Fritz Müller `http://rw4.cs.uni-saarland.de/~mueller/uglyproof.ps.gz`
supplementary material for "On Berry's conjectures about the stable order in PCF"
The first version of section 4. Game Terms: The proof of the game term theorem without reducibility predicate.

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 [6, 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 a new construct for the PCF language, for every $i \geq 0$:
If $M, N_0, \ldots, N_i \colon \iota$, then $\mathsf{case}_i\, M N_0 \ldots N_i \colon \iota$.
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. The reduction rule for case_i is:

$$\mathsf{case}_i\, n N_0 \ldots N_i \to N_n, \text{ for } 0 \leq n \leq i$$

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

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

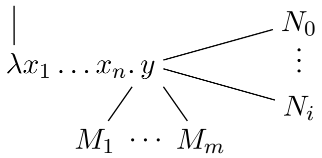
This is the "filter" as it appears in the finite projection term Ψ_i^ι. So case_i does not enhance the expressiveness of PCF. It is merely a "macro" that is used as short expression for the filter term above, to keep the unity of the filter term in the transformation to game terms.

Definition 4.1. *Game terms of grade i*, $i \geq 0$, are the well-typed closed PCF-case-terms that are furthermore produced by the following grammar:

$$\begin{aligned} M, N ::= {} & \bot^\sigma, \ \sigma \text{ any type} \\ & \lambda x_1 \ldots x_n.m, \ m \leq i \text{ integer constant}, \ n \geq 0 \\ & \lambda x_1 \ldots x_n.\, \mathsf{case}_i(y M_1 \ldots M_m) N_0 \ldots N_i, \ n, m \geq 0 \end{aligned}$$

Please note that $\lambda x_1 \ldots x_n.$ vanishes for $n = 0$. All game terms of grade i represent finite terms of grade i. (Please remember that a term $M \colon \sigma$ is a finite term of grade i if it is *closed* and $\Psi_i^\sigma M \cong M$.)

We will use a graphical representation of game terms in the next sections:
A subterm $\lambda x_1 \dots x_n.\, \mathsf{case}_i(yM_1 \dots M_m)N_0 \dots N_i$ 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 $M_1, \dots, M_m$ are the *legs* of y; the $N_0, \dots, N_i$ 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, see the examples in the next section.

Our aim is to prove the following theorem:

Theorem 4.2 (Game Term Theorem)**.** *If $M \prec N$ are finite terms of grade i, then there are game terms $M' \prec N'$ of grade i with $M' \cong M$ and $N' \cong N$.*

This means that the refutation of Berry's second conjecture may be restricted to game terms. We prove the theorem in two steps: First we reduce M, N with the rules β and $\mathsf{Y}M \to M(\mathsf{Y}M)$ until we get normal form approximations (with full semantics) that do not contain Y. Then we apply a transformation gt_i^σ (to be explained below) to get the desired game terms.

The lemmas for the two steps are:

Lemma 4.3 (Approximation Lemma)**.** *(explained below)*
If $M \prec N$ are finite terms, then there are $M' \prec N'$ with $M \to_{\beta Y \perp}^* M'$, $N \to_{\beta Y}^* N'$, $\omega(M') \cong M$, $\omega(N') \cong N$.
Then also $\omega(M') \prec \omega(N')$, and these terms do not contain Y.

Lemma 4.4 (Game Term Lemma)**.** *If $M \prec N$ are finite terms of type σ and grade i without Y, then $\mathrm{gt}_i^\sigma(M) \prec \mathrm{gt}_i^\sigma(N)$, $\mathrm{gt}_i^\sigma(M) \cong M$, $\mathrm{gt}_i^\sigma(N) \cong N$, and $\mathrm{gt}_i^\sigma(M)$, $\mathrm{gt}_i^\sigma(N)$ are game terms of grade i.*

The game term theorem follows immediately from these two lemmas. (Note that equivalent terms have the same grade.) The subsections 4.1 and 4.2 prove them; this is a boring exercise in rewriting bureaucracy and can be skipped. The theorem will be used in the proofs of the counter-examples. Finally subsection 4.3 introduces infinite game terms.

4.1. **Proof of the Approximation Lemma.** First the definitions for the lemma:
The map $\omega\colon \mathrm{PCF} \to \mathrm{PCF}$ is taken from [4, 3] and called the *immediate syntactic value*:

$$\omega(M) = \begin{cases} \lambda x_1 \dots x_n.u\, \omega(M_1) \dots \omega(M_m), & \text{if } M = \lambda x_1 \dots x_n.uM_1 \dots M_m \\ & \text{with } u \text{ a variable or constant,} \\ & \text{i.e. } M \text{ is in head normal form} \\ \perp \text{ else} & \end{cases}$$

Please note that in this definition u is not $\perp$ or Y.
$\to_{\beta Y}$ is a one-step reduction with the β-rule or the rule $\mathsf{Y}M \to M(\mathsf{Y}M)$.
$\to_{\beta Y\perp}$ is a one-step reduction with those two rules or the rule $\perp M \to \perp$.
As is known from [4, 3], if $M \to^*_{\beta Y} N$, then $\omega(M) \prec \omega(N)$. As a simple extension also: If $M \to^*_{\beta Y\perp} N$, then $\omega(M) \prec \omega(N)$.

