Leftarrow$" we assume that there is a k-ary relation R such that $\exists_{n+1} \in R$ for all $n \in \mathbf{N}$, but $\exists \notin R$. Then it exists a tuple $(f_1, \ldots, f_k) \in R^{\iota^1}$ with $\exists_{n+1}(\vec{f}) \in R^\iota$ for all $n \in \mathbf{N}$ and $\exists(\vec{f}) \notin R^\iota$, i.e. $\exists_{n+1}(\vec{f}) \neq \exists(\vec{f})$ holds for all $n \in \mathbf{N}$. Hence, there is an $1 \leq i \leq k$ such that $\exists_{n+1}(f_i) = \perp$ for all $n \in \mathbf{N}$ and $\exists(f_i) = 0$. But this is impossible. $\Box$

As shown above all functions $\exists_{n+1}$ have the same *dop*. Therefore, we get: $\exists$ and $\exists_1$ are invariant under the same sequentiality relations in spite of $\exists_1 <_p \exists$.

The following result is from Sieber (cp. [Sie92]):

Theorem 9.2 *In the case $D_\iota = \mathbf{B}_\bot$ every logically sequential function with order ≤ 2 is PCF-definable.*

We begin the proof with

Lemma 9.3 *Let $\sigma = (\sigma_1, \ldots, \sigma_n, \iota) \in \Sigma$ be a type with $ord(\sigma) \leq 2$, $f \in D_\sigma$ a logically sequential function and $(d_{i,1}, \ldots, d_{i,n}) \in D_{\sigma_1} \times \ldots \times D_{\sigma_n}$ tuples for all $1 \leq i \leq m$ and $m \in \mathbf{N}$ arbitrary. Then there is a closed PCF-term M^σ such that $f(d_{i,1}, \ldots, d_{i,n}) = [\![M]\!](d_{i,1}, \ldots, d_{i,n})$ for all $1 \leq i \leq m$.*

Proof: We define an m-ary logical relation R by $(e_1, \ldots, e_m) \in R^\iota :\Leftrightarrow$ there is a closed PCF-term M^σ with $e_i = [\![M]\!](d_{i,1}, \ldots, d_{i,n})$ for all $1 \leq i \leq m$. It suffices to verify $(f(d_{1,1}, \ldots, d_{1,n}), \ldots, f(d_{m,1}, \ldots, d_{m,n})) \in R^\iota$.

1. $R \in SR$:

   For all $k \in \mathbf{N}$ and $1 \leq i \leq m$ we have $[\![\lambda x_1^\iota, \ldots, x_n^\iota.\underline{k}]\!](d_{i,1}, \ldots, d_{i,n}) = [\![\underline{k}]\!]$, thus $k \in R^\iota$. Let c be a PCF-constant of order 1, i.e. $[\![c]\!] \in D_{\iota^r}$, and $(u_1^{(1)}, \ldots, u_m^{(1)}), \ldots, (u_1^{(r)}, \ldots, u_m^{(r)}) \in R^\iota$. There are closed PCF-terms $\underline{U}^k$ for all $1 \leq k \leq r$ such that $u_i^{(k)} = [\![\underline{U}^k]\!](d_{i,1}, \ldots, d_{i,n})$ for all $1 \leq i \leq m$. This implies

   $$[\![c]\!](\vec{u}_i) = [\![\lambda x_1^\iota, \ldots, x_n^\iota.c(\underline{U}^1(x_1, \ldots, x_n), \ldots, \underline{U}^r(x_1, \ldots, x_n))]\!](d_{i,1}, \ldots, d_{i,n}).$$

   Therefore, $([\![c]\!](\vec{u}_1), \ldots, [\![c]\!](\vec{u}_m)) \in R^\iota$ holds, i.e. $[\![c]\!] \in R^{\iota^r}$. So all PCF-constants but the fixed point operators Y_σ are proved to be invariant under R. Theorem 3.4 implies $R \in SR$.

2. For all $1 \leq j \leq n$ we have $(d_{1,j}, \ldots, d_{m,j}) \in R^{\sigma_j}$:

   In case $\sigma_j = \iota$ one has $d_{i,j} = [\![\lambda x_1^\iota, \ldots, x_n^\iota.x_j]\!](d_{i,1}, \ldots, d_{i,n})$ for all $1 \leq i \leq m$. In case $\sigma_j = \iota^r$ choose for $1 \leq k \leq r$ an arbitrary $\vec{u}^{(k)} \in R^\iota$ and the corresponding term $\underline{U}^k$ as in item 1. Then

   $$d_{i,j}(\vec{u}_i) = [\![\lambda x_1^\iota, \ldots, x_n^\iota.x_j(\underline{U}^1(x_1, \ldots, x_n), \ldots, \underline{U}^r(x_1, \ldots, x_n))]\!](d_{i,1}, \ldots, d_{i,n})$$

   holds for all $1 \leq i \leq m$, i.e. $(d_{1,j}(\vec{u}_1), \ldots, d_{m,j}(\vec{u}_m)) \in R^\iota$.

Since f is logically sequential, item 1 implies $f \in R^\sigma$. Applying item 2 concludes the proof. $\square$

The previous lemma says that for every function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ with $ord(f) \leq 2$ a PCF-term exists whose semantics coincides on an arbitrary finite part of the definition domain with f. All domains D_σ can be approximated by finite ones in monotone ascending way:

Lemma 9.4 *For all $\sigma \in \Sigma$ there is a monotone sequence of finite projections $(p_i^\sigma \in [D_\sigma \to D_\sigma] \mid i \in \mathbf{N})$, i.e. for all $i \in \mathbf{N}$ we have $p_i^\sigma \circ p_i^\sigma = p_i^\sigma$, the set $p_i^\sigma(D_\sigma)$ is finite, and $p_i^\sigma \sqsubseteq p_{i+1}^\sigma$ holds. Moreover, $id_{D_\sigma} = \bigsqcup_{i\in\mathbf{N}} p_i^\sigma$.*

Proof: In the case $D_\iota = \mathbf{B}_\perp$ set $p_i^\sigma := id_{D_\sigma}$. So we look at $D_\iota = \mathbf{N}_\perp$. Obviously, the functions

$$p_i^\iota : d \mapsto \begin{cases} d & , \ d \in \{0, \ldots, i\} \\ \perp & , \ \text{otherwise} \end{cases}$$

for $i \in \mathbf{N}$ have the required properties. In case $\sigma = \sigma_1 \to \sigma_2$ set $p_i^\sigma(f) := p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1}$ for $f \in D_\sigma$. $p_i^\sigma \in [D_\sigma \to D_\sigma]$ is evident. Moreover, for all $i \in \mathbf{N}$ and $f \in D_\sigma$ we have $(p_i^\sigma \circ p_i^\sigma)(f) = p_i^{\sigma_2} \circ p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1} \circ p_i^{\sigma_1} = p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1} = p_i^\sigma(f)$ and by induction hypothesis $p_i^\sigma(f) = p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1} \sqsubseteq p_{i+1}^{\sigma_2} \circ f \circ p_{i+1}^{\sigma_1} = p_{i+1}^\sigma(f)$. Let $p_i^{\sigma_1}(D_{\sigma_1}) := \{s_1, \ldots, s_r\}$ be finite, then so is $p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1}(D_{\sigma_1}) := \{t_1, \ldots, t_k\}$. Define $T_j := (p_i^{\sigma_2})^{-1}(t_j) \subseteq D_{\sigma_2}$ for $1 \leq j \leq k$. We get $\sharp p_i^\sigma(D_\sigma) = \sharp\{p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1} \mid f \in D_\sigma\} \leq k^r$, because only the mapping by f from $\{s_1, \ldots, s_r\}$ to $\{T_1, \ldots, T_k\}$ must be considered. By induction hypothesis one has $p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1} \sqsubseteq f$ for all $i \in \mathbf{N}$ and $f \in D_\sigma$. Thus, $\bigsqcup_{i\in\mathbf{N}} p_i^\sigma$ is well-defined. We obtain

$$\bigsqcup_{i\in\mathbf{N}} p_i^\sigma(f) = \bigsqcup_{i\in\mathbf{N}} p_i^{\sigma_2} \circ f \circ p_i^{\sigma_1} = \bigsqcup_{i,j\in\mathbf{N}} p_i^{\sigma_2} \circ f \circ p_j^{\sigma_1} = \bigsqcup_{i\in\mathbf{N}} p_i^{\sigma_2} \circ f \circ \bigsqcup_{j\in\mathbf{N}} p_j^{\sigma_1} = f. \qquad \square$$

All projections are definable:

$$p_i^\iota = [\![\lambda x^\iota.cond(x \dot{\leq} \underline{i}, x, \Omega^\iota)]\!]$$
$$p_i^\sigma = [\![\lambda y^\sigma, x^{\sigma_1}.\underline{p}_i^{\sigma_2}(y(\underline{p}_i^{\sigma_1}(x)))]\!].$$

Applying this construction to lemma 9.3 we get

Theorem 9.5 *Let $\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$ be a type with $ord(\sigma) \leq 2$ and $f \in D_\sigma$ a logically sequential function. Then f is a lub of a directed set of PCF-definable elements.*

Proof: Because of $f = \bigsqcup_{i\in\mathbf{N}} p_i^\sigma(f)$ it remains to show that $p_i^\sigma(f)$ is definable for all $i \in \mathbf{N}$. For all $g \in D_\sigma$ and $d_1 \in D_{\sigma_1}, \ldots, d_n \in D_{\sigma_n}$ we have

$$\begin{aligned}(p_i^\sigma(g))(d_1)\ldots(d_n) &= p_i^{(\sigma_2,\ldots,\sigma_n,\iota)} \circ \underbrace{g \circ p_i^{\sigma_1}(d_1)}_{\in D_{(\sigma_2,\ldots,\sigma_n,\iota)}} \ldots (d_n) \\ &= p_i^{(\sigma_3,\ldots,\sigma_n,\iota)} \circ g(p_i^{\sigma_1}(d_1)) \circ p_i^{\sigma_2}(d_2) \ldots (d_n) \\ &= \ldots = p_i^\iota \circ g(p_i^{\sigma_1}(d_1))(p_i^{\sigma_2}(d_2)) \ldots (p_i^{\sigma_n}(d_n)). \ [1]\end{aligned}$$

Since $p_i^{\sigma_1}(D_{\sigma_1}) \times \ldots \times p_i^{\sigma_n}(D_{\sigma_n})$ is finite, lemma 9.3 implies the existence of a closed PCF-term M such that for all $d_1 \in D_{\sigma_1}, \ldots, d_n \in D_{\sigma_n}$:

$$f(p_i^{\sigma_1}(d_1), \ldots, p_i^{\sigma_n}(d_n)) = [\![M]\!](p_i^{\sigma_1}(d_1), \ldots, p_i^{\sigma_n}(d_n)) \quad [2].$$

Therefore, we get

$$\begin{aligned} p_i^{\sigma}(f) &= [\![\lambda d_1^{\sigma_1}, \ldots, d_n^{\sigma_n}.\underline{p}_i^{\iota}(\underline{f}(\underline{p}_i^{\sigma_1}(d_1), \ldots, \underline{p}_i^{\sigma_n}(d_n)))]\!] \quad \text{by } [1] \\ &= [\![\lambda d_1^{\sigma_1}, \ldots, d_n^{\sigma_n}.\underline{p}_i^{\iota}(M(\underline{p}_i^{\sigma_1}(d_1), \ldots, \underline{p}_i^{\sigma_n}(d_n)))]\!] \quad \text{by } [2] \\ &= p_i^{\sigma}([\![M]\!]) = [\![\underline{p}_i^{\sigma}(M)]\!]. \qquad \Box \end{aligned}$$

Corollary 9.6 *Let $\sigma = (\sigma_1, \ldots, \sigma_n, \iota)$ be a type of order ≤ 2 and $f \in \hat{D}_\sigma$ a logically sequential function. Then f is PCF-definable.*

Proof: Because of $f \sqsubseteq \bigsqcup_{i \in \mathbf{N}} p_i^{\sigma}(f)$ and the compactness of f there is an $i_0 \in \mathbf{N}$ such that $f \sqsubseteq p_{i_0}^{\sigma}(f)$. Thus, $f = p_{i_0}^{\sigma}(f)$ holds. $\Box$

This corollary implies directly theorem 9.2, as in the case $D_\iota = \mathbf{B}_\perp$ every $f \in \bigcup_{\sigma \in \Sigma} D_\sigma$ is compact.

