Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Theoretical Computer Science 173 (2007) 203–219
Electronic Notes in Theoretical Computer Science
www.elsevier.com/locate/entcs

Sequentiality and the CPS Semantics of Fresh Names

J. Laird[1,2]

Dept. of Informatics
University of Sussex
UK

Abstract

We investigate the domain-theoretic denotational semantics of a CPS calculus with fresh name declaration. This is the target of a fully abstract CPS translation from the nu-calculus with first-class continuations. We describe a notion of "FM-categorical" model for our calculus, with a simple interpretation of name generation due to Shinwell and Pitts. We show that full abstraction fails (at order two) in the simplest instance of such a model (FM-cpos) because of the presence of parallel elements. Accordingly, we define a sequential model — FM-biorders, based on "bistable FM-bicpos" and bistable functions — and prove that it is fully abstract up to order four (our main result), but that full abstraction fails at order five.

Keywords: Continuation-passing style, Fresh names, Sequentiality, FM-sets, Full Abstraction.

1 Introduction

This paper is a study of the denotational semantics of freshly generated names in a continuation-passing style (CPS) setting. Freshly generated names are a key element of many computational effects, and are also intrinsically interesting; they may be used to represent secrets such as cryptographic keys, for example. The behaviour of names in a functional setting is rather subtle, and has been studied, via prototypical calculi such as the nu-calculus [7], using both operational and denotational techniques.

An approach to the denotational semantics of naming which has been advocated by Pitts and others is via Fraenckel-Mostowski (FM) set-theory. In essence, a FM-set is an action of the group of permutations of a countable set of atoms (representing names) in which each element of the carrier set depends on only finitely many atoms (its *support*): if we interpret terms as elements of the carrier set then the

[1] Supported by UK EPSRC grant GR/S72181
[2] Email: jiml@sussex.ac.uk

doi:10.1016/j.entcs.2007.02.035

group action corresponds to substitution of one name for another. So, for example, we may interpret the type of names as the FM-set (or flat FM-cpo) N of natural numbers with the canonical permutation group action. Shinwell and Pitts [6] have observed that there is a simple and natural continuation-passing-style interpretation of name generation in such a setting: if we take the CPS "answer-object" to be the two-point set (or order) $\Sigma = \{\top, \bot\}$ then new-name generation should correspond to a function from Σ^N into Σ. Any element of Σ^N — a function from N into Σ — must take the same value on every argument which is not in its support, so the result of supplying a new name to f must be that value.

The limitation of this approach, when confined to the FM-cpo model may be seen when we consider the questions of universality (what "junk" does it contain?) and full abstraction (which equivalences does it reflect accurately?). The presence of parallel elements in the model, but not the language, means that full abstraction will fail as for PCF, moreover some of the simplest equivalences between terms which create and generate new names are broken. In particular, we have a function $p : \Sigma^N \to \Sigma$ such that $p(f) = \top$ if there exists n such that $f(n) = \top$, effectively allowing the names recognized by f to be "guessed".

The main goal of this paper is to investigate a *sequential* CPS model of naming, obtained by working in a category of bistable "FM-bicpos". These are obtained simply by extending FM-cpos with additional structure such that functions which preserve that structure (bistable functions) are strongly sequential. Bistable Bicpos were introduced in [2,4], where they were shown to give a fully abstract model of a functional language with control operators (SPCF). Here we shall give the semantics of a CPS *target* language with a name generating operation. There is a fully abstract translation into this calculus from (for example) the nu-calculus extended with first-class continuations.

It is straightforward to show that our calculus may be soundly interpreted in any "FM-category" with finite products and Σ-exponentials, and satisfying some simple axioms. We then study the completeness properties of the bistable bicpo interpretation, showing that full abstraction holds for terms up to fourth order (i.e. terms with types containing up to four nestings of the continuation $\neg$-operator). However, full abstraction fails at fifth order, showing that even in this extensional setting, sequentiality is not sufficient to obtain full abstraction.