Lemma 4.5. *For all finite M there is N with $M \to^*_{\beta Y} N$ and $\omega(N) \cong M$.*

Proof. For the fully abstract cpo-model (and therefore for all f-models) the approximation continuity theorem [4] 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 $[\![M]\!] = [\![\omega(N)]\!]$. $\square$

In the following we assume some usual notion of position in a term (as a symbol string that denotes the path to the position). The subterm of M at position u is denoted M/u.

Lemma 4.6. *If $M \prec N$ and $N \to_{\beta Y} N'$, then simply $M \prec N'$ or there is M' with $M \to_{\beta Y\perp} M'$ and $M' \prec N'$.*

Proof. Let u in N be the position of the reduction $N \to_{\beta Y} N'$. Let the reduction be of the form $(\lambda x.P)L \to P[x := L]$. There are some cases for the position u:
If u is not in M then simply $M \prec N'$.
If $M/u = \perp$ then also $M \prec N'$.
If $M/u = \perp L'$ then reduce $\perp L' \to \perp$ in M.
If $M/u = (\lambda x.P')L'$, then $P' \prec P$, $L' \prec L$. Reduce $(\lambda x.P')L' \to P'[x := L']$ in M to get $M' \prec N'$.
The proof for a reduction of the form $\mathsf{Y}P \to P(\mathsf{Y}P)$ is similar. $\square$

Lemma 4.7. *If $M \prec N$ and $M \to_{\beta Y\perp} M'$, then simply $M' \prec N$, or there is N' with $N \to_{\beta Y} N'$ and $M' \prec N'$.*

Proof. If the reduction is with the β-rule or the Y-rule, then it can be performed also in N to get N', as $M \prec N$.
If the reduction is with the rule $\perp P \to \perp$, then simply $M' \prec N$. $\square$

Lemma 4.8. *The reduction relation $\to_{\beta Y\perp}$ is confluent.*

Proof. This can be proved e.g. with the Main Theorem of [7], see also [5, Theorem 10.4.15, page 576]: The rules $\mathsf{Y}M \to M(\mathsf{Y}M)$ and $\perp M \to \perp$ 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. $\square$

Proof of the Approximation Lemma 4.3:
By lemma 4.5 there is M^* with $M \to^*_{\beta Y} M^*$ and $M \cong \omega(M^*)$; and there is N^* with $N \to^*_{\beta Y} N^*$ and $N \cong \omega(N^*)$.
By lemma 4.6 there is M^+ with $M \to^*_{\beta Y\perp} M^+$ and $M^+ \prec N^*$.
By confluence there is M' with $M^+ \to^*_{\beta Y\perp} M'$ and $M^* \to^*_{\beta Y\perp} M'$.
By lemma 4.7 there is N' with $N^* \to^*_{\beta Y} N'$ and $M' \prec N'$.
We have $M \to^*_{\beta Y} M^* \to^*_{\beta Y\perp} M'$ and $N \to^*_{\beta Y} N^* \to^*_{\beta Y} N'$ and $M' \prec N'$.
As $\omega(M^*) \prec \omega(M')$, we get $\omega(M') \cong M$.

As $\omega(N^*) \prec \omega(N')$, we get $\omega(N') \cong N$.
(Remark: In fact already M^+ and N^* could have been taken as the desired terms M', N', but for this it would be hard to prove that M^+ delivers already the full normal form approximation to M.) $\square$

4.2. **Proof of the game term lemma.** Let us first ask if the β-Y-normal form finite terms that we get from the approximation lemma already have a structure that is close to game terms. Finite terms have to filter or restrict their input and output values on all type levels and therefore should have some kind of eta-expanded structure. But there are exceptions: First there may be redundant subterms in dead branches that can be safely replaced by $\perp$; these subterms may have any structure. But it does not suffice to eliminate these redundant subterms, as this example shows: Let $\sigma_1 = (\iota \to \iota) \to \iota$, $\sigma_2 = \iota \to \iota$, $\sigma = \sigma_1 \to \sigma_2 \to \iota$.

$$X = \lambda g^{\sigma_1} f^{\sigma_2}.\,\mathsf{if}\ f0\ \mathsf{then\ if}\ g(\lambda x^{\iota}.\,\mathsf{if}\ x\ \mathsf{then}\ (\mathsf{if}\ gf\ \mathsf{then}\ 0\ \mathsf{else}\ \perp)\ \mathsf{else}\ \perp)\ \mathsf{then}\ 0\ \mathsf{else}\ \perp$$
$$\mathsf{else}\ \perp$$

Note that X is finite of grade 0, although the f in the subterm gf is not eta-expanded and "filtered". (And there is no redundant subterm that can be replaced by $\perp$.) This f could be safely replaced by $\lambda y.\,\mathsf{if}\ y\ \mathsf{then}\ 0\ \mathsf{else}\ \perp$, but this is not an intensionally equivalent replacement. We do not further explore this way of transformation by replacements, but will impose the game term structure on our terms from outside.

The production of game terms by the transformation gt_i^σ is done in three steps. We use X as our running example in these steps.

First transformation step.
Only for the transformation we introduce new language constructs:
If $M\colon \sigma$, then $\Phi_i^\sigma M\colon \sigma$ and $\bar{\Phi}_i^\sigma M\colon \sigma$.
Φ_i^σ and $\bar{\Phi}_i^\sigma$ represent the term Ψ_i^σ, they are declared to have the same semantic denotation. But in the process of transformation we restrict or enhance the reduction "power" of these new symbols to certain rules that are appropriate to the actual transformation step. Of course, these rules preserve the semantics.