10 Chains and Antichains

Up to now we have given a description of only finitely many *dops*. But there are infinitely many ones, even ordered sequences of them.

Definition 10.1 *A sequence $\{\langle f_i \rangle \in \bigcup_{\sigma \in \Sigma} D_\sigma \mid i \in \mathbf{N}\}$ is called a* descending, *resp. an* ascending chain *of dops, if $f_{i+1} <_p f_i$, resp. $f_{i+1} >_p f_i$ holds for all $i \in \mathbf{N}$. A sequence $\{\langle f_i \rangle \in \bigcup_{\sigma \in \Sigma} D_\sigma \mid i \in \mathbf{N}\}$ is called an* antichain *of dops, if $f_i \mid_p f_j$ holds for all $i \neq j$.*

Definition 10.2 *A function $f \in \bigcup_{\sigma \in \Sigma} D_\sigma$ is said to be* p-complete *for $A \subseteq \bigcup_{\sigma \in \Sigma} D_\sigma$, if $g \leq_p f$ holds for all $g \in A$, and there is a function $g_0 \in A$ such that $f \equiv_p g_0$.*

Theorem 10.3
1. *In the cases $D_\iota = \mathbf{N}_\perp$ and $D_\iota = \mathbf{B}_\perp$ it exists an descending chain above $\exists_1$, which por is p-complete for.*
2. *In the case $D_\iota = \mathbf{N}_\perp$ it exists a chain descending from por $\oplus \exists$ and above $\exists$.*
3. *In the cases $D_\iota = \mathbf{N}_\perp$ and $D_\iota = \mathbf{B}_\perp$ it exists a descending chain, which gu is p-complete for and there are only logically sequential dops below.*

The **proof** of the previous theorem covers the following pages. We start with a generalization of V:

Definition 10.4 *For all $n, m \in \mathbf{N}$ with $n < m < 2n$ set*

$$V_{n,m} \in D_{\iota^m} : \ (x_1, \dots, x_m) \mapsto \begin{cases} x & , \ ex.\ I \subseteq \{1, \dots, m\} : (\sharp I \geq n \ \wedge \ f.a.\ i \in I : x_i = x) \\ \perp & , \ otherwise. \end{cases}$$

The function $V_{n,m}$ checks, whether n of the m arguments coincide, and assigns this value to the arguments. Obviously, it is $V = V_{2,3}$.

$\triangleright$ For all $n \geq 1$ we have $por \leq_p V_{n+1,2n+1}$:

$$por = [\![\lambda x^\iota, y^\iota . \underline{V}_{n+1,2n+1}(\underbrace{Z(x), \dots, Z(x)}_{n-\text{times}}, \underbrace{Z(y), \dots, Z(y)}_{n-\text{times}}, \underline{0})]\!].$$

Lemma 10.5 *For all $n, m, z \in \mathbf{N}$ with $n < m < 2n$ and $z \geq 2$ we have*

$$V_{n,m} \in \tilde{z} \Leftrightarrow z \leq \lceil \frac{m}{m-n} \rceil - 1.$$

Proof: "$\Rightarrow$": Let $z \geq \lceil \frac{m}{m-n} \rceil$ and $k := m \bmod (m-n) \leq m-n$. Because of $z(m-n) \geq m$ a $\perp$ is standing in each argument column of the following table, but no one in the value column, i.e. $V_{n,m}$ is not invariant under $\tilde{z}$:

$$
\begin{array}{rl|c}
 & x_1 \qquad\qquad \dots \qquad\qquad x_m & V_{n,m}(\vec{x}) \\ \hline
\lceil \frac{m}{m-n} \rceil \text{ rows} & \left\{ \begin{array}{l}
0\dots0\ 0\dots0 \quad 0\dots0 \quad 0\dots0\ 0\dots0 \quad \underbrace{\perp\dots\perp}_{(m-n)-\text{times}} \\
0\dots0\ 0\dots0 \quad 0\dots0 \quad 0\dots0 \quad \underbrace{\perp\dots\perp}_{(m-n)-\text{times}} \quad 0\dots0 \\
\qquad\qquad \vdots \\
\underbrace{0\dots0}_{k-\text{times}} \quad \perp\dots\perp \quad \perp\dots\perp \quad \underbrace{0\dots0 \quad 0\dots0 \quad 0\dots0}_{(n-k)-\text{times}}
\end{array} \right. & \begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \end{array} \\
(z - \lceil \frac{m}{m-n} \rceil) - \text{times} & \left\{ \begin{array}{l}
\underbrace{\perp\dots\perp\ \perp\dots\perp}_{(m-n)-\text{times}} \quad 1\dots1 \quad 1\dots1 \quad 1\dots1 \quad 1\dots1 \\
\qquad\qquad \vdots \\
\underbrace{\perp\dots\perp\ \perp\dots\perp}_{(m-n)-\text{times}} \quad 1\dots1 \quad 1\dots1 \quad 1\dots1 \quad 1\dots1
\end{array} \right. & \begin{array}{c} 1 \\ \vdots \\ 1 \end{array}
\end{array}
$$

"$\Leftarrow$": In the following table let $z \leq \lceil \frac{m}{m-n} \rceil - 1$ and $\vec{x}_1, \dots, \vec{x}_m \in \tilde{z}$:

x_1	$\dots$	x_m	$V_{n,m}(\vec{x})$
$x_1^{(1)}$	$\dots$	$x_m^{(1)}$	e_1
$\vdots$		$\vdots$	$\vdots$
$x_1^{(z)}$	$\dots$	$x_m^{(z)}$	e_z

We assume $\perp \notin \{e_1, \dots, e_z\}$. Because of $z(m-n) < m$ there is some argument column which contains no $\perp$ and in which the arguments coincide with the corresponding value.

Therefore, all arguments in this column coincide, i.e. $e_1 = \ldots = e_z$ holds and we get $V_{n,m} \in \tilde{z}$. □

Consequences:

- For $m \geq n \geq 3$ we have $V_{n-1,n} \notin \tilde{m}$ because of $m \geq \lceil \frac{n}{n-(n-1)} \rceil = n$.
- For $m \leq n-1$ and $n \geq 3$ we have $V_{n-1,n} \in \tilde{m}$ because of $m \leq \lceil \frac{n}{n-(n-1)} \rceil - 1 = n-1$.
- For all $n \geq 3$ we have $V_{n,n+2} \notin \tilde{n}$ because of $n = 2x \Rightarrow x \geq 1 \Rightarrow 2x \geq x+1 = \lceil \frac{2x+2}{2} \rceil = \lceil \frac{n+2}{n+2-n} \rceil$ and $n = 2x+1 \Rightarrow x \geq 1 \Rightarrow 2x+1 \geq x+2 \geq \lceil \frac{2x+3}{2} \rceil = \lceil \frac{n+2}{n+2-n} \rceil$.

$\triangleright$ Thus, we obtain $V_{n,n+1} \in \tilde{m} \Leftrightarrow m \leq n$, which implies $V = V_{2,3} \not\leq_p V_{3,4} \not\leq_p V_{4,5} \not\leq_p \ldots$ and $V_{n,n+1} \not\leq_p \exists$.

$\triangleright$ $V_{n,m} \not\leq_p V'$: We have

$$
\begin{array}{c|c}
x_1 \qquad \ldots \qquad x_m & V_{n,m}(\vec{x}) \\
\hline
\underbrace{0 \quad \ldots \quad 0}_{n-\text{times}} \perp \ldots \perp & 0 \\
\perp \ldots \perp \underbrace{0 \quad \ldots \quad 0}_{n-\text{times}} & 0 \\
\perp \ldots \perp \underbrace{0 \ldots 0}_{k-\text{times}} \underbrace{\perp \ldots \perp}_{(n-k)-\text{times}} & \perp,
\end{array}
$$

because $2n > m$ implies $k \geq 1$ and $n < m$ implies $k < n$. So one gets $V_{n,m} \notin (1,2 \mid 3)$ and $V' \in (1,2 \mid 3)$.

A helpful combinatory lemma is

Lemma 10.6 *Let $m, n \in \mathbf{N}$ be arbitrary with $n-1 < m-1 < 2(n-1)$. Then any familiy A with m elements contains n equal elements if, and only if, in each of the m subfamilies $A_1, \ldots, A_m$ of A with $m-1$ elements an element $\hat{a}_1, \ldots$,resp. $\hat{a}_m$ appears $(n-1)$-times and, moreover, $\hat{a}_1 = \ldots = \hat{a}_m$ holds.*

Proof:

"$\Rightarrow$": Let $A = (\underbrace{a_1, \ldots, a_1}_{n-\text{times}}, a_{m-n}, \ldots, a_m)$, $A_1 = (\underbrace{a_1, \ldots, a_1}_{(n-1)-\text{times}}, a_{m-n}, \ldots, a_m)$, $\ldots$, $A_m = (\underbrace{a_1, \ldots, a_1}_{n-\text{times}}, a_{m-n}, \ldots, a_{m-1})$. Then we have $\hat{a}_1 = \ldots = \hat{A}_m = a_1$.

"$\Leftarrow$": Let $A_m = (\underbrace{\hat{a}_m, \ldots, \hat{a}_m}_{(n-1)-\text{times}}, a_n, \ldots, a_{m-1})$ and $A_1 = (\underbrace{\hat{a}_m, \ldots, \hat{a}_m}_{(n-2)-\text{times}}, a_n, \ldots, a_m)$. One should notice that $n-1 < m-1 < 2(n-1)$ implies $n \geq 3$. In case $\hat{a}_m = a_m$ we have found a subfamiliy of A with n elements. So we assume that $\hat{a}_m \neq a_m$ holds. Then it remains to verify that there is an $n \leq i \leq m-1$ such that $a_i = \hat{a}_m$.

- Case $m-(n-1)<n-1$: The $n-1$ equal elements in A_1 are not contained in $\{a_n,\dots,a_m\}$. Thus, the assertion is proved.

- Case $m-(n-1)\geq n-1$: We have $n\leq\frac{m}{2}+1$. By assumption $\frac{m-1}{2}<n-1$ holds. m is not odd, because $m=2x+1$ implies $x<n-1\leq x+\frac{1}{2}$. Thus, $m=2x$ holds and we have $x-\frac{1}{2}<n-1\leq x$ and $n=x+1$. $A_1=(\underbrace{\hat{a}_m,\dots,\hat{a}_m}_{(x-1)-\text{times}},a_{x+1},\dots,a_{2x})$ contains $n-1$ equal elements. These can not be $a_{x+1},\dots,a_{2x}$, otherwise $\hat{a}_m=\hat{a}_1=a_{2x}$ holds. The case that there is an $x+1\leq i\leq 2x-1$ with $a_i=\hat{a}_m$ is left. $\square$

$\triangleright$ Consequence: $V_{n,m}\leq_p V_{n-1,m-1}$, because

$$
\begin{aligned}
V_{n,m}=[\![\lambda x_1^\iota,\dots,x_m^\iota.cond(&\underline{V}_{n-1,m-1}\ (x_1,\dots,x_{m-1})\doteq\underline{V}_{n-1,m-1}(x_1,x_3\dots,x_m)\\
&\qquad\qquad\underline{\triangle}\cdots\underline{\triangle}\\
&\underline{V}_{n-1,m-1}(x_1,\dots,x_{m-2},x_m)\doteq\underline{V}_{n-1,m-1}(x_2,\dots,x_m),\\
&\underline{V}_{n-1,m-1}(x_1,\dots,x_{m-1}),\\
&\Omega^\iota)]\!]
\end{aligned}
$$

holds.

$\triangleright$ Further relations between *dops* of the $V_{n,m}$-functions (provided that the indices are admissible):

- $V_{n,m}\leq_p V_{n,m+1}$: $V_{n,m}=[\![\lambda x_1^\iota,\dots,x_m^\iota.(x_1,\dots,x_m,\Omega^\iota)]\!]$.

- $V_{n,m}\leq_p V_{n-1,m}$ because of previous $V_{n-1,m-1}\leq_p V_{n-1,m}$.

- $V_{n,n+1}<_p V_{n,n+2}$ because of $V_{n,n+1}\leq_p V_{n,n+2}$ and $V_{n,n+1}\in\tilde{n}$, but $V_{n,n+2}\notin\tilde{n}$.

