ON BERRY'S CONJECTURES ABOUT THE STABLE ORDER IN PCF

FRITZ MÜLLER

Saarland University, Department of Computer Science, Campus E1.3, 66123 Saarbrücken, Germany, http://rw4.cs.uni-saarland.de/~mueller
e-mail address: (λx.muellerxcs.uni-saarland.de)@

Abstract. PCF is a sequential simply typed lambda calculus language. There is a unique order-extensional fully abstract cpo-model of PCF, built up from equivalence classes of terms. In 1979, Gérard Berry defined the stable order in this model and proved that the extensional and the stable order together form a bicpo. He made the following two conjectures:
1) "Extensional and stable order form not only a bicpo, but a bidomain."
We refute this conjecture by showing that the stable order is not bounded complete, already for finitary PCF of second-order types.
2) "The stable order of the model has the syntactic order as its image: If a is less than b in the stable order of the model, for finite a and b, then there are normal form terms A and B with the semantics a, resp. b, such that A is less than B in the syntactic order."
We give counter-examples to this conjecture, again in finitary PCF of second-order types, and also refute an improved conjecture: There seems to be no simple syntactic characterization of the stable order. But we show that Berry's conjecture is true for unary PCF.

For the preliminaries, we explain the basic fully abstract semantics of PCF in the general setting of (not-necessarily complete) partial order models (f-models). And we restrict the syntax to "game terms", with a graphical representation.

1. Introduction

PCF is a simple functional programming language, a call-by-name typed lambda calculus with integers and booleans as ground types, some simple sequential operations on the ground types, and a fixpoint combinator. The concept of PCF was formed by Dana Scott in 1969, see the historical document [27]. It is used as a prototypical programming language to explore the relationship between operational and denotational semantics, see the seminal paper of Gordon Plotkin [24].

The (operational) *observational preorder* $M \sqsubseteq_{op} N$ of two terms (of equal type) is defined as: For all contexts $C[\ ]$ of integer type, if $C[M]$ reduces to the integer n, then $C[N]$ also reduces to the same n. The denotational semantics (the model) assigns to every

1998 ACM Subject Classification: F.3.2[Logics and Meanings of Programs]: Semantics of Programming Languages — denotational semantics, operational semantics, F.4.1[Mathematical Logic]: Lambda Calculus and Related Systems.

Key words and phrases: functional program, typed lambda calculus, PCF, denotational semantics, fully abstract model, non-cpo model, game semantics, stable function, stable order, dI-domain, bicpo, bidomain, syntactic order.

DOI:10.2168/LMCS-???

term M an element $[\![M]\!]$ of a partial order $(D, \sqsubseteq)$ (usually a complete partial order, cpo) as meaning. The model is said to be *(order) fully abstract* if the two orders coincide: $M \sqsubseteq_{op} N \iff [\![M]\!] \sqsubseteq [\![N]\!]$. The standard model of Scott domains and continuous functions is adequate (i.e. the direction $\Longleftarrow$ of the coincidence), but not fully abstract, because the semantic domains contain finite elements that are not expressible as terms, like the parallel or function. First Robin Milner [16] constructed in 1977 a unique fully abstract order-extensional cpo-model of PCF that can be built up from equivalence classes of terms by some ideal completion. The problem to construct a fully abstract model of PCF that does not use the syntax of terms (the "full abstraction problem") was the driving force of the subsequent developments, see also the handbook article [22].

In 1979 Gérard Berry published his PhD thesis [4] with the translated title "Fully abstract and stable models of typed lambda-calculi", which is the main basis of our work. In order to sort out functions like the parallel or from the semantic domains, to get "closer" to the fully abstract model, he gave the definition of stable function: A function f is *stable* if for the computation of some finite part of the output a deterministic minimal part of the input is needed. In the case that there are only finitely many elements smaller than a finite element, this definition is equivalent to the definition of a *conditionally multiplicative function* f: If a and b are compatible, then $f(a \sqcap b) = f a \sqcap f b$. To make the operation of functional application of stable functions itself stable, Berry had to replace the pointwise order of functions, the extensional order, by the new stable order: Two functions are in the *stable order*, $f \leq g$, if for all $x \leq y$: $f x = f y \sqcap g x$. This entails the pointwise order, but it demands in addition that g must not output some result for input x that f outputs only for greater y.

Side remark: Stability is a universal concept that was independently (re)discovered in many mathematical contexts. So Jean-Yves Girard found it in the logical theory of dilators and then transferred it to domain theory (qualitative domain, coherence space) to give a model of polymorphism (system F) [9], thereby independently reinventing Berry's stable functions and stable order, see also the textbook [10], chapter 8 and appendix A. For a general theory of stability and an extensive bibliography see [31].

Now Berry had a model (of PCF) of stable functions with the stable order. But this model did not respect the old (pointwise) extensional order of the standard model and so had new unwanted elements not contained in the standard model. To get a proper subset of the standard model, he introduced bicpo models. A *bicpo* is a set with two orders, an extensional and a stable one, both forming cpos and being connected in some way. He augmented Milner's fully abstract cpo model by the stable order and proved that it consists of bicpos and its functions are conditionally multiplicative. In section 3 we show in addition that its stable order forms stable bifinite domains and therefore its functions are also stable and can be represented by *traces*, i.e. sets of tokens (or events) like in [7]. E.g. the function $[\![\lambda f.\, \mathsf{if}(\mathsf{zero}(f 0))\, \mathsf{then}\, 0\, \mathsf{else}\, \bot]\!]$ can be represented by the trace consisting of the tokens $\{0 \mapsto 0\} \mapsto 0$ and $\{\bot \mapsto 0\} \mapsto 0$. Functions are in the stable order, $f \leq g$, iff the trace of f is a subset of the trace of g.

In his thesis Berry made the following two conjectures that we refute:

1) "Extensional and stable order in the fully abstract cpo-model of PCF form not only a bicpo, but a bidomain."

This would mean (among other things) that the stable order is bounded complete and distributive. We give counter-examples in finitary PCF of second-order types to this conjecture. The idea is that the stable lub of two stably bounded elements a and b may entail a new token that was not present in a or b. This new token must be used in the syntax to separate a subterm denoting a from a subterm denoting b that cannot be unified in a common term. Therefore distributivity is not fulfilled, stable lubs are not taken pointwise. And worse: There may be a choice between different new tokens to be entailed, then there is a choice between different minimal stable upper bounds of a and b, but there is no stable lub. The minimal stable upper bounds are pairwise stably incompatible, and the extensional lub $a \sqcup b$ is one of them.

2) The extensional order of the fully abstract model coincides with the (syntactic) observational preorder. This leads to the question: Is there a syntactic characterization also for the stable order? Berry made the conjecture:
"The stable order of the model has the syntactic order as its image:
If $a \leq b$ in the stable order, for finite a and b, then there are normal form terms A and B with $[\![A]\!] = a$ and $[\![B]\!] = b$, such that $A \prec B$ in the syntactic order."

Berry proved the converse direction: If $A \prec B$, then $[\![A]\!] \leq [\![B]\!]$, and proved the conjecture for first-order types.

Our simplest counter-example to this conjecture is a situation of four terms $A \prec B \cong C \prec D$, where $\cong$ is observational equivalence, so that $[\![A]\!] \leq [\![D]\!]$, but there is no way to find terms $A' \cong A$, $D' \cong D$ with $A' \prec D'$. The elimination of some token of D depends on the prior elimination of some other token, so that two $\prec$-steps are necessary to get from D down to A.

We further give examples where such a chain of $\prec$-steps (with intermediate $\cong$-steps) of any length is necessary. This proposes an improved conjecture, the "chain conjecture": Instead of $A \prec B$ we demand the existence of a chain between A and B. But we also refute this conjecture. Although stable order and syntactic order are connected, there seems to be no simple syntactic characterization of the stable order in PCF.

All our counter-examples for both conjectures are in finitary PCF of second-order types. They all share a common basic idea: We have a term $M \colon (\iota \to \iota \to \iota) \to \iota$ with two tokens (among others) which are in the simplest form like the tokens $\{\bot\bot{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}$ and $\{\bot\mathsf{0}{\mapsto}\mathsf{0}, \mathsf{11}{\mapsto}\mathsf{1}\}{\mapsto}\mathsf{0}$. The function call that realizes $\bot\bot{\mapsto}\mathsf{0}$ resp. $\bot\mathsf{0}{\mapsto}\mathsf{0}$ is at the top level of M, the function call for $\mathsf{11}{\mapsto}\mathsf{1}$ is nested below. We want to eliminate the token $\{\bot\bot{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}$. For this the function call for $\mathsf{11}{\mapsto}\mathsf{1}$ must be "lifted" to the top level, but this is not possible due to other tokens of M that have to stay.

The necessary ingredients for the counter-examples are: at least second-order type with some functional parameter of arity at least 2, at least two different ground values $\mathsf{0}$ and $\mathsf{1}$, and the need for nested function calls.

If we restrict the calculus to a single ground value $\mathsf{0}$, we get unary PCF, and in this case both of Berry's conjectures are true: The fully abstract model is a bidomain, in fact it is the standard semantical bidomain construction, proved by Jim Laird in [12]. And we prove that the syntactic order is the image of the stable order, using Laird's proof that every type in unary PCF is a definable retract of some first-order type.

The need for nested function calls is the result of a "restriction" of PCF: There is no operator to test if a function demands a certain argument, so that this information could be used in an if-then-else. Jim Laird has shown that in a language with such control operators (SPCF) nested function calls can be eliminated, and also every type of SPCF is a definable retract of a first-order type [13]. Therefore I am convinced, though I do not prove it here, that also for SPCF the syntactic order is the image of the stable order.

The above mentioned “restriction” of PCF is generally the reason for many irregularities of the semantics of PCF and the difficulty of the full abstraction problem. An important result is the undecidability of finitary PCF [14]. This means that the observational equivalence of two terms of finitary PCF is undecidable, and also the question whether there is a term for a functional value table. As remarked in the introduction to [7], this result restricts the possible fully abstract models of PCF to be not “finitary” in some sense. There have been several solutions for semantical fully abstract models of PCF: A model of continuous functions restricted by Kripke logical relations [21], and game semantics [1, 11, 18]. In game semantics a term of PCF is modeled by a strategy of a game, i.e. by a process that performs a dialogue of questions and answers with the environment, the opponent. These strategies are still intensional; the fully abstract model is formed by a quotient, the extensional collapse. The strategies can be identified with PCF Böhm trees of a certain normal form, see also [2, section 6.6]. We call these Böhm trees “game terms” and prove that it is sufficient to formulate all our results in the realm of game terms, esp. that if two terms are syntactically ordered, then there are equivalent game terms so ordered. This simplifies the proofs of the counter-examples. We also introduce a graphical notation for game terms that facilitates the handling of larger examples.

It was an open problem whether the game model is isomorphic to Milner’s fully abstract cpo-model, i.e. whether its domains are cpos. This problem was solved by Dag Normann [19]: Its domains are not cpos, i.e. there are directed sets that have no lub. Then Vladimir Sazonov made a first attempt to build a general theory for these non-cpo domains [25, 26, 20]. His main insight was that functions are continuous only with respect to certain lubs of directed sets that he calls “natural lubs”; these are the hereditarily pointwise lubs.

We want to place our results in the context of these new, more general models. For the semantic preliminaries we give a simple definition of a set of well-behaved (not-necessarily complete) partial order fully abstract models of PCF: These *f-models* are sets of ideals of finite elements, such that application is defined and every PCF-term has a denotation. Sazonov’s natural lubs correspond to our *f-lubs*, which are defined with respect to the finite elements.

I found the counter-example to Berry’s second conjecture around the year 1990, but did not yet publish it. As far as I know, nobody else tackled Berry’s problems. The reason for this seems to be that they were simply forgotten. The stable order in the fully abstract model was never explored after Berry; a reason may be that he never prepared a journal version of his thesis, which is not easily accessible. The recommended introduction to our subject is the report “Full abstraction for sequential languages: The state of the art” [5], which contains the thesis in condensed form, but lacks most proofs. There is also an article [3] published by Berry before his thesis, which is not recommended, because section 4.5 (bidomains) is wrong (different definition of bidomain, the first conjecture is stated as theorem). An excellent general introduction to domains, stability and PCF (and many other things) is the textbook [2]. But for the stable order in the fully abstract model of PCF the only detailed source remains Berry’s thesis.

Here is the structure of the paper. The counter-examples are given in the order of their discovery, i.e. in the order of increasing complexity.

2. Syntax of PCF.

3. Semantics of PCF: non-complete partial order f-models:
   We introduce f-models as general (not-necessarily complete) partial order fully abstract models of PCF and give the properties of the stable order in this general context. (The order-extensional fully abstract cpo-model of PCF is a special case.)
4. Game terms:
   We describe the construction of game terms by the finite projections and give a graphical notation for game terms.
   The expert who is interested only in the counter-examples may skip the introductory sections 2-4; reading only the definition of game terms and their graphical notation at the beginning of section 4.
5. The syntactic order is not the image of the stable order:
   We prove Berry's second conjecture for first-order types, give a counter-example in a second-order type (a chain of length 2), and prove the existence of chains of any least length.
6. The stable order is not bounded complete: no bidomain:
   We prove Berry's first conjecture for first-order types. In a second-order type we give an example of a stable lub that does not fulfill distributivity, and an example of two stably bounded elements without stable lub.
7. Refutation and improvement of the chain-conjecture:
   We refute the improved second conjecture that the stable order entails a chain of terms. We propose in turn an improvement of the chain conjecture, based on the complementary syntactic relation of strictification.
8. Unary PCF:
   We prove Berry's second conjecture for unary PCF, with the aid of Jim Laird's definable retractions from any type to some first-order type [12].
9. Outlook.

2. Syntax of PCF

In this section we give the syntactic definitions of PCF [24, 5, 2]. The programming language PCF is a simply typed lambda calculus with arithmetic and fixpoint operators. It usually comes with two ground types ι (integers) and o (booleans). We simplify the language and use only the ground type ι (integers); the booleans are superfluous and can be coded as integers, the intensional structure of the terms stays the same.

The **types** are formed by ι and function types $\sigma \to \tau$ for types σ and τ.

The typed **constants** are:
$\mathsf{0}, \mathsf{1}, \mathsf{2}, \ldots : \iota$, the integers;
$\mathsf{suc}, \mathsf{pre} \colon \iota \to \iota$, successor and predecessor function;
$\mathsf{if}\ \_\ \mathsf{then}\ \_\ \mathsf{else}\ \_ \colon \iota \to \iota \to \iota \to \iota$, this conditional tests if the first argument is $\mathsf{0}$.
(We write e.g. $\mathsf{if}\ x\ \mathsf{then}\ y$ for the application of this function to only two arguments.)

The PCF **terms** comprise the constants and the typed constructs by the following rules:
$\bot^\sigma \colon \sigma$ for any type σ, the undefined term.
$x^\sigma \colon \sigma$ for any variable x^σ.
If $M \colon \tau$, then $\lambda x^\sigma.M \colon \sigma \to \tau$, lambda abstraction.

If $M\colon \sigma \to \tau$ and $N\colon \sigma$, then $MN\colon \tau$, function application.
If $M\colon \sigma \to \sigma$, then $\mathsf{Y}M\colon \sigma$, Y is the fixpoint operator.

PCF^σ is the set of all PCF terms of type σ, and PCF^σ_c is the set of the closed terms of these.
Type annotations of $\perp$ and of variables will often be omitted.
We use the (semantic) symbol $\perp$ also as syntactic term, instead of the usual Ω.
We define the *syntactic order* $\prec$ (also called $\perp$-match order in the literature) on terms of the same type:
$M \prec N$ iff N can be obtained by replacing some occurrences of $\perp$ in M by terms.

The **reduction rules** are (where n is a variable for integer constants):
$(\lambda x.M)N \to M[x := N]$, the usual β-reduction;
$\mathsf{Y}M \to M(\mathsf{Y}M)$;
$\mathsf{suc}\, n \to (n+1)$;
$\mathsf{pre}\, n \to (n-1)$, for $n \geq 1$;
$\mathsf{if}\ 0\ \mathsf{then}\ M\ \mathsf{else}\ N \to M$;
$\mathsf{if}\ n\ \mathsf{then}\ M\ \mathsf{else}\ N \to N$, for $n \geq 1$.

The *reduction relation* $\to$ is one step of reduction by these rules in any term context. It is confluent. $\to^*$ is the reflexive, transitive closure of $\to$.

A *program* is a closed term of type ι.
The *operational (observational) preorder* $\sqsubseteq_{op}$ on terms of the same type is defined as:
$M \sqsubseteq_{op} N$ (M is *operationally less defined than* N) iff
$P[M] \to^* n$ implies $P[N] \to^* n$ for all contexts $P[\ ]$ such that $P[M]$ and $P[N]$ are both programs.
The *operational equivalence* is defined as: $M \cong N$ iff $M \sqsubseteq_{op} N$ and $N \sqsubseteq_{op} M$.

3. Semantics of PCF: non-complete partial order f-models

This section gives an exposition of the fully abstract semantics of PCF with the stable order, as far as it is needed to understand the results of this paper. The proofs are omitted, as they are easy and/or already known in some form.

The order-extensional fully abstract cpo-model of PCF was first constructed by Robin Milner [16] based on terms of an SKI-combinator calculus. Later Gérard Berry's thesis [4] constructed this model based on the proper λ-terms. This model is the ideal completion of the finite elements; every directed set has a lub.

Then came the fully abstract game models of PCF [1, 11, 18]. The elements of these models can be represented by the (infinite) Böhm trees of PCF. It was an open problem whether the game model is isomorphic to Milner's model, i.e. whether its domains are cpos.

This problem was solved by Dag Normann [19]: Its domains are not cpos, i.e. there are directed sets that have no lub. Then Vladimir Sazonov made a first attempt to build a general theory for these non-cpo domains [25, 26, 20]. His main insight was that functions are continuous only with respect to certain lubs of directed sets that he calls "natural lubs"; these are the hereditarily pointwise lubs.

We want to place our results in the context of these new, more general models. Therefore we give a simple definition of a set of well-behaved (not-necessarily complete) partial order fully abstract models of PCF: These *f-models* are sets of ideals of finite elements, such that application is defined and every PCF-term has a denotation. Sazonov's natural lubs correspond to our *f-lubs*, which are defined with respect to the finite elements.

We state the usual properties for these f-models; the essence of their proofs is already contained in Berry's construction. Our aim is the definition of the stable order and of conditionally multiplicative (cm) functions. All functions in f-models are cm. We can further show, in addition to Berry, that the domains have property I under the stable order and therefore the functions are stable and we can work with their traces.

We need the following PCF terms, the *finite projections* on type σ of *grade* i, $\Psi_i^\sigma : \sigma \to \sigma$:

$$\begin{aligned}\Psi_i^\iota &= \lambda x^\iota.\,\mathsf{if}\ x\ \mathsf{then}\ 0\ \mathsf{else}\ \mathsf{if}\ \mathsf{pre}^1\,x\ \mathsf{then}\ 1\ \mathsf{else}\ldots\mathsf{if}\ \mathsf{pre}^i\,x\ \mathsf{then}\ i\ \mathsf{else}\ \bot\\ \Psi_i^{\sigma\to\tau} &= \lambda f^{\sigma\to\tau}.\lambda x^\sigma.\Psi_i^\tau(f(\Psi_i^\sigma x))\end{aligned}$$

We also need the following terms for the glb functions on all types, $\inf^\sigma : \sigma \to \sigma \to \sigma$, here in a liberal syntax:

$$\begin{aligned}\inf^\iota &= \lambda x^\iota y^\iota.\,\mathsf{if}\ x = y\ \mathsf{then}\ x\ \mathsf{else}\ \bot\\ &= \lambda x^\iota y^\iota.\,\mathsf{if}\ x\ \mathsf{then}\ \mathsf{if}\ y\ \mathsf{then}\ 0\ \mathsf{else}\ \bot\\ &\qquad\qquad\qquad \mathsf{else}\ \mathsf{suc}(\inf^\iota(\mathsf{pre}\,x)(\mathsf{pre}\,y))\\ \inf^{\sigma\to\tau} &= \lambda f^{\sigma\to\tau}g^{\sigma\to\tau}.\lambda x^\sigma.\inf^\tau(fx)(gx)\end{aligned}$$

When applied to a closed term $M\colon \sigma$, the function term Ψ_i^σ serves as a "filter" that lets only pass integer values $\leq i$ as input or output to M. This serves to define the finite elements of the intended model.