1.1 Related Work

Several models for functional languages with freshly generated names have been proposed, including (equivalent) functor category and FM-set models for the nu-calculus described by Stark in [7], the FM-cpo CPS models of mini-FreshML [6], and games models [3] and [1], also based on nominal sets. Both extensional and intensional models give limited information about equivalence in a functional setting with fresh names, in the extensional cases, because full abstraction fails at low-level types, in the more intensional (games) models because full abstraction depends either on a quotient, or on allowing names to be leaked through the store. Although our semantics is not fully abstract, it does capture a substantial frag-

$$\frac{}{\Gamma\vdash v{:}B}\, v \in \{tt, ff\} \qquad \frac{}{\Gamma\vdash n{:}N}\, n \in \mathbb{N} \qquad \frac{}{\Gamma\vdash C{:}\Sigma}\, C \in \{\top, \bot\}$$

$$\frac{\Gamma\vdash s{:}N \qquad \Gamma\vdash t{:}N}{\Gamma\vdash s{=}t{:}B} \qquad \frac{\Gamma\vdash s{:}B \qquad \Gamma\vdash t{:}B}{\Gamma\vdash \mathsf{bop}(s,t){:}B}$$

$$\frac{}{\Gamma,x{:}T\vdash T} \qquad \frac{\Gamma\vdash s{:}\neg P \qquad \Gamma\vdash t{:}P}{s\,t{:}\Sigma}$$

$$\frac{}{\Gamma\vdash \mathtt{new}{:}\neg\neg N} \qquad \frac{\Gamma\vdash r{:}B \qquad \Gamma\vdash s{:}\Sigma \qquad \Gamma\vdash t{:}\Sigma}{\mathsf{If}\ r\ \mathsf{then}\ s\ \mathsf{else}\ t{:}\Sigma}$$

$$\frac{\Gamma\vdash s_i{:}T_i\ :\ i{<}n}{\Gamma\vdash \langle s_i \mid i{<}n\rangle{:}\Pi_{i<n}T_i} \qquad \frac{\Gamma,x_0{:}T_1,\ldots,x_{n-1}{:}T_{n-1}\vdash s{:}\Sigma}{\Gamma\vdash \lambda(x_0,\ldots,x_{n-1}).s{:}\neg\Pi_{i<n}T_i}$$

Table 1
Term-Formation for the CPS-nu Calculus

ment of functional naming without requiring additional powerful techniques such as logical relations.

2 A CPS calculus with name-generation

The CPS-nu-calculus is in essence a simply-typed λ-calculus in which terms take either a basic "answer-type" Σ, a value type — B (boolean) and N (names) and continuations — or a product of value types. The value and product types are given by the following grammar:

$$T ::= B \mid N \mid \neg P \qquad\qquad P ::= \Pi_{i<n}T_i$$

Value types are included in the product types as unary products; we write U for the empty product (unit type) and define $(\Pi_{i<m}S_i)\times(\Pi_{i<m}T_i)$ to be $\Pi_{i<m+n}R_i$, where $R_i = S_i$ for $i < m$ and $R_i = T_{i+m}$ for $i > m$. The *order* of a type is its continuation-nesting depth — so N, B have order 0, $\Pi_{i<n}T_i$ has order $\max\{o(T_i) \mid i \leq n\}$ and $\neg P$ has order $o(P)+1$.

Terms are formed over contexts of variables of value type by λ-abstraction and application, together with: constants $\top$ (error) and $\bot$ (divergence) at return type, a set of names (constants) $\{n \mid n \in \mathbb{N}\}$, a conditional, a new-name generator $\mathtt{new} : \neg\neg N$ and boolean expresions formed from tt, ff and equality testing on names. We write $\nu x.s$ for $\mathtt{new}\,\lambda x.s$. The term-formation/typing rules are given in Table 1.