- $V_{n+1,2n+1}\equiv_p por$: $\geq_p$ has already been shown. Moreover, we have

  $V_{3,5}=[\![\lambda x_1^\iota,\dots,x_5^\iota.\underline{por}_{\binom{5}{3}}(x_1\doteq x_2\,\underline{\triangle}\,x_2\doteq x_3,x_1\doteq x_2\,\underline{\triangle}\,x_2\doteq x_4,\dots,x_3\doteq x_4\,\underline{\triangle}\,x_4\doteq x_5)]\!]$.

  In analogous way one gets more generally $V_{n+1,2n+1}\leq_p por_{\binom{2n+1}{n+1}}\equiv_p por$.

- In the case $D_\iota=\mathbf{B}_\perp$ we have $V_{n,m}\leq_p V_{n+1,m+2}$ by

  $$V_{n,m}=[\![\lambda x^\iota,\dots,x_m^\iota.\underline{V}_{n+1,m+2}(x_1,\dots,x_m,\underline{0},\underline{1})]\!]$$

  and $por\equiv_p V_{2n,3n}$ by

  $$por=[\![\lambda x_1^\iota,x_2^\iota.\underline{V}_{2n,3n}(\underbrace{Z(x_1),\dots,Z(x_1)}_{n-\text{times}},\underbrace{Z(x_2),\dots,Z(x_2)}_{n-\text{times}},\underbrace{\underline{0},\dots,\underline{0}}_{n-\text{times}})]\!].$$

  Therefore, $\frac{m}{n}\geq\frac{3}{2}\Rightarrow por\equiv_p V_{n,m}$ holds, because, as shown above, a decreasing of n or an increasing of m effects an increasing w.r.t. the *dop*.

Some special relations between these *dops* can be seen in the table

	$V_{2,3}$	$V_{3,4}$	$V_{4,5}$	$V_{4,6}$	$V_{5,6}$	$V_{5,7}$	$V_{5,8}$	$V_{6,7}$	$V_{6,8}$	$V_{6,9}$	$V_{6,10}$	$V_{7,8}$	$V_{7,9}$	$V_{7,10}$	$V_{7,11}$
$\tilde{3}$	$\notin$	$\in$	$\in$	$\notin$	$\in$	$\in$	$\notin$	$\in$	$\in$	$\notin$	$\notin$	$\in$	$\in$	$\in$	$\notin$
$\tilde{4}$	$\notin$	$\notin$	$\in$	$\notin$	$\in$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\notin$	$\in$	$\in$	$\notin$	$\notin$
$\tilde{5}$	$\notin$	$\notin$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\notin$
$\tilde{6}$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\notin$
$\tilde{7}$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\notin$	$\in$	$\notin$	$\notin$	$\notin$

(to read as: $V_{2,3} \notin \tilde{3}, V_{3,4} \in \tilde{3}$ etc.) and with the help of the constructions

1. $M_{3,4} := \lambda \dot{x}_1^\iota, \ldots, \dot{x}_4^\iota.cond(conv(\underline{V}_{4,6}(\dot{x}_1, \ldots, \dot{x}_4, \dot{x}_1, \dot{x}_2) \,\underline{\wedge}\, conv(\underline{V}_{4,6}(\dot{x}_1, \ldots, \dot{x}_4, \dot{x}_3, \dot{x}_4)),$
   $\underline{V}_{4,6}(\dot{x}_1, \ldots, \dot{x}_4, \dot{x}_1, \dot{x}_3), \Omega^\iota)$
2. $M_{4,5} := \lambda \dot{x}_1^\iota, \ldots, \dot{x}_5^\iota.cond(conv(\underline{V}_{5,7}(\dot{x}_1, \ldots, \dot{x}_5, \dot{x}_1, \dot{x}_2) \,\underline{\wedge}\, conv(\underline{V}_{5,7}(\dot{x}_1, \ldots, \dot{x}_5, \dot{x}_3, \dot{x}_4)),$
   $\underline{V}_{5,7}(\dot{x}_1, \ldots, \dot{x}_5, \dot{x}_1, \dot{x}_2), \Omega^\iota)$
3. $M_{5,6} := \lambda \dot{x}_1^\iota, \ldots, \dot{x}_6^\iota.cond(conv(\underline{V}_{6,8}(\dot{x}_1, \ldots, \dot{x}_6, \dot{x}_1, \dot{x}_2) \,\underline{\wedge}\, conv(\underline{V}_{6,8}(\dot{x}_1, \ldots, \dot{x}_6, \dot{x}_3, \dot{x}_4)$
   $\underline{\wedge} conv(\underline{V}_{6,8}(\dot{x}_1, \ldots, \dot{x}_6, \dot{x}_5, \dot{x}_6)),$
   $\underline{V}_{6,8}(\dot{x}_1, \ldots, \dot{x}_6, \dot{x}_1, \dot{x}_2), \Omega^\iota).$

$\triangleright$ Without reckoning all cases explicitly one has

1. $[\![M_{3,4}]\!](\vec{x}) \neq \bot \Rightarrow (x_1 = x_3 \;\vee\; V_{3,4}(\vec{x}) \neq \bot) \;\wedge\; (x_1 = x_2 \;\vee\; V_{3,4}(\vec{x}) \neq \bot) \;\wedge\; (x_3 = x_4 \;\vee\; V_{3,4}(\vec{x}) \neq \bot) \Rightarrow V_{3,4}(\vec{x}) \neq \bot$ and $V_{3,4}(\vec{x}) \neq \bot \Rightarrow [\![M_{3,4}]\!](\vec{x}) = V_{3,4}(\vec{x})$, i.e. $[\![M_{3,4}]\!] = V_{3,4}$ and therefore $V_{3,4} <_p V_{4,6}$.
2. $[\![M_{4,5}]\!](\vec{x}) \neq \bot \Rightarrow ((ex.\ i \in \{3,4,5\} : x_1 = x_2 = x_i) \;\vee\; V_{4,5}(\vec{x}) \neq \bot) \;\wedge\; ((ex.\ i \in \{1,2,5\} : x_3 = x_4 = x_i) \;\vee\; V_{4,5}(\vec{x}) \neq \bot) \Rightarrow V_{4,5}(\vec{x}) \neq \bot$ and $V_{4,5}(\vec{x}) \neq \bot \Rightarrow [\![M_{4,5}]\!](\vec{x}) = V_{4,5}(\vec{x})$, i.e. $[\![M_{4,5}]\!] = V_{4,5}$ and therefore $V_{4,5} <_p V_{5,7}$.
3. $V_{5,6} <_p V_{6,8}$ analogous to the previous cases.

A generalization of these relations to the whole $V_{n,m}$-subsemilattice is still lacking.

To analyze the relation between $\exists_1$ and $V_{n,m}$ we define for all $m, n \in \mathbf{N}$ with $1 \leq m \leq n+1$ and $n \geq 1$:

$$\exists_{m,n} \in D_{\iota^1,\iota},\ f \mapsto \begin{cases} 0 & \text{, there are m zeros of f in } \{0, \ldots, n\} \\ 1 & ,\ f(\bot) = 1 \\ \bot & \text{, otherwise.} \end{cases}$$

$\triangleright$ For all $i \in \mathbf{N}$ we have

$$\exists_{1,n} = \exists_n = [\![\lambda f^{\iota^1}.\underline{\exists}_{i+1,n+1}(\lambda x^\iota.cond(x \dot{\geq} \underline{n+1}, \underline{0}, f(x)))]\!],$$

i.e. $\exists_n \leq_p \exists_{i+1,n+i}$ and therefore $\exists_{m,n} \geq_p \exists_{n-m+1} \equiv_p \exists_n$ holds. A function f has m zeros in $\{0,\dots,n\}$ if, and only if, there is a zero in each subset of $\{0,\dots,n\}$ with $n-m+2$ elements. So we get $\exists_{m,n} \leq_p \exists_n$ and hence $\exists_{m,n} \equiv_p \exists_n$.

$$\exists_{n,n} = [\![\lambda f^\iota.cond(\underline{V}_{n,n+1}(f(\underline{0}),\dots,f(\underline{n})), \underline{0}, cond(\underline{V}_{n,n+1}(f(\underline{0}),\dots,f(\underline{n})) \doteq \underline{1}, f(\Omega^\iota), \Omega^\iota))]\!]$$

implies $\exists_n \equiv_p \exists_{n,n} \leq_p V_{n,n+1} <_p V_{n-1,n}$. We obtain $\exists_n \equiv_p \exists_{m+1} <_p V_{m,m+1}$.

A chain satisfying the conditions in theorem 10.3 item 1 is given by

Corollary 10.7 *$\{V_{n,n+1} \mid n \geq 2\}$ forms a descending chain of dops above $\exists_1$, for which por is p-complete, i.e.*

$$\exists_1 <_p \dots <_p V_{5,6} <_p V_{4,5} <_p V_{3,4} <_p V_{2,3} \equiv_p por.$$

holds. The diagrams 6 and 7 show the relations between the dops. □

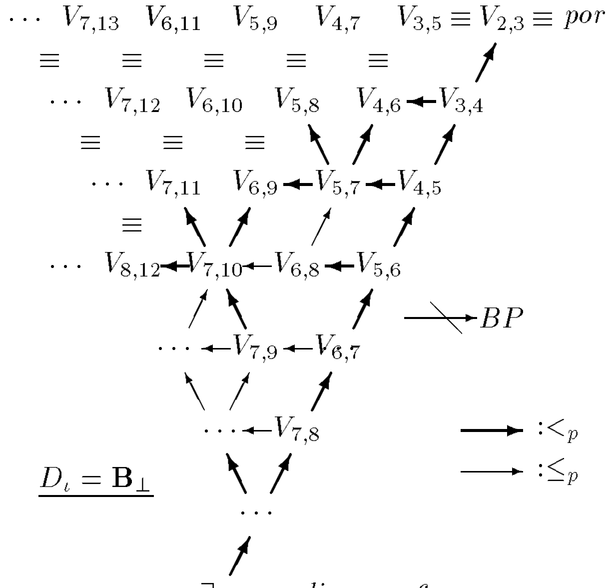

diagram 6

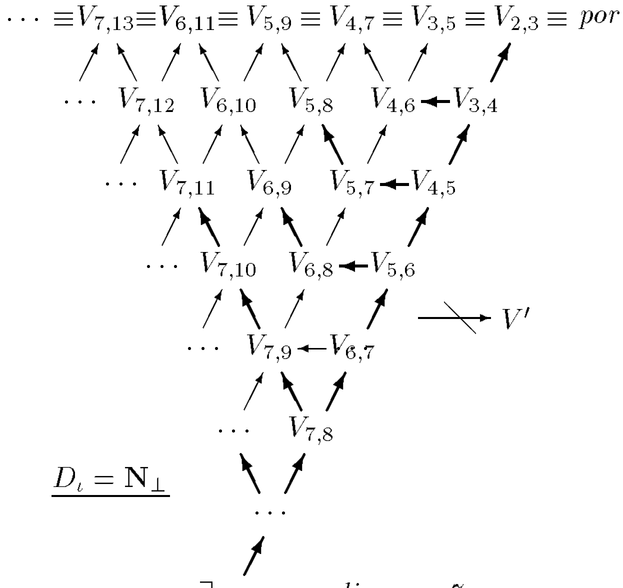

diagram 7

By lifting we get a chain like demanded by theorem 10.3 item 2:

Corollary 10.8 *In the case $D_\iota = \mathbf{N}_\perp$ the relations between the dops represented in diagram 9 hold.*

Proof: We have $V_{n,n+1} \,|_p\, \exists$ for all $n \geq 2$, because $\exists \not\leq_p por$ implies $\exists \not\leq_p V_{n,n+1}$, and $V_{n,n+1} \notin \widetilde{n+1}$ and $\exists \in \widetilde{n+1}$ imply $V_{n,n+1} \not\leq_p \exists$. Furthermore, $\exists \in \widetilde{n+2}$, $V_{n+1,n+2} \in \widetilde{n+2}$ and $V_{n,n+1} \notin \widetilde{n+2}$ hold and therefore it is $V_{n,n+1} \oplus \exists >_p V_{n+1,n+2} \oplus \exists$. □

The chain in theorem 10.3 item 3 is constructed by a generalization of *gu*.

$$\begin{tikzcd}
\exists \arrow[r] & \cdots \arrow[r] & V_{5,6}\oplus\exists \arrow[r] & V_{4,5}\oplus\exists \arrow[r] & V_{3,4}\oplus\exists \arrow[r] & V_{2,3}\oplus\exists \equiv por\oplus\exists \\
\exists_{n+1} \arrow[r] \arrow[u] & \cdots \arrow[r] & V_{5,6} \arrow[r] \arrow[u] & V_{4,5} \arrow[r] \arrow[u] & V_{3,4} \arrow[r] \arrow[u] & V_{2,3} \equiv por \arrow[u]
\end{tikzcd}$$

diagram 8

Definition 10.9 *For any $n \geq 3$ we define $gu_n \in D_{\iota^n}$ by*

$$\begin{aligned} tr(gu_n) := &\{((\perp, 0, \ldots, 0, n+1), 0) \mid n \in \mathbf{N}\} \cup \{((n+1, \perp, 0, \ldots, 0), 0) \mid n \in \mathbf{N}\} \\ &\cup \ldots \cup \{((0, \ldots, 0, n+1, \perp), 0) \mid n \in \mathbf{N}\}. \end{aligned}$$

Obviously, $gu = gu_3$ holds.