In the first transformation step, Φ_i^σ is applied to the terms $M \prec N$ of the game term lemma. Here the reduction rules for Φ_i^σ, $\bar{\Phi}_i^\sigma$ are for $\sigma = \sigma_1 \to \ldots \to \sigma_n \to \iota$:

$$\Phi_i^\sigma M \to \lambda x_1^{\sigma_1} \ldots x_n^{\sigma_n}.\Phi_i^\iota(M(\bar{\Phi}_i^{\sigma_1} x_1)\ldots(\bar{\Phi}_i^{\sigma_n} x_n)),\ \text{for } n \geq 1$$
$$\bar{\Phi}_i^\sigma M \to \lambda y_1^{\sigma_1} \ldots y_n^{\sigma_n}.\,\mathsf{case}_i(M(\Phi_i^{\sigma_1} y_1)\ldots(\Phi_i^{\sigma_n} y_n))0\ldots i,\ \text{for } n \geq 0$$

Φ_i^σ and $\bar{\Phi}_i^\sigma$ alternate when going down the type: Φ_i^σ filters the output by applying Φ_i^ι, $\bar{\Phi}_i^\sigma$ filters the input by applying case_i. They build up the fully η-expanded term structure around the term M. Note that at this stage there is no rule for Φ_i^ι.

We can show the joint effect of the two alternating steps:
Let $M\colon \sigma$, $\sigma = \sigma_1 \to \ldots \to \sigma_n \to \iota$, $\sigma_j = \sigma_{j1} \to \cdots \to \sigma_{jk_j} \to \iota$, then

$$\Phi_i^\sigma M \to^* \lambda x_1^{\sigma_1} \ldots x_n^{\sigma_n}.\Phi_i^\iota[M(\lambda y_{11} \ldots y_{1k_1}.\,\mathsf{case}_i(x_1(\Phi_i^{\sigma_{11}} y_{11})\ldots(\Phi_i^{\sigma_{1k_1}} y_{1k_1}))0\ldots i)$$
$$\ldots(\lambda y_{n1} \ldots y_{nk_n}.\,\mathsf{case}_i(x_n(\Phi_i^{\sigma_{n1}} y_{n1})\ldots(\Phi_i^{\sigma_{nk_n}} y_{nk_n}))0\ldots i)]$$

The reduction rules for Φ_i^σ and $\bar{\Phi}_i^\sigma$ preserve the semantics. We call $\varphi_i^\sigma(M)$ the result of the complete reduction of $\Phi_i^\sigma M$. $\varphi_i^\sigma(M)$ is like the result of the first substitution of M in $\Psi_i^\sigma M$, only the "filters" are replaced by the expressions with Φ_i^ι and case_i. We call PCF-terms without Y, but possibly with case_i and Φ_i^ι shortly *PCF-i-terms.* We get $\varphi_i^\sigma(M) \prec \varphi_i^\sigma(N)$.

For our example we get:

$$\varphi_0^\sigma(X) = \lambda x_1^{\sigma_1} x_2^{\sigma_2}.\Phi_0^\iota[X(\lambda y^{\iota\to\iota}.\,\mathsf{case}_0(x_1^{\sigma_1}(\lambda x_1^\iota.\Phi_0^\iota(y^{\iota\to\iota}(\mathsf{case}_0\, x_1^\iota \mathsf{0}))))\mathsf{0})$$
$$(\lambda y_1^\iota.\,\mathsf{case}_0(x_2^{\sigma_2}(\Phi_0^\iota y_1^\iota))\mathsf{0})]$$

Second transformation step.

Here we first analyze the general structure of the PCF-i-terms $\varphi_i^\sigma(M)$ that will lead us to game terms.

First we look at our example $\varphi_0^\sigma(X)$ and show what definitions we will introduce later. We call the variables $x_1^{\sigma_1}$, $x_2^{\sigma_2}$ and x_1^ι *input variables*, as they are the only variables that are used for the communication of the term with the outside world. They are the only variables that will survive the transformation, all other variables are "internal" and will be eliminated by beta reduction.

$x_1^{\sigma_1}$ and $x_2^{\sigma_2}$ are input variables of *depth* 1, because they communicate directly. x_1^ι is an input variable of *depth* 2, because it depends on the input variable occurrence $x_1^{\sigma_1}$ of depth 1 (the *parent* of x_1^ι); it communicates with the future argument that will be substituted for $x_1^{\sigma_1}$. And to be precise, we will speak not of input variables, but always of *input variable occurrences (ivo).* ivos should be in a specific conext, e.g. the occurrence of $x_1^{\sigma_1}$ is in the context

$$\mathsf{case}_0(x_1^{\sigma_1}(\lambda x_1^\iota.\Phi_0^\iota \ldots)\ldots$$

This context is called the *input region* of the ivo. Roughly: A PCF-i-term is in *input form* if every ivo is in an input region.

Now to the proper definitions. Again we need a notion of position u, v, w in PCF-terms as strings of symbols that denote the path to the position. $u \leq v$ means that u is some ancestor (prefix string) of v. M/u is the subterm of M at position u. The occurrence of a symbol x (mostly a variable) at position u in some term will be denoted $x|u$, where for a variable x the occurrence is not in a lambda binder λx. We also need a notion of region in a term which is quite obvious, but which I could not find in the literature:
A *region in a term* M is some (M, u, U), where u is a position in M, the root of the region, and U is a convex set of positions in M with u the least element of U. (U is convex means: If $v_1 \leq w \leq v_2$ and $v_1, v_2 \in U$, then $w \in U$.)
A region in a term M could be seen as a truncated subterm of M, but retained in its place inside M.
A symbol occurrence $x|v$ *is in a region* (M, u, U), if $v \in U$ and $M/v = x$.
A region (M, u', U') *is in a region* (M, u, U) if $U' \subseteq U$.