Definition 3.1. A term $M\colon \sigma$ is a *finite term of grade* i if it is closed and $M \cong \Psi_i^\sigma M$.
$\mathcal{F}_i^\sigma = \{\, [\Psi_i^\sigma M]_{op} \mid M \in \mathrm{PCF}_c^\sigma \,\}$ is the set of *finite elements of grade* i of type σ, where $[X]_{op}$ is the equivalence class of term X under the operational equivalence $\cong$.
$\mathcal{F}^\sigma = \bigcup_i \mathcal{F}_i^\sigma$ is the set of *finite elements* of type σ.

The finite elements are partially ordered by the extension of the operational preorder $\sqsubseteq_{op}$ to equivalence classes.
An *ideal* of finite elements of type σ is a set $S \subseteq \mathcal{F}^\sigma$ such that: $S \neq \emptyset$ and
$a, b \in S \Longrightarrow \exists c \in S.\ a \sqsubseteq_{op} c$ and $b \sqsubseteq_{op} c$,
and $a \in S, b \in \mathcal{F}^\sigma$ and $b \sqsubseteq_{op} a \Longrightarrow b \in S$.
$I(\mathcal{F}^\sigma)$ is the set of ideals of finite elements of type σ.
There is an operation apply on ideals of finite elements. For $f \in I(\mathcal{F}^{\sigma\to\tau})$, $d \in I(\mathcal{F}^\sigma)$:

$$\mathrm{apply}(f, d) = {\downarrow}\{\, f'd' \mid f' \in f, d' \in d \,\} \in I(\mathcal{F}^\tau),$$

where $f'd' = [MN]_{op}$ for $M \in f'$, $N \in d'$. $\mathrm{apply}(f, d)$ is simply written fd.
From now on $a \in \mathcal{F}^\sigma$ is identified with the ideal ${\downarrow}\{a\}$, the downward closure w.r.t. $\sqsubseteq_{op}$ of $\{a\}$. So we have the embedding $\mathcal{F}^\sigma \subseteq I(\mathcal{F}^\sigma)$.

Definition 3.2. An *f-model* of PCF ("f" means: based on finite elements) is a collection of $D^\sigma \subseteq I(\mathcal{F}^\sigma)$ for every type σ, each D^σ ordered by inclusion $\subseteq$ written $\sqsubseteq$,
such that for $f \in D^{\sigma\to\tau}$, $d \in D^\sigma$: $fd \in D^\tau$,
and such that every closed term $M\colon \sigma$ has its denotation in D^σ: ${\downarrow}\{\, [\Psi_i^\sigma M]_{op} \mid i \geq 0 \,\} \in D^\sigma$.
The lubs w.r.t. $\sqsubseteq$ will be written $\sqcup$ and $\bigsqcup$, the glbs $\sqcap$ and $\bigsqcap$.

All f-models coincide on their part of the finite elements w.r.t. both extensional $\sqsubseteq$ and stable $\leq$ order. In the following sections, propositions will mostly deal with finite elements. The propositions are valid for all f-models if not otherwise stated.

To every f-model we can associate the *semantic map* $[\![\;\;]\!]\colon \mathrm{PCF}^\sigma \to \mathrm{ENV} \to D^\sigma$, where ENV is the set of environments ρ that map every variable x^σ to some $\rho(x^\sigma) \in D^\sigma$. If $M : \sigma$ is a term with the free variables $x_1, \ldots, x_n$, then

$$[\![M]\!]\rho = \downarrow\!\{\, [\Psi_i^\sigma M[x_1 := N_1, \ldots, x_n := N_n]]_{op} \mid i \geq 0, [N_j]_{op} \in \rho(x_j) \,\}.$$

For closed terms M we also write $[\![M]\!]$ for $[\![M]\!]\bot$.

There are three outstanding examples of f-models: There is the least f-model that consists of just the ideals denoting closed PCF-terms. There is the greatest f-model consisting of all ideals; this is Milner's and Berry's cpo-model. And there is the game model consisting of all denotations of (infinite) PCF-Böhm-trees, i.e. the sequential functionals. By Normann's result [19] we know that the game model is properly between the least and the greatest f-models.

Now we will collect the most important properties of f-models. In the following the D^σ are the domains of some f-model.

Lemma 3.3. *Every $\mathcal{F}_i^\sigma$ has finitely many elements.*
The semantics of the $\inf^\sigma$*-terms are the glb-functions with respect to the order $\sqsubseteq$; we write $\sqcap$ for these functions.*
If $d, e \in \mathcal{F}_i^\sigma$, then $d \sqcap e \in \mathcal{F}_i^\sigma$.
If $d, e \in \mathcal{F}_i^\sigma$ are compatible (bounded), i.e. there is some $a \in D^\sigma$ with $d \sqsubseteq a$ and $e \sqsubseteq a$, then there is a lub $d \sqcup e \in \mathcal{F}_i^\sigma$.

With this lemma we can prove:

Proposition 3.4. *All $D^{\sigma\to\tau}$ are order-extensional, i.e. :*

$$\begin{aligned} \textit{If } f, g \in D^{\sigma\to\tau}, \textit{ then } f \sqsubseteq g &\iff \forall d \in D^\sigma.\ fd \sqsubseteq gd \\ &\iff \forall d \in \mathcal{F}^\sigma.\ fd \sqsubseteq gd \end{aligned}$$

Elements of $D^{\sigma\to\tau}$ will be identified with the corresponding functions. apply and these functions are all monotone. They are continuous with respect to certain directed lubs, the f-lubs.

Definition 3.5. The directed set $S \subseteq D^\sigma$ *has the f-lub* $s \in D^\sigma$, written $S \to s$, iff s is an upper bound of S and for all finite $x \sqsubseteq s$ there is some $y \in S$ with $x \sqsubseteq y$. (This is equivalent to: s is the set-theoretical union of S. s is also the lub of S w.r.t. $\sqsubseteq$.)
A function $f\colon D^\sigma \to D^\tau$ is *f-continuous*, iff it is monotone and respects f-lubs of directed sets $S \subseteq D^\sigma$, i.e. if $S \to s$, then $fS \to fs$. (With $fS = \{\, fx \mid x \in S \,\}$.)

Proposition 3.6. *The* apply *operation is f-continuous on the domain $D^{\sigma\to\tau} \times D^\sigma$. (With component-wise order and pairs of finite elements as finite elements.) Therefore* apply *is f-continuous in each argument, and the functions of $D^{\sigma\to\tau}$ are f-continuous.*

In [20] it is shown that in the game model there are lubs of directed sets that are not f-lubs; and that there are finite elements that are not compact in the usual sense with respect to general directed lubs.

The f-lubs are exactly the directed lubs for which all functions are continuous: If we have a directed lub that is not an f-lub, then this lub contains a finite element that is not contained in the directed set. The PCF-function that "observes" (or "tests") this finite element is a function that is not continuous for the directed set.

In the greatest f-model all lubs of directed sets are f-lubs. If $S \to s$ in the greatest f-model, then the same holds in all f-models that contain s and the elements of S.

In an f-model we can define natural lubs in the sense of Sazonov as hereditarily pointwise lubs. Then a directed set S has the f-lub s iff S has the natural lub s.

Side remark: Here we must also mention the "rational chains" of Escardó and Ho [8]. These are ascending sequences of PCF terms that can be defined syntactically by a PCF procedure. The denotations (in any f-model) of the elements of a rational chain always form a directed set with an f-lub (natural lub). The converse does not hold generally.

Proposition 3.7. *The semantic map of an f-model fulfills the usual equations, i.e. the constants have their intended meanings, and:*

$$\begin{aligned} [\![\lambda x.M]\!]\rho d &= [\![M]\!]\rho[x := d] \\ [\![MN]\!]\rho &= [\![M]\!]\rho[\![N]\!]\rho \\ [\![\mathsf{Y}M]\!]\rho &= \bigsqcup_{n\geq 1}(([\![M]\!]\rho)^n\bot) \end{aligned}$$

Proposition 3.8 (Berry, 3.6.11 in [4])**.** *Define the functions $\psi_i^\sigma = [\![\Psi_i^\sigma]\!]\bot \colon D^\sigma \to D^\sigma$. For all σ, (ψ_i^σ) is an increasing sequence of finite projections with f-lub the identity* id*:*

$$\begin{aligned} \psi_i^\sigma &\sqsubseteq \mathrm{id} \\ \psi_i^\sigma \circ \psi_i^\sigma &= \psi_i^\sigma, \ \textit{with} \circ \textit{function composition} \\ \psi_i^\sigma &\sqsubseteq \psi_{i+1}^\sigma \\ \{\, \psi_i^\sigma \mid i \geq 0 \,\} &\to \mathrm{id} \\ \psi_i^\sigma(D^\sigma) &= \mathcal{F}_i^\sigma \end{aligned}$$

Proposition 3.9. *Every f-model is fully abstract for PCF: For all terms M, N of the same type*

$$(\forall \rho \in \mathrm{ENV}.\ [\![M]\!]\rho \sqsubseteq [\![N]\!]\rho) \Longleftrightarrow M \sqsubseteq_{op} N.$$

In the rest of this section we will define the stable order in f-models and collect the corresponding properties that will be needed in this paper.

The definition of the stable order $\leq$ is given by Berry [4, 4.8.6, page 4-93] for the fully abstract cpo-model as follows:

$$\begin{aligned} \text{For } d, e \in D^\iota &: d \leq e \Longleftrightarrow d \sqsubseteq e \\ \text{For } f, g \in D^{\sigma\to\tau} &: f \leq g \Longleftrightarrow \forall x \in D^\sigma.\ fx \leq gx \text{ and} \\ &\qquad \forall x, y \in D^\sigma.\ x \uparrow_\leq y \Longrightarrow fx \sqcap gy = fy \sqcap gx \end{aligned}$$

(Here $\uparrow_\leq$ means compatibility w.r.t. $\leq$.)
This definition serves as well for our f-models, but I prefer the equivalent (w.r.t. the full type hierarchy) form:

Definition 3.10 (stable order $\leq$)**.**

$$\begin{aligned} \text{For } d, e \in D^\iota &: d \leq e \Longleftrightarrow d \sqsubseteq e \\ \text{For } f, g \in D^{\sigma\to\tau} &: f \leq g \Longleftrightarrow \forall x, y \in \mathcal{F}^\sigma.\ x \leq y \Longrightarrow fx = fy \sqcap gx \end{aligned}$$

The order $\leq$ is extended pointwise to environments from ENV, here used in the definition of $\leq$ on denotations:

$$\text{For } f, g \in \text{ENV} \to D^\sigma : f \leq g \Longleftrightarrow \forall \rho, \varepsilon \in \text{ENV}.\ \rho \leq \varepsilon \Longrightarrow f\rho = f\varepsilon \sqcap g\rho$$

The lubs w.r.t. $\leq$ will be written $\vee$ and $\bigvee$, the glbs $\wedge$.

Note that $\sqcap$ is by definition the glb w.r.t. the *extensional* order $\sqsubseteq$. But we can prove the following:

Proposition 3.11. *In any actual f-model the following holds:*
For $f, g \in D^\sigma$: If f, g are $\leq$-compatible in the greatest f-model, then $f \sqcap g$ is also the glb w.r.t. $\leq$. (Note: If f, g are $\leq$-compatible in the actual f-model, then they are also compatible in the greatest f-model.)
If $f \leq g$ then $f \sqsubseteq g$. $\leq$ is a partial order on D^σ.

$$\begin{aligned} \textit{For } f, g \in D^{\sigma\to\tau} : f \leq g \Longleftrightarrow\ & \forall x \in D^\sigma.\ fx \leq gx \textit{ and} \\ & \forall x, y \in D^\sigma.\ x \leq y \Longrightarrow fx = fy \sqcap gx \end{aligned}$$

The definition of $\leq$ can be given in "uncurried" form with vectors of arguments, the order $\leq$ extended componentwise:

$$\begin{aligned} \textit{For } f, g \in D^{\sigma_1\to\ldots\to\sigma_n\to\iota} : f \leq g \Longleftrightarrow\ & \forall x_1, y_1 \in D^{\sigma_1}, \ldots, x_n, y_n \in D^{\sigma_n}. \\ & (x_1, \ldots, x_n) \leq (y_1, \ldots, y_n) \Longrightarrow \\ & fx_1 \ldots x_n = fy_1 \ldots y_n \sqcap gx_1 \ldots x_n \end{aligned}$$

Proof. The proof that $f \sqcap g$ is the glb w.r.t. $\leq$ (for $\leq$-compatible f, g) is by induction on the type σ. It uses only the definition of $\leq$ and that $\sqcap$ is the glb w.r.t. $\sqsubseteq$, no stability (or conditional multiplicativity) is used. $\square$

Definition 3.12. $f \in D^{\sigma\to\tau}$ is *conditionally multiplicative (cm)* if

$$\forall x, y \in \mathcal{F}^\sigma.\ x \uparrow_\leq y \Longrightarrow f(x \sqcap y) = fx \sqcap fy$$

Analogously for denotations $f \in \text{ENV} \to D^\sigma$.

This definition can also be given in "uncurried" form: $f \in D^{\sigma_1\to\ldots\to\sigma_n\to\iota}$ is cm iff

$$\begin{aligned} \forall x_1, y_1 \in D^{\sigma_1}, \ldots, x_n, y_n \in D^{\sigma_n}.\ & (x_1, \ldots, x_n) \uparrow_\leq (y_1, \ldots, y_n) \Longrightarrow \\ & f(x_1 \sqcap y_1) \ldots (x_n \sqcap y_n) = fx_1 \ldots x_n \sqcap fy_1 \ldots y_n \end{aligned}$$

Theorem 3.13 (Berry, 4.8.10 in [4])**.** *In an f-model, all functions from domains $D^{\sigma\to\tau}$ are cm. All denotations $[\![M]\!]$ are cm.*

Proof. Berry first proves the property cm for the denotations of normal form terms by induction on the size of the type. Then it is extended to all functions by continuity. $\square$

Proposition 3.14 (Berry [4], syntactic monotony w.r.t. $\leq$)**.**
For every context $C[\quad]$ with hole of type σ, and terms $M, N : \sigma$:
If $[\![M]\!] \leq [\![N]\!]$ then $[\![C[M]]\!] \leq [\![C[N]]\!]$.
Therefore, for terms $M, N : \sigma$: If $M \prec N$ then $[\![M]\!] \leq [\![N]\!]$.

We will also write $M \leq N$ for $[\![M]\!] \leq [\![N]\!]$.

Now we show property I of $(D^\sigma, \leq)$ and the representation of all functions by traces, which is not contained in Berry's thesis.

Proposition 3.15. *For the finite projections we have: $\psi_i^\sigma \leq \psi_{i+1}^\sigma$ and $\psi_i^\sigma \leq \mathrm{id}$.*
The $\mathcal{F}_i^\sigma$ are downward closed w.r.t. $\leq$: If $d \in \mathcal{F}_i^\sigma$, $e \in D^\sigma$ and $e \leq d$, then $e \in \mathcal{F}_i^\sigma$.
Therefore the domains $(D^\sigma, \leq)$ have the property I: There are only finitely many elements under each finite element.

Proof. The proof of $\psi_i^\sigma \leq \mathrm{id}$ is by induction on the type σ; the induction step is in the proof of proposition 12.4.4 in the section on stable bifinite domains of [2, page 287]. The downward closedness of $\mathcal{F}_i^\sigma$ is an easy consequence and can be found at the same place. $\square$

Because of property I, all our functions of $D^{\sigma\to\tau}$ (which are cm) are also stable, and therefore can be represented by traces. We chose the trace of the uncurried form.

Definition 3.16. Let $f \in D^{\sigma_1\to\ldots\to\sigma_n\to\iota}$, $n \geq 0$, $x_i \in D^{\sigma_i}$ and $fx_1\ldots x_n = j$ for some integer j.
Then there are $y_i \in \mathcal{F}^{\sigma_i}$, $y_i \leq x_i$, with $fy_1\ldots y_n = j$ and $(y_1,\ldots,y_n)$ is the $\leq$-least vector with this property. (This is the meaning of: f is stable.)
In this case we say that $y_1{\mapsto}\ldots{\mapsto}y_n{\mapsto}j$ is a *token* of f.
The set of all tokens of f is called the *trace* of f, written $\mathcal{T}(f)$.
The y_i in the token will be represented by traces again. We will use a liberal syntax for tokens and traces, writing $\perp$ for the trace $\emptyset$, $\mathsf{0}$ for the trace $\{\mathsf{0}\}$ of $\mathsf{0}$, and also $\mathsf{00}{\mapsto}\mathsf{0}$ for the token $\{\mathsf{0}\}{\mapsto}\{\mathsf{0}\}{\mapsto}\mathsf{0}$. If M is a closed term, we write simply $\mathcal{T}[\![M]\!]$ for the trace of its denotation $[\![M]\!]\perp$.

Proposition 3.17. *For $f, g \in D^\sigma$: $f \leq g$ iff $\mathcal{T}(f) \subseteq \mathcal{T}(g)$.*
If f, g are $\leq$-compatible in the greatest f-model, then $\mathcal{T}(f \sqcap g) = \mathcal{T}(f) \cap \mathcal{T}(g)$.
$f \in D^\sigma$ is finite of grade i, $f \in \mathcal{F}_i^\sigma$, iff all numbers in the trace of f are $\leq i$.

4. Game Terms

Berry's conjectures demand the existence of certain finite PCF-terms. In this section we show that we may restrict these finite terms to terms in a certain standard normal form that we call *game terms*. This will simplify the proofs of the counter-examples, and is also an interesting result itself. Game terms first appeared in the literature on game semantics as terms representing game strategies; in [1, section 3.2] they were called (finite and infinite) "evaluation trees", in [11, section 7.3] "finite canonical forms" that correspond to compact innocent strategies, and in [2, section 6.6] "PCF Böhm trees". The textbook article on "PCF Böhm trees" comes closest to our approach, as it introduces a semantics in the form of Böhm trees and has to solve similar problems in the needed syntactic transformations. But we do not employ a (game or other) semantics, i.e. we do not interpret the PCF-constants by infinite strategies or Böhm trees; our approach is purely syntactic. We take a finite PCF-term, apply an operator that resembles the finite projection Ψ_i^σ and reduce the resulting term to its game term form. We show that the transforming reductions respect the syntactic order $\prec$ (used in the refutation of Berry's second conjecture), and this will also enable us to proceed to infinite game terms. We also introduce a graphical representation of game terms that makes the behaviour of terms better visible.