Lemma 10.10 *For all $n \geq 3$ we have $gu_n \in (1, \ldots, m-1 \mid m) \Leftrightarrow m \leq n$.*

Proof: "$\Rightarrow$": For $m > n$ the table

	$x_1 \ \ldots \ x_n$	$gu_n(\vec{x})$
	$\perp 0 \ldots 01$	0
	$1 \perp 0 \ldots 0$	0
	$\ddots \ \ddots$	$\vdots$
$(m-n)$ – times	$0 \ldots 01 \perp$	0
	$\vdots$	$\vdots$
	$0 \ldots 01 \perp$	0
	$1 \ \ldots \ 1$	$\perp$,

holds, i.e. $gu_n \notin (1, \ldots, m-1 \mid m)$.

"$\Leftarrow$": In the following table let $m \leq n$ and $\vec{x}_1, \ldots, \vec{x}_n \in (1, \ldots, m-1 \mid m)$:

x_1	$\ldots$	x_n	$gu_n(\vec{x})$
$x_1^{(1)}$	$\ldots$	$x_m^{(1)}$	0
$\vdots$		$\vdots$	$\vdots$
$x_1^{(m-1)}$	$\ldots$	$x_n^{(m-1)}$	0
$x_1^{(m)}$	$\ldots$	$x_n^{(m)}$	e_m

We have to show $e_m = 0$. We call the respective $m-1$ upper elements of an argument column the corresponding "top column". Because of $n > m-1$ there is a top column without $\perp$, i.e. all arguments in the column coincide. If $n+1$ is this value, $n-2$ of the top columns contain only the value 0 and the assertion is proved. Assume there is no top column containing only values of the form $n+1$. Because of the "rotationsymmetry" of the function w.r.t to the arguments we can further assume that the first column is constant 0. Since we have $e_1 = 0$, there is a $2 \le k < n$ with $x_k^{(1)} = n+1$. The k-th top column is not constant $n+1$. So there is an $2 \le i_1 < m$ with $x_k^{(i_1)} = \perp$. Because of $e_k = 0$ we have $x_{k-1}^{(i_1)} = m+1$. Again there is an $2 \le i_2 < m$ such that $i_2 \ne i_1$, $x_{k-1}^{(i_2)} = \perp$ and $x_{k-2}^{(i_2)} = l+1$. In case $k > m$ these conclusions go on till: There is an $2 \le i_{m-1} < m$ with $i_{m-1} \notin \{i_1, \ldots, i_{m-2}\}$ and $x_{k-m+2}^{(i_{m-1})} = \perp$. But such an i_{m-1} does not exist. In case $k \le m$ we obtain a contradiction at the step: There is an $2 \le i_{k-1} < m$ such that $i_{k-1} \notin \{i_1, \ldots, i_{k-2}\}$ and $x_1^{(i_{k-1})} = r+1$. By assumption we have $x_1^{(i_{k-1})} = 0$ namely. $\square$

$\triangleright$ Consequence: For all $n \ge 3$ we have $gu_n \not\le_p gu_{n+1}$.

Lemma 10.11 *$gu_{n+1} <_p gu_n$ holds for all $n \ge 3$.*

Proof: It remains to verify $gu_{n+1} \le_p gu_n$.

Defining $M := \lambda \dot{x}_1^\iota, \ldots, \dot{x}_{n+1}^\iota . \underline{gu}_n(\dot{x}_1 \underline{\triangle} \dot{x}_{n+1}, \dot{x}_2, \ldots, \dot{x}_{n-1}, cond(\dot{x}_n, \dot{x}_{n+1}, cond(\dot{x}_1, \underline{1}, \Omega^\iota)))$.

we get

1. $gu_{n+1}(\vec{x}) \ne \perp \Rightarrow [\![M]\!](\vec{x}) = gu_{n+1}(\vec{x})$
2. $[\![M]\!](\vec{x}) \ne \perp \Rightarrow gu_{n+1}(\vec{x}) = [\![M]\!](\vec{x})$.

ad 1:

$$
\begin{aligned}
&[\![M]\!](\perp, 0, \ldots, 0, m+1) = gu_n(\perp, 0, \ldots, 0, m+1) = 0,\\
&[\![M]\!](m+1, \perp, 0, \ldots, 0) = gu_n(1, \perp, 0, \ldots, 0) = 0,\\
&[\![M]\!](0, \ldots, 0, m+1, \perp, 0, \ldots, 0) = gu_n(0, \ldots, 0, m+1, \perp, 0, \ldots, 0) = 0 \text{ (just for } n \ge 4),\\
&[\![M]\!](0, \ldots, 0, m+1, \perp, 0) = gu_n(0, \ldots, 0, m+1, \perp) = 0,\\
&[\![M]\!](0, \ldots, 0, m+1, \perp) = gu_n(\perp, 0, \ldots, 0, 1) = 0.
\end{aligned}
$$

ad 2:

Defining $x_1' := \dot{x}_1 \underline{\triangle} \dot{x}_{n+1}$, $x_i' := \dot{x}_i$ for $2 \le i \le n-1$ and $x_n' := cond(\dot{x}_n, \dot{x}_{n+1}, cond(\dot{x}_1, \underline{1}, \Omega^\iota))$ we obtain

$$
\begin{aligned}
([\![x_i']\!])_{1\le i\le n} = ([\![x_1']\!], 0, \ldots, 0, m+1) &\Rightarrow ([\![x_i]\!])_{1\le i\le n+1} = ([\![x_1]\!], 0, \ldots, 0, m+1)\\
&\qquad \vee ([\![x_i]\!])_{1\le i\le n+1} = (0, \ldots, 0, m+1, [\![x_{n+1}]\!]),\\
([\![x_i']\!])_{1\le i\le n} = (m+1, [\![x_2']\!], 0, \ldots, 0) &\Rightarrow ([\![x_i]\!])_{1\le i\le n+1} = (m+1, [\![x_2']\!], 0, \ldots, 0, 0),\\
([\![x_i']\!])_{1\le i\le n} = (0, \ldots, 0, m+1, [\![x_i']\!], 0, \ldots, 0)&\\
&\Rightarrow ([\![x_i]\!])_{1\le i\le n+1} = (0, \ldots, 0, m+1, [\![x_i']\!], 0, \ldots, 0, 0),\\
([\![x_i']\!])_{1\le i\le n} = (0, \ldots, 0, m+1, [\![x_n']\!]) &\Rightarrow ([\![x_i]\!])_{1\le i\le n+1} = (0, \ldots, 0, m+1, [\![x_n]\!], 0).
\end{aligned}
$$

Thus, $[\![M]\!] = gu_{n+1}$ holds. □

With the following lemma the proof of theorem 10.3 is done.

Lemma 10.12 *Every function $f \in \bigcup_{\sigma\in\Sigma} D_\sigma$ with $f <_p gu_n$ for all $n \geq 3$ is logically sequential.*

Proof: Without essentially modifying lemma 10.10 can be generalized to:

$$\text{For all } k \in \mathbf{N} \text{ we have } gu_n \in (1, \ldots, m-1 \mid m, \ldots, m+k) \Leftrightarrow m \leq n.$$

Invariance under a relation $S^n_{A,B}$ depends only on the size of A, but not on the concretely chosen indices out of B. So we get for all $n, k, r, m \in \mathbf{N}$ and for all permutations $i \in \Sigma_k$:

$$((k \geq m > r \ \wedge \ r < n) \ \vee \ k \geq m = r) \Rightarrow gu_n \in S^k_{\{i_1,\ldots,i_r\},\{i_1,\ldots,i_m\}}.$$

Let $R \in SR$ be arbitrary with arity k. With existing only finitely many relations of the form $S^k_{A,B}$ one has the representation $R^\iota = \bigcap_{1\leq i\leq r} S^k_{A_i,B_i}$. Defining

$$\tilde{r}(R) := \max\{\sharp A_i \mid \sharp B_i > \sharp A_i, 1 \leq i \leq r\} + 1$$

we have $gu_{\tilde{r}(R)} \in S^k_{A_i,B_i}$ for all $1 \leq i \leq r$. Because of $ord(gu_{\tilde{r}(R)}) = 1$ we get $gu_{\tilde{r}(R)} \in R$ and therefore $f \in R$. □

Corollary 10.6 implies

Corollary 10.13 *In the case $D_\iota = \mathbf{B}_\perp$ we have for any function f with $f <_p gu_n$ for all $n \geq 3$: f is PCF-definable or $ord(f) \geq 3$.* □

So far we have constructed descending chains, only. A further generalization of gu due to Bucciarelli (cp. [Buc95]) for $D_\iota = \mathbf{B}_\perp$ yields ascending and descending chains as well as antichains. We introduce now the required notions:

Definition 10.14 *A set $A = \{(x_1^{(1)}, \ldots, x_1^{(n)}), \ldots, (x_k^{(1)}, \ldots, x_k^{(n)})\} \subseteq \mathbf{B}^n_\perp$ is said to be* (linearly) coherent, *if $(x_1^{(i)}, \ldots, x_k^{(i)}) \in \tilde{k}$ for all $1 \leq i \leq n$. The set of all coherent subsets of $\mathbf{B}^n_\perp$ is called $\mathcal{C}(\mathbf{B}^n_\perp)$.*

Definition 10.15 *A continuous function $f : \mathbf{B}^n_\perp \to \mathbf{B}^m_\perp$ is said to be* (linearly) strong stable, *if $f(A) \in \mathcal{C}(\mathbf{B}^m_\perp)$ and $f(\sqcap A) = \sqcap f(A)$ whenever $A \in \mathcal{C}(\mathbf{B}^n_\perp)$. Here $\sqcap A$ is standing for the greatest lower bound (glb) of A w.r.t. to the partial order of $\mathbf{B}^n_\perp$.*

Notice that every subset $A \neq \emptyset$ of a b-complete cpo X has a glb, that is $\sqcap A = \bigsqcup\{x \in X \mid f.a.\ a \in A :\ a \sqsubseteq x\}$.

These functions, compatible with coherent sets, are definable by PCF-terms:

Lemma 10.16 *Every strong stable function $f : \mathbf{B}_\perp^n \to \mathbf{B}_\perp$ is PCF-definable.*

Proof: By corollary 9.6 we have to show that f is invariant under each sequentiality relation. Because of $ord(f) = 1$ this holds, if f is invariant under each relation $S_{A,B}^k$. Let $\vec{v}_i := (v_i^{(1)}, \ldots, v_i^{(k)}) \in S_{A,B}^k$ for all $1 \le i \le n$ and $f(\vec{v}^{(j)}) \ne \perp$ for all $j \in A$. Then we have $(v_i^{(j)} \mid j \in A) \in \widetilde{\sharp A}$ for all $1 \le i \le n$, thus $\{\vec{v}^{(j)} \mid j \in A\} \in \mathcal{C}(\mathbf{B}_\perp^n)$. The strong stability of f implies $\{f(\vec{v}^{(j)}) \mid j \in A\} \in \mathcal{C}(\mathbf{B}_\perp)$. By assumption we have $f(\vec{v}^{(j)}) \ne \perp$ for all $j \in A$. Therefore, $f(\vec{v}^{(j_1)}) = f(\vec{v}^{(j_2)})$ holds for all $j_1, j_2 \in A$. We get $\perp \ne \sqcap\{f(\vec{v}^{(j)}) \mid j \in A\} = f(\sqcap\{(\vec{v}^{(j)}) \mid j \in A\}) =: f(\alpha_1, \ldots, \alpha_n)$. In case $\alpha_i \ne \perp$ one has $v_i^{(j)} \sqsupseteq \alpha_i$ for all $j \in A$ and by assumption $v_i^{(j)} \sqsupseteq \alpha_i$. We obtain $f(v_1^{(j)}, \ldots, v_n^{(j)}) \sqsupseteq f(\alpha_1, \ldots, \alpha_n) \ne \perp$ for all $j \in B$, i.e. $f(\vec{v}^{(j_1)}) = f(\vec{v}^{(j_2)})$ holds for all $j_1, j_2 \in B$. □

The coherent subsets in the trace of a function are interesting:

Definition 10.17 *For any function $f : \mathbf{B}_\perp^n \to \mathbf{B}_\perp$ with $tr(f) = \{(v_i, b_i) \mid 1 \le i \le k\}$ the set*

$$H(f) := \{\{i_1, \ldots, i_l\} \subseteq \{1, \ldots, k\} \mid \sharp\{i_1, \ldots, i_l\} \ge 2, \{v_{i_1}, \ldots, v_{i_l}\} \in \mathcal{C}(\mathbf{B}_\perp^n)\}$$

is called the hypergraph *of f. Let $f' : \mathbf{B}_\perp^m \to \mathbf{B}_\perp$ be any function with $tr(f') = \{(v_i', b_i') \mid 1 \le i \le k'\}$. Then the mapping $\varphi : \{1, \ldots, k\} \to \{1, \ldots, k'\}$ is said to be a* morphism *from $H(f)$ to $H(f')$, if $\varphi(A) \in H(f')$ for all $A \in H(f)$.*

$H(BP) = \{\{1, 2, 3\}\}$ und $H(por) = \{\{1, 2\}, \{1, 2, 3\}\}$ are examples for hypergraphs.