If we have an expression *exp* of a term pattern with term variables $M, N, \ldots$ (e.g. $(\lambda x.M)N$), then we say that some region (P, u, U) *is of the form exp*, if V is the set of positions of non-term-variable symbols in *exp*, $U = \{\, uv \mid v \in V \,\}$ and for every $v \in V$: the symbol of *exp* at v is the same as the symbol of P at uv.

Now we describe the general form of the terms $\varphi_i^\sigma(M)$ from the first transformation step.

Definition 4.9. Let $M\colon \sigma_1 \to \ldots \to \sigma_n \to \iota$ be a closed PCF-i-term. We define *input variable occurrences (ivo) of depth* d in M by induction on d:

- $x|u$ is an *input variable occurrence of depth* 1 in M if M is of the form $M = \lambda x_1 \dots x_n.\Phi_i^\iota M'$ and $x = x_j$ for some j and $M/u = x$ and this occurrence is bound by x_j in the binder $\lambda x_1 \dots x_n$.
- $x|u$ is an *input variable occurrence of depth* $d+1$ in M if there is an ivo $y|v$ of depth d in M (the *parent ivo* of $x|u$) such that $y|v$ is the occurrence of y in a region of M of the form
$$\mathsf{case}_i(y(\lambda x_{11} \dots x_{1k_1}.\Phi_i^\iota M_1) \dots (\lambda x_{m1} \dots x_{mk_m}.\Phi_i^\iota M_m))N_0 \dots N_i$$
(this is called the *input region of* $y|v$), and there is $h \leq m$ and $j \leq k_h$ such that $x = x_{hj}$ and u is in the region of M_h in M, $M/u = x$ and this occurrence of x is bound by x_{hj} in the binder $\lambda x_{h1} \dots x_{hk_h}$.

An *input region* is always understood to be the input region of an ivo.
M is in *input form of grade* i if M is of the form $M = \lambda x_1 \dots x_n.\Phi_i^\iota M'$ and every ivo $y|v$ and every occurrence of case_i in M is in an input region (as above), and every occurrence of Φ_i^ι in M' is in an input region.

The outcoming terms $\varphi_i^\sigma(M) \prec \varphi_i^\sigma(N)$ of the first step are input forms of grade i. The input form will be invariant under the following transformation by some reduction rules. For the result of this second step we aim at pre-game terms:

Definition 4.10. A *pre-game term* of grade i is a well-typed closed PCF-i-term of the form:
$$\lambda x_1 \dots x_n.\Phi_i^\iota(\text{integer-term of grade } i), n \geq 0$$
where an *integer-term* of grade i is produced by the grammar:
$$\begin{aligned} M, N ::= {} & \bot^\iota \\ & n, \text{ integer constant} \\ & \mathsf{case}_i(y(\lambda x_{11} \dots x_{1k_1}.\Phi_i^\iota M_1) \dots (\lambda x_{m1} \dots x_{mk_m}.\Phi_i^\iota M_m))N_0 \dots N_i \end{aligned}$$

A pre-game term is in input form, its definition is based on input regions.

We introduce reduction rules $\to_c$ for PCF-i-terms.
First the PCF-rules for constants:
$$\begin{aligned} \mathsf{suc}\, n &\to_c (n+1) \\ \mathsf{pre}\, n &\to_c (n-1), \text{ for } n \geq 1 \\ \mathsf{if}\, 0\, \mathsf{then}\, M\, \mathsf{else}\, N &\to_c M \\ \mathsf{if}\, n\, \mathsf{then}\, M\, \mathsf{else}\, N &\to_c N, \text{ for } n \geq 1 \end{aligned}$$
Then some rules over case_i:
$$\begin{aligned} \mathsf{suc}(\mathsf{case}_i\, M N_0 \dots N_i) &\to_c \mathsf{case}_i\, M(\mathsf{suc}\, N_0) \dots (\mathsf{suc}\, N_i) \\ \mathsf{pre}(\mathsf{case}_i\, M N_0 \dots N_i) &\to_c \mathsf{case}_i\, M(\mathsf{pre}\, N_0) \dots (\mathsf{pre}\, N_i) \\ \mathsf{if}(\mathsf{case}_i\, M N_0 \dots N_i)\, \mathsf{then}\, Q\, \mathsf{else}\, R &\to_c \mathsf{case}_i\, M(\mathsf{if}\, N_0\, \mathsf{then}\, Q\, \mathsf{else}\, R) \dots (\mathsf{if}\, N_i\, \mathsf{then}\, Q\, \mathsf{else}\, R) \end{aligned}$$

Then some rules for $\perp$:

$$\begin{aligned}
\mathsf{pre}\,\mathsf{0} &\to_c \perp \\
\mathsf{suc}\perp &\to_c \perp \\
\mathsf{pre}\perp &\to_c \perp \\
\mathsf{if}\perp\mathsf{then}\ M\ \mathsf{else}\ N &\to_c \perp \\
\perp M &\to_c \perp
\end{aligned}$$

Note that case_i is regarded by these rules as a constructor (like the integer constants) on which the evaluators $\mathsf{suc}, \mathsf{pre}, \mathsf{if}$ work. There is no rule for case_i as evaluator, and no rule with Φ_i^ι.