2.1 Operational Semantics

The operational semantics (Table 2) is given by a termination predicate, $\Downarrow$, on pairs s, k of a term s of return type and a value k such that every name occurring in s is less than k. Note that a closed term $s : B$ is simply a formula of propositional logic

$$\frac{}{\top,k\Downarrow} \qquad \frac{s\,\langle k\rangle,k{+}1\Downarrow}{\mathtt{new}\,\langle s\rangle,k\Downarrow} \qquad \frac{s[\vec{t}/\vec{x}],k\Downarrow}{(\lambda(\vec{x}).s)\,\langle\vec{t}\rangle,k\Downarrow}$$

$$\frac{s,k\Downarrow}{\mathsf{If}\ r\ \mathsf{then}\ s\ \mathsf{else}\ t,k\Downarrow}\ |r| = tt \qquad \frac{t,k\Downarrow}{\mathsf{If}\ r\ \mathsf{then}\ s\ \mathsf{else}\ t,k\Downarrow}\ |r| = ff$$

Table 2
Operational Semantics of the CPS-nu-calculus

over a set of atoms consisting of equality statements $n = m$ for natural numbers n, m, and thus has a standard interpretation as a boolean value $|s|$. We write $s \Downarrow$ if there exists k such that $s, k \Downarrow$ and define standard notions of observational approximation — $s \lesssim t$ if for any compatible context $C[\_]$ of return type, $C[s] \Downarrow$ implies $C[t] \Downarrow$ — and equivalence — $s \approx t$ if $s \lesssim t$ and $t \lesssim s$.

Lemma 2.1 (Context Lemma) *For any terms $s, t : \neg P$, $s \lesssim t$ if and only if for all terms $L : P$, $M\,L \Downarrow$ implies $N\,L \Downarrow$.*

Proof. Follows the standard proof for e.g. PCF [5]. □

Lemma 2.2 *If s, t are terms without explicit names then $s \lesssim t$ if for any name-free compatible context $C[\_]$, $C[s] \Downarrow$ implies $C[t] \Downarrow$.*

Proof. Given name-free terms which are distinguished by a context $C[\_]$, we may obtain a distinguishing context by replacing all explicit names in $C[\_]$ by variables and declaring them with `new`. □

As in the nu-calculus there are some useful examples of equivalences in our calculus which capture aspects of name generation in the calculus, and can be used to test any candidate models. These are related to nu-calculus equivalences considered by Stark [7], although they are not CPS translates of the latter. The two key instances of such equivalences are:

(i) At the type $\neg\neg\neg N$: $\lambda\kappa.\nu n.\kappa\,\lambda y.\mathsf{If}\ x = n\ \mathsf{then}\ \top\ \mathsf{else}\bot \simeq \lambda y.\bot$.

(ii) At the type $\neg\neg\neg\neg(N \times \neg U)$:
$\lambda k.\nu n.k\,\langle\lambda f.\nu m.f\,\langle m, \lambda a.f\,\langle n, \bot\rangle\rangle\rangle \simeq \lambda k.k\,\langle\lambda f.\nu m.f\,\langle m, \bot\rangle\rangle$

Informally, (1) holds because any argument of type $\neg\neg N$ supplied must apply its argment to a name which is not equal to n. (2) holds because any argument supplied cannot "recognize" n and therefore cannot apply $\lambda f.\nu\, m.f\,\langle m, \lambda a.f\,\langle n, \bot\rangle\rangle$ to an argument which can distinguish n from the fresh name m. We will prove both equivalences formally, using semantic means.

$$\overline{s = t} = \lambda\kappa.\overline{s}\,\langle\lambda a.\overline{t}\,\langle\lambda b.\kappa\,\langle a = b\rangle\rangle\rangle \qquad \overline{v} = \lambda\kappa.\kappa\,\langle v\rangle \quad v \in \{tt, ff\}$$

$$\overline{\lambda x.s} = \lambda\kappa.\kappa\,\langle\lambda(x,z).\overline{s}\,\langle z\rangle\rangle \qquad \overline{s\,t} = \lambda\kappa.\overline{s}\,\lambda(a).\overline{t}\,\lambda b.a\,\langle b,\kappa\rangle$$