A first relation between *dops* and hypergraphs we get by

Lemma 10.18 *For all functions $f : \mathbf{B}_\perp^n \to \mathbf{B}_\perp$ and $g : \mathbf{B}_\perp^m \to \mathbf{B}_\perp$ such that $H(f) \ne \emptyset \ne H(g)$ and $\min\{\sharp A \mid A \in H(f)\} < \min\{\sharp A \mid A \in H(g)\}$ we have $f \not\le_p g$.*

Helpful for the proof is

Lemma 10.19 *Let $A \in \mathcal{C}(\mathbf{B}_\perp^n)$ and $B \subseteq \mathbf{B}_\perp^n$ an Egli-Milner lower bound of A, i.e.*

$$(f.a.\ a \in A :\ ex.\ b \in B :\ b \sqsubseteq a) \text{ and } (f.a.\ b \in B :\ ex.\ a \in A :\ b \sqsubseteq a).$$

Then we have $B \in \mathcal{C}(\mathbf{B}_\perp^n)$.

Proof: Let $A := \{v^{(1)}, \ldots, v^{(k)}\}$, $B := \{w^{(1)}, \ldots, w^{(k')}\}$ such that $w_j^{(i')} \ne \perp$ for all $1 \le i' \le k'$ for an arbitrary $1 \le j \le n$. Then for all $1 \le i \le k$ there is an i' with $w_j^{(i')} \sqsubseteq v_j^{(i)}$, i.e. $v_j^{(i)} \ne \perp$ holds for all $1 \le i \le k$. $A \in \mathcal{C}(\mathbf{B}_\perp^n)$ implies $v_j^{(i_1)} = v_j^{(i_2)} \ne \perp$ for all $1 \leq i_1, i_2 \leq k$. By assumption for all $1 \leq i' \leq k'$ there is an i with $\perp \neq w_j^{(i')} \sqsubseteq v_j^{(i)}$, i.e. $w_j^{(i')} = v_j^{(i)}$. We obtain $w_j^{(i'_1)} = w_j^{(i'_2)}$ for all $1 \leq i'_1, i'_2 \leq k'$. $\square$

Proof of lemma 10.18:

Define $k := \min\{\sharp A \mid A \in H(f)\}$ and $S := (1, \ldots, k \mid k+1)$. We show that $g \in S$, but $f \notin S$.

Let $\{\vec{v}_1, \ldots, \vec{v}_k\} \in \mathcal{C}(\mathbf{B}_\perp^n) \cap \pi_1(tr(f))$. We set for $1 \leq j \leq n$:

$$\vec{w}^{(j)} := (v_1^{(j)}, \ldots, v_k^{(j)}, \sqcap\{v_k^{(j)} \mid 1 \leq j \leq k\}).$$

We have $\vec{w}^{(j)} \in S$, but $f(\vec{w}^{(1)}, \ldots, \vec{w}^{(n)}) = (b_1, \ldots, b_k, \perp) \notin S$. Therefore it is $f \notin S$.

Assuming there is a set $U := \{\vec{u}_1, \ldots, \vec{u}_{k+1}\} \subseteq \mathbf{B}_\perp^m$ such that $(u_1^{(j)}, \ldots, u_{k+1}^{(j)}) \in S$ for all $1 \leq j \leq m$ and $g(\vec{u}_1, \ldots, \vec{u}_{k+1}) \notin S$, we get $U \in \mathcal{C}(\mathbf{B}_\perp^m)$ and $g(\vec{u}_l) \neq \perp$ for all $1 \leq l \leq k$, i.e. $\vec{u}_l$ has a lower bound in $\pi_1(tr(g))$. Hence, it exists an Egli-Milner lower bound A of U in $\pi_1(tr(g))$ with $\sharp A \leq k$. The previous lemma implies $A \in \mathcal{C}(\mathbf{B}_\perp^m)$. In case $\sharp A = 1$ there is a $\vec{v} \in \pi_1(tr(g))$ such that $\vec{u}_i \sqsupseteq \vec{v}$ for all $1 \leq i \leq k$. For any $1 \leq j \leq m$ we get ($ex.\ 1 \leq j \leq k :\ u_i^{(j)} = \perp) \Rightarrow u_{k+1}^{(j)} \sqsupseteq v^{(j)} = \perp$ and ($f.a.\ 1 \leq i \leq k :\ u_i^{(j)} \neq \perp) \Rightarrow v^{(j)} \sqsubseteq u_1^{(j)} = \ldots = u_{k+1}^{(j)}$, i.e. $\vec{u}_{k+1} \sqsupseteq \vec{v}$ and thus $g(\vec{u}_1) = \ldots = g(\vec{u}_{k+1})$ and $(g(\vec{u}_1), \ldots, g(\vec{u}_{k+1})) \in S$. This contradicts the assumption. So $1 < \sharp A \leq k$ must hold. But this is a contradiction to the conditions of the lemma. We conclude $g \in S$. $\square$

The inversion does not hold, e.g. we have $BP \not\leq_p \mathcal{O}$, but $3 = \min\{\sharp A \mid A \in H(BP)\} > \min\{\sharp A \mid A \in H(\mathcal{O})\} = 2$ because of $H(\mathcal{O}) = \{\{1,2\},\{1,2,3\}\}$. To use lemma 10.18 Bucciarelli defined special subsets of $\mathcal{P}_{fin}(\{1, \ldots, m\})$ and functions whose hypergraphs are these sets.

Definition 10.20 *For any $m \geq n \geq 3$ we write $h_{n,m}$ for the set $\{A \subseteq \{1, \ldots, m\} \mid \sharp A \geq n\}$.*

The existence of a morphism from $h_{n,m}$ to $h_{n',m'}$ can be expressed in combinatorial way:

Lemma 10.21 *For any $m \geq n \geq 3$ and $m' \geq n' \geq 3$ the following conditions are equivalent:*

1. *There is a morphism $\varphi : \{1, \ldots, m\} \to \{1, \ldots, m'\}$ from $h_{n,m}$ to $h_{n',m'}$.*

2.

$$C^{m,n',m'} := \min\{n'-1, m \bmod m'\}(\lceil \frac{m}{m'} \rceil + 1) + \max\{0, n'-1-(m \bmod m')\}\lceil \frac{m}{m'} \rceil > n$$

Proof:

φ is a morphism from $h_{n,m}$ to $h_{n',m'}$.
$\Leftrightarrow$ For all $\{i_1,\ldots,i_k\} \subseteq \{1,\ldots,m\}$ with $\sharp\{i_1,\ldots,i_k\} \geq n$ we have $\sharp\{\varphi(i_1),\ldots,\varphi(i_k)\} \geq n'$.
$\Leftrightarrow$ For all $B \subseteq \{1,\ldots,m'\}$ with $\sharp B < n'$ we have $\sharp\varphi^{-1}(B) < n$.
$\Leftrightarrow$ $\max\{\sharp\varphi^{-1}(B) \mid B \subseteq \{1,\ldots,m'\}, \sharp B = n'-1\} < n$.

Hence, there is a morphism from $h_{n,m}$ to $h_{n',m'}$ if, and only if

$$n > \min_{\varphi:\{1,\ldots,m\}\to\{1,\ldots,m'\}} \max\{\sharp\varphi^{-1}(B) \mid B \subseteq \{1,\ldots,m'\}, \sharp B = n'-1\}.$$

Now we need a function $\varphi_0 : \{1,\ldots,m\} \to \{1,\ldots,m'\}$, so that the maximal number of elements in $\{1,\ldots,m\}$ being mapped onto $n'-1$ pairwise different elements is minimal, i.e. φ_0 has to be "as injective as possible". For example we can choose

$$\varphi_0 := i \mapsto ((i-1) \bmod m') + 1, \text{ d.h. } 1 \mapsto 1,\ \ldots,\ m' \mapsto m',\ m'+1 \mapsto 1,\ \ldots\ .$$

Then we have

$$k := \max\{\sharp\varphi_0^{-1}(B) \mid B \subseteq \{1,\ldots,m'\}, \sharp B = n'-1\} = \sharp\varphi_0^{-1}(\{1,\ldots,n'-1\}).$$

- Case $n'-1 \leq m \bmod m'$:

  $\varphi_0(\{m-(m \bmod m')+1,\ldots,m\})$ covers $\{1,\ldots,n'-1\}$, i.e.

  $$k = (n'-1)\lceil\frac{m}{m'}\rceil = (n'-1)(\lfloor\frac{m}{m'}\rfloor+1)$$

  holds.

- Case $n'-1 > m \bmod m'$:

  We have

  $$k = (n'-1)\lfloor\frac{m}{m'}\rfloor + m \bmod m' = (m \bmod m')(\lfloor\frac{m}{m'}\rfloor+1) + (n'-1-(m \bmod m'))\lfloor\frac{m}{m'}\rfloor.$$

We summarize:

$$k = \min\{n'-1, m \bmod m'\}(\lceil\frac{m}{m'}\rceil+1) + \max\{0, n'-1-(m \bmod m')\}\lceil\frac{m}{m'}\rceil = C^{m,n',m'}.$$

□

Construction of functions $f_{n,m}$ with hypergraph $H(f_{n,m}) = h_{n,m}$ for $m \geq n \geq 3$:

Set $k(n,m) := \sum_{i=2}^{n-1}\binom{m}{i}$. That is the number of subsets of $\{1,\ldots,m\}$ with $2 \leq i \leq n-1$ elements. We define the trace of $f_{n,m} : \mathbf{B}_\perp^{k(n,m)} \to \{\perp, 0\} \subset \mathbf{B}_\perp$ with m elements represented as a matrix by the following criterion: The m rows are standing for the elements $\{1,\ldots,m\}$, and the $k(n,m)$ columns for all subsets of $\{1,\ldots,m\}$ with $2 \leq i \leq$

$n-1$ elements, whereat, viewed from above, the first contained element is marked with 0, all further with 1 and each not contained element with $\perp$:

$$\begin{array}{ccccccccccccc|c}
x_1 & & & & & & \cdots & & & & & x_{k(n,m)} & f_{n,m}(\vec{x}) \\ \hline
 & & & & & & & & 0 & & & \perp & 0 \\
0 & 0 & & \perp & 0 & & \perp & & 1 & & & \vdots & \\
1 & \perp & & \vdots & 1 & & \vdots & (n-2) & \vdots & & & \perp & \\
\perp & 1 & \cdots & \perp & 1 & \cdots & \perp & \text{-times} & 1 & \cdots & & 0 & \vdots \\
\perp & \perp & & \perp & \perp & & 0 & & \perp & & & 1 & \\
\vdots & \vdots & & 0 & \vdots & & 1 & & \vdots & & (n-2) & \vdots & \\
\perp & \perp & & 1 & \perp & & 1 & & \perp & & \text{-times} & 1 & 0
\end{array}$$

The function gu coincides with $f_{3,3}$. For all $A \subseteq \{1,\ldots,m\}$ with $\sharp A \geq n$ the corresponding submatrix of $\pi_1(tr(f_{n,m}))$ contains an $\perp$ in each column, i.e. the set is coherent. Whenever $2 \leq \sharp A < n$ there is a column $(0,1,\ldots,1) \notin \widetilde{\sharp A}$ in the corresponding submatrix, i.e. A is not coherent. Therefore, we get

$$f.a.\ A \subseteq \pi_1(tr(f_{n,m})):\ A \in \mathcal{C}(\mathbf{B}_\perp^n) \Leftrightarrow \sharp A \geq n.$$

That means $H(f_{n,m}) = h_{n,m}$.