First we introduce an additional new construct for the PCF language, for every $i \geq 0$:
If $M, N_0, \ldots, N_i \colon \iota$, then $\mathsf{case}_i\, MN_0\ldots N_i \colon \iota$.
Please note that case_i is not a constant, but the whole case-expression is a new construct of

the language, it is no application. We call the new terms (PCF-)case-terms, and a case-term with all case-expressions as case_i for fixed i we call case_i-term. The reduction rule for case_i is:

$$\mathsf{case}_i\, nN_0 \dots N_i \to N_n, \text{ for } 0 \le n \le i$$

The case-expression is equivalent to a PCF-term:

$$\mathsf{case}_i\, MN_0 \dots N_i \cong \mathsf{if}\ M\ \mathsf{then}\ N_0\ \mathsf{else\ if\ pre}^1\, M\ \mathsf{then}\ N_1\ \mathsf{else} \dots \mathsf{if\ pre}^i\, M\ \mathsf{then}\ N_i\ \mathsf{else}\ \bot$$

This is the "filter" as it appears in the finite projection term Ψ_i^ι. So case_i does not enhance the expressiveness of PCF. It is merely a "macro" that is used as short expression for the filter term above, to keep the unity of the filter term in the transformation to game terms.

The syntactic order $\prec$ is defined on case-terms as follows:

$$\mathsf{case}_i\, MN_0 \dots N_i \prec \mathsf{case}_j\, M'N'_0 \dots N'_j \text{ iff } i \le j,\ M \prec M' \text{ and } N_k \prec N'_k \text{ for } 0 \le k \le i.$$

This is equivalent to the syntactic order on the macro expansions of the case-expressions.

Definition 4.1. *Game terms* are the well-typed PCF-case-terms that are furthermore produced by the following grammar:

$$\begin{aligned} M, N ::= {} & \bot^\sigma,\ \sigma \text{ any type} \\ & \lambda x_1 \dots x_n.m,\ m \text{ integer constant},\ n \ge 0 \\ & \lambda x_1 \dots x_n.\, \mathsf{case}_i(yM_1 \dots M_m)N_0 \dots N_i,\ y \text{ variable},\ n, m, i \ge 0 \end{aligned}$$

Please note that $\lambda x_1 \dots x_n.$ vanishes for $n = 0$, so needed for the N_k of type ι.

A *game term of grade* i, $i \ge 0$, is a game term that is a case_i-term (every case is case_i) with all integer constants $\le i$. (This entails that a closed game term of grade i is a finite term of grade i.)

A *game term of pregrade* i, $i \ge 0$, is a game term that is furthermore produced by the following grammar for the non-terminal N:

$$\begin{aligned} N ::= {} & \bot^\sigma,\ \sigma \text{ any type} \\ & \lambda x_1 \dots x_n.m,\ m \text{ integer constant},\ n \ge 0 \\ & \lambda x_1 \dots x_n.\, \mathsf{case}_i(yM_1 \dots M_m)N_0 \dots N_i,\ y \text{ variable, all } M_k \text{ game term of grade } i, \\ & \qquad n, m \ge 0 \end{aligned}$$

(A game term of pregrade i is a case_i-term.)

Informally, we call the positions in a game term of integer constants at the top level, i.e. where this integer serves as output of the term, *output positions*. So a game term of pregrade i is a game term such that for all integer constants m that are *not* in output position it is $m \le i$. (So the integers at output positions are not restricted.)

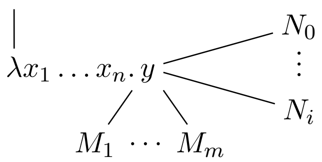
We define a notion for the replacement of integers in output positions of game terms.

Definition 4.2. Let P, L be game terms, $L\colon \iota$ and $l \ge 0$. We define $P\lceil l := L\rceil$ by recursion on P:

$$\begin{aligned} \bot\lceil l := L\rceil &= \bot \\ (\lambda x_1 \dots x_n.l)\lceil l := L\rceil &= \lambda x_1 \dots x_n.L \\ (\lambda x_1 \dots x_n.m)\lceil l := L\rceil &= \lambda x_1 \dots x_n.m, \text{ for } m \ne l \\ (\lambda x_1 \dots x_n.\, \mathsf{case}_i(yM_1 \dots M_m)N_0 \dots N_i)\lceil l := L\rceil &= \lambda x_1 \dots x_n.\, \mathsf{case}_i(yM_1 \dots M_m) \\ & \qquad N_0\lceil l := L\rceil \dots N_i\lceil l := L\rceil \end{aligned}$$

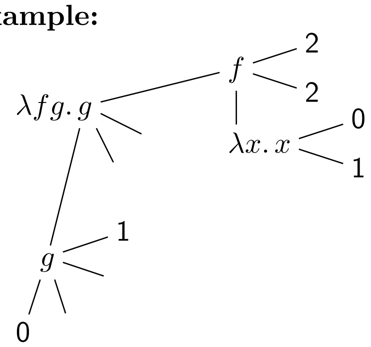
We also write multiple replacements, e.g. $P\lceil l := L_l \text{ for } l \geq 0\rceil$. These multiple replacements are done in parallel, the whole replacement moves down the term.

We will use a graphical representation of game terms in the next sections:
A subterm $\lambda x_1 \ldots x_n.\, \mathsf{case}_i(yM_1 \ldots M_m)N_0 \ldots N_i$ is represented in the graph by a node of the form:



The upper parent of this node is connected to the λ; if the λ is missing, the upper or left parent is connected to the y. The $M_1, \ldots, M_m$ are the *legs* of y; the $N_0, \ldots, N_i$ are the *arms* of y. A leg or arm that points to a $\perp$ is mostly represented simply by a leg or arm pointing to empty space. This graphical representation makes the behaviour of game terms much better visible.

Example:



This is the representation of the term:

$$\lambda fg.\, \mathsf{case}_1[g(\mathsf{case}_1(g\mathsf{0}\perp)\mathsf{1}\perp)\perp][\mathsf{case}_1(f(\lambda x.\, \mathsf{case}_1\, x\mathsf{01}))\mathsf{22}]\perp$$

of type $((\iota \to \iota) \to \iota) \to (\iota \to \iota \to \iota) \to \iota$. It is a game term of *pregrade* 1. The output positions are the two positions of the number 2. If we replace the number 2 at the output positions by $\perp$, $\mathsf{0}$ or $\mathsf{1}$, then we get a game term of *grade* 1.

Game terms are the real "medium" in which to investigate Berry's problems: First, if one seeks terms M which have many semantically different syntactic parts $N \prec M$, according to Berry's second conjecture, then one is naturally led to game terms, because they have a very fine syntactic structure. Second, they simplify the proofs of the counter-examples. The conditional always appears together with a variable, cutting down the cases to be analysed and simplifying the induction hypotheses considerably.

In the next subsection we develop a map gt_i^σ from finite terms to equivalent game terms such that $M \prec N\colon \sigma$ entails $\mathrm{gt}_i^\sigma(M) \prec \mathrm{gt}_j^\sigma(N)$, where M, N are of grade i resp. j, $i \leq j$. This means that the refutation of Berry's conjectures may be restricted to game terms. In the following subsection we extend our result to infinite game terms. They are needed for a full formulation of Berry's conjectures for first-order types (where they are valid).

4.1. **Finite Game Term Theorem.** We are given finite terms $M \prec N$ and want to find equivalent game terms. First we must get rid of the Ys in the terms.

The map $\omega\colon \mathrm{PCF}^\sigma \to \mathrm{PCF}^\sigma$ (for all types σ) is taken from [5, 4] and called the *immediate syntactic value*:

$$\omega(M) = \begin{cases} \lambda x_1 \dots x_n . u\, \omega(M_1) \dots \omega(M_m), & \text{if } M = \lambda x_1 \dots x_n . u M_1 \dots M_m \\ & \text{with } u \text{ a variable or constant,} \\ & \text{i.e. } M \text{ is in head normal form} \\ \perp \text{ else} \end{cases}$$

Please note here that a constant is $\mathsf{suc}, \mathsf{pre}, \mathsf{if}, 0, 1, 2, \dots$ A constant is not $\perp$ or Y.
$\to_{\beta Y}$ is the one-step reduction with the β-rule or the rule $\mathsf{Y}M \to M(\mathsf{Y}M)$ in any context.
As is known from [5, 4], if $M \to_{\beta Y}^* N$, then $\omega(M) \prec \omega(N)$.

Lemma 4.3 (Approximation Lemma)**.** *For every finite term M there is a term N' such that $M \to_{\beta Y}^* N$ for some N, $N' \prec \omega(N)$, $M \cong N'$ and N' is the $\prec$-least term with this property. This unique N' is called* $\mathrm{approx}(M)$*.*

Proof. For the fully abstract cpo-model (and therefore for all f-models) the approximation continuity theorem [5, theorem 4.3.1] is valid:

$$\{\, [\![\omega(N)]\!] \mid M \to_{\beta Y}^* N \,\} \to [\![M]\!].$$

The set on the left is directed and M is finite, therefore there is N with $M \to_{\beta Y}^* N$ and $[\![M]\!] = [\![\omega(N)]\!]$.

Now assume the type of M, N is $\sigma_1 \to \dots \to \sigma_n \to \iota$. Take any vector of closed terms $A_1\colon \sigma_1, \dots, A_n\colon \sigma_n$ with $\omega(N) A_1 \dots A_n \to^* m$ (integer constant).

By syntactic stability [4, theorem 2.8.8] [5, theorem 3.6.7] there is a $\prec$-least term $N^* \prec \omega(N)$ with $N^* A_1 \dots A_n \to^* m$. Take as N' the $\prec$-lub of all these N^*. ☐

Lemma 4.4. *For all finite terms $M \prec N$ it is* $\mathrm{approx}(M) \prec \mathrm{approx}(N)$*.*

Proof. Let M' be a term with $M \to_{\beta Y}^* M'$ and $M \cong \omega(M')$.
As the β-rule and the Y-rule do not involve $\perp$, all these reductions $M \to_{\beta Y}^* M'$ can also be done in N. (If $A \prec B$ and $A \to_{\beta Y} A'$, then there is B' with $B \to_{\beta Y} B'$ and $A' \prec B'$.)
So there is N' with $N \to_{\beta Y}^* N'$ and $M' \prec N'$, and of course $\omega(M') \prec \omega(N')$.

By confluence of $\to_{\beta Y}$ there is N'' with $N' \to_{\beta Y}^* N''$ and $N \cong \omega(N'')$.
It is $\omega(N') \prec \omega(N'')$, therefore $\omega(M') \prec \omega(N'')$.
$\mathrm{approx}(M)$ is the least term X with $X \prec \omega(N'')$ and $M \sqsubseteq_{op} X$.
$\mathrm{approx}(N)$ fulfills the two conditions for X, therefore $\mathrm{approx}(M) \prec \mathrm{approx}(N)$. ☐

Now we have finite terms $\mathrm{approx}(M) \prec \mathrm{approx}(N)$ without Y. The next step is to apply a Ψ_i^σ-like operator to the terms and reduce according to some reduction rules to game terms. The proof can be done in different ways:

In my first version I proved the termination of the reductions, formulated an invariant of the (eta-expanded) term structure, proved the invariance under the reductions and that they lead to game terms. This resulted in an induction on the reduction sequence, the induction step done by induction on the term, causing much rewriting bureaucracy. (This ugly proof is available as supplementary material from my home page.)

Here we will see a more elegant half-sized proof based on an induction on the term from the beginning, with the aid of a reducibility predicate (see e.g. [24, theorem 3.1]).

(Jim Laird also uses a reducibility predicate to produce eta-expanded normal forms of a simply typed λ-calculus with lifting (without inconsistent values) [12, proposition 4.2].)

To produce the game terms we define for every $i \geq 0$ a big-step reduction relation $M \downarrow_i N$ on case_i-terms. The mere existence of the game terms could be proved without $\downarrow_i$, but we want to give an explicit deterministic algorithm. (Determinism is easily built into big-step reduction.) The values for $\downarrow_i$, i.e. the terms that we consider as the results of reductions, are the game terms of pregrade i.

Here are the rules for $\downarrow_i$. In the hypothesis of a rule the abbreviation $M \downarrow_i N\ gi$ means "$M \downarrow_i N$ and N is a game term of grade i", $M \downarrow_i N\ pi$ means "$M \downarrow_i N$ and N is a game term of pregrade i".

(0) $n \downarrow_i n$ for all integer constants n

(1) $\dfrac{M[y := M_1]M_2 \ldots M_m \downarrow_i P}{(\lambda y.M)M_1 M_2 \ldots M_m \downarrow_i P}$, for $m \geq 1$ (2) $\bot M_1 \ldots M_m \downarrow_i \bot$, for $m \geq 0$

(3) $\dfrac{A \downarrow_i A'\ pi}{\mathsf{suc}\, A \downarrow_i A'\lceil m := m+1 \text{ for } m \geq 0\rceil}$ (4) $\dfrac{A \downarrow_i A'\ pi}{\mathsf{pre}\, A \downarrow_i A'\lceil \mathsf{0} := \bot,\ m := m-1 \text{ for } m \geq \mathsf{1}\rceil}$

(5) $\dfrac{A_k \downarrow_i A'_k\ pi, \text{ for } k = 1,2,3}{\mathsf{if}\ A_1\ \mathsf{then}\ A_2\ \mathsf{else}\ A_3 \downarrow_i A'_1\lceil \mathsf{0} := A'_2,\ m := A'_3 \text{ for } m \geq 1\rceil}$

(6) $\dfrac{A \downarrow_i A'\ pi, A' \neq \bot}{\lambda x_1 \ldots x_n.\, \mathsf{case}_i\, A\mathsf{0} \ldots i \downarrow_i \lambda x_1 \ldots x_n.A'\lceil k := \bot \text{ for } k > i\rceil}$, for $n \geq 0$

(7) $\dfrac{A \downarrow_i \bot}{\lambda x_1 \ldots x_n.\, \mathsf{case}_i\, A\mathsf{0} \ldots i \downarrow_i \bot}$, for $n \geq 0$

(8) $\dfrac{A_k \downarrow_i A'_k\ gi, \quad \text{for } 1 \leq k \leq m}{\mathsf{case}_i(xA_1 \ldots A_m)\mathsf{0} \ldots i \downarrow_i \mathsf{case}_i(xA'_1 \ldots A'_m)\mathsf{0} \ldots i}$, for $m \geq 0$

Remarks: Not for all case_i-terms $M : \sigma$ there is a value V with $M \downarrow_i V$, but there will be a value V with $\Psi_i^\sigma M \downarrow_i V$ for Ψ_i^σ suitably defined. The reduction relations are complete enough for the purposes of the following proofs. So to understand the reductions at this stage, just check the soundness of each rule separately, according to the following lemma, and do not bother about completeness. When you go through the subsequent proofs, you will see that exactly these rules are needed, no more, no less.

Lemma 4.5 (soundness of the reduction relations $\downarrow_i$)**.** *For all case_i-terms M, M': If $M \downarrow_i M'$, then $[\![M]\!] = [\![M']\!]$ and M' is a value (i.e. a game term of pregrade i).*

Proof. Translate each reduction rule into a rule with semantic equivalence instead of the reduction relation: Translate statements $A \downarrow_i A'$ into ($[\![A]\!] = [\![A']\!]$ and A' is a value), and keep the statements gi and pi. Then check each translated rule for validity. $\square$

Now we come to the reducibility predicate. We pack all that we want to prove into its definition: the compatibility of the transformation with the order $\prec$ and even the uniqueness of the reduction $\downarrow_i$.

Definition 4.6 (reducibility predicate)**.** Let $i \leq j$, A a case_i-term and B a case_j-term of type $\sigma = \sigma_1 \to \ldots \to \sigma_n \to \iota$, $n \geq 0$.
$A \prec B \colon \sigma$ are (i,j)-*transformable*, written $A \prec B \colon \sigma(i,j)$,
iff for all $A_l \prec B_l \colon \sigma_l(i,j)$, $1 \leq l \leq n$, there are game terms $A', B' \colon \iota$ of pregrade i resp. j with $AA_1 \ldots A_n \downarrow_i A'$ and $BB_1 \ldots B_n \downarrow_j B'$,
A' and B' are unique for these reductions, and furthermore $A' \prec B'$.

Note that this definition does not take care of the free variables of A, B. Note also that it does not demand the *grade* i, j of A', B', but the *pregrade*. So it will be applicable to general terms that do not restrict the integer constants, in lemma 4.9.

Lemma 4.7. *If $A \prec B \colon \sigma(i,j)$, then $\perp \prec B \colon \sigma(k,j)$ for all $k \leq j$.*

Proof. Easy consequence of the definition of the reducibility predicate and of rule (2) for $\perp$-application. ☐

For the next lemma we need a notion of simultaneous substitution for PCF-terms that properly renames bound variables. We take Allen Stoughton's definitions [29].

A *substitution* is a function s, t from variables to terms (of the type of the variable). The substitution $s[x := N]$ is defined by $(s[x := N])x = N$ and $(s[x := N])y = sy$ for $y \neq x$. id is the identity substitution.

If x is a variable, M a term, s a substitution, then we define

$$\mathrm{new}\, xMs = \{\, y \mid y \text{ variable and for all } z \in FV(M) - \{x\}.\ y \notin FV(sz)\,\},$$

where $FV(X)$ is the set of free variables of term X.

The *simultaneous substitution* Ms of sx for the free occurrences of x in M, for all x, is defined by structural recursion on M:

$$\begin{aligned}
xs &= sx, \text{ for every variable } x\\
cs &= c, \text{ for every constant } c\\
(MN)s &= (Ms)(Ns)\\
(\lambda x.M)s &= \lambda y.(M(s[x := y])), \text{ with } y = \mathrm{choice}(\mathrm{new}\, xMs),
\end{aligned}$$

where choice is a fixed function that chooses some variable y from the argument set of variables.

We suppose that the normal substitution (in the β-rule) behaves like this:

$$P[y := N] = P(\mathrm{id}[y := N]).$$

Lemma 4.8. *For terms M, N, substitution s and variables x, y with $y =$ choice(new xMs) we have:*

$$(M(s[x := y]))[y := N] = M(s[x := N])$$

Proof. Follows from theorem 3.2 of [29]. ☐

Lemma 4.9. *Let $A \prec B \colon \sigma$ be PCF-terms without* Y.
Let $\{x_1^{\tau_1}, \ldots, x_m^{\tau_m}\}$ be a superset of the free variables of B.
For $1 \le k \le m$ let $A'_k \prec B'_k \colon \tau_k(i,j)$ be case_i*- resp.* case_j*-terms that are (i,j)-transformable.*
Define the substitutions $s = \mathrm{id}[x_1 := A'_1] \ldots [x_m := A'_m]$ and $t = \mathrm{id}[x_1 := B'_1] \ldots [x_m := B'_m]$.
Then $As \prec Bt \colon \sigma(i,j)$.

Proof. By induction on the term B. (Note: PCF-terms are without case.)
Case $B = B^* B_0$, $B^* \colon \sigma_0 \to \sigma_1 \to \ldots \to \sigma_n \to \iota$, for $n \ge 0$:
First let $A = A^* A_0$.
By the induction hypothesis we get $A^*s \prec B^*t \colon \sigma_0 \to \ldots \sigma_n \to \iota(i,j)$ and $A_0 s \prec B_0 t \colon \sigma_0(i,j)$.
Let $A_l \prec B_l \colon \sigma_l(i,j)$ for $1 \le l \le n$.
By the reducibility predicate there are game terms $A' \prec B' \colon \iota$ of pregrade i resp. j with

$$(A^*s)(A_0 s)A_1 \ldots A_n \downarrow_i A'$$
$$(B^*t)(B_0 t)B_1 \ldots B_n \downarrow_j B'$$

So $As \prec Bt \colon \sigma(i,j)$.

Now let $A = \bot$. By the same argument we have $Bt \prec Bt \colon \sigma(j,j)$, therefore by lemma 4.7: $\bot \prec Bt \colon \sigma(i,j)$.