$\to_c$ is the reduction by some rule for $\to_c$ in any context. $\to_{\beta c}$ is the reduction by $\to_c$ or the β-rule in any context.

Lemma 4.11. *The reduction $\to_{\beta c}$ is confluent on PCF-i-terms.*

Proof. This can again be proved with the Main Theorem of [7]: The rules $\to_c$ are confluent on the applicative terms (they are orthogonal), they are left-linear and not variable-applying. Therefore their combination with the β-rule is confluent. □

Lemma 4.12. *Every PCF-i-term M has a unique normal form for the reduction $\to_{\beta c}$, called* $\mathrm{nf}(M)$*.*

Proof. As first step there is a normal form M' of M for β-reduction, as we are in a simply typed λ-calculus. If N is a β-normal form and $N \to_c N'$, then N' is also a β-normal form; as no rule for $\to_c$ can generate a new β-redex. We $\to_c$-reduce M' to a normal form for $\to_c$, and so we also get a normal form for $\to_{\beta c}$ reduction. This normal form is unique because of confluence.

It remains to show that $\to_c$ is (strongly) normalizing. This can be proved by an interpretation num of PCF-i-terms in numbers ≥ 2:

$$\begin{aligned}
\mathrm{num}(M) &= 2, \text{ for } M \text{ a variable, a constant or } \perp \\
\mathrm{num}(MN) &= \mathrm{num}(M) \cdot \mathrm{num}(N) \\
\mathrm{num}(\lambda x.M) &= \mathrm{num}(M) \\
\mathrm{num}(\Phi_i^\iota M) &= \mathrm{num}(M) \\
\mathrm{num}(\mathsf{case}_i\, M N_0 \dots N_i) &= \mathrm{num}(M) + \mathrm{num}(N_0) + \dots + \mathrm{num}(N_i)
\end{aligned}$$

For every reduction rule $\to_c$, the interpretation by num of the right side is strictly less than the interpretation of the left side. Contexts are monotonic for num. Therefore $\to_c$ is strongly normalizing.

Remark: It would have been nice to show strong normalization for $\to_{\beta c}$, but this is not so easy, as there seems to be no theorem in the literature that would be immediately applicable to our case: Note that the $\to_c$-rules are not first-order, $\perp M \to_c \perp$ is higher-order; and that the if-case-rule is duplicating the variables Q and R. □

Lemma 4.13. *The input form is invariant under reductions $\to_{\beta c}$: If M is a PCF-i-term in input form and $M \to_{\beta c} M'$, then M' is also in input form.*

Proof. We have to prove:

(1) M' is of the form $M' = \lambda x_1 \dots x_n.\Phi_i^\iota M''$.

(2) Every ivo $x'|u'$ in M' is in an input region.
(3) Every occurrence $\mathsf{case}_i\,|u'$ in M' and every occurrence $\Phi_i^\iota|u'$ in M'' is in an input region.
Proposition (1) can be easily seen: The reduction rules $\to_{\beta c}$ do not change this form of M.

For the rest we need the notion of descendant and ancestor in reductions. In a reduction $M \to_{\beta c} M'$ a symbol occurrence in M that is not in the redex has the corresponding occurrences in M' as descendants. A symbol occurrence in M that is in a subterm corresponding to a term variable N of the rule's left side has the corresponding occurrences in M' that are determined by the Ns in the right side as descendants. (The symbol is the same or may change if a variable is renamed in β-reduction. This case is specified in more detail below.) There is a special treatment of the three rules for case_i: Here moreover the left occurrence of case_i has the right occurrence of case_i as descendant. If occurrence u has occurrence u' as descendant, then u is the ancestor of u'.

To prove proposition (2), we first have to prove:
(4) For every ivo $x'|u'$ in M' the ancestor $x|u$ in M is an ivo.

This is proved for each reduction rule separately by induction on the depth of $x'|u'$. The proofs are easy and boring; we only give here the proof for β-reduction $(\lambda y.P)L \to P[y := L]$, with the root of the redex at position r in M. We define the following regions of M and M' that correspond to each other:

region of M	region of M'
$\bar{M} = M$ without redex at r	$\bar{M}' = M'$ without subterm at r
$\bar{P} =$ subterm P of redex, without the free ys	$\bar{P}' =$ subterm of M' at r, without the substituted Ls
$\bar{L} =$ subterm L of redex	$\bar{L}'_j =$ the jth copy of the substituted L

Every position in one of the three regions of M has as descendant the corresponding position in the corresponding region(s) of M'.

Now we prove proposition (4) by induction on the depth of $x'|u'$:
Depth of $x'|u'$ is 1: The ancestor $x|u$ in M is also bound by the same outer λ, so it is an ivo.
Depth of $x'|u'$ is $d+1$:
$x'|u'$ has a parent $z'|v'$ of depth d in M'.
$z'|v'$ has an ancestor $z|v$ in M, which is an ivo by induction hypothesis. We have to prove that $x|u$ has $z|v$ as parent in M.
The input region of $z|v$ is entirely either in $\bar{M}$, $\bar{P}$ or $\bar{L}$. Then the region of $z'|v'$ is entirely in the corresponding $\bar{M}'$, $\bar{P}'$ or some $\bar{L}'_j$. Here are the three cases:

(a) If $z'|v'$ is in $\bar{M}'$, then $x'|u'$ is in $\bar{M}'$, $\bar{P}'$ or some $\bar{L}'_j$.
Since $z'|v'$ is parent of $x'|u'$, $x|u$ is in the same position relative to the input region of $z|v$, and $z|v$ is parent of $x|u$.
(b) If $z'|v'$ is in $\bar{P}'$, then $x'|u'$ is in $\bar{P}'$.
($x'|u'$ cannot be in some $\bar{L}'_j$, because a free variable x' in $\bar{L}'_j$ could not become bound in $\bar{P}'$ in the input region of $z'|v'$.) Then the relative position of the variables is again preserved by the reduction, and $z|v$ is parent of $x|u$.
(c) If $z'|v'$ is in some $\bar{L}'_j$, then $x'|u'$ is in $\bar{L}'_j$. Again $z|v$ is parent of $x|u$.