The relations between the *dops* of these functions can be expressed in dependence on the indices m, n, m', n':

Theorem 10.22 *For all $m \geq n \geq 3$ we have $f_{n,m} <_p f_{n',m'} \Leftrightarrow n > C^{m,n',m'}$.*

Proof: "$\Leftarrow$": Set $k := \sum_{i=2}^{n-1}\binom{m}{i}$, $k' := \sum_{i=2}^{n'-1}\binom{m}{i}$, $A := \{v_1,\ldots,v_m\} := \pi_1(tr(f_{n,m}))$ and $B := \{w_1,\ldots,w_{m'}\} := \pi_1(tr(f_{n',m'}))$. By the previous lemma there is a morphism φ from $h_{n,m}$ to $h_{n',m'}$.

1. auxiliary lemma: $tr(F) := \{(v_i, w_{\varphi(i)} \mid 1 \leq i \leq m\}$ defines a strong stable function $F : \mathbf{B}_\perp^k \to \mathbf{B}_\perp^{k'}$.

Proof: Let $C \in \mathcal{C}(\mathbf{B}_\perp^k)$.

- $F(C) \in \mathcal{C}(\mathbf{B}_\perp^{k'})$:

  If $(\underbrace{\perp,\ldots,\perp}_{k'-\text{times}}) \in F(C)$, then $F(C)$ clearly is coherent. In case $(\perp,\ldots,\perp) \notin F(C)$ there exists an Egli-Milner lower bound C' of C such that $C' \subseteq \pi_1(tr(F))$ and $F(C) = F(C') = \{w_{\varphi(i)} \mid v_i \in C'\}$. By lemma 10.19 we have $C' \in \mathcal{C}(\mathbf{B}_\perp^{k'})$. Since φ is a morphism, we get $\sharp F(C') \geq n'$ and $F(C') \subseteq \pi_1(tr(f_{n',m'}))$. Hence $F(C') \in \mathcal{C}(\mathbf{B}_\perp^{k'})$.

- $F(\sqcap C) = \sqcap F(C)$:

  $(\perp, \ldots, \perp) \in F(C)$ obviously implies $\sqcap F(C) = (\perp, \ldots, \perp) = F(\sqcap C)$. Therefore, let $(\perp, \ldots, \perp) \notin F(C)$ and C' as in the previous item. In case $\sharp C' = 1$, i.e. $C' = \{v_i\}$, we have $v_i \sqsubseteq \sqcap C$ and $F(C) = \{F(v_i)\}$, thus $F(v_i) = \sqcap F(C) \sqsupseteq F(\sqcap C) \sqsupseteq F(v_i)$, i.e. $\sqcap F(C) = F(\sqcap C)$. In case $\sharp C' \geq 2$ we have $F(C') \in \mathcal{C}(\mathbf{B}_\perp^{k'})$ and $\sharp F(C') \geq 2$. Moreover, $F(C') \in \pi_1(tr(f_{n',m'}))$ implies $\sharp F(C') \geq n'$. Hence, each column of $F(C')$ contains a $\perp$ and we obtain $F(\sqcap C) \sqsubseteq \sqcap F(C) = \sqcap F(C') = (\perp, \ldots, \perp)$, i.e. $F(\sqcap C) = \sqcap F(C)$.

$\square$(1. auxiliary lemma)

2. auxiliary lemma: For all $1 \leq i \leq n$ the projection $\pi_i : \mathbf{B}_\perp^n \to \mathbf{B}_\perp$, $(x_1, \ldots, x_n) \mapsto x_i$ is strong stable.

Proof: Analogous to theorem 2.12 the functions π_i are continuous. Whenever $\{v_1, \ldots, v_k\} \in \mathcal{C}(\mathbf{B}_\perp^n)$ one has $(v_1^{(j)}, \ldots, v_k^{(j)}) \in \tilde{k}$ for all $1 \leq j \leq n$ and thus $\pi_i(\{v_1, \ldots, v_k\}) = \{v_1^{(i)}, \ldots, v_k^{(i)}\}$ $\in \mathcal{C}(\mathbf{B}_\perp)$. Furthermore, $\pi_i(\sqcap\{v_1, \ldots, v_k\}) = \sqcap\{v_1^{(i)}, \ldots, v_k^{(i)}\} = \sqcap\, \pi_i(\{v_1, \ldots, v_k\})$.

$\square$(2. auxiliary lemma)

3. auxiliary lemma: Strong stability is preserved by composition.

Proof: Obviously. $\square$

The auxiliary lemmata imply that for all $1 \leq i \leq k'$ the functions $g_i := \pi_i \circ \mathbf{B}_\perp^k \to \mathbf{B}_\perp$ are strong stable and hence by lemma 10.16 PCF-definable. Define

$$M := \lambda y^{\iota^{k'}}, x_1^\iota, \ldots, x_k^\iota . y(\underline{g}_1(x_1, \ldots, x_k), \underline{g}_2(x_1, \ldots, x_k), \ldots, \underline{g}_{k'}(x_1, \ldots, x_k)).$$

For all $v \in \mathbf{B}_\perp^k$ we have

$f_{n',m'} \circ F(v) = 0 \Leftrightarrow (ex.\ j \leq m' :\ F(v) \sqsupseteq w_j) \Leftrightarrow (ex :\ i \leq m :\ F(v) \sqsupseteq w_{\varphi(i)})$
$\Leftrightarrow (ex.\ i \leq m :\ v \sqsupseteq v_i) \Leftrightarrow f_{n,m}(v) = 0.$

By identifying F with $g_1 \times \ldots \times g_{k'}$ we have shown $f_{n',m'} \circ (g_1 \times \ldots \times g_{k'}) = f_{n,m}$. We obtain $[\![M]\!](f_{n',m'}) = f_{n,m}$.

"$\Rightarrow$": Defining an $(m+1)$-ary sequentiality relation by

$$R := \Big(\bigcap_{A \subseteq \{1,\ldots,m\}, \sharp A = n} S_{A,A}^{m+1}\Big) \cap (1, \ldots, m \mid m+1)$$

we will verify that $f_{n,m} \notin R$, but $f_{n',m'} \in R'$.

In $\pi_1(tr(f_{n,m})) := \{(x_1^{(1)}, \ldots, x_1^{(k(n,m))}), \ldots, (x_m^{(1)}, \ldots, x_m^{(k(n,m))})\}$ by construction every column $\{(x_1^{(i)}, \ldots, x_m^{(i)})\}$ with $1 \leq i \leq k(n,m)$ contains the element $\perp$ at least $(m - n + 1)$ times. So for all $1 \leq i \leq k(n,m)$ every subset of the tuple $w_i := (x_1^{(i)}, \ldots, x_m^{(i)})$

with n elements contains a $\perp$, i.e. these tuples are invariant under R. But we have $f_{n,m}(w_1, \ldots, w_{k(n,m)}) = (0, \ldots, 0, \perp) \notin R$, and hence $f_{n,m} \notin R$.

Assume that $f_{n',m'} \notin R$. Then there are $\vec{y}^{(1)}, \ldots, \vec{y}^{(k(n',m'))} \in R$ with $f_{n',m'}(\vec{y}_1, \ldots, \vec{y}_{m+1}) \notin R$. As $f_{n',m'}$ merely assumes the values 0 and $\perp$, the value tuple is invariant under each relation $S^{m+1}_{A,A}$. Therefore, only the relation $(1, \ldots, m \mid m+1)$ can make problems, i.e. we would have $f_{n',m'}(\vec{y}_1) = \ldots = f_{n',m'}(\vec{y}_m) = 0 \;\wedge\; f_{n',m'}(\vec{y}_{m+1}) = \perp$. Then for all $1 \leq i \leq m$ there is a $\vec{w}_{\varphi(i)} \in \pi_1(tr(f_{n',m'}))$ with $\vec{y}_i \sqsupseteq \vec{w}_{\varphi(i)}$ for a suitable function $\varphi : \{1, \ldots, m\} \to \{1, \ldots, m'\}$. Because of $n \geq C^{m,n',m'}$ the function φ is no morphism, i.e. it exists a set $B \subseteq \{1, \ldots, m'\}$ such that $\sharp B = n' - 1$ and $\sharp\varphi^{-1}(B) \geq n$. Thus, there are n elements $\vec{y}_{i_1}, \ldots, \vec{y}_{i_n}$ with $1 \leq i_j \leq m$ and moreover $\vec{w}_1, \ldots, \vec{w}_k \in \pi_1(tr(f_{n',m'}))$ for some $1 \leq k \leq n' - 1$ such that $\{\vec{w}_1, \ldots, \vec{w}_k\}$ is Egli-Milner lower than $\{\vec{v}_{i_1}, \ldots, \vec{v}_{i_n}\}$.

- Case $k \geq 2$:

  $\{\vec{y}_{i_1}, \ldots, \vec{y}_{i_n}\}$ is not coherent, otherwise $\{\vec{w}_1, \ldots, \vec{w}_k\}$ were also coherent by lemma 10.19 and because of $\{\vec{w}_1, \ldots, \vec{w}_k\} \subseteq \pi_1(tr(f_{n',m'}))$ we would have $k \geq n'$. Hence, there is an $1 \leq h \leq k(n', m')$ such that $\{y^{(h)}_{i_1}, \ldots, y^{(h)}_{i_n}\} = \{0, 1\}$. We get $(y^{(h)}_1, \ldots, y^{(h)}_{m+1}) \notin S^{m+1}_{\{i_1,\ldots,i_n\},\{i_1,\ldots,i_n\}}$, a contradiction to the assumption.

- Case $k = 1 \;\wedge\; ex.\ 1 \leq j \leq m :\ \vec{y}_j \not\sqsupseteq \vec{w}_1$:

  Here we have $\vec{w} \in \pi_1(tr(f_{n',m'}))$ with $\vec{w} \neq \vec{w}_1$ and $\vec{y}_j \sqsupseteq \vec{w}$. Like in the previous case $\{\vec{v}_{i_1}, \ldots, \vec{v}_{i_n}, \vec{v}_j\}$ is not coherent, otherwise $\{\vec{w}_1, \vec{w}\}$ would be coherent and hence $n' \leq 2$. Again there is an $1 \leq h \leq k(n, m)$ with $\vec{y}^{(h)} \notin S^{m+1}_{\{i_1,\ldots,i_n\},\{i_1,\ldots,i_n\}}$, a contradiction to the assumption.

- Case $k = 1 \;\wedge\; f.a.\ 1 \leq i \leq m :\ \vec{y}_i \sqsupseteq \vec{w}_1$:

  We have $y^{(j)}_1, \ldots, y^{(j)}_m \sqsupseteq w^{(j)}_1$ for all $1 \leq j \leq k(n', m')$. $\vec{y}^{(j)} \in S^{m+1}_{\{i_1,\ldots,i_n\},\{i_1,\ldots,i_n\}}$ implies

  $$f.a.\ 1 \leq i \leq m : y^{(j)}_i = \perp \Rightarrow w^{(j)}_1 = \perp$$

  and

  $$(f.a.\ 1 \leq i \leq m :\ y^{(j)}_i \neq \perp) \Rightarrow y^{(j)}_{m+1} = y^{(j)}_1,$$

  i.e. $y^{(j)}_{m+1} \sqsupseteq w^{(j)}_1$. So we get $\vec{y}_{m+1} \sqsupseteq \vec{w}_1$ and therefore $f_{n',m'}(\vec{y}_{m+1}) = 0$, again a contradiction.