Case $B = \lambda x.B^* \colon \sigma_1 \to \ldots \to \sigma_n \to \iota$, $n \ge 1$:
First let $A = \lambda x.A^*$.
Let $A_l \prec B_l \colon \sigma_l(i,j)$ for $1 \le l \le n$.
By the induction hypothesis for B^* we get

$$A^*(s[x := A_1]) \prec B^*(t[x := B_1]) \colon \sigma_2 \to \ldots \to \sigma_n \to \iota(i,j).$$

Therefore there are game terms $A', B' \colon \iota$ of pregrade i resp. j with

$$(A^*(s[x := A_1]))A_2 \ldots A_n \downarrow_i A'$$
$$(B^*(t[x := B_1]))B_2 \ldots B_n \downarrow_j B',$$

with A', B' unique and $A' \prec B'$.
By lemma 4.8 and the definition of substitution we get:

$$A^*(s[x := A_1]) = (A^*(s[x := y]))[y := A_1], \text{ for } y = \mathrm{choice}(\mathrm{new}\, x A^* s)$$
$$B^*(t[x := B_1]) = (B^*(t[x := z]))[z := B_1], \text{ for } z = \mathrm{choice}(\mathrm{new}\, x B^* t)$$
$$(\lambda x.A^*)s = \lambda y.A^*(s[x := y])$$
$$(\lambda x.B^*)t = \lambda z.B^*(t[x := z])$$

Then it reduces

$$(A^*(s[x := y]))[y := A_1]A_2 \ldots A_n \downarrow_i A', \text{ and therefore by rule (1):}$$
$$(\lambda y.A^*(s[x := y]))A_1 A_2 \ldots A_n \downarrow_i A', \text{ therefore}$$
$$(\lambda x.A^*)s A_1 A_2 \ldots A_n \downarrow_i A'.$$

Analogously:

$$(\lambda x.B^*)t B_1 B_2 \ldots B_n \downarrow_j B'.$$

These reductions are unique, and $A' \prec B'$. So $As \prec Bt \colon \sigma(i,j)$.

Now let $A = \bot$. By the same argument we have $Bt \prec Bt\colon \sigma(j,j)$, therefore by lemma 4.7: $\bot \prec Bt\colon \sigma(i,j)$.

Cases $B = x$ (variable), $B = n$ (integer constant), $B = \bot$ are clear.
For $B = n$ rule (0) is used, for $B = \bot$ rule (2).
For the subcases $A = \bot$ lemma 4.7 is used.

Case $B = \mathsf{if}$:
First let $A = \mathsf{if}$.
Let $A_l \prec B_l\colon \iota(i,j)$ for $1 \le l \le 3$.
Then there are $A_l \downarrow_i A'_l$ and $B_l \downarrow_j B'_l$ (A'_l, B'_l unique) with $A'_l \prec B'_l$, for $1 \le l \le 3$.
It reduces by rule (5):

$$\mathsf{if}\, A_1\, \mathsf{then}\, A_2\, \mathsf{else}\, A_3 \downarrow_i A'_1\lceil \mathsf{0} := A'_2,\ m := A'_3 \text{ for } m \ge 1\rceil$$
$$\mathsf{if}\, B_1\, \mathsf{then}\, B_2\, \mathsf{else}\, B_3 \downarrow_j B'_1\lceil \mathsf{0} := B'_2,\ m := B'_3 \text{ for } m \ge 1\rceil$$

Both reductions are unique and the results are in relation $\prec$.
Now let $A = \bot$. By lemma 4.7 it is $\bot \prec \mathsf{if}\colon \sigma(i,j)$.

Cases $B = \mathsf{suc}$, $B = \mathsf{pre}$: analogous to $B = \mathsf{if}$.
For $B = \mathsf{suc}$ rule (3) is used, for $B = \mathsf{pre}$ rule (4). □

Next we prove a lemma that introduces the terms Ψ_i^σ into the transformation. For the rest of this section we redefine the finite projection terms Ψ_i^σ as equivalent case_i-terms:

$$\Psi_i^{\sigma_1\to\ldots\to\sigma_n\to\iota} = \lambda f.\lambda x_1\ldots x_n.\,\mathsf{case}_i[f(\Psi_i^{\sigma_1}x_1)\ldots(\Psi_i^{\sigma_n}x_n)]\mathsf{0}\ldots i, \text{ for } n \ge 0.$$

Lemma 4.10. *For all types $\sigma = \sigma_1\to\ldots\to\sigma_n\to\iota$ the following three propositions are valid:*
(1) For all $A \prec B\colon \sigma(i,j)$ it is

$$A(\Psi_i^{\sigma_1}x_1)\ldots(\Psi_i^{\sigma_n}x_n) \prec B(\Psi_j^{\sigma_1}x_1)\ldots(\Psi_j^{\sigma_n}x_n)\colon \iota(i,j).$$

(2) For all $A \prec B\colon \sigma(i,j)$ there are $A', B'\colon \sigma$ with $\Psi_i^\sigma A \downarrow_i A'$ and $\Psi_j^\sigma B \downarrow_j B'$ such that both are unique for this reduction, and furthermore $A' \prec B'$ and they are game terms of grade *i resp. j.*
(3) For all variables x^σ and $i \le j$: $\Psi_i^\sigma x^\sigma \prec \Psi_j^\sigma x^\sigma\colon \sigma(i,j)$.

Proof. By simultaneous induction on the type σ.
(1) By the induction hypothesis for (3) we get $\Psi_i^{\sigma_k}x_k \prec \Psi_j^{\sigma_k}x_k\colon \sigma_k(i,j)$, for $1 \le k \le n$, and the proposition follows.
(2) The proposition (1) means that there are game terms $A'', B''\colon \iota$ with pregrade i resp. j such that $A(\Psi_i^{\sigma_1}x_1)\ldots(\Psi_i^{\sigma_n}x_n) \downarrow_i A''$ and $B(\Psi_j^{\sigma_1}x_1)\ldots(\Psi_j^{\sigma_n}x_n) \downarrow_j B''$, with A'', B'' unique for this reduction and $A'' \prec B''$.
If $A'' = \bot$ then it reduces by rule (7):

$$\lambda x_1\ldots x_n.\,\mathsf{case}_i[A(\Psi_i^{\sigma_1}x_1)\ldots(\Psi_i^{\sigma_n}x_n)]\mathsf{0}\ldots i \downarrow_i \bot$$

and therefore by rule (1): $\Psi_i^\sigma A \downarrow_i \bot$.
If also $B'' = \bot$, then likewise $\Psi_j^\sigma B \downarrow_j \bot$ and the proposition follows.
(We still have $A'' = \bot$.) If $B'' \ne \bot$ then it reduces by rule (6):

$$\lambda x_1\ldots x_n.\,\mathsf{case}_j[B(\Psi_j^{\sigma_1}x_1)\ldots(\Psi_j^{\sigma_n}x_n)]\mathsf{0}\ldots j \downarrow_j \lambda x_1\ldots x_n.B''\lceil k := \bot \text{ for } k > j\rceil = B'$$

and therefore by rule (1): $\Psi_j^\sigma B \downarrow_j B'$, B' is a game term of grade j, and the proposition follows.

If $A'' \neq \perp$ and $B'' \neq \perp$, then we get like the last reduction by rules (6) and (1):

$$\Psi_i^\sigma A \downarrow_i \lambda x_1 \ldots x_n.A''\lceil k := \perp \text{ for } k > i\rceil = A'$$

$$\Psi_j^\sigma B \downarrow_j \lambda x_1 \ldots x_n.B''\lceil k := \perp \text{ for } k > j\rceil = B'$$

Both reductions are unique, it is $A' \prec B'$ and they are game terms of grade i resp. j.

(3) We have to prove that for all $A_l \prec B_l \colon \sigma_l(i,j)$, $1 \leq l \leq n$, there are game terms $A' \prec B'$ of pregrade i resp. j with $(\Psi_i^\sigma x)A_1 \ldots A_n \downarrow_i A'$ and $(\Psi_j^\sigma x)B_1 \ldots B_n \downarrow_j B'$ (with uniqueness of the reductions).

By the induction hypothesis of (2) for all l there are game terms $A'_l \prec B'_l \colon \sigma_l$ of grade i resp. j with $\Psi_i^{\sigma_l} A_l \downarrow_i A'_l$ and $\Psi_j^{\sigma_l} B_l \downarrow_j B'_l$ (with uniqueness of the reductions).

It reduces by rule (8)

$$\mathsf{case}_i[x(\Psi_i^{\sigma_1} A_1) \ldots (\Psi_i^{\sigma_n} A_n)]0 \ldots i \downarrow_i \mathsf{case}_i[xA'_1 \ldots A'_n]0 \ldots i = A'$$

and therefore by rule (1):

$$(\Psi_i^\sigma x)A_1 \ldots A_n \downarrow_i A'$$

Likewise it reduces by rules (8) and (1):

$$(\Psi_j^\sigma x)B_1 \ldots B_n \downarrow_j \mathsf{case}_j[xB'_1 \ldots B'_n]0 \ldots j = B'$$

A', B' are even game terms of *grade* i resp. j. The reductions are unique. It is $A' \prec B'$. □

Definition 4.11. Let A be a case_i-term without Y with $A \prec A \colon \sigma(i,i)$.
The unique game term A' of grade i with $\Psi_i^\sigma A \downarrow_i A'$ is called $\mathrm{proj}_i^\sigma(A)$.
For every finite term $M \colon \sigma$ we get $\mathrm{approx}(M)$ without Y with $\mathrm{approx}(M) \prec \mathrm{approx}(M) \colon \sigma(i,i)$ by lemma 4.9. (Note that finite terms are closed.)
We define the map $\mathrm{gt}_i^\sigma(M) = \mathrm{proj}_i^\sigma(\mathrm{approx}(M))$, for $M \colon \sigma$ finite term of grade i.

Theorem 4.12 (Game Term Theorem)**.** *If $i \leq j$ and $M \prec N \colon \sigma$ are finite PCF-terms of grade i resp. j, then $\mathrm{gt}_i^\sigma(M) \prec \mathrm{gt}_j^\sigma(N)$ are game terms of grade i resp. j with $M \cong \mathrm{gt}_i^\sigma(M)$ and $N \cong \mathrm{gt}_j^\sigma(N)$.*

Proof. By lemma 4.3 and 4.4 we get $\mathrm{approx}(M) \prec \mathrm{approx}(N)$ without Y. By lemma 4.9 it is $\mathrm{approx}(M) \prec \mathrm{approx}(N) \colon \sigma(i,j)$. By lemma 4.10(2) $\mathrm{proj}_i^\sigma(\mathrm{approx}(M)) \prec \mathrm{proj}_j^\sigma(\mathrm{approx}(N))$ are game terms of grade i resp. j. Furthermore $M \cong \Psi_i^\sigma(\mathrm{approx}(M)) \cong \mathrm{proj}_i^\sigma(\mathrm{approx}(M))$ and likewise for N. □

4.2. **Infinite game terms.**

Definition 4.13. An *infinite game term* of type σ is an ideal of game terms of type σ (of any grade), under the ordering $\prec$. (Infinite game terms can be construed as Böhm trees with infinite case-expressions, which we write as $\mathsf{case}_\infty\, MN_0N_1 \ldots$.) The order $\prec$ on infinite game terms is the subset order of the ideals. The semantics (in some f-model) of an infinite game term is the lub of the semantics of the members of its ideal, if the lub exists in the f-model.

Definition 4.14. Let $M \colon \sigma$ be a closed PCF-term.
$\Psi_0^\sigma M \prec \Psi_1^\sigma M \prec \Psi_2^\sigma M \prec \ldots$ is an ascending chain of finite terms with ascending grade.
Define $\mathrm{gt}^\sigma(M)$ as the lub (in the order of infinite game terms) of the ascending chain of game terms $\mathrm{gt}_0^\sigma(\Psi_0^\sigma M) \prec \mathrm{gt}_1^\sigma(\Psi_1^\sigma M) \prec \mathrm{gt}_2^\sigma(\Psi_2^\sigma M) \prec \ldots$.

Theorem 4.15 (Infinite Game Term Theorem)**.** *If $M \prec N : \sigma$ are closed PCF-terms, then $\mathrm{gt}^\sigma(M) \prec \mathrm{gt}^\sigma(N)$ are infinite game terms with $[\![M]\!] = [\![\mathrm{gt}^\sigma(M)]\!]$ and $[\![N]\!] = [\![\mathrm{gt}^\sigma(N)]\!]$ in any f-model.*

Proof. By proposition 3.8 it is $[\![\Psi^\sigma_i M]\!] \to [\![M]\!]$, therefore $[\![M]\!] = [\![\mathrm{gt}^\sigma(M)]\!]$, and likewise $[\![N]\!] = [\![\mathrm{gt}^\sigma(N)]\!]$. As $\mathrm{gt}^\sigma_i(\Psi^\sigma_i M) \prec \mathrm{gt}^\sigma_i(\Psi^\sigma_i N)$ for all i, we get $\mathrm{gt}^\sigma(M) \prec \mathrm{gt}^\sigma(N)$. □

5. The syntactic order is not the image of the stable order

Berry's second conjecture in its finite form says that the stable order of the order-extensional fully abstract cpo-model of PCF (our greatest f-model) has the syntactic order as its image:
If $a \leq b$ for finite a, b in the model, then there are normal form terms A, B with $[\![A]\!] = a$, $[\![B]\!] = b$ and $A \prec B$.
(The choice of the greatest f-model is not important, as all f-models coincide on their finite parts.)

In this section we will first show that Berry's second conjecture is valid in first-order types. Then we give our simplest counter-example in finitary PCF of second-order type, a chain of length 2. We also give examples of chains of any finite length.

For first-order types Berry's conjecture can be strengthened to the infinite case:

Theorem 5.1 (Berry, Theorem 4.1.7 and 4.8.14 in [4])**.** *Let σ be a first-order type, and $b \in D^\sigma$ in the greatest f-model. Then there is an infinite game term B with $b = [\![B]\!]$. Furthermore, for all such infinite game terms B and every subset $t \subseteq \mathcal{T}(b)$ there is an infinite game term $A \prec B$ with $\mathcal{T}[\![A]\!] = t$. (As infinite game term, A has a denotation in the greatest f-model.)*

Proof. Let $\sigma = \iota \to \iota \to \ldots \to \iota$ with $n \geq 1$ arguments. In [4, 4.1.7] Berry shows that $b \in D^\sigma$, as the lub of a growing sequence of finite sequential functions, is itself sequential. Therefore: If b is not some constant function, then b is strict in some j-th argument. So B can be recursively constructed as infinite game term (with case_∞ the infinite case) in the form:

$$B = \lambda x_1 \ldots x_n.\, \mathsf{case}_\infty\, x_j B_1 B_2 \ldots,$$

where B_i is a term with free variables $x_1, \ldots x_{j-1} x_{j+1} \ldots x_n$ for the residual function b_i given by

$$b_i x_1 \ldots x_{j-1} x_{j+1} \ldots x_n = b x_1 \ldots x_{j-1} i x_{j+1} \ldots x_n.$$

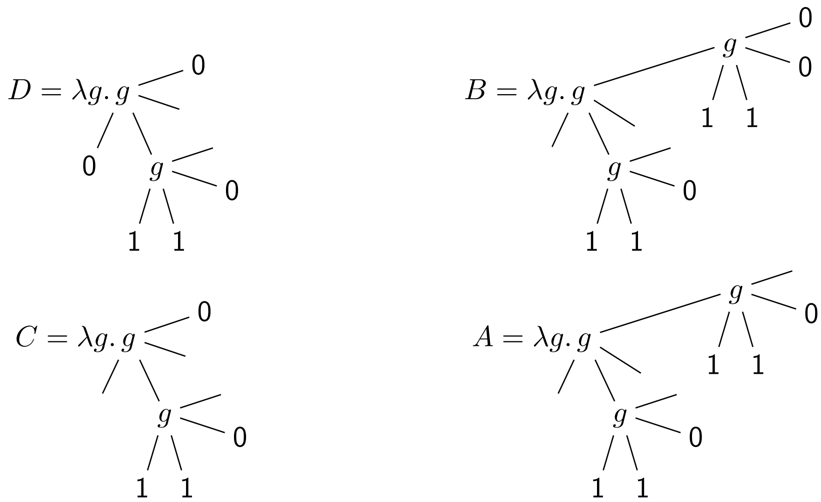
In [4, 4.8.14] Berry shows that A can be constructed in the same manner B was constructed, i.e. following the same choice of the variables for which the function is strict. We can describe the construction of A differently by using traces: The tokens of the trace $\mathcal{T}[\![B]\!]$ correspond exactly to the branches of B that output a result, i.e. do not lead to $\perp$. We simply choose $A \prec B$ by setting those branches of B that do not correspond to a token in t to the empty output $\perp$. □

We conjecture that Berry's second conjecture is also true for second-order types with parameters of arity at most one:

Conjecture 5.2. Let $\sigma = \sigma_1 \to \ldots \to \sigma_n \to \iota$ with $\sigma_i = \iota$ or $\sigma_i = \iota \to \iota$ for all i. Let $b \in \mathcal{F}_i^\sigma$ be a finite element of grade i.
Then there is a game term B of grade i for b, $b = [\![B]\!]$, such that for every subset $t \subseteq \mathcal{T}(b)$ that is secured in the sense of definition 2 of [7] there is $A \prec B$ with $\mathcal{T}[\![A]\!] = t$. (The trace of every semantic element is secured, so Berry's second conjecture would be fulfilled for these types.)

The proof of this conjecture is in preparation. It needs a new theory of (PCF-)terms that would exceed the frame of this paper.

5.1. **Refutation of Berry's second conjecture: A chain of least length 2.** Our simplest counter-example to Berry's second conjecture is in finitary PCF of second-order type $(\iota \to \iota \to \iota) \to \iota$. We consider the following game terms A, B, C, D:



$$\begin{array}{l}
D = \lambda g.\ \mathsf{case}_1(g\,0\,(\mathsf{case}_1(g\,1\,1)\bot\,0))\,0\,\bot\\
C = \lambda g.\ \mathsf{case}_1(g\bot(\mathsf{case}_1(g\,1\,1)\bot\,0))\,0\,\bot\\
B = \lambda g.\ \mathsf{case}_1(g\bot(\mathsf{case}_1(g\,1\,1)\bot\,0))(\mathsf{case}_1(g\,1\,1)\,0\,0)\bot\\
A = \lambda g.\ \mathsf{case}_1(g\bot(\mathsf{case}_1(g\,1\,1)\bot\,0))(\mathsf{case}_1(g\,1\,1)\bot\,0)\bot
\end{array}$$

For illustration (not for the proof) we give the trace semantics of these terms:

$$\left.\begin{array}{l}
\left.\begin{array}{l}
A\left\{\begin{array}{l}
\{1\,1\mapsto 1,\ \ \bot\,0\mapsto 0\}\mapsto 0\\
\{\bot\,1\mapsto 1,\ \ \bot\,0\mapsto 0\}\mapsto 0
\end{array}\right.\\
\ \{\qquad\quad\ \ \bot\bot\mapsto 0\}\mapsto 0
\end{array}\right\} B \cong C\\
\ \{\qquad\quad\ \ 0\,\bot\mapsto 0\}\mapsto 0\\
\ \{1\,1\mapsto 1,\ \ 0\,0\mapsto 0\}\mapsto 0\\
\ \{1\,\bot\mapsto 1,\ \ 0\,0\mapsto 0\}\mapsto 0\\
\ \{\bot\,1\mapsto 1,\ \ 0\,0\mapsto 0\}\mapsto 0
\end{array}\right\} D$$

We have $A \prec B \cong C \prec D$, therefore $[\![A]\!] \leq [\![D]\!]$. We will prove that this chain of two steps of $\prec$ cannot be replaced by one single step.