This ends the proof of proposition (4).

Now we can prove proposition (2) with the aid of (4) by inspection of each reduction rule, here only for β-reduction, with the naming of the regions above. The ancestor $x|u$ in M of $x'|u'$ is an ivo. Its input region is entirely in $\bar{M}$, $\bar{P}$ or $\bar{L}$. The reduction transfers this input region into the input region of $x'|u'$ in the corresponding $\bar{M}'$, $\bar{P}'$ or some $\bar{L}'_j$.

To prove proposition (3), we first have to prove a proposition that goes in the opposite direction to (4):
(5) For every occurrence $\mathsf{case}_i\,|u$ in M (in an input region), every descendant $\mathsf{case}_i\,|u'$ in M' is in an input region. (And the same for occurrencies of $\Phi_i^\iota|u$ in an input region.)

This is proved for each reduction rule separately by induction on the depth of the (input variable of the) input region in M. The induction step has to show that the pair of an input region and its parent is transfered by a reduction step to corresponding pairs. (The induction step of proposition (4) was in the opposite direction.) We omit this proof.

The proof of proposition (3) is then easy: $\mathsf{case}_i\,|u'$ in M' has an ancestor $\mathsf{case}_i\,|u$ in M, because we have defined the ancestors/descendants of the case-rules this way. And $\Phi_i^\iota|u'$ has an ancestor because no reduction rule creates a new occurrence of Φ_i^ι. $\square$

Lemma 4.14. *Every closed PCF-i-term M that is in normal form w.r.t. $\to_{\beta c}$ and in input form of grade i is a pre-game term of grade i.*

Proof. M is of the form $M = \lambda x_1 \dots x_n.\Phi_i^\iota M'$. We have to prove that M' is an integer-term of grade i. For this we show:

Every subterm occurrence $N\colon \iota$ in M' whose occurrences of free variables are input variable occurrences in M is an integer-term of grade i or it is a subterm occurrence of the form $y\dots$ or $\Phi_i^\iota\dots$ whose root is in an input region of M.

We prove this claim by induction on the term N. There are several cases for N, according to its leftmost subterm:
$N = \bot^\iota$ or $N = n$, n integer constant, are clear.
$N = \mathsf{case}_i\dots$: Then N has the form of an input region:

$$N = \mathsf{case}_i(y(\lambda x_{11}\dots x_{1k_1}.\Phi_i^\iota M_1)\dots(\lambda x_{m1}\dots x_{mk_m}.\Phi_i^\iota M_m))N_0\dots N_i$$

All free variable occurrencies of M_j are input variable occurrencies in M (as y is an ivo). Then by the induction hypothesis all M_j are integer-terms. Also the N_j. So N is an integer-term.
$N = (\lambda x.P)\dots$: Not possible because N is normal form, β-rule.
$N = \bot^{\sigma\to\tau}\dots$: Not possible because N is normal form, rule $\bot P \to_c \bot$.
$N = y\dots$: The occurrence of y must be an input variable occurrence, so it must appear in an input region.
$N = \Phi_i^\iota\dots$: The occurrence of Φ_i^ι must be in an input region.
$N = \mathsf{if}\dots$, $N = \mathsf{suc}\dots$ or $N = \mathsf{pre}\dots$:
We repeatedly go down the first argument of the if or suc or pre until we find at the bottom in N some subterm $\mathsf{if}\,N'PQ$, $\mathsf{suc}\,N'$ or $\mathsf{pre}\,N'$ with N' a term of one of the forms that are listed above for N.
The cases $N' = \bot^\iota$, $N' = n$, $N' = \mathsf{case}_i\dots$ are not possible because N is in normal form; for each of these cases there is a rule in $\to_c$.
The cases $N' = (\lambda x.P)\dots$ and $N' = \bot^{\sigma\to\tau}\dots$ are also not possible because N is normal form.
The cases $N' = y\dots$ and $N' = \Phi_i^\iota\dots$ are not possible because y and Φ_i^ι must occur in an

input region.
This proves the claim. From the claim we conclude that M' is an integer-term of grade i. $\square$

Lemma 4.15. *If $M\colon\sigma$ is a finite PCF-term of grade i without Y, then $\mathrm{nf}(\varphi_i^\sigma(M))$ is a pre-game term equivalent to M. (On the supposition that case_i and Φ_i^ι have the declared semantics.)*

Proof. $\varphi_i^\sigma(M)$ is in input form, therefore $\mathrm{nf}(\varphi_i^\sigma(M))$ is also in input form by lemma 4.13. By lemma 4.14 it is a pre-game term of grade i. It is $\Psi_i^\sigma M \cong M$, therefore $\varphi_i^\sigma(M) \cong M$. The reductions $\rightarrow_{\beta c}$ preserve the semantics, therefore $\mathrm{nf}(\varphi_i^\sigma(M)) \cong M$. $\square$

Now we still have to prove that our transformation so far respects the syntactic order on terms.