We obtain $f_{n',m'} \in R$. □

With the help of this relation one can construct a subsemilattice of the *dops* including ascending chains, descending ones and antichains. Notice that the order of every function is 1.

$$\begin{tikzcd}
f_{3,3} \arrow[r] & f_{3,4} \arrow[r] & f_{3,5} \arrow[r] & f_{3,6} \arrow[r] & f_{3,7} \arrow[r] & f_{3,8} \arrow[r] & \cdots \\
f_{4,4} \arrow[u] \arrow[r] & f_{4,5} \arrow[u] \arrow[r] & f_{4,6} \arrow[u] \arrow[r] & f_{4,7} \arrow[u] \arrow[r] & f_{4,8} \arrow[u] \arrow[r] & f_{4,9} \arrow[u] \arrow[r] & \cdots \\
f_{5,5} \arrow[u] \arrow[r] & f_{5,6} \arrow[u] \arrow[r] & f_{5,7} \arrow[u] \arrow[r] & f_{5,8} \arrow[u] \arrow[r] & f_{5,9} \arrow[u] \arrow[r] & f_{5,10} \arrow[u] \arrow[r] & \cdots \\
f_{6,6} \arrow[u] \arrow[r] & f_{6,7} \arrow[u] \arrow[r] & f_{6,8} \arrow[u] \arrow[r] & f_{6,9} \arrow[u] \arrow[r] & f_{6,10} \arrow[u] \arrow[r] & f_{6,11} \arrow[u] \arrow[r] & \cdots \\
f_{7,7} \arrow[u] \arrow[r] & f_{7,8} \arrow[u] \arrow[r] & f_{7,9} \arrow[u] \arrow[r] & f_{7,10} \arrow[u] \arrow[r] & f_{7,11} \arrow[u] \arrow[r] & f_{7,12} \arrow[u] \arrow[r] & \cdots \\
\vdots \arrow[u] & \vdots \arrow[u] & \vdots \arrow[u] & \vdots \arrow[u] & \vdots \arrow[u] & \vdots \arrow[u] &
\end{tikzcd}$$

diagram 9

Corollary 10.23 *For the functions $f_{n,m}$ as defined above we have the relations between their dops as shown in diagram 9.*

Proof:

- For all $m \geq 3, n \geq 0$ we have

  $$C^{m+n,m,m+n+1} = \min\{m-1, m+n\} \cdot 1 + \max\{0, m-1-(m+n)\} \cdot 0 = m-1 < m$$

  and

  $$C^{m+n+1,m,m+n} = \min\{m-1, 1\} \cdot 2 + \max\{0, m-2\} \cdot 1 = 2+m-2 = m,$$

  i.e. $f_{m,m} <_p f_{m,m+1} <_p f_{m,m+2} <_p \ldots$.

- For all $m \geq n \geq 3$ we have

  $$C^{m+1,n,m} = \min\{n-1, 1\} \cdot 2 + \max\{0, n-2\} \cdot 1 = 2+n-2 = n < n+1$$

  and

  $$C^{m,n+1,m+1} = \min\{n, m\} \cdot 1 + \max\{0, n-m\} \cdot 0 = n,$$

  i.e. $f_{3,n} >_p f_{4,n+1} >_p f_{5,n+2} >_p \ldots$. □

We get antichains, if the "gradient" of the indices $\frac{\Delta m}{\Delta n}$ for a sequence $(f_{n_i,m_i} \mid i \in \mathbf{N})$ is large enough:

Lemma 10.24 *For all $m \geq n \geq 3$ and $m' \geq n' \geq 3$ with $\frac{m'-m}{n'-n} > \frac{m'}{n'-1}$ we have $f_{n,m} \not\leq_p f_{n',m'}$.*

Proof: Defining $b := m \bmod m'$ and $k := \lceil \frac{m}{m'} \rceil$ we get

$$\begin{aligned} C^{m,n',m'} &= \min\{n'-1, b\}(k+1) + \max\{0, n'-1-b\}k \\ &= \begin{cases} (n'-1)(k+1) & , n'-1 \leq b \\ b + (n'-1)k & , n'-1 > b \end{cases} \\ &= \min\{n'-1, b\} + (n'-1)k \end{aligned}$$

We have to verify $C^{m,n',m'} \geq n$.

1. Case $m < m'$:

   We have $b = m$ and $k = 0$. Therefore, $C^{m,n',m'} = \min\{n'-1, m\}$ holds. Because of $\frac{m'-m}{n'-n} > \frac{m'}{n'-1} > 1$ it is $n' > n$, thus $n'-1 \geq n$. By definition it is $m \geq n$, i.e. $\min\{n'-1, m\} \geq n$.

2. Case $m > m' \ \wedge \ n'-1 \geq b$:

$$\begin{aligned} C^{m,n',m'} & b + (n'-1)k \\ &> b + n' - 1 + (k-1)m'\tfrac{n'-n}{m'-m} && , (\tfrac{n'-1}{m'} > \tfrac{n'-n}{m'-m}) \\ &\geq b\tfrac{n'-n}{m'-m} + n' - 1 + (k-1)m'\tfrac{n'-n}{m'-m} && , (\tfrac{n'-n}{m'-m} < 1) \\ &= (b + km' - m')\tfrac{n'-n}{m'-m} + n' - 1 \\ &= (m - m')\tfrac{n-n'}{m-m'} + n' - 1 \\ &= n - 1 \end{aligned}$$

3. Case $m > m' \ \wedge \ n'-1 < b$:

$$\begin{aligned} C^{m,n',m'} &= (n'-1)(k+1) \\ &= \tfrac{n-1}{m}m'(k+1) && , (\tfrac{m'-m}{n'-n} > \tfrac{m'}{n'-1}) \\ &= \tfrac{n-1}{m}m && , (\tfrac{m}{m'} < k+1) \\ &= n - 1 \end{aligned}$$

□

From this derives the existence of antichains:

Corollary 10.25 *For all $x \geq 3$ and $p, q \in \mathbf{N}$ with $\frac{p}{q} > \frac{x}{x-1}$ the sequence*

$$\{f_{x+n, x+\frac{p}{q}n} \mid n \in q\mathbf{N}\}$$

forms an antichain.

Proof: For all $n, m \in \mathbf{N}$ and $r > \frac{x}{x-1}$, provided that the indices are natural numbers, $f_{x+n,x+rn} \mid_p f_{x+m,x+rm}$ holds because of

$$\frac{x+rn-(x+rm)}{x+n-(x+m)} = r > \frac{x+rm}{x+m-1} \Leftrightarrow r > \frac{x}{x-1} \Leftrightarrow r > \frac{x+rn}{x+n-1}$$

Then lemma 10.24 concludes the proof. $\square$

The transformations in the previous corollary permit a further generalization:

For all $x \in \mathbf{Q}, x > 1$ and $p, q \in \mathbf{N}$ with $\frac{p}{q} > \frac{x}{x-1}$

$$\{f_{x+s,x+\frac{p}{q}s} \mid s \in \mathbf{Q}^+, x+s, x+\frac{p}{q}s \in \{n \in \mathbf{N} \mid n \geq 3\}\}$$

forms an antichain.

Concrete examples for antichains according to the previous corollary are:

- $x = 3, \frac{p}{q} = 2: \ \{f_{3,3}, f_{4,5}, f_{5,7}, f_{6,9}, \ldots\}$
- $x = 4, \frac{p}{q} = \frac{3}{2} > \frac{4}{3}: \ \{f_{4,4}, f_{6,7}, f_{8,10}, \ldots\}$
- $x = \frac{3}{2}, \frac{p}{q} = 5 > 3: \ \{f_{3,9}, f_{4,14}, f_{5,19}, \ldots\}$.

11 Stability

This short section is intended to insert the functions of order 1 which Berry has considered in order to achieve full abstraction of PCF, in the subsemilattice of the *dobs* in the case $D_\iota = \mathbf{B}_\perp$.

Definition 11.1 *A function $f \in D_{(\sigma,\tau)}$ is said to be* stable, *if for all $x, y \in D_\sigma$:*

$$x \uparrow^\sigma y \Rightarrow f(x \sqcap y) = f(x) \sqcap f(y).$$

Stable functions of order 1 can be characterized with the help of their traces, resp. their hypergraphs:

Lemma 11.2 *For any continuous function $f \in \bigcup_{\sigma \in \Sigma} D_\sigma$ of order 1 the following conditions are equivalent:*

1. *f is stable.*
2. *For all $v_1, v_2 \in \pi_1(tr(f))$ we have $v_1 \not\uparrow v_2$.*
3. *For all $A \in H(f)$ we have $\sharp A \neq 2$.*

Proof: "1 ⇒ 2": Assume there are $v_1, v_2 \in \pi_1(tr(f))$ with $v_1 \uparrow v_2$. Then it is $v_1 \not\sqsubseteq v_2$, $v_2 \not\sqsubseteq v_1$ and $f(v_1) = f(v_2)$. Thus, we have $v_1 \sqcap v_2 \sqsubset v_1$ and $f(v_1 \sqcap v_2) = f(v_1) \sqcap f(v_2) \neq \bot$, contradicting $v_2 \in \pi_1(tr(f))$.

"2 ⇒ 1": Assuming $x, y \in D_{\iota^n}$ such that $x \uparrow y$ and $f(x \sqcap y) \sqsubset f(x) \sqcap f(y)$, we get $f(x \sqcap y) = \bot$ and $f(x) = f(y) \neq \bot$, i.e. there are $v_1, v_2 \in \pi_1(tr(f))$ with $v_1 \sqsubseteq x$ and $v_2 \sqsubseteq y$, therefore $v_1 \uparrow v_2$. This contradicts item 2.

"2 ⇔ 3": We have

$$\{v^{(1)}, v^{(2)}\} \in \mathcal{C}(\mathbf{B}_\bot^n) \Leftrightarrow (f.a.\ 1 \leq j \leq n:\ v_j^{(1)} = \bot \vee v_j^{(2)} = \bot \vee v_j^{(1)} = v_j^{(2)}) \Leftrightarrow v^{(1)} \uparrow v^{(2)}.$$

□

Due to [Cur93] the *dop* of BP is the greatest of all stable functions of order 1. The specified proof comes from Nickau and Bucciarelli:

Theorem 11.3 *In the case $D_\iota = \mathbf{B}_\bot$ any function f of order 1 is stable if, and only if, $f \leq_p BP$ holds.*

Proof: "⇒": Let $f \in D_{\iota^n}$ with $tr(f) := \{(\vec{v}_1, b_1), \ldots, (\vec{v}_r, b_r)\}$. We show $f \leq_p BP$ by induction on $\sharp tr(f)$:

For any $1 \leq i \leq r$ the test $\vec{x} \sqsupseteq \vec{v}_i$ obviously is PCF-definable. Therefore, in the case $r = 1$ the function f is PCF-definable by $\lambda\vec{x}.cond(\vec{x} \sqsupseteq \vec{v}_1, b_1, \Omega^\iota)$.

For all $i \neq j$ we have $\vec{v}_i \not\uparrow \vec{v}_j$, i.e. there is a $1 \leq k \leq n$ with $v_i^{(k)} \not\uparrow v_j^{(k)}$. In the case $r = 2$ let w.l.o.g. $v_1^{(k)} = 0$ and $v_2^{(k)} = 1$. Then it is

$$f = [\![\lambda\vec{x}.cond(x_k, cond(\vec{x} \sqsupseteq \vec{v}_1, b_1, \Omega^\iota), cond(\vec{x} \sqsupseteq \vec{v}_2, b_2, \Omega^\iota))]\!].$$

Now let $r \geq 3$. We split the trace of f into $T_1 \dot\cup T_2 \dot\cup T_3$ with $\sharp T_i \geq 1$ for all $1 \leq i \leq 3$. By lemma 11.2 item 2 and by defining

$$\begin{aligned}
&tr(f_1) := \pi_1(T_2) \times 1 \cup \pi_1(T_3) \times 0,\\
&tr(f_2) := \pi_1(T_1) \times 0 \cup \pi_1(T_3) \times 1,\\
&tr(f_3) := \pi_1(T_1) \times 1 \cup \pi_1(T_2) \times 0,\\
&tr(g_1) := \{x \in tr(f) \mid \pi_1(x) \in \pi_1(T_1)\}\\
&tr(g_2) := \{x \in tr(f) \mid \pi_1(x) \in \pi_1(T_2) \cup \pi_1(T_3)\}
\end{aligned}$$

we obtain five stable functions, which are by induction hypothesis definable with the help of BP. We call the corresponding terms F_1, F_2, F_3, resp. G_1, G_2. Defining

$$G := \lambda\vec{x}.\underline{BP}(F_1(\vec{x}), F_2(\vec{x}), F_3(\vec{x}))$$

we have

$$[\![G]\!](\vec{z}) = \begin{cases} 1 & , f_1(\vec{z}) \neq \bot \\ 0 & , f_2(\vec{z}) \neq \bot \neq f_3(\vec{z}) \\ \bot & , \text{sonst.} \end{cases}$$

Because of

$$\begin{aligned} &f(\vec{z}) = \bot \Rightarrow G(\vec{z}) = BP(\bot, \bot, \bot) = \bot \text{ and} \\ &f(\vec{z}) \neq \bot \Rightarrow \vec{z} \in tr(g_1) \ \vee \ \vec{x} \in tr(g_2) \\ &\qquad \Rightarrow (G(\vec{z}) = 0 \ \wedge \ G_1(\vec{z}) = f(\vec{z})) \ \vee \ (G(\vec{z}) = 1 \ \wedge \ G_2(\vec{z}) = f(\vec{z})) \end{aligned}$$

we get $f = [\![\lambda \vec{x}.cond(G(\vec{x}, G_1(\vec{x}), G_2(\vec{x})))]\!]$.