$$\overline{\mathsf{If}\ s\ \mathsf{then}\ t_1\ \mathsf{else}\ t_2} = \lambda\kappa.\overline{s}\,\langle\lambda a.\mathsf{If}\ a\ \mathsf{then}\ \overline{t_1}\,\langle\kappa\rangle\ \mathsf{else}\ \overline{t_2}\,\langle\kappa\rangle\rangle \qquad \overline{x} = \lambda\kappa.\kappa\,\langle x\rangle.$$

$$\overline{\texttt{call/cc}\ s} = \lambda\kappa.\overline{s}\,\langle\lambda a.a\,\langle\lambda(b,c).\kappa\,\langle b\rangle, \kappa\rangle \qquad \overline{\texttt{new}} = \texttt{new}$$

Table 3
CPS Translation of the control nu-calculus

2.2 CPS Translation

We may use the CPS-nu-calculus to interpret languages such as the nu-calculus by CPS translation. Here we shall give such a (fully abstract) translation for the nu-calculus extended with first-class continuations. Our source calculus is a simply-typed, call-by-value λ-calculus over the base types ν, o (names and booleans), with constants: $tt : o$ and $ff : o$, equality testing of names, a conditional, new-name generation $\texttt{new} : \nu$ and call-with-current-continuation $\texttt{call/cc} : ((T \Rightarrow S) \Rightarrow T) \Rightarrow T$.

We may interpret this by a standard CPS translation $\overline{(\_)}$ into the CPS-nu-calculus, sending each base type to the corresponding value type and $S \Rightarrow T$ to $\neg(\overline{S} \times \neg\overline{T})$. The translation taking terms-in-context $x_1 : S_1, \ldots, x_n : S_n \vdash M : T$ to $x_1 : \overline{S_1}, \ldots, x_n : \overline{S_n} \vdash M : \neg(\neg T)$ is given in Table 3. For closed terms $s : o$, we may write $s \Downarrow tt$ if $\overline{s}\,\langle\lambda b.\mathsf{If}\ b\ \mathsf{then}\ \top\ \mathsf{else}\ \bot\rangle, 0 \Downarrow$ — it is straightforward to show that this agrees with the standard operational semantics for terms of the nu-calculus. Thus we may derive a notion of observational equivalence for terms of the control nu-calculus, with respect to which CPS translation is sound by definition; it is also complete and thus fully abstract (Corollary 3.7).

3 CPS Semantics of Naming

The CPS interpetation of name generation introduced by Shinwell and Pitts [6] was sketched in the introduction. Here, we give a more formal and general account of the semantics of our CPS calculus in any "FM-enriched" category with the requisite structure.

For a countable set of "atoms" X, let G be the topological group of automorphisms on X, with the product topology. An action of G on a set A is continuous (with respect to the discrete topology on A) if and only if for every element $a \in A$, there is a finite subset $k \subseteq X$ such that $\pi(x) = x$ for all $x \in k$ implies $\pi \cdot a = a$. Let $\nu(a)$, the support of a, be the least such subset. A FM-set $(A, \cdot)$ is a set A with a continuous G-action on it, a FM-order is a FM-set $(A, \cdot)$ with a partial order on A such that $x \leq y$ iff $\pi \cdot x \leq \pi \cdot y$. A FM-order D is a FM-cpo if every directed set X with bounded support (i.e. such that $\bigcup\{\nu(x) \mid x \in X\}$ is finite) has a least upper bound.

A FM-category is a category $\mathcal{C}$ enriched with a continuous G-action — i.e. every hom-set is a FM-set — such that $(\pi \cdot f); (\pi \cdot g) = \pi \cdot (f; g)$ and $\pi \cdot \mathsf{id} = \mathsf{id}$. A morphism

f is *invariant* if $\nu(f) = \varnothing$. The basic example of a FM-category is that of FM-sets (i.e. objects are FM-sets, morphisms from A to B are functions $f : A \to B$ with $\pi \cdot f$ defined $(\pi \cdot f)(a) = \pi \cdot (f(\pi^{-1} \cdot a))$. Similarly, we have FM-categories of FM-orders and monotone functions and FM-cpos and continuous functions. Note that all of these categories are Cartesian closed. We may also construct FM-categories of games and strategies [3,1].