Proof of the equivalence $B \cong C$: For any argument g, if Cg converges (i.e. reduces to an integer constant), then the subterm $g\mathsf{11}$ of C converges also. (There are only two possibilities for g: either $\mathcal{T}(g) = \{\perp\perp\mapsto\mathsf{0}\}$, or g demands its second argument.) Therefore it is possible to safely replace the result $\mathsf{0}$ in C by the term $\mathsf{case}_1(g\mathsf{11})\mathsf{00}$, i.e. to "lift" $g\mathsf{11}$ to the top level.

It is important to notice that this transformation cannot be performed with D: Here there are more possibilities for g to make Dg converge. It might be that $\mathcal{T}(g) = \{\mathsf{0}\perp\mapsto\mathsf{0}\}$, then the subterm $g\mathsf{11}$ does not converge.

The intuition of the example: We start with term D, working downwards step by step to A eliminating tokens of the trace. First the token $\{\mathsf{0}\perp\mapsto\mathsf{0}\}\mapsto\mathsf{0}$ is eliminated getting C (and the other tokens with g demanding its first argument $\mathsf{0}$). Then it becomes possible to lift $g\mathsf{11}$, we get $B \cong C$. Next we eliminate the token $\{\perp\perp\mapsto\mathsf{0}\}\mapsto\mathsf{0}$ in B to get A. This is done by "forcing" the evaluation of the second argument of g, by demanding that g delivers different results for different arguments.

Proposition 5.3. *Let A, D be the game terms of grade 1 above. There are no game terms A', D' of grade 1 with $A' \prec D'$ and $A' \cong A$, $D' \cong D$. Then by the game term theorem 4.12 there are no PCF-terms A', D' with this property. Since we have seen that $[\![A]\!] \leq [\![D]\!]$, the proposition refutes Berry's second conjecture.*

Proof. As game terms of grade 1, A' and D' should be of the form $\lambda g.S$, where $S\colon \iota$ is a game term possibly with the only free variable g. We abbreviate $S[g := M]$ as $S[M]$.

Let $R, P, Q\colon \iota \to \iota \to \iota$ be the following terms:

$$\begin{aligned} R &= \lambda xy.\,\mathsf{case}_1\, y\, \mathsf{0}(\mathsf{case}_1\, x \perp \mathsf{1}), & \mathcal{T}[\![R]\!] &= \{\mathsf{11}\mapsto\mathsf{1},\ \ \perp\mathsf{0}\mapsto\mathsf{0}\} \\ P &= \lambda xy.\mathsf{0}, & \mathcal{T}[\![P]\!] &= \{\perp\perp\mapsto\mathsf{0}\} \\ Q &= \lambda xy.\,\mathsf{case}_1\, x\, \mathsf{0}\perp, & \mathcal{T}[\![Q]\!] &= \{\mathsf{0}\perp\mapsto\mathsf{0}\} \end{aligned}$$

We will prove: For any terms S, S' of the form above,

$$\text{if } S' \prec S \text{ and } S[Q] \to^* \mathsf{0} \text{ and } S'[R] \to^* \mathsf{0}, \text{ then } S'[P] \to^* \mathsf{0}.$$

The proposition follows from this claim, as $DQ \to^* \mathsf{0}$ and $AR \to^* \mathsf{0}$, but not $AP \to^* \mathsf{0}$.

The proof of the claim is by induction on the term S:

The cases $S = \perp, \mathsf{0}, \mathsf{1}$ are clear.

Let $S = \mathsf{case}_1(gS_1S_2)S_3S_4$ and $S' \prec S$ with $S' = \mathsf{case}_1(gS_1'S_2')S_3'S_4'$. (The remaining case $S' = \perp$ is clear.)

Suppose $S[Q] \to^* \mathsf{0}$ and $S'[R] \to^* \mathsf{0}$. Then $S_1[Q] \to^* \mathsf{0}$.

R and Q are compatible in the Scott model of all continuous functions, the "parallel or" is an upper bound. Expressed differently, R and Q are compatible in the sense that they produce compatible integer results for the same argument. Therefore the semantics of $S_1[R]$ and $S_1[Q]$ must be compatible, so it is not possible that $S_1[R] \to^* \mathsf{1}$.

As $S_1' \prec S_1$, it is also not possible that $S_1'[R] \to^* \mathsf{1}$.

Therefore $(gS_1'S_2')[R] \to^* \mathsf{0}$ (it must converge to get $S'[R] \to^* \mathsf{0}$).

Hence $S'[R] \to^* S_3'[R] \to^* \mathsf{0}$.

On the other side we have $S[Q] \to^* S_3[Q] \to^* \mathsf{0}$.

Together we have $S_3[Q] \to^* \mathsf{0}$ and $S_3'[R] \to^* \mathsf{0}$, and by the induction hypothesis for S_3

follows: $S'_3[P] \to^* \mathsf{0}$.
Therefore $S'[P] \to^* S'_3[P] \to^* \mathsf{0}$. $\square$

Remark 5.4. As we base our proof on game terms, we gave a special induction hypothesis for the combination of case_1 and g. The proof for general normal form terms is more complicated as it must work with if and g separately and use a more general induction hypothesis, i.e. one proves by induction on S:

$$\text{If } S' \prec S, \text{ then } [\![S[Q]]\!] = [\![S'[R]]\!] = [\![S'[P]]\!] \text{ or } [\![S[Q]]\!] = \bot \text{ or } [\![S'[R]]\!] = \bot$$

This has on the surface the *form* of the Sieber sequentiality logical relation $S^3_{\{1,2\}\{1,2,3\}}$, see [28]. (It is $(d_1, d_2, d_3) \in S^3_{\{1,2\}\{1,2,3\}}$ iff $d_1 = d_2 = d_3$ or $d_1 = \bot$ or $d_2 = \bot$.) This form on the surface is responsible for the fact that the induction hypothesis goes up through the case $S = \mathsf{if}\ S_1\ \mathsf{then}\ S_2\ \mathsf{else}\ S_3$. But for the proof of the case $S = gS_1S_2$ the specific semantics of R, P, Q and the fact $S' \prec S$ are needed.

So a sequentiality relation alone is not sufficient to prove this counter-example: a logical relation is a *semantic* means to prove the undefinability of a function. But here we must prove the undefinability of $S' \prec S$ for two functions $[\![A]\!] \leq [\![D]\!]$, where both functions separately are definable. At first sight this necessitates a *syntactic* proof. But we could ask the question: Are there *semantic* means to prove this? Are there necessary semantic conditions for the syntactic order that are stronger than the condition of stable order? See also the remark in the last section "Outlook".

5.2. **Chains of any length.**

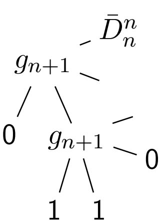
We have seen an example of a chain of two $\prec$-steps. Generally:

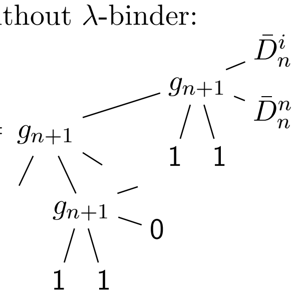
Definition 5.5. Let $a \leq b$ be finite elements in an f-model.
A *chain of length* $n \geq 1$ *between* a *and* b is a pair of sequences of terms $(C_i), (D_i)$ with $1 \leq i \leq n$ and $a = [\![C_1]\!]$, $b = [\![D_n]\!]$ and $C_i \prec D_i$, $D_i \cong C_{i+1}$.
If $a = b$, then we say there is a chain of length 0 between a and b.
A chain is *of least length* n if there is no shorter chain.

By the game term theorem, if there is a chain of PCF-terms, then there is an equivalent chain of game terms.

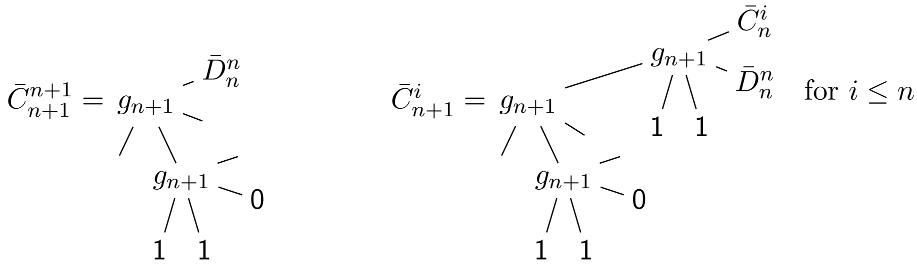
Now we construct examples of chains of least length $n+1$ for any finite $n \geq 0$, by a sequential composition of n copies of our first example, each copy for a different argument g_i. For every $n \geq 0$ let σ_n be the type $(\iota \to \iota \to \iota) \to \ldots \to (\iota \to \iota \to \iota) \to \iota$ with n parameters. For every n we define two sequences of game terms $C^i_n, D^i_n \colon \sigma_n$ with $0 \leq i \leq n$.
First we define by induction on n the versions $\bar{C}^i_n, \bar{D}^i_n$ without λ-binder:

$\bar{D}^0_0 = 0$

$\bar{D}^{n+1}_{n+1} =$ 

$\bar{D}^i_{n+1} =$  for $i \leq n$

$\bar{C}^0_0 = \bot$



We define $C^i_n = \lambda g_n \dots g_1.\bar{C}^i_n$ and $D^i_n = \lambda g_n \dots g_1.\bar{D}^i_n$.
For all $n \geq 0$, $0 \leq i \leq n$: $C^i_n \prec D^i_n$. The proof is an easy induction on n.
For all $n \geq 1$, $i < n$: $D^i_n \cong C^{i+1}_n$. Proof by induction on n:
For $n = 1$, $i = 0$ we have that D^0_1 is the term B, and C^1_1 the term C of our former example, both only with g replaced by g_1.
For $n := n + 1$:
For $i = n$ we have $D^n_{n+1} \cong C^{n+1}_{n+1}$ by the same argument as in our former example for $B \cong C$.
For $i < n$ we get $D^i_{n+1} \cong C^{i+1}_{n+1}$ by the induction hypothesis.

All together for any $n \geq 0$ we get a chain of length $n + 1$ between $[\![C^0_n]\!]$ and $[\![D^n_n]\!]$:

$$C^0_n \prec D^0_n \cong C^1_n \prec D^1_n \dots D^{n-1}_n \cong C^n_n \prec D^n_n.$$

We want to prove that this chain has the least length.

First the intuition of the example: We use the terms R, P, Q of the proof of proposition 5.3 and name their traces:

$$r = \mathcal{T}[\![R]\!] = \{11 \mapsto 1, \bot 0 \mapsto 0\}, \quad p = \mathcal{T}[\![P]\!] = \{\bot\bot \mapsto 0\}, \quad q = \mathcal{T}[\![Q]\!] = \{0\bot \mapsto 0\}$$

The trace of D^n_n contains all tokens $p \dots pq \dots q \mapsto 0$, with j arguments p, $0 \leq j \leq n$. These tokens are in the upper branch of D^n_n. We work down from D^n_n eliminating all these tokens in $n + 1$ steps.

In the j-th step ($0 \leq j \leq n$) the token $p \dots pq \dots q \mapsto 0$, with j arguments p, is eliminated in D^{n-j}_n. (In D^{n-j}_n all the tokens of this form with less arguments p have already been eliminated.) If $j < n$ we proceed as follows: Following the upper branches in D^{n-j}_n we come to an occurrence of the variable g_{n-j}. It is the root of a subterm $\bar{D}^{n-j}_{n-j}$, its upper arm is $\bar{D}^{n-j-1}_{n-j-1}$. The elimination is by setting the first argument of this g_{n-j} to $\bot$, getting C^{n-j}_n. Only then it is possible to lift the lower $g_{n-j}11$ to the top level, getting D^{n-j-1}_n. There the new $g_{n-j}11$ at the top level gets two arms which are copies of $\bar{D}^{n-j-1}_{n-j-1}$. The lower arm (of these two) stays the same in the following transformations (it contains the token $p \dots prq \dots q \mapsto 0$ with j arguments p). The upper arm undergoes further eliminations of tokens $p \dots pq \dots q \mapsto 0$. These further eliminations are only possible after the separation of the two arms.

Finally in the n-th step the 0 which stands at the end of the upper branches of D^0_n is set to $\bot$ getting C^0_n, eliminating the token $p \dots p \mapsto 0$.

Proposition 5.6. *Let $n \geq 0$ and C^i_n, D^i_n be the terms defined above. Then the chain*

$$C^0_n \prec D^0_n \cong C^1_n \prec D^1_n \dots D^{n-1}_n \cong C^n_n \prec D^n_n$$

between $[\![C^0_n]\!]$ and $[\![D^n_n]\!]$ has the least length $n + 1$.

Proof. We assume $n \geq 1$ and suppose any chain between C_n^0 and D_n^n and look at an intermediate $\prec$-step of this chain, i.e. we have the situation

$$C_n^0 \leq M \prec N \leq D_n^n.$$

We assume that some token of the form $p \dots pq \dots q \mapsto \mathsf{0}$ is eliminated in this step. Let t be such token with the minimal number j of arguments p, and assume $j < n$.
Then we have

$$NP \dots PQ \dots Q \to^* \mathsf{0}, \text{ and } MP \dots PRQ \dots Q \to^* \mathsf{0},$$

because $C_n^0 \leq M$ (both with j arguments P).
We can abstract the $(j+1)$st argument in these terms and build the terms

$$N' = \lambda g.NP \dots PgQ \dots Q \text{ and } M' = \lambda g.MP \dots PgQ \dots Q.$$

It is $M' \prec N'$. We can transform M', N' to game terms and apply the argument in the proof of proposition 5.3 to deduce: $M'P \to^* \mathsf{0}$.
So $MP \dots PPQ \dots Q \to^* \mathsf{0}$ (with $j+1$ arguments P).
As $Q \sqsubseteq_{op} P$, we also have $MP \dots PQ \dots Q \to^* \mathsf{0}$ for all $k \geq j+1$ arguments P.
All these arguments of M are minimal w.r.t. the stable order, because they are also minimal for D_n^n and it is $M \leq D_n^n$.
Therefore every token $p \dots pq \dots q \mapsto \mathsf{0}$ with $k \geq j+1$ arguments p is in M.
This shows that from the tokens of the form $p \dots pq \dots q \mapsto \mathsf{0}$ only the token t is eliminated in the step $M \prec N$. (For $j = n$ this is trivially the case.) As there are $n+1$ of these tokens to be eliminated, the chain must have at least $n+1$ steps. □

Our example of a chain of least length $n+1$ has n functional parameters g_i of arity 2 and is of grade 1. We could transform it into an "equivalent" example with only one functional parameter g of arity 3 and terms of grade n, by coding $g_i MN$ as $giMN$.

Our results suggest an improvement of Berry's second conjecture:

Conjecture 5.7 (Chain Conjecture)**.** If $a \leq b$ are finite elements in an f-model, then there is a chain between a and b.
We will refute also this conjecture in section 7.

6. The stable order is not bounded complete: no bidomain

Gérard Berry showed that the fully abstract order-extensional cpo-model of PCF (our greatest f-model) together with the stable order forms a bicpo, and conjectured that it is also a bidomain (Berry's first conjecture). Here we repeat the definitions of both structures. We prove the conjecture for first-order types. Then we refute the general conjecture. Our first example is the stable lub of two finite elements for which the distributive law is not valid. Our second example consists of two finite elements with stable upper bound but without stable lub. Both examples are in PCF of second-order type of grade 2.

Definition 6.1 (Berry: 4.7.2 in [4])**.** A *bicpo* is a structure $(D, \sqsubseteq, \leq, \perp)$ such that:

(1) The structure $(D, \sqsubseteq, \perp)$ is a cpo with least element $\perp$ and with a continuous glb-function $\sqcap$.
(2) The structure $(D, \leq, \perp)$ is a cpo with least element $\perp$ such that $a \leq b \Longrightarrow a \sqsubseteq b$ and for all $\leq$-directed sets S the two lubs are equal: $\bigvee S = \bigsqcup S$.
(3) The function $\sqcap$ is $\leq$-monotonic. (With (1) and (2) it follows that it is $\leq$-continuous.)

(4) For all $\leq$-directed sets S and S': If for all $a \in S$, $a' \in S'$ there are $b \in S$, $b' \in S'$ with $a \sqsubseteq b$, $a' \sqsubseteq b'$, $b \leq b'$, then $\bigsqcup S \leq \bigsqcup S'$.

In a bicpo: For all $a \uparrow_{\leq} b$, $a \sqcap b$ is also the glb w.r.t. $\leq$.

Theorem 6.2 (Berry: 4.8.10 in [4])**.** *The domains $(D^\sigma, \sqsubseteq, \leq, \perp)$ of the fully abstract order-extensional cpo-model of PCF are bicpos.* □

Definition 6.3 (Berry: 4.4.10 in [4])**.** A cpo $(D, \leq, \perp)$ is *distributive* if

(1) it is bounded complete
(This means that for $a \uparrow_{\leq} b$ there is a lub $a \vee b$. And this entails with completeness that there is also a glb $a \wedge b$ for all a, b, even for $\leq$-incompatible ones.)
and

(2) for all $a, b, c \in D$ with $b \uparrow_{\leq} c$: $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$.

Definition 6.4 (Berry: 4.7.9 in [4])**.** A bicpo $(D, \sqsubseteq, \leq, \perp)$ is *distributive* if $(D, \leq, \perp)$ is distributive and for all $a \uparrow_{\leq} b$: $a \vee b$ is also the lub w.r.t. $\sqsubseteq$.

(Please note that in a distributive bicpo only for $a \uparrow_{\leq} b$ it must be $a \wedge b = a \sqcap b$.)

Definition 6.5 (Berry: 4.7.12 in [4])**.** A distributive bicpo $(D, \sqsubseteq, \leq, \perp)$ is a *bidomain* if there is a $\leq$-growing sequence $(\psi_i)_{i \geq 1}$ of finite projections w.r.t. $\leq$ and with lub $\bigvee \psi_i = \mathrm{id}$. (This means: $\psi_i \colon D \to D$ is continuous w.r.t. $\sqsubseteq$ and $\leq$, $\psi_i \leq \mathrm{id}$, $\psi_i \circ \psi_i = \psi_i$, $\psi_i \leq \psi_{i+1}$, $\psi_i(D)$ finite, $\bigvee \psi_i = \mathrm{id}$.)

In this definition the sequence (ψ_i) is also a $\sqsubseteq$-growing sequence of finite projections w.r.t. $\sqsubseteq$ and with lub id. Together with the the glb-function $\sqcap$ it follows that $(D, \sqsubseteq, \perp)$ is a Scott domain, a bounded complete ω-algebraic cpo.

As we have explained in proposition 3.8 and 3.15, the conditions for (ψ_i) in the definition of bidomain are fulfilled for the fully abstract order-extensional cpo-model (and furthermore for all f-models) by the projections ψ_i^σ. In fact the $(D^\sigma, \leq)$ are stable ω-bifinite domains for the cpo-model, in the sense of definition 12.4.3 of [2].

To be precise, the condition of distributivity of the stable order was not conjectured by Berry in his thesis; there he remained agnostic. But in the state-of-the-art paper [5] we can read: "Unfortunately we are not able to show that the domains of the fully abstract model are bidomains, although we definitely believe it; the problem is to show that the $\leq_{cm}$-lubs are taken pointwise."