Lemma 4.16. *If $M \prec N$ are PCF-i-terms and $N \rightarrow_{\beta c} N'$, then simply $M \prec N'$, or there is M' with $M \rightarrow_{\beta c} M'$ and $M' \prec N'$.*

Proof. There are three possible cases for the left side region L of the redex in N:
(1) L is not in M, i.e. it is under some $\perp$-occurrence in M. Then simply $M \prec N'$.
(2) L is fully in M. Then a reduction with the same rule at the same position is done in M, $M \rightarrow_{\beta c} M'$, and it is $M' \prec N'$.
(3) L is partially in M. Then there is a rule with the partial left side region that can be applied in M, $M \rightarrow_{\beta c} M'$, and it is $M' \prec N'$. (Partial means: w.r.t. the syntactic order $\prec$.)

As an example for the last case take for $N \rightarrow_{\beta c} N'$ the rule

$$\mathsf{suc}(\mathsf{case}_i\, PN_0 \dots N_i) \rightarrow_c \mathsf{case}_i\, P(\mathsf{suc}\, N_0) \dots (\mathsf{suc}\, N_i)$$

and assume that of the left side region only $\mathsf{suc}\,\perp$ is in M. Then we use the rule $\mathsf{suc}\,\perp \rightarrow_c \perp$ in M. Please note that in this example the partial left side region could not be $\mathsf{suc}(\perp PN_0 \dots N_i)$, because case_i is not a constant. $\square$

Lemma 4.17. *If $M, N\colon\sigma$ are finite PCF-terms of grade i without Y, and $M \prec N$, then $\mathrm{nf}(\varphi_i^\sigma(M)) \prec \mathrm{nf}(\varphi_i^\sigma(N))$.*

Proof. We have seen that $\varphi_i^\sigma(M) \prec \varphi_i^\sigma(N)$. By the preceding lemma 4.16 there is some M' with $\varphi_i^\sigma(M) \rightarrow_{\beta c}^* M'$ and $M' \prec N' = \mathrm{nf}(\varphi_i^\sigma(N))$. We prove that M' is also a $\rightarrow_{\beta c}$-normal form. One can check for all reduction rules except the last one ($\perp P \rightarrow_c \perp$): If M' contains the left side region, then N' contains it too (except occurrencies of $\perp$). In each case this is not possible because N' is a pre-game term.

Now assume that M' contains a redex $\perp P$. Then N' must have an application at the same position, and this is only possible in the first argument of a case_i, in an input region. Back in M': The redex must also occur in a case_i-expression $\mathsf{case}_i(\perp PM_1 \dots M_m)N_0 \dots N_i$. But this is not possible, as M' is in input form. (Note again that $\Phi_i^\iota \dots$ and $\mathsf{case}_i \dots$ are constructs.) This proves that M' is in normal form, $M' = \mathrm{nf}(\varphi_i^\sigma(M))$. $\square$

For our example we get:

$$\begin{aligned}\mathrm{nf}(\varphi_0^\sigma(X)) = \lambda x_1^{\sigma_1} x_2^{\sigma_2}.\Phi_0^\iota[\,&\mathsf{case}_0(x_2^{\sigma_2}\mathsf{0})(\mathsf{case}_0(x_1^{\sigma_1}(\lambda x_1^\iota.\Phi_0^\iota(\mathsf{case}_0\, x_1^\iota\\ &(\mathsf{case}_0(x_1^{\sigma_1}(\lambda x_1^\iota.\Phi_0^\iota(\mathsf{case}_0(x_2^{\sigma_2}(\Phi_0^\iota(\mathsf{case}_0\, x_1^\iota \mathsf{0})))\mathsf{0})))\mathsf{0})\\ &)))\mathsf{0})]\end{aligned}$$

Third transformation step.

In the third transformation step we let the Φ_i^ι spread over the case_i-expressions and filter the integers, and we replace the η-expanded $\perp$s. We first apply these reduction rules to a pre-game term until we get a normal form.

$$\begin{aligned}
\Phi_i^\iota(\mathsf{case}_i\, M N_0 \dots N_i) &\to \mathsf{case}_i\, M(\Phi_i^\iota N_0)\dots(\Phi_i^\iota N_i)\\
\Phi_i^\iota \perp^\iota &\to \perp^\iota\\
\Phi_i^\iota n &\to n,\ \text{for integer constant } n \le i\\
\Phi_i^\iota n &\to \perp,\ \text{for integer constant } n > i
\end{aligned}$$

Then we apply this rule to the normal form in all possible places:

$$\lambda x_1 \dots x_n. \perp^\iota \to \perp,\ \text{with maximal number of variables } x_j$$

The result of the whole transformation on pre-game term M is called filter(M). If M is a pre-game term of grade i, then filter(M) is a game term of grade i with filter$(M) \cong M$. If $M \prec N$ are pre-game terms of grade i, then filter$(M) \prec$ filter(N). We define for a finite PCF-term $M\colon \sigma$ of grade i without Y:

$$\mathrm{gt}_i^\sigma(M) = \mathrm{filter}(\mathrm{nf}(\varphi_i^\sigma(M)))$$

As final result we get the game term lemma above, and from this the game term theorem.