Henceforth we shall take the set X of atoms to be $\mathbb{N}$. Let $\mathcal{C}$ be a FM-category with finite products. We shall say that $\mathcal{C}$ has *boolean*, *naming* and Σ-objects if it contains objects B, N, Σ such that:

- B is a disjoint coproduct of the terminal object with itself.
- $\mathcal{C}(1, N)$ consists of distinct maps $\{\overline{i} \mid i \in \mathbb{N}\}$ such that $\pi \cdot \overline{i} = \overline{\pi(i)}$, and there is a "decidable equality" morphism $\mathsf{eq} : N \times N \to B$ such that $\langle \overline{n}, \overline{m} \rangle; \mathsf{eq} = \mathtt{in}_1(*)$ if and only if $n = m$.
- $\mathcal{C}(1, \Sigma)$ consists of distinct, invariant maps $\{\bot, \top\}$ such that $f : N \to \Sigma = g : N \to \Sigma$ if and only if $\overline{n}; f = \overline{n}; g$ for all n.

In the category of FM-sets, B and Σ are just two-point sets and N is the set of natural numbers with canonical G-action. In the category of FM-cpos, B and N have the discrete order and $\bot \sqsubseteq \top$ in Σ. The interpretation of the new-name generation constant new, is derived from the following observation.

Lemma 3.1 *Given $f : N \to \Sigma$ and $\overline{m}, \overline{n} : 1 \to N$, $\overline{m}; f = \overline{n}; f$ if and only if $m \in \nu(f) \iff n \in \nu(f)$.*

Proof. Suppose without loss of generality that $\overline{m}; f = \top$ for some $m \notin \nu(f)$. Let $[m \leftrightarrow n] \in G$ be the automorphism which swaps m and n and leaves all other points alone. If $n \notin \nu(f)$ then $[m \leftrightarrow n]$ is in the stabilizer of f so $\overline{m}; f = ([n \leftrightarrow m] \cdot \overline{n}); f = \overline{n}; ([n \leftrightarrow m] \cdot f) = \overline{n}; f = \top$. Hence $\{l \mid \overline{l}; f = \bot\} \subseteq \nu(f)$.

Given $\pi \in G$, suppose $\pi \cdot \overline{n} = n$ for all n such that $\overline{n}; f = \bot$. Then for any n, if $\overline{n}; f = \bot$ then $\overline{n}; (\pi \cdot f) = (\pi^{-1} \cdot \overline{n}); f = \overline{n}; f = \bot$ and if $\overline{n}; (\pi \cdot f) = (\pi^{-1} \cdot \overline{n}); f = \bot$ then $\pi(\pi^{-1}(n)) = n = \pi(n)$ and so $\overline{n}; f = \bot$. Thus π is in the stabilizer of f and so $\nu(f) \subseteq \{n \mid \overline{n}; f = \bot\}$ and $\overline{m}; f = \overline{n}; f$ if and only if $m \in \nu(f)$ iff $n \in \nu(f)$ as required. □

Definition 3.2 Suppose $\mathcal{C}$ is an FM-category with Σ-exponentials — i.e. a Σ-object, and for any object A, an exponential Σ^A of Σ by A. (So each morphism $f : N \to \Sigma$ has a "name" $\ulcorner f \urcorner : 1 \to \Sigma^A$.) Let ϵ be a choice function on $\mathcal{P}(\mathbb{N})$. A morphism $\mathsf{new} : \Sigma^N \to \Sigma$ in $\mathcal{C}$ is a *name generator* if for any $f : N \to \Sigma$, $\ulcorner f \urcorner; \mathsf{new} = \overline{\epsilon(\nu(f)^c)}; f$.