First we clarify the situation for first-order types:

Theorem 6.6. *Let σ be a first-order type and $(D^\sigma, \sqsubseteq, \leq, \perp)$ be the corresponding domain of any f-model.*
The finite elements of D^σ fulfill distributivity w.r.t. $\leq$ in D^σ in the following sense:
For $a, b \in \mathcal{F}^\sigma$ the glb in D^σ exists and is given by $\mathcal{T}(a \wedge b) = \mathcal{T}(a) \cap \mathcal{T}(b)$.
For $a, b \in \mathcal{F}^\sigma$ with $a \uparrow_{\leq} b$ the lub in D^σ exists and is given by $\mathcal{T}(a \vee b) = \mathcal{T}(a) \cup \mathcal{T}(b)$. It is taken pointwise and it is also the lub w.r.t. $\sqsubseteq$.
Then the distributive law is fulfilled by set theory on traces.

If D^σ contains a denotation for every infinite game term of type σ (this is the case for the game model and every greater f-model), then D^σ is the domain of the greatest f-model. In this case all elements $a, b \in D^\sigma$ fulfill distributivity in the sense above. Therefore D^σ is a bidomain in this case.

Proof. Let $a, b \in \mathcal{F}^\sigma$.
We can apply theorem 5.1 and get a game term A with $a = [\![A]\!]$, and a game term $C \prec A$ with $\mathcal{T}[\![C]\!] = \mathcal{T}(a) \cap \mathcal{T}(b)$. Define $a \wedge b = [\![C]\!]$; it is finite and therefore in D^σ.

Now let $a \uparrow_{\leq} b$, i.e. there is some d with $a \leq d$ and $b \leq d$. By theorem 5.1 there are an infinite game term D with $d = [\![D]\!]$, and finite game terms A, B with $a = [\![A]\!]$, $b = [\![B]\!]$, $A \prec D$, $B \prec D$. Take the syntactical lub E of A and B. It is $\mathcal{T}[\![E]\!] = \mathcal{T}[\![A]\!] \cup \mathcal{T}[\![B]\!]$, because in first-order game terms branches correspond to tokens. Define $a \vee b = [\![E]\!]$; it is finite and therefore in D^σ. This lub is pointwise on the uncurried argument and therefore also the lub w.r.t. $\sqsubseteq$.

If D^σ contains a denotation for every infinite game term of type σ, then by theorem 5.1 D^σ is exactly the domain of the greatest f-model. The construction of $a \wedge b$ and $a \vee b$ for any $a, b \in D^\sigma$ is as above, only with infinite game terms. $\square$

Conjecture 6.7. For all types of the form $\sigma = \sigma_1 \to \ldots \to \sigma_n \to \iota$, with $\sigma_i = \iota$ or $\sigma_i = \iota \to \iota$, Berry's first conjecture is valid, i.e. D^σ is a bidomain in the greatest f-model.

The proof of this conjecture is in preparation. It relies on the conjecture 5.2.

Now we prove some properties of stable upper bounds (sub) in f-models. (These are properties that are also valid in stable bifinite domains, see lemma 12.4.7 in [2].)

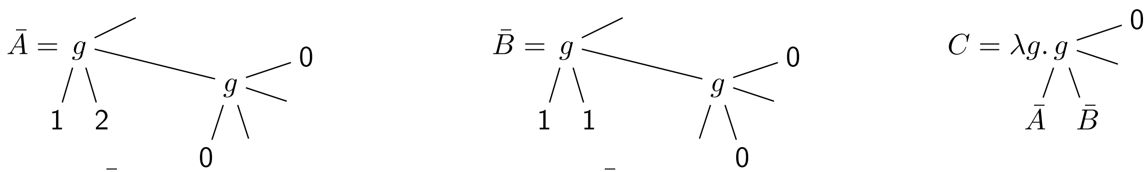
Theorem 6.8. *Let D^σ be a domain of an f-model, $\sigma = \sigma_1 \to \ldots \to \sigma_n \to \iota$, $n \geq 0$. Let X be a finite set of finite elements of D^σ that has a stable upper bound (sub) in D^σ. Let m be the maximal grade of the elements of X. For every sub x of X there is a unique minimal (w.r.t. $\leq$) sub y of X with $y \leq x$. Every minimal sub of X is finite of grade m; they are pairwise $\leq$-incompatible. The extensional lub $\bigsqcup X$ is one of those.*

Proof. Let x be a sub of X. Then the projection $\psi^\sigma_m x$ is also a sub of X. Let Z be the set of all subs z of X with $z \leq \psi^\sigma_m x$; it is a non-empty finite set of finite elements. Then $y = \bigsqcap Z$ is the desired unique minimal sub of X with $y \leq x$.

Let a, b be two minimal subs of X that are $\leq$-compatible. Then $a \sqcap b$ is also a sub of X, therefore $a = b$.

Let $g = \bigsqcup X$ and h some sub of X. We have to show that $f \leq g$ for every $f \in X$.
This is clear for $n = 0$, in the type ι.
Now let $n > 0$ and $\vec{x}$, $\vec{y}$ be two vectors of arguments of type $\sigma_1 \times \ldots \times \sigma_n$ with $\vec{x} \leq \vec{y}$.
We have to show that $f\vec{x} = f\vec{y} \sqcap g\vec{x}$.
It is $f\vec{x} = f\vec{y} \sqcap h\vec{x} \sqsupseteq f\vec{y} \sqcap g\vec{x}$. And $f\vec{x} \sqsubseteq f\vec{y} \sqcap g\vec{x}$ is clear.
This shows that g is a sub of X; of course it is also minimal w.r.t. $\leq$. $\square$

6.1. **A stable lub without distributivity.** Our first counter-example to Berry's first conjecture is of type $(\iota \to \iota \to \iota) \to \iota$ and of grade 2. We consider the following game terms A, B, C, where we use a case_1 for a case_2 with the third arm $\perp$:



$A = \lambda g.\bar{A}$ $B = \lambda g.\bar{B}$

Here are the traces of these terms:

$$\left.\begin{array}{l} A\left\{\begin{array}{ll} \{\mathsf{0}\,\bot\mapsto\mathsf{0}, & \mathsf{1}\,\mathsf{2}\mapsto\mathsf{1}\}\mapsto\mathsf{0} \\ \{\mathsf{0}\,\bot\mapsto\mathsf{0}, & \mathsf{1}\,\bot\mapsto\mathsf{1}\}\mapsto\mathsf{0} \end{array}\right. \\ B\left\{\begin{array}{ll} \{\bot\,\mathsf{0}\mapsto\mathsf{0}, & \mathsf{1}\,\mathsf{1}\mapsto\mathsf{1}\}\mapsto\mathsf{0} \\ \{\bot\,\mathsf{0}\mapsto\mathsf{0}, & \bot\,\mathsf{1}\mapsto\mathsf{1}\}\mapsto\mathsf{0} \end{array}\right. \\ \quad\;\; \{\bot\bot\mapsto\mathsf{0} \qquad\qquad\quad \}\mapsto\mathsf{0} \end{array}\right\} C$$

It is $A \leq C$ and $B \leq C$. We will show that C is the stable lub of A and B.

The intuition of the example: A and B do not contain the token $\{\bot\bot\mapsto\mathsf{0}\}\mapsto\mathsf{0}$, because their two occurrencies of g are forced to evaluate their first resp. second argument, to get different results for different arguments. (This is the same trick that was used in the preceding section.) C adds to the tokens of A and B just the token $\{\bot\bot\mapsto\mathsf{0}\}\mapsto\mathsf{0}$, to separate $\bar{A}$ and $\bar{B}$. (Note that a g for which Cg converges cannot demand both its arguments $\mathsf{00}$.) Therefore this lub does not fulfill distributivity. In C it is not possible to lift a differing term gMN to the top level that would eliminate that token, because the five occurrences of g in C cannot be "unified" to a common term that would always converge.

Proposition 6.9. *Let A, B, C be the game terms above. $[\![C]\!]$ is the stable lub of $a = [\![A]\!]$ and $b = [\![B]\!]$. Let d be the finite element with the trace $\{\{\bot\bot\mapsto\mathsf{0}\}\mapsto\mathsf{0}\}$. Then $d \wedge (a \vee b) \neq (d \wedge a) \vee (d \wedge b)$. This refutes Berry's first conjecture.*

Proof. By the game term theorem 4.12 and the preceding theorem 6.8, every minimal sub of A and B can be represented by a game term of grade 2. Such a game term is of the form $\lambda g.S$, where $S\colon \iota$ is a game term possibly with the only free variable g. We abbreviate $S[g := M]$ as $S[M]$.

We use the following terms as arguments:

$$\begin{aligned} Q &= \lambda xy.\,\mathsf{case}_1\,x\mathsf{0}(\mathsf{case}_2\,y\bot\bot\mathsf{1}) & \mathcal{T}[\![Q]\!] &= \{\mathsf{0}\bot\mapsto\mathsf{0},\ \ \mathsf{12}\mapsto\mathsf{1}\} \\ R &= \lambda xy.\,\mathsf{case}_1\,y\mathsf{0}(\mathsf{case}_1\,x\bot\mathsf{1}) & \mathcal{T}[\![R]\!] &= \{\bot\mathsf{0}\mapsto\mathsf{0},\ \ \mathsf{11}\mapsto\mathsf{1}\} \\ P &= \lambda xy.\mathsf{0} & \mathcal{T}[\![P]\!] &= \{\bot\bot\mapsto\mathsf{0}\} \end{aligned}$$

Q and R are compatible in the sense that they produce compatible results for the same argument. We will prove that for any term S of the form above:

$$\text{If } S[Q] \to^* \mathsf{0} \text{ and } S[R] \to^* \mathsf{0}, \text{ then } S[P] \to^* \mathsf{0}.$$

The proof is by induction on the term S: The cases $S = \bot, \mathsf{0}, \mathsf{1}, \mathsf{2}$ are clear.
Let $S = \mathsf{case}_2(gS_1S_2)S_3S_4S_5$.
For $S[Q] \to^* \mathsf{0}$ it must be $S_1[Q] \to^* \mathsf{0}$ or $S_2[Q] \to^* \mathsf{2}$.

(1) case $S_1[Q] \to^* \mathsf{0}$:
For $S[R] \to^* \mathsf{0}$ it must be $S_2[R] \to^* \mathsf{0}$ or $S_1[R] \to^* \mathsf{1}$.
 (1.1) case $S_2[R] \to^* \mathsf{0}$:
 We have $S[Q] \to^* S_3[Q] \to^* \mathsf{0}$ and $S[R] \to^* S_3[R] \to^* \mathsf{0}$.
 By the induction hypothesis for S_3 we get $S_3[P] \to^* \mathsf{0}$, therefore $S[P] \to^* \mathsf{0}$.
 (1.2) case $S_1[R] \to^* \mathsf{1}$:
 This is not possible, as Q and R are compatible in the sense above.

(2) case $S_2[Q] \to^* \mathsf{2}$:
For $S[R] \to^* \mathsf{0}$ it must be $S_2[R] \to^* \mathsf{0}$ or $S_2[R] \to^* \mathsf{1}$.
Both cases are not possible, as Q and R are compatible in the sense above.

So we have shown that for every $\sqsubseteq$-upper bound D of grade 2 of A and B it must be $DP \to^* \mathsf{0}$. For a $\leq$-upper bound it cannot be $D\bot \to^* \mathsf{0}$. Therefore P is a $\leq$-minimal argument to fulfill $DP \to^* \mathsf{0}$. This means: Any minimal stable upper bound of A and B must contain the token $\{\bot\bot\mapsto\mathsf{0}\}\mapsto\mathsf{0}$. So C is the stable lub of A and B. (It is also the $\sqsubseteq$-lub.) $\square$

Remark 6.10 (alternative proof with Sieber sequentiality relation)**.** Because we work in the proof above on game terms, the induction hypothesis is simpler and the proof shorter than a proof by induction on general terms. A short purely semantic proof for general terms is possible with a Sieber sequentiality logical relation [28].

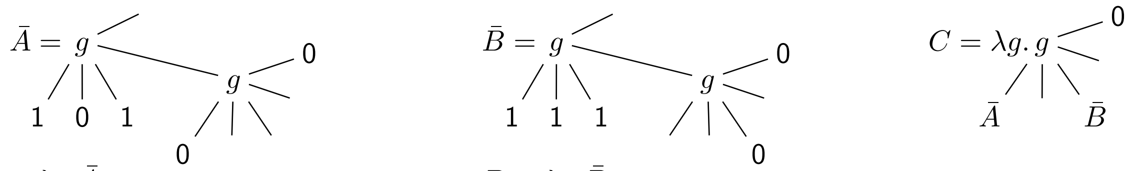
We can show that there is no definable function that fulfills the value table $[\![Q]\!]\mapsto\mathsf{0}$, $[\![R]\!]\mapsto\mathsf{0}$, $[\![P]\!]\mapsto n$ for $n \neq \mathsf{0}$. We use the sequentiality relation $rel = S^3_{\{1,2\}\{1,2,3\}}$.
For $d_1, d_2, d_3\colon \iota$ it is $(d_1, d_2, d_3) \in rel$ iff $d_1 = \bot$ or $d_2 = \bot$ or $d_1 = d_2 = d_3$.
First, the output column $(\mathsf{0}, \mathsf{0}, n)$ of the value table is not in this relation.
Then we have to show that $([\![Q]\!], [\![R]\!], [\![P]\!]) \in rel$ (on the type $\iota \to \iota \to \iota$).
Suppose we have

$$\begin{aligned}
[\![Q]\!]a_1b_1 &= c_1,\\
[\![R]\!]a_2b_2 &= c_2,\\
[\![P]\!]a_3b_3 &= c_3
\end{aligned}$$

and suppose $(c_1, c_2, c_3) \notin rel$. We have to show that $(a_1, a_2, a_3) \notin rel$ or $(b_1, b_2, b_3) \notin rel$.
It must be $c_3 = 0$.
It cannot be $c_1 = \bot$, so it must be $c_1 = 0$ or $c_1 = 1$:
If $c_1 = 0$, then it cannot be $c_2 = 0$, so it must be $c_2 = 1$, then $a_1 = 0$, $a_2 = 1$, therefore $(a_1, a_2, a_3) \notin rel$, end of proof for $c_1 = 0$.
If $c_1 = 1$, then it is $c_2 = 0$ or $c_2 = 1$:
If $c_2 = 0$, then $b_1 = 2$, $b_2 = 0$, therefore $(b_1, b_2, b_3) \notin rel$.
If $c_2 = 1$, then $b_1 = 2$, $b_2 = 1$, therefore $(b_1, b_2, b_3) \notin rel$.

It is no surprise that we have to perform a case analysis of similar complexity as in the proof above. But it is interesting that the whole proof of this remark can be done mechanically by the computer program written by Allen Stoughton [30]. For a general system of ground constants, this program takes a value table of a second-order function and returns either a term defining such a function or a logical relation proving its undefinability.

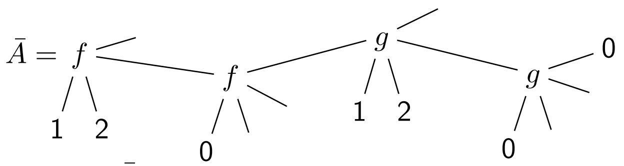
Our counter-example is of grade 2 with g of arity 2. There is an "equivalent" example of grade 1 with g of arity 3:



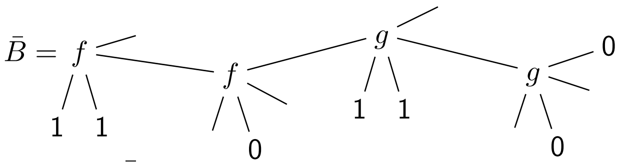
$A = \lambda g.\bar{A}$ $\qquad\qquad$ $B = \lambda g.\bar{B}$

Conjecture 6.11. In $\mathcal{F}_1^{(\iota\to\iota\to\iota)\to\iota}$, the finite elements of grade 1 of the type $(\iota\to\iota\to\iota)\to\iota$, Berry's first conjecture is valid; this subdomain is a bidomain. (This is a finite combinatorial problem and could be solved by a computer program.)

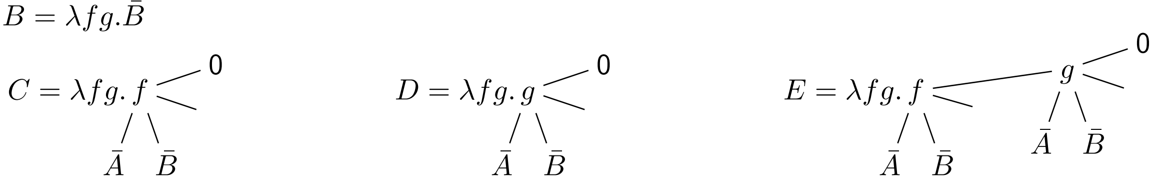
6.2. **Two elements without stable lub.** Now to our counter-example to bounded completeness of the stable order. It is of type $(\iota \to \iota \to \iota) \to (\iota \to \iota \to \iota) \to \iota$ and of grade 2. It employs the trick of our last example twice to two functional parameters. Consider the following game terms A, B, C, D, E, where we use a case_1 for a case_2 with the third arm $\perp$.



$A = \lambda fg.\bar{A}$



$B = \lambda fg.\bar{B}$



The traces of the terms are:

$$
\begin{aligned}
\mathcal{T}[\![A]\!] &= \{\,0\,\perp\mapsto 0\,,\ 1\ \begin{smallmatrix}2\\ \perp\end{smallmatrix}\mapsto 1\,\}\mapsto\{\,0\,\perp\mapsto 0\,,\ 1\ \begin{smallmatrix}2\\ \perp\end{smallmatrix}\mapsto 1\,\}\mapsto 0\\
\mathcal{T}[\![B]\!] &= \{\perp 0\mapsto 0\,,\ \begin{smallmatrix}1\\ \perp\end{smallmatrix}\ 1\mapsto 1\,\}\mapsto\{\perp 0\mapsto 0\,,\ \begin{smallmatrix}1\\ \perp\end{smallmatrix}\ 1\mapsto 1\,\}\mapsto 0\\
\mathcal{T}[\![C]\!] &= \mathcal{T}[\![A]\!] \cup \mathcal{T}[\![B]\!] \cup \{\{\perp\perp\mapsto 0\,\}\mapsto\perp \mapsto 0\,\}\\
\mathcal{T}[\![D]\!] &= \mathcal{T}[\![A]\!] \cup \mathcal{T}[\![B]\!] \cup \{\perp \mapsto\{\perp\perp\mapsto 0\,\}\mapsto 0\,\}\\
\mathcal{T}[\![E]\!] &= \mathcal{T}[\![A]\!] \cup \mathcal{T}[\![B]\!] \cup \{\{\perp\perp\mapsto 0\,\}\mapsto\{\perp\perp\mapsto 0\,\}\mapsto 0\,\}
\end{aligned}
$$

The token of $\mathcal{T}[\![A]\!]$ entails three more tokens: (1) with the first indicated 2 replaced by $\perp$, (2) with the second indicated 2 replaced by $\perp$, (3) with both replaced by $\perp$. Likewise for the token of $\mathcal{T}[\![B]\!]$. (These entailments are due to securedness, see the definition 2 of [7].)

C, D, E are three stable upper bounds of A and B; we will show that they are just the minimal stable upper bounds. E is the $\sqsubseteq$-lub of A and B.

The intuition of the example: In an upper bound of A and B, both have to be separated by some function call at the top level; because A and B cannot be "unified". There are three ways to choose the separator: f or g or (both f and g), realized by C, D, E resp.