In our example Φ_0^ι just walks through the term and we get the result in our graphical notation:

$$\begin{array}{llllll}
\mathrm{gt}_0^\sigma(X) = \mathrm{filter}(\mathrm{nf}(\varphi_0^\sigma(X))) = \lambda x_1^{\sigma_1} x_2^{\sigma_2}. & x_2^{\sigma_2} - & x_1^{\sigma_1} \text{ --- } \mathsf{0} & & & \\
& | & | & & & \\
& \mathsf{0} & \lambda x_1^\iota.\, x_1^\iota - & x_1^{\sigma_1} \text{ --- } \mathsf{0} & & \\
& & & | & & \\
& & & \lambda x_1^\iota.\, & x_2^{\sigma_2} \text{ --- } \mathsf{0} & \\
& & & & | & \\
& & & & x_1^\iota \text{ --- } \mathsf{0} &
\end{array}$$

4.3. **Infinite game terms.** We will need infinite game terms to fully formulate Berry's conjectures for first-order types, where they become true. We briefly introduce them here. First we generalize the syntactic order $\prec$ to terms with differing case-expressions:

$$\mathsf{case}_i\, M N_0 \dots N_i \prec \mathsf{case}_j\, M' N_0' \dots N_j' \text{ iff } i \le j \text{ and } M \prec M' \text{ and } N_n \prec N_n' \text{ for } 0 \le n \le i.$$

This definition comes out when the normal $\prec$ is applied to the macro expansions of the two case-expressions as PCF-terms.

Definition 4.18. 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.) The order $\prec$ on infinite game terms is the subset order of the ideals. The semantics 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.

For our infinite game term theorem we can use the machinery of subsection 4.2. We only need to adapt lemma 4.16 and lemma 4.17 to terms of different grade and to the new cases of the syntactic order of case-expressions.

For our auxiliary constructs we postulate:

$$\Phi_i^\sigma M \prec \Phi_j^\sigma N \text{ and } \bar{\Phi}_i^\sigma M \prec \bar{\Phi}_j^\sigma N \text{ for } i \le j \text{ and } M \prec N.$$

Then we get $\varphi_i^\sigma(M) \prec \varphi_j^\sigma(N)$.

Lemma 4.19. *If $i \leq j$, M is a PCF-i-term, N is a PCF-j-term, $M \prec N$ and $N \to_{\beta c} N'$, then simply $M \prec N'$, or there is M' with $M \to_{\beta c} M'$ and $M' \prec N'$.*

Proof. The proof is literally the same as the proof of lemma 4.16. Only in case (3) there are new partial left side regions for the reduction rules with case; e.g. $\mathsf{suc}(\mathsf{case}_i\, PN_0 \dots N_i) \prec \mathsf{suc}(\mathsf{case}_j\, P'N'_0 \dots N'_j)$. □

Lemma 4.20. *If $i \leq j$ and $M, N\colon \sigma$ are finite PCF-terms of grade i resp. j without Y, and $M \prec N$, then $\mathrm{nf}(\varphi_i^\sigma(M)) \prec \mathrm{nf}(\varphi_j^\sigma(N))$.*

Proof. The proof is literally the same as the proof of lemma 4.17. Only difference: If M' contains a left side region like $\mathsf{suc}(\mathsf{case}_i\, PN_0 \dots N_i)$, then N' contains at the same position the larger region $\mathsf{suc}(\mathsf{case}_j\, P'N'_0 \dots N'_j)$. □

Lemma 4.21. *If $i \leq j$ and $M, N\colon \sigma$ are finite PCF-terms of grade i resp. j without Y, and $M \prec N$, then $\mathrm{gt}_i^\sigma(M) \prec \mathrm{gt}_j^\sigma(N)$, $\mathrm{gt}_i^\sigma(M) \cong M$, $\mathrm{gt}_j^\sigma(N) \cong N$, and $\mathrm{gt}_i^\sigma(M), \mathrm{gt}_j^\sigma(N)$ are game terms of grade i resp. j.*

Proof. This follows from the game term lemma in the preceding subsection, from the preceding lemma, and from $\mathrm{filter}(M') \prec \mathrm{filter}(N')$ for $M' \prec N'$ with M', N' pre-game terms of ascending grade. □

Theorem 4.22. *For every closed PCF-term there is an infinite game term that has an equivalent semantics in all f-models.*

Proof. Let M be a closed PCF-term of type σ. By the approximation continuity theorem (see the proof of lemma 4.5), there is a sequence of terms $(M_i)_{i \geq 0}$ without Y such that $M_i \prec M_{i+1}$ and $[\![M_i]\!]_{i \geq 0} \to [\![M]\!]$ holds in all f-models.

Then for all i: $\Psi_i^\sigma M_i$ is a finite term, $\Psi_i^\sigma M_i \prec \Psi_{i+1}^\sigma M_{i+1}$ and $[\![\Psi_i^\sigma M_i]\!]_{i \geq 0} \to [\![M]\!]$. By the preceding lemma we get $\mathrm{gt}_i^\sigma(\Psi_i^\sigma M_i) \cong \Psi_i^\sigma M_i$ and $\mathrm{gt}_i^\sigma(\Psi_i^\sigma M_i) \prec \mathrm{gt}_{i+1}^\sigma(\Psi_{i+1}^\sigma M_{i+1})$. So this is an ascending sequence of game terms with the semantics of M. □

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. *Modèles complètement adéquats et stables des lambda-calculs typés.* PhD thesis, Université Paris VII, 1979.

[4] 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.

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

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

[7] 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`.