"$\Leftarrow$": By $f \leq_p BP$ and lemma 10.18 we have $\min\{\sharp A \mid A \in H(f)\} \geq \min\{\sharp A \mid A \in H(BP)\} = 3$. Lemma 11.1 concludes the proof. □

Bucciarelli's functions can be inserted in the semilattice more exactly:

Corollary 11.4 *For all $m \leq n \leq 3$ we have $f_{n,m} <_p \mathcal{O}$.*

Proof:

- $f_{n,m} \leq_p \mathcal{O}$:

  More generally: For any $f \in D_{\iota^n}$ such that $\sharp tr(f) < \infty$ and $\sharp Im(f) \leq 2$, i.e. f at most assumes one value $\bot \neq b \in D_\iota$, f is sequentially reducible on $\mathcal{O}$.

  Proof: For $tr(f) = \{((v_{1,1}, \ldots, v_{1,n}), b), \ldots, ((v_{m,1}, \ldots, v_{m,n}), b)\}$ we have

  $$f = [\![\lambda x_1^\iota, \ldots, x_m^\iota.cond(\underline{\mathcal{O}}(\vec{x} \sqsupseteq \underline{v}_1, \ldots, \vec{x} \sqsupseteq \underline{v}_m), \underline{b}, \Omega^\iota)]\!],$$

  whereat $\vec{x} \sqsupseteq \underline{v}_i$ is standing for $x_{j_1} \doteq \underline{v}_{i,j_1} \underline{\wedge} \ldots \underline{\wedge} x_{j_{r(i)}} \doteq \underline{v}_{i,j_{r(i)}}$ and $\{x_{j_1}, \ldots, x_{j_{r(i)}}\} := \{x_j \mid 1 \leq j \leq n, v_{j,i} \neq \bot\}$ for all $1 \leq i \leq m$.

- $f_{n,m} \not\geq_p \mathcal{O}$:

  By lemma 11.1 all functions $f_{n,m}$ are stable. Therefore, it is $f_{n,m} \leq_p BP$. Because of $BP \,|_p\, \mathcal{O}$ we have the desired relation.

□

12 Epilogue

In this paper we use mainly the sequentiality relations in order to compare the degrees of parallelism with each other. However, in section 10 we give an example for two *dops* that can not be distinguished by these relations. So the subsemilattice of the *dops* can not be analysed completely just with the help of this method. Jung and Tiuryn show in [JT93] that for the simply typed λ-calculus with just one ground type ι and without constants elements of an arbitrary type frame are λ-definable if, and only if, they are invariant under all Kripke logical relations. Kripke logical relations are generalized logical relations:

Definition 12.1 *Let $\mathcal{C}$ be a small subcategory of the category $\underline{SET}$ and $\mathcal{A}$ a typing frame for the simply typed λ-calculus with just one ground domain A_ι and without constants. Then a family of relations $(R^w_\sigma)^{w\in Obj(\mathcal{C})}_{\sigma\in\Sigma}$ is called a familiy of* Kripke logical relations with varying arity *w.r.t. $\mathcal{C}$ and $\mathcal{A}$, if the following conditions hold:*

- *For all $w \in Obj(\mathcal{C})$ it is $R^w_\iota \subseteq A^{\sharp w}_\iota$.*
- *For all $v, w \in Obj(\mathcal{C})$ and $f \in Mor(v, w)$ we have $(x_j)_{j\in w} \in R^w_\iota \Rightarrow (x_{f(i)})_{i\in v} \in R^w_\iota$.*
- *For all $\sigma, \tau \in \Sigma$ and $w \in Obj(\mathcal{C})$ it is*

$$\begin{aligned} R^w_{\sigma\to\tau} = \{(g_j)_{j\in w} \mid & (f.a.\ j \in w:\ g_j \in A_{\sigma\to\tau}) \\ & \wedge (f.a.\ f \in Mor(v, w), (x_i)_{i\in v} \in R^v_\sigma:\ (g_{f(i)}(x_i))_{i\in v} \in R^v_\tau)\}. \end{aligned}$$

Choosing $\mathcal{C}$ with $\sharp Obj(\mathcal{C}) = 1$ and $\sharp Mor(w, w) = 1$ one gets a $\sharp w$-ary logical relation as defined in 3.1. Hearn and Riecke have transferred this generalization to definition 3.3 (cp. [OR94]):

Definition 12.2 *A* Kripke sequentiality relation *R is a Kripke logical relation meeting the additional conditions:*

- *For all $w \in Obj(\mathcal{C})$ and $\sigma \in \Sigma$ it is $\bot \in R^w_\sigma$.*
- *For all $w \in Obj(\mathcal{C})$ and $\sigma \in \Sigma$ it is R^w_σ closed under directed lubs.*
- *For all $w \in Obj(\mathcal{C})$ the relation R^w_ι is an intersection of relations $S^w_{A,B}$ with $A \subseteq B \subseteq w$, whereat*

$$S^w_{A,B} = \{g : w \to D_\iota \mid (ex.\ i \in A:\ g(i) = \bot)\ \vee\ (f.a.\ i, j \in B:\ g(i) = g(j))\}.$$

It can be verified that all PCF-terms are invariant under the Kripke sequentiality relations. It is yet unknown, whether the inversion holds, i.e. whether every element being invariant under all Kripke sequentiality relations is PCF-definable. At least can an element like that be represented (without restriction of the order) as directed lub of PCF-definable elements. It is open, whether the Kripke sequentiality relations can achieve more than the finite sequentiality relations w.r.t. the characterization of the PCF-definability.

Stoughton's algorithm is a handy procedure for finding out, whether $f \leq_p g$ does hold or not. However, this is only possible for a finite ground domain and restricted order of the functions, to be precise for $ord(f) \leq 2$ and $ord(g) \leq 1$. In [Loa93] the reader can found a proof that there is no general decision algorithm for λ-definability in case of an arbitrary finite ground domain: Loader reduces the relative λ-definability problem, i.e. whether a function f is λ-definable with the help of a function g (cp. the definition 4.1 of $\leq_p$), by encoding to a word problem, i.e. given a rewriting system (Semi-Thue system) is it decidable, whether a word w can be derived from an word v. The latter problem is undecidable because of the undecidability of the halting problem of Turing machines. We want to mention once more result to this point (see [JT93]): In case of a finite ground domain and a *finite* set of variables the λ-definability is decidable. Results for the PCF-definability are as yet lacking.

In [Saz76] Sazonov lists relations between *dops* without proof. The results referring to computable functions are mainly proved in this paper. Indeed, big problems cause the T_f-functionals, defined in section 8. Because of the large order a specific analysing of T_f for invariance under sequentiality relations is not possible. Especially, it remains open, whether $[\mathcal{O}]_p, [T_{\exists_{n+1}}]_p$ and $[T_{por}]_p$ are three different degrees, as it is indicated by Sazonov. For large orders results are hardly known.

Results for the ground domain $\mathbf{N}_\perp$ are a further open question. What parallelism do functions have which take properties of the natural numbers into account like infinity and divisibility, e.g. the following generalization of V' for $n \geq 3$?

$$V'_n \in D_{\iota^n},\ (x_1, \ldots, x_n) \mapsto \begin{cases} nm & , x_1 = \ldots = x_{n-1} = nm \\ nm+1 & , x_1 = \ldots = x_{n-2} = x_n = nm+1 \\ \vdots & \\ nm+n-1 & , x_2 = \ldots = x_n = nm+n-1 \end{cases}$$

Chains and antichains of *dops* have been constructed. Which structural properties of the semilattice can be given in addition, e.g. analogous to the semilattice of the Turing degrees? In which case does the glb of two *dops* exists? Are there minimal *dops*? Is the semilattice dense, i.e. is there for any two *dops* a, b with $a <_p b$ a *dop* c such that $a <_p c <_p b$?

Notation Index

References

[Bar84] H.P.Barendregt. *The Lambda Calculus, Its Syntax and Semantics.* North-Holland. Amsterdam, 1984.

[Ber78] G.Berry. *Stable models of typed lambda calculi.* In G.Ausiello and C.Bohm, eds., Automata, Languages and Programming, LNCS 62, 72-89. Springer, 1978.

[Ber79] G.Berry. *Modèles complètement adéquats et stables des lambda-calculs typés.* Technical report. Université Paris VII, Thèse de Doctorat d'Etat, 1979.

[Buc95] A.Bucciarelli. *Degrees of parallelism in the continuous type hierarchy.* To appear in Proceedings of the Ninth International Conference on Mathematical Foundations of Programming Semantics, 1995.

[Cur93] P.-L.Curien. *Categorical Combinators, Sequential Algorithms and Functional Programming.* Progress in Theoretical Computer Science. Birkhäuser, 1993.

[Gun92] C.A.Gunter. *Semantics of Programming Languages: Structures and Techniques.* Foundations of Computing Series. MIT Press, 1992.

[HS86] J.R.Hindley and J.P.Seldin. *Introduction to Combinators and λ-Calculus.* London Mathematical Society Student Texts. Cambridge University Press, 1986.

[JT93] A.Jung and J.Tiuryn. *A New Characterization of Lambda Definability.* In M.Bezem and J.F.Groote, eds., International Conference on Typed Lambda Calculi and Applications, LNCS 664, 230-244. Springer, 1993.

[Loa94] R.Loader. *The Undecidability of λ-Definability.* In M.Zeleny, ed., The Church Festschrift. CSLI/University of Chicago Press, 1994.

[Mil77] R.Milner. *Fully abstract models of typed lambda-calculus.* Theoretical Computer Science 4, 1-22, 1977.

[OR94] P.W.O'Hearn and J.G.Riecke. *Kripke Logical Relations and PCF.* To appear in Information and Computation, 1994.

[Ong94] C.-H.L.Ong. *Correspondence between Operational and Denotational Semantics: The Full Abstraction Problem for PCF.* Handbook of Logic in Computer Science 3, Oxford University Press, 1994.

[Plo77] G.D.Plotkin. *LCF considered as a programming language.* Theoretical Computer Science 5, 223-255, 1977.

[Plo80] G.D.Plotkin. *Lambda-definability in the full type hierarchy.* In J.Seldin and J.Hindley, eds., To H.B.Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism, 363-374. Academic Press, 1980.

[Saz76] V.Yu.Sazonov. *Degrees of parallelism in computations.* Mathematical Foundations of Computer Science, LNCS 45, 517-523. Springer, 1976.

[Sco69] D.S.Scott. *A type-theoretical alternative to CUCH, ISWIN and OHWY.* Unpublished manuscript. Oxford, 1969.

[Sie92] K.Sieber. *Reasoning about Sequential Functions via Logical Relations.* In M.P.Fourman, P.T.Johnstone und A.M.Pitts, eds., Applications of Categories in Computer Science, LMS Lecture Note Series 177, 258-269. Cambridge University Press, 1992.

[Spr92] D.Spreen. *Berechenbarkeit und Komplexitatstheorie.* Lecture script, Siegen, 1992.

[Sto94] A.Stoughton. *Mechanizing Logical Relations.* To appear in Proceedings of the Ninth International Conference on Mathematical Foundations of Programming Semantics, 1994.

[Tra75] M.B.Trachtenbrot. *On representation of sequential and parallel functions.* Proceedings of the Fourth Symposium on Mathematical Foundations of Computer Science, LNCS 32, 411-417. Springer, 1975.

[Wei80] K.Weihrauch and Th.Deil. *Berechenbarkeit auf cpo-s.* Schriften zur angewandten Mathematik und Informatik Nr.63, Rheinisch-Westfälische Technische Hochschule Aachen. Aachen, 1980.