Proposition 6.12. *Let A, B, C, D, E be the game terms above. $[\![C]\!], [\![D]\!], [\![E]\!]$ are the minimal stable upper bounds of $[\![A]\!]$ and $[\![B]\!]$. So $[\![A]\!]$ and $[\![B]\!]$ have no stable lub. (This again refutes Berry's first conjecture.)*

Proof. By theorem 6.8, every minimal sub of A and B is of grade 2. By the game term theorem, we restrict to game terms of grade 2. These game terms must have the form $\lambda fg.S$. We use the terms Q, R, P of the proof of proposition 6.9. Our claim is: For every term S of the form above,

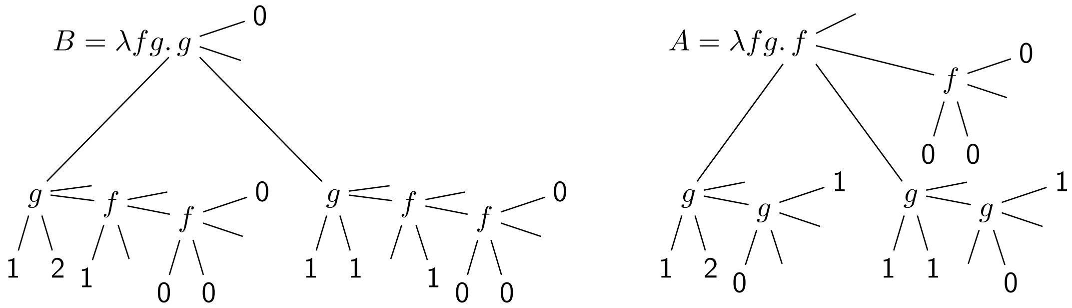
$$\text{if } S[f := Q, g := Q] \to^* \mathsf{0} \text{ and } S[f := R, g := R] \to^* \mathsf{0}, \text{ then } S[f := P, g := P] \to^* \mathsf{0}.$$

The proof of the claim is by induction on the term S and follows exactly the proof of proposition 6.9. There is only one additional case $S = \mathsf{case}_2(fS_1S_2)S_3S_4S_5$ of the same scheme.

So we have shown that for every $\sqsubseteq$-upper bound F of grade 2 of A and B it must be $FPP \to^* \mathsf{0}$. For a $\leq$-upper bound it cannot be $F\bot\bot \to^* \mathsf{0}$. Hence the minimal arguments to fulfill $FPP \to^* \mathsf{0}$ must be (P, P), $(P, \bot)$ or $(\bot, P)$. This is fulfilled by E, C, D respectively. $\square$

7. Refutation and improvement of the chain conjecture

The chain conjecture 5.7 said that for finite elements $a \leq b$ there is a chain between a and b, see the definition 5.5 of chain. We give here a counter-example in the type $(\iota \to \iota \to \iota) \to (\iota \to \iota \to \iota) \to \iota$ of grade 2. Consider the following game terms A, B:



Here are the traces of these terms:

$$\left.\begin{array}{l} A\left\{\begin{array}{llll} \{\mathsf{0\,0}\mapsto\mathsf{0}, & \mathsf{1}\bot\mapsto\mathsf{1}\}\mapsto\{\mathsf{0}\bot\mapsto\mathsf{0}, & \mathsf{1\,2}\mapsto\mathsf{1}\}\mapsto\mathsf{0} \\ \quad\bot & & \quad\bot & \\ \{\mathsf{0\,0}\mapsto\mathsf{0}, & \bot\mathsf{1}\mapsto\mathsf{1}\}\mapsto\{\bot\mathsf{0}\mapsto\mathsf{0}, & \mathsf{1\,1}\mapsto\mathsf{1}\}\mapsto\mathsf{0} \\ \bot & & \bot & \end{array}\right. \\ \quad\;\bot \qquad\qquad\qquad\quad \mapsto\{\bot\bot\mapsto\mathsf{0} \qquad\qquad\quad \}\mapsto\mathsf{0} \end{array}\right\} B$$

The first token entails three more tokens: (1) with the indicated $\mathsf{0}$ replaced by $\bot$, (2) with the indicated 2 replaced by $\bot$, (3) with both replaced by $\bot$. Likewise for the second token.

It is $A \leq B$. B contains just one more token t than A. Assume that there is a chain between $[\![A]\!]$ and $[\![B]\!]$. Then t is eliminated in a definite step $A' \prec B'$ of the chain, with $A \cong A'$ and $B \cong B'$. We will show that such $A' \prec B'$ do not exist.

The intuition of the example: It is derived from the example of subsection 6.1. A and B are like the term C of that example. For B: In the left leg of the upper g the subterm $g\mathsf{0}\bot$ (of C) is replaced by the subterm demanding the first argument of f. In the right leg the subterm $g\bot\mathsf{0}$ (of C) is replaced by the subterm demanding the second argument of f. This ensures that not both legs (of the upper g) can be evaluated. There is again no term with g that could be lifted to the top level and that would eliminate the token $\bot\mapsto\{\bot\bot\mapsto\mathsf{0}\}\mapsto\mathsf{0}$. Therefore there is no $\prec$-step leading from A to B. But the subterms with f can be lifted to the top replacing the upper g of B (as "separator" of $g\mathsf{12}$ and $g\mathsf{11}$), so we get A with that token eliminated. Here the subterms $g0\perp$ and $g\perp 0$ of the former example C appear again; they must appear to ensure that A gets the first eight tokens of B and ensure that not both legs of the upper f can be evaluated.

Proposition 7.1. *Let A, B be the game terms of grade 2 above. There are no game terms A', B' of grade 2 with $A' \prec B'$ and $A' \cong A$, $B' \cong B$. Then by the game term theorem there are no PCF-terms A', B' with this property. This refutes the chain conjecture.*

Proof. As game terms of grade 2, A' and B' should be of the form $\lambda fg.S$, where $S\colon \iota$ is a game term possibly with the only free variables f, g. We abbreviate $S[f := M, g := N]$ as $S[M, N]$.
We use the terms of the proof of proposition 6.9 as arguments for g:

$$
\begin{aligned}
Q &= \lambda xy.\,\mathsf{case}_1\, x0(\mathsf{case}_2\, y\perp\perp 1) & \mathcal{T}[\![Q]\!] &= \{0\perp\mapsto 0,\ \ 12\mapsto 1\}\\
R &= \lambda xy.\,\mathsf{case}_1\, y0(\mathsf{case}_1\, x\perp 1) & \mathcal{T}[\![R]\!] &= \{\perp 0\mapsto 0,\ \ 11\mapsto 1\}\\
P &= \lambda xy.0 & \mathcal{T}[\![P]\!] &= \{\perp\perp\mapsto 0\}
\end{aligned}
$$

We use the following terms as arguments for f:

$$
\begin{aligned}
Q' &= \lambda xy.\,\mathsf{case}_1\, x(\mathsf{case}_1\, y0\perp)1 & \mathcal{T}[\![Q']\!] &= \{00\mapsto 0,\ \ 1\perp\mapsto 1\}\\
R' &= \lambda xy.\,\mathsf{case}_1\, y(\mathsf{case}_1\, x0\perp)1 & \mathcal{T}[\![R']\!] &= \{00\mapsto 0,\ \ \perp 1\mapsto 1\}
\end{aligned}
$$

The pairs (Q', Q) and (R', R) are compatible in the sense that their replacement into the same integer term leads to compatible results.

We will prove that for any terms S, S' of the form above:

$$\text{If } S' \prec S \text{ and } S[\perp, P] \to^* 0, \quad S'[Q', Q] \to^* 0, \quad S'[R', R] \to^* 0, \text{ then } S'[\perp, P] \to^* 0.$$

The proposition follows immediately from this claim.

The proof is by induction on the term S: The cases $S = \perp, 0, 1, 2$ are clear.
Let $S = \mathsf{case}_2(gS_1S_2)S_3S_4S_5$ and $S' \prec S$ with $S' = \mathsf{case}_2(gS_1'S_2')S_3'S_4'S_5'$. (The case $S' = \perp$ is clear.)
Assume the three conditions of the claim.
Let $(gS_1'S_2')[Q', Q] \to^* q$ and $(gS_1'S_2')[R', R] \to^* r$, both terms must converge to integer constants.
From the compatibility of (Q', Q) and (R', R) follows the compatibility of q and r, so either $q = r = 0$ or $q = r = 1$.
As (Q', Q) and (R', R) are compatible, it cannot be $S_2'[Q', Q] \to^* 2$ and $S_2'[R', R] \to^* 1$.
Therefore $q = r = 0$.
Then we get $S_3[\perp, P] \to^* 0$, $S_3'[Q', Q] \to^* 0$, $S_3'[R', R] \to^* 0$.
By the induction hypothesis for S_3 we conclude $S_3'[\perp, P] \to^* 0$, hence $S'[\perp, P] \to^* 0$. This fulfills the claim.
Now let $S = \mathsf{case}_2(fS_1S_2)S_3S_4S_5$.
Then $S[\perp, P] \cong \perp$, so the claim is fulfilled. □

The refutation of the chain conjecture shows that already for second-order types the correspondence of stable and syntactic order is destroyed; there seems to be no simple syntactic characterization of the stable order. But certainly the two orders are related, but in which sense? A weaker conjecture that is now open is the following:

Conjecture 7.2 (Maximality Conjecture)**.** Every PCF-term without Y that is syntactically maximal (i.e. contains no $\perp$) is also stably maximal.

The existence of chains of any length suggests a kind of "metric" on finite elements $a \leq b$: If there is a chain between a and b of least length n, then the distance of a and b is n. If there is no chain, then the distance is ∞. But it might be doubted if this is meaningful, or if a transition $A \leq B$ like the example above (without chain) should also be counted as some kind of elementary step of finite distance.

The example $A \leq B$ above shows us that the syntactic order $\prec$ is not enough to give a syntactic description of the stable order; there are more "syntactic" relations needed. We can imagine that A is produced from B by "forcing" the upper g in B to be strict in one of its two arguments, so that the token $\perp \mapsto \{\perp\perp\mapsto 0\} \mapsto 0$ is eliminated.

We tentatively propose an improved chain conjecture with such a new syntactic relation of "strictification". For this we have to extend PCF with a new operator. The theory of this extension has still to be properly developed; so all propositions in the rest of this section have the status of conjectures.

In [23] Luca Paolini extends PCF with two new operators, one of them called $\mathsf{strict?}$ of type $(\iota \to \iota) \to \iota$. Suppose the operational semantics is given by an evaluation procedure eval. Then $\mathsf{strict?}$ obeys the rules:

$$\text{If } \operatorname{eval}(M\mathsf{0})\!\downarrow \text{ and } \operatorname{eval}(M\perp)\!\uparrow \text{ then } \operatorname{eval}(\mathsf{strict?}\,M) = \mathsf{0}$$
$$\text{If } \operatorname{eval}(M\mathsf{0})\!\downarrow \text{ and } \operatorname{eval}(M\perp)\!\downarrow \text{ then } \operatorname{eval}(\mathsf{strict?}\,M) = \mathsf{1}$$

Here $X\!\downarrow$ means that X evaluates to some integer constant, $X\!\uparrow$ is the negation. Paolini also gives an effective evaluation for $\mathsf{strict?}$.

We use instead a new constant $\mathsf{str}\colon (\iota \to \iota) \to \iota$ that is the "strict half" of $\mathsf{strict?}$, i.e. we have the only rule:

$$\text{If } \operatorname{eval}(M\mathsf{0}) = \mathsf{0} \text{ and } \operatorname{eval}(M\perp)\!\uparrow \text{ then } \operatorname{eval}(\mathsf{str}\,M) = \mathsf{0}$$

str can be expressed by a term with $\mathsf{strict?}$, but $\mathsf{strict?}$ cannot be expressed by str. Note that our str is finite. An effective evaluation could also be given for str. ($\mathsf{str}\,M$ tests if $M\mathsf{0}$ evaluates to $\mathsf{0}$ and in this process checks if M demands its argument $\mathsf{0}$.)

On the extended language (PCF+str) the operational equivalence $\cong$ is defined in the usual way by observation through program contexts. It is extensional, i.e. $M \cong N$ iff for all $M' \cong N'$ it is $MM' \cong NN'$. There is a fully abstract semantics $[\![\quad]\!]$ given by equivalence classes of terms; these equivalence classes are construed as functions. These functions are stable; we can define a trace semantics $\mathcal{T}[\![\quad]\!]$ in the usual way, with the stable order $\leq$ as the inclusion relation on traces. All denotations are monotonic w.r.t. the stable order $\leq$.

str has the trace semantics

$$\mathcal{T}[\![\mathsf{str}]\!] = \{\{\mathsf{0}\mapsto\mathsf{0}\}\mapsto\mathsf{0}\}$$

Note that the token $\{\mathsf{0}\mapsto\mathsf{0}\}\mapsto\mathsf{0}$ expresses the fact that the argument function $\{\mathsf{0}\mapsto\mathsf{0}\}$ is strict, its argument $\mathsf{0}$ is needed. Note that str is not monotonic w.r.t. the extensional order of PCF; it is $[\![\mathsf{str}]\!]\{\mathsf{0}\mapsto\mathsf{0}\} = \mathsf{0}$, but $[\![\mathsf{str}]\!]\{\perp\mapsto\mathsf{0}\} = \perp$. It is

$$\mathcal{T}[\![\mathsf{str}]\!] \subseteq \mathcal{T}[\![\lambda f.\,\mathsf{if}\ f\mathsf{0}\ \mathsf{then}\ \mathsf{0}\ \mathsf{else}\ \perp]\!] = \{\{\mathsf{0}\mapsto\mathsf{0}\}\mapsto\mathsf{0},\ \{\perp\mapsto\mathsf{0}\}\mapsto\mathsf{0}\}$$

All semantic elements preserve compatibility in the following sense. Let us define the relation $\uparrow_h$ of *hereditary compatibility* on denotations: for integers it is $m \uparrow_h n$ if $m = \perp$ or $n = \perp$ or $m = n$. For functions it is $f \uparrow_h g$ if for all $x \uparrow_h y$: $fx \uparrow_h gy$. All our functions f of (PCF+str) have the property that $f \uparrow_h f$. Paolini's operator $\mathsf{strict?}$ does not have it.

With str we can define functions $\mathrm{strictify}_n\colon \sigma_n \to \sigma_n$, where $\sigma_n = (\iota \to \ldots \to \iota \to \iota)$ with $n \geq 1$ arguments. E.g. $\mathrm{strictify}_2\colon (\iota \to \iota \to \iota) \to (\iota \to \iota \to \iota)$,

$\mathrm{strictify}_2 = \lambda gxy.\, \mathsf{if}(\mathsf{str}[\lambda z.\, \mathsf{if}\ g(\mathsf{if}\ z\ \mathsf{then}\ x\ \mathsf{else}\ \bot)(\mathsf{if}\ z\ \mathsf{then}\ y\ \mathsf{else}\ \bot)\ \mathsf{then}\ 0\ \mathsf{else}\ 0])\ \mathsf{then}\ gxy\ \mathsf{else}\ \bot$

$\mathrm{strictify}_2\, gxy$ tests if gxy converges and $g\bot\bot$ diverges, and outputs gxy in this case. So $\mathrm{strictify}_2\, gxy$ "forces" g to be strict in one of its two arguments. If it is not, then the output is $\bot$.

Let us replace in the example term B above the upper occurrence of g by $(\mathrm{strictify}_2\, g)$ to get a new term B'. Then $[\![B']\!] = [\![A]\!]$, in the semantics of (PCF+str). A is a "strictification" of B.

If M is a term of (PCF+str), then $\mathrm{unstr}(M)$ is defined as the term M with all occurrences of str replaced by $\lambda f.\, \mathsf{if}\ f0\ \mathsf{then}\ 0\ \mathsf{else}\ \bot$. So $\mathrm{unstr}(M)$ is a PCF-term and $M \leq \mathrm{unstr}(M)$, in the semantics of the extended language.

Now we can define our complementary "syntactic" relation.

Definition 7.3. Let M, N be PCF-terms of the same type. M is a *strictification* of N, written $M \prec^s N$, if there is a (PCF+str)-term M' with $[\![M]\!] = [\![M']\!]$ (in the semantics of (PCF+str)) and $[\![\mathrm{unstr}(M')]\!] = [\![N]\!]$ (in the semantics of PCF).

Note that for PCF-terms M, N: $(M \prec^s N \Longrightarrow [\![M]\!] \leq [\![N]\!])$ and $(M \cong N \Longrightarrow M \prec^s N)$.

Conjecture 7.4 (improved chain conjecture)**.** In PCF we have: For all finite elements $a \leq b$ there is a sequence (M_i) of terms with $1 \leq i \leq n$, $[\![M_1]\!] = a$, $[\![M_n]\!] = b$, and for every $i < n$ it is $M_i \prec M_{i+1}$ or $M_i \prec^s M_{i+1}$.

A proof of this conjecture would be non-trivial and should first be tried on second-order types. (It might be that types higher than second-order need new higher-type strictness operators that cannot be defined from str.) Perhaps the situation should first be clarified in the realm of (PCF+str) and a conjecture of this kind should be proved there.

Our (PCF+str) is the "weakest" sequential extension of PCF with a control operator. It is properly included in (PCF+$\mathsf{strict?}$), this in turn is included in (PCF+H), the sequentially realizable functionals of John Longley [15]; see section 9 in [23] for an overview of such extensions of PCF. (PCF+H) is included in SPCF (mentioned in the introduction), which is no more extensional. For all these extensions of PCF it would be interesting to give syntactic characterizations of the stable order. First it should be clarified if all types are definable retracts of some lower order types, as is the case for (PCF+H) and SPCF. This could make the proofs easier, as we will see for unary PCF in the following section.

8. Unary PCF

Here we will prove Berry's conjectures for unary PCF, with the aid of Jim Laird's results [12]. Unary PCF is the calculus of PCF without Y and with the only constant $\mathsf{0}$ and case_0-expressions. Its semantics is given by the finite elements of $\mathcal{F}_0^\sigma$ for all σ, with the orders $\sqsubseteq$ and $\leq$.

We first repeat the general closure properties of the $\mathcal{F}_i^\sigma$, seen as embedded in the D^σ of an f-model, taken from lemma 3.3, proposition 3.15 and theorem 6.8.

Proposition 8.1. *The $\mathcal{F}_i^\sigma$ are finite and downward closed w.r.t. $\leq$.*
For $a, b \in \mathcal{F}_i^\sigma$, $a \sqcap b \in \mathcal{F}_i^\sigma$ is the glb w.r.t. $\sqsubseteq$ in D^σ and $\mathcal{F}_i^\sigma$. For $a \uparrow_\leq b$ it is also the glb w.r.t. $\leq$.

For $a, b \in \mathcal{F}_i^\sigma$ that are $\sqsubseteq$-bounded in D^σ, $a \sqcup b \in \mathcal{F}_i^\sigma$ is the lub w.r.t. $\sqsubseteq$ in D^σ and $\mathcal{F}_i^\sigma$. For a finite set $X \subseteq \mathcal{F}_i^\sigma$ that has a stable upper bound, all minimal stable upper bounds of X are in $\mathcal{F}_i^\sigma$. The extensional lub $\bigsqcup X$ is one of those. If X has a stable lub, then it is $\bigsqcup X$.

To apply Laird's results on definable retractions, we augment unary PCF with product types $\sigma \times \tau$. The constructs of the whole language are:
$\mathsf{0}\colon \iota$, $\bot^\sigma\colon \sigma$, $x^\sigma\colon \sigma$
If $M\colon \tau$, then $\lambda x^\sigma.M\colon \sigma \to \tau$.
If $M\colon \sigma \to \tau$ and $N\colon \sigma$, then $MN\colon \tau$.
If $M, N\colon \iota$, then $\mathsf{case}_0\, MN\colon \iota$.
If $M\colon \sigma$ and $N\colon \tau$, then $\langle M, N\rangle\colon \sigma \times \tau$.
If $M\colon \sigma \times \tau$, then $\pi_1 M\colon \sigma$ and $\pi_2 M\colon \tau$.

The reduction rules are:
$(\lambda x.M)N \to M[x := N]$
$\mathsf{case}_0\, \mathsf{0}M \to M$
$\pi_1\langle M, N\rangle \to M$
$\pi_2\langle M, N\rangle \to N$

This section needs the products only as auxiliary constructions for the first-order types that are the targets of Laird's retractions. In this section the underlying language is always the augmented unary PCF with products if products are not explicitly excluded.

Laird defines in [12] a categorical notion of *standard model* of unary PCF together with order-extensionality and partial extensional order at each type. He defines *parallel composition* as the function f with $f\langle\bot,\bot\rangle = \bot$, $f\langle\bot,\mathsf{0}\rangle = f\langle\mathsf{0},\bot\rangle = \mathsf{0}$, $f\langle\mathsf{0},\mathsf{0}\rangle = \mathsf{0}$. A model is *universal* at type τ if every element of τ is the denotation of a term.

Definition 8.2 (Laird, definition 3.4 in [12])**.** Given types σ, τ, a *definable retraction from σ to τ* (in a model $\mathcal{M}$) (written $\mathrm{Inj} : \sigma \trianglelefteq \tau : \mathrm{Proj}$ or just $\sigma \trianglelefteq \tau$) is a pair of (closed) terms $\mathrm{Inj}\colon \sigma \to \tau$ and $\mathrm{Proj}\colon \tau \to \sigma$ such that $[\![\lambda x.\, \mathrm{Proj}(\mathrm{Inj}\, x)]\!] = \mathrm{id}$ in $\mathcal{M}$.

Lemma 8.3 (Laird, lemma 3.10 in [12])**.** *For any type τ there is a natural number n such that there is a definable retraction from τ to some binary product form of $(\iota \to \iota)^n$; the same retraction for any standard order-extensional model without parallel composition.*

Theorem 8.4 (Laird, theorem 3.11 in [12])**.** *Any standard model of unary PCF which is order-extensional and excludes parallel composition is universal.*

We can build the stable biorder model of unary PCF as a collection of bicpos $(E^\sigma, \sqsubseteq, \leq)$ for every type σ: We start with $E^\iota = \{\bot, \mathsf{0}\}$ and $\bot \sqsubseteq \mathsf{0}$, $\bot \leq \mathsf{0}$.
$E^{\sigma\times\tau} = E^\sigma \times E^\tau$ with the usual $\sqsubseteq$ and $\leq$.
$E^{\sigma\to\tau}$ is the set of stable and monotone functions $f\colon (E^\sigma, \sqsubseteq, \leq) \to (E^\tau, \sqsubseteq, \leq)$. (If $x \sqsubseteq y$ then $fx \sqsubseteq fy$. If $x \leq y$ then $fx \leq fy$. If $x \uparrow_\leq y$ then $f(x \sqcap y) = fx \sqcap fy$. Continuity conditions are not necessary as the domains are finite.) $E^{\sigma\to\tau}$ is ordered by the usual $\sqsubseteq$ and $\leq$.

$(E^\sigma, \sqsubseteq, \leq)$ is not only a bicpo, but a distributive bicpo where the stable lub of two $\leq$-compatible functions is defined pointwise, by proposition 4.7.10 in Berry's thesis [4]. (If $f \uparrow_\leq f'$, then $(f \vee f')x = fx \vee f'x$.) Therefore the stable lub of two elements is also defined by union on traces.

The stable biorder model fulfills the conditions of theorem 8.4, therefore it is universal (and fully abstract). This means that $(E^\sigma, \sqsubseteq, \leq)$ is isomorphic to $(\mathcal{F}_0^\sigma, \sqsubseteq, \leq)$ for types σ

without products. In the following the semantics of unary PCF-terms is always taken in the model $(E^\sigma, \sqsubseteq, \leq)$. All this proves Berry's first conjecture for unary PCF:

Theorem 8.5 (Laird [12]). *For every type σ without products, the structure $(\mathcal{F}_0^\sigma, \sqsubseteq, \leq)$ is a distributive bicpo (hence also a bidomain as it is finite).*
For $a, b \in \mathcal{F}_0^\sigma$ with $a \uparrow_\leq b$, $a \vee b$ is given by $\mathcal{T}(a \vee b) = \mathcal{T}(a) \cup \mathcal{T}(b)$ and this lub is taken pointwise for functions a, b.

With the aid of Laird's definable retractions we can prove a strong form of Berry's second conjecture for unary PCF, based on the fact that it is valid for first-order types. First we need two lemmas on the reduction.

Lemma 8.6. *The reduction $\rightarrow$ on unary PCF with products is confluent and strongly normalizing. Therefore it has unique normal forms. The normal form of a term of a type without products does not contain any product subterm.*

Proof. The confluence can be proved with the main theorem of [17], see also [6, theorem 10.4.15, page 576]: The rules of $\rightarrow$ without the β-rule are confluent on the applicative terms (i.e. the terms without λ), as they are orthogonal; they are left-linear and not variable-applying. Therefore their combination with the β-rule is confluent.

For the proof of strong normalization there seems to be no theorem in the literature that would provide an easy modular check for the simply typed λ-calculus with algebraic rewrite rules of our form.

Therefore we take the proof of strong normalization of the simply typed λ-calculus with products in the textbook [10, chapter 6] for the only atomic type ι and augment it by the constant $\mathsf{0}$ and case_0-expressions. The proof stays literally the same. The only thing we have to add is a proof that if M, N are strongly normalizable, then $\mathsf{case}_0\, MN$ is so; in the proof that all terms are reducible. $\square$

Lemma 8.7. *Let ω be the following map on unary PCF-terms (where $n, m \geq 0$):*

$$\begin{aligned}
\omega(\lambda x_1 \dots x_n.\mathsf{0}) &= \lambda x_1 \dots x_n.\mathsf{0} \\
\omega(\lambda x_1 \dots x_n.y M_1 \dots M_m) &= \lambda x_1 \dots x_n.y\, \omega(M_1) \dots \omega(M_m), \text{ for } y \text{ variable} \\
\omega(\lambda x_1 \dots x_n.\, \mathsf{case}_0\, MN) &= \lambda x_1 \dots x_n.\, \mathsf{case}_0\, \omega(M)\, \omega(N), \\
&\qquad \text{if } \omega(M) = \mathsf{case}_0 \dots \text{ or } \omega(M) = y \dots \text{ with } y \text{ variable} \\
\omega(\lambda x_1 \dots x_n.\langle M, N\rangle) &= \lambda x_1 \dots x_n.\langle \omega(M), \omega(N)\rangle \\
\omega(\lambda x_1 \dots x_n.\pi_1 M) &= \lambda x_1 \dots x_n.\pi_1\, \omega(M), \text{ if } \omega(M) = y \dots \text{ with } y \text{ variable} \\
\omega(\lambda x_1 \dots x_n.\pi_2 M) &= \lambda x_1 \dots x_n.\pi_2\, \omega(M), \text{ if } \omega(M) = y \dots \text{ with } y \text{ variable} \\
\omega(M) &= \perp, \text{ in all other cases}
\end{aligned}$$

$\omega(M)$ is a normal form prefix of M, it pushes $\perp$s upwards.
If M is a normal form, then $\omega(M) \cong M$.
If $M \rightarrow^* N$, then $\omega(M) \prec \omega(N)$.
If $M \prec N$, then $\omega(M) \prec \omega(N)$.
We define $\mathrm{nf}(M) = \omega($*the normal form of* $M)$.
For all $M \prec N$ it is $\mathrm{nf}(M) \prec \mathrm{nf}(N)$.

Proof. The first four propositions are clear, we prove here the last one; the proof is similar to the one of lemma 4.4.

Let M', N' be the normal forms of M, N.
As the reduction rules for $\to$ do not involve $\bot$, all the reductions $M \to^* M'$ can also be done in N. (If $A \prec B$ and $A \to A'$, then there is B' with $B \to B'$ and $A' \prec B'$.)
So there is N'' with $N \to^* N''$ and $M' \prec N''$.
By confluence of $\to$ it is $N'' \to^* N'$.
Then we get $\mathrm{nf}(M) = \omega(M') \prec \omega(N'') \prec \omega(N') = \mathrm{nf}(N)$. □

Theorem 8.8. *For every type σ without products, for every $a \in \mathcal{F}_0^\sigma$ there is a game term $A\colon \sigma$ with $a = [\![A]\!]$ such that for every $b \leq a$ there is $B \prec A$ with $b = [\![B]\!]$.*

Proof. By Laird's lemma 8.3 there is a number n and a definable retraction $\mathrm{Inj}\colon \sigma \trianglelefteq \tau\colon \mathrm{Proj}$, with τ some binary product form of $(\iota \to \iota)^n$.
Let A' be a term for a, $[\![A']\!] = a$.
Let $A'' = \mathrm{nf}(\mathrm{Proj}(\mathrm{Inj}\, A'))$. A'' does not contain any subterm of product type.
By the game term theorem 4.12 we get the desired game term $A = \mathrm{gt}_0^\sigma(\Psi_0^\sigma A'')$ with $A \cong A''$, so $[\![A]\!] = a$.
Let $C = \mathrm{nf}(\mathrm{Inj}\, A')$. $C = \langle C_1, \ldots, C_n\rangle$ in some binary pair form, where $C_i \cong \lambda x.\bot$ or $\lambda x.\mathsf{0}$ or $\lambda x.x$.
Let $b \leq a$. Then $[\![\mathrm{Inj}]\!] b \leq [\![\mathrm{Inj}]\!] a = [\![C]\!]$.
For every i, if $x \leq [\![C_i]\!]$ then $x = [\![C_i]\!]$ or $x = \bot$. Therefore there is $B' \prec C$ with $[\![B']\!] = [\![\mathrm{Inj}]\!] b$.
Let $B'' = \mathrm{nf}(\mathrm{Proj}\, B')$. It is $A'' = \mathrm{nf}(\mathrm{Proj}\, C)$. Therefore $B'' \prec A''$.
By the game term theorem 4.12 there is a game term $B = \mathrm{gt}_0^\sigma(\Psi_0^\sigma B'')$ with $B \cong B''$ and $B \prec A$.
We have $b = [\![\mathrm{Proj}]\!]([\![\mathrm{Inj}]\!] b) = [\![\mathrm{Proj}]\!][\![B']\!] = [\![B]\!]$. □

Remark: Please note that Laird's retractions are incredibly intelligent, because they must introduce in the term $A'' = \mathrm{nf}(\mathrm{Proj}(\mathrm{Inj}\, A'))$ some nestings of variables that were not present in A', to fulfill the proposition of the theorem.

It is a nice exercise (of three pages) to compute an example: Take $\sigma = (\iota \to \iota \to \iota) \to \iota$ and $A' = \lambda g.g\mathsf{0}\mathsf{0}\colon \sigma$. The trace of A' is

$$\mathcal{T}[\![A']\!] = \{\{\bot\bot{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}, \{\mathsf{0}\bot{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}, \{\bot\mathsf{0}{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}, \{\mathsf{0}\mathsf{0}{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}\}.$$

Going through Laird's proof of lemma 8.3, we get complicated terms $\mathrm{Inj}\colon \sigma \trianglelefteq \tau\colon \mathrm{Proj}$ with $\tau = (((\iota \to \iota) \times (\iota \to \iota)) \times (\iota \to \iota)) \times ((\iota \to \iota) \times \iota)$.
We compute the normal forms:

$$\mathrm{Inj}\, A' \to^* C = \langle\langle\langle \lambda x.x, \lambda x.x\rangle, \lambda x.x\rangle, \langle \lambda x.x, \underline{\mathsf{0}}\rangle\rangle$$

$$\mathrm{Proj}(\mathrm{Inj}\, A') \to^* A'' = \lambda g.\, \mathsf{case}_0[g(g\mathsf{0}(g\underline{\mathsf{0}}\mathsf{0}))\mathsf{0}][g\mathsf{0}(g\underline{\mathsf{0}}\mathsf{0})]$$

This term is much more expanded than needed.

If we replace the underlined $\underline{\mathsf{0}}$ in C by $\bot$, we get a term A'' with both underlined $\underline{\mathsf{0}}$ replaced by $\bot$. The trace of this new term A'' is $\{\{\bot\bot{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}, \{\mathsf{0}\bot{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}, \{\bot\mathsf{0}{\mapsto}\mathsf{0}\}{\mapsto}\mathsf{0}\}$.
Note that there was no syntactically lesser term than A' with this trace.

Remark: Another recommended exercise for the reader is to encode our first counter-example (to Berry's second conjecture) of subsection 5.1 in unary PCF. The booleans are encoded by the type $\beta = \iota \to \iota \to \iota$ as usual. The value $\mathsf{0}$ is represented by $\lambda xy.x$, $\mathsf{1}$ is represented by $\lambda xy.y$. There are three more inhabitants of β: $\bot$, $\lambda xy.\, \mathsf{case}_0\, xy$ and $\lambda xy.\mathsf{0}$. The example is now of type $(\beta \to \beta \to \beta) \to \beta$. The term D can be given an expanded form such that $A \prec B = C \prec D$. In D the top boolean $\lambda xy.\mathsf{0}$ is used (in one position) as the lub of $\lambda xy.x$ and $\lambda xy.y$.

9. Outlook

We have seen one trick to produce several examples which show that the stable order in PCF is not so regular as Berry had expected. These counter-examples have as necessary ingredients: at least two incompatible values and at least a second-order type with at least arity two of some functional parameter. To be precise, we still have to show that Berry's conjectures are valid in all second-order types with functional parameters of only arity one, see conjectures 6.7 and 5.2.

With the refutation of the chain conjecture in section 7 we have shown that there is no simple characterization of the stable order in terms of the syntactic order. In fact the counter-example shows that there is not only the syntactic order that causes the stable order, but that there are other syntactic relations needed with this property. Such another relation was identified as the relation of "strictification", and an improved chain conjecture 7.4 was tentatively proposed.

There should be some kind of full syntactic account of the stable order, at least for second-order types. For any type there should be syntactic conditions that are necessary for the relation $A \leq B$ of terms. These should at least prove the maximality conjecture 7.2: Every PCF-term without Y that is syntactically maximal is also stably maximal.

It would also be interesting to find syntactic characterizations of the stable order in extensions of PCF by sequential control operators, i.e. in (PCF+str), (PCF+$\mathsf{strict?}$), (PCF+H) and SPCF, see the remarks at the end of section 7.

In this paper we have treated the problem of the syntactic characterization of the stable order, but Berry originally had in mind the semantic characterization of the syntactic order. In the light of the results of this paper this seems to be a problem of similar difficulty. One should first seek necessary conditions for the syntactic order that are stronger than the stable order.

Acknowledgement

I thank Reinhold Heckmann for carefully reading drafts of this paper and many discussions. I thank Reinhard Wilhelm and the members of his chair for their support. I thank the anonymous referees for their valuable suggestions.

References

[1] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. Full abstraction for PCF. *Information and Computation*, 163:409–470, 2000.
[2] Roberto M. Amadio and Pierre-Louis Curien. *Domains and Lambda-Calculi*. Cambridge University Press, 1998.
[3] Gérard Berry. Stable models of typed λ-calculi. In *5. ICALP'78, LNCS 62*, pages 72–89. Springer, 1978.
[4] Gérard Berry. *Modèles complètement adéquats et stables des lambda-calculs typés*. PhD thesis, Université Paris VII, 1979.
[5] Gérard Berry, Pierre-Louis Curien, and Jean-Jacques Levy. Full abstraction for sequential languages: The state of the art. In Maurice Nivat and John Reynolds, editors, *Algebraic Methods in Semantics*, pages 59–132. Cambridge University Press, 1985.

[6] Inge Bethke. Lambda calculus. In Terese, editor, *Term Rewriting Systems*, pages 548–587. Cambridge University Press, 2003.

[7] Pierre-Louis Curien, Gordon Plotkin, and Glynn Winskel. Bistructures, bidomains and linear logic. In Gordon Plotkin, Colin Stirling, and Mads Tofte, editors, *Proof, Language, and Interaction. Essays in Honour of Robin Milner*. MIT Press, 2000.

[8] Martin Escardó and Weng Kin Ho. Operational domain theory and topology of sequential programming languages. *Information and Computation*, 207:411–437, 2009.

[9] Jean-Yves Girard. The system F of variable types, fifteen years later. *Theoretical Computer Science*, 45:159–192, 1986.

[10] Jean-Yves Girard, Paul Taylor, and Yves Lafont. *Proofs and Types*. Cambridge University Press, 1989.

[11] J. Martin E. Hyland and C.-H. Luke Ong. On full abstraction for PCF. *Information and Computation*, 163:285–408, 2000.

[12] Jim Laird. Sequentiality in bounded biorders. *Fundamenta Informaticae*, 65:173–191, 2005.

[13] Jim Laird. On the expressiveness of affine programs with non-local control: The elimination of nesting in SPCF. *Fundamenta Informaticae*, 77:511–531, 2007.

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

[15] John Longley. The sequentially realizable functionals. *Annals of Pure and Applied Logic*, 117:1–93, 2002.

[16] Robin Milner. Fully abstract models of typed λ-calculi. *Theoretical Computer Science*, 4:1–22, 1977.

[17] Fritz Müller. Confluence of the lambda calculus with left-linear algebraic rewriting. *Information Processing Letters*, 41:293–299, 1992. Please use the electronic version: `http://rw4.cs.uni-saarland.de/~mueller/lconfluence.ps.gz`.

[18] Hanno Nickau. Hereditarily sequential functionals. In *Logical Foundations of Computer Science, LNCS 813*, pages 253–264. Springer, 1994.

[19] Dag Normann. On sequential functionals of type 3. *Mathematical Structures in Computer Science*, 16:279–289, 2006.

[20] Dag Normann and Vladimir Yu. Sazonov. The extensional ordering of the sequential functionals. *Annals of Pure and Applied Logic*, 163:575–603, 2012.

[21] Peter W. O'Hearn and Jon G. Riecke. Kripke logical relations and PCF. *Information and Computation*, 120:107–116, 1995.

[22] C.-H. Luke Ong. Correspondence between operational and denotational semantics: the full abstraction problem for PCF. In S. Abramsky, D. Gabbay, and T.S.E. Maibaum, editors, *Handbook of Logic in Computer Science, Volume 4*, pages 269–356. Oxford University Press, 1995.

[23] Luca Paolini. A stable programming language. *Information and Computation*, 204:339–375, 2006.

[24] Gordon D. Plotkin. LCF considered as a programming language. *Theoretical Computer Science*, 5:223–256, 1977.

[25] Vladimir Sazonov. An inductive definition and domain theoretic properties of fully abstract models for PCF and PCF+. *Logical Methods in Computer Science*, 3:1–50, 2007.

[26] Vladimir Sazonov. Natural non-dcpo domains and f-spaces. *Annals of Pure and Applied Logic*, 159:341–355, 2009.

[27] Dana S. Scott. A type-theoretical alternative to ISWIM, CUCH, OWHY. *Theoretical Computer Science*, 121:411–440, 1993. Originally written and distributed in 1969.

[28] Kurt Sieber. Reasoning about sequential functions via logical relations. In *Applications of Categories in Computer Science*. Cambridge University Press, 1992.

[29] Allen Stoughton. Substitution revisited. *Theoretical Computer Science*, 59:317–325, 1988.

[30] Allen Stoughton. Mechanizing logical relations. In *Mathematical Foundations of Programming Semantics 1993, LNCS 802*, pages 359–377. Springer, 1994.

[31] Paul Taylor. An algebraic approach to stable domains. *Journal of Pure and Applied Algebra*, 64:171–203, 1990.