By Lemma 3.1, $\ulcorner f \urcorner; \mathsf{new} = \overline{n}; f$ if and only if $n \notin \nu(f)$. Since each $f : 1 \to \Sigma^N$ has finite support, $\{x \mid f(x) = \top\}$ is thus either finite or co-finite, and we have $\mathsf{new}; \ulcorner f \urcorner = \top$ if and only if $\{n \mid \overline{n}; f = \bot\}$ is finite. Thus in the category of FM-sets we may define a name generator: $\mathsf{new}(f) = \top$ if and only if $\{n \mid \overline{n}; f = \bot\}$ is finite. This is also a well-defined morphism in the categories of FM-orders and FM-cpos: to show that it is bounded continuous: suppose $F \subseteq \Sigma^N$ is directed and

has bounded support, and suppose $\mathsf{new}(f) = \bot$ for all $f \in F$. Then $(\bigsqcup F)(x) = \top$ iff there exists $f \in F$ such that $f(x) = \top$ iff there exists $f \in F$ such that $x \in \nu(f)$. Hence by boundedness of the support of F, $(\bigsqcup F)(x) = \top$ for finitely many x and so $\mathsf{new}(\bigsqcup F) = \bot$ as required.

Given a FM-category with boolean and naming objects, finite products and Σ-exponentials, a name-generator and decidable equality, let $[\![B]\!] = B$, $[\![N]\!] = N$, $[\![\neg T]\!] = \Sigma^{[\![T]\!]}$ and $[\![\Pi_{i<n} T_i]\!] = \Pi_{i<n} [\![P_i]\!]$. The categorical structure yields direct interpretations of the operations and constants, and a straightforward proof of soundness with respect to the operational semantics. We establish the following by a simple structural induction.

Lemma 3.3 *If $n \in \nu([\![s]\!])$ then n occurs in s.*

Proposition 3.4 (Soundness) *For any closed term, if $s, k \Downarrow$ then $[\![s]\!] = \top$.*

Proof. By induction on derivation. For the `new` rule, suppose $s\,\langle \underline{k} \rangle, k+1 \Downarrow$ and so $\langle [\![s]\!], \overline{k} \rangle; \mathsf{app} = \top$ by induction hypothesis. By assumption, k does not occur in s, and so $k \notin \nu([\![s]\!])$ by Lemma 3.3, and so $[\![\mathtt{new}\, s]\!] = [\![s]\!]; \mathsf{new} = \langle [\![s]\!], \overline{k} \rangle; \mathsf{app} = \top$ as required. □

Proposition 3.5 (Adequacy) *If $[\![s]\!] = \top$ then $s \Downarrow$.*

Proof. By a standard computability predicate argument. □

Thus any FM-categorical model of the calculus is equationally sound — $[\![s]\!] = [\![t]\!]$ implies $s \approx t$ — and any FM-order-enriched model (with $\bot \leq \top$) is inequationally sound — $[\![s]\!] \sqsubseteq [\![t]\!]$ implies $s \lesssim t$. As a first application of our semantics, we may use it to prove that the CPS translation of the control nu-calculus is fully abstract.

Proposition 3.6 *For every name-free term $s : \overline{T}$ of the CPS-nu-calculus there exists a term $\widehat{s} : T$ of the control nu-calculus (extended with constants $\top, \bot : o$) such that $[\![\overline{\widehat{s}}]\!] = [\![s]\!]$.*

Proof. We prove by induction on length that for any β-normal term $x_1 : \overline{T_1}, \ldots, x_m : \overline{T_m}, y_1 : \neg\overline{S_1}, \ldots, y_n : \neg\overline{S_m} \vdash s : \Sigma$ there is a term $x_1 : T_1, \ldots, x_m : T_m, y_1 : S_1 \to B, \ldots, y_n : S_n \to B \vdash \widehat{s} : B$ such that $[\![\overline{\widehat{s}}]\!](\bot) = [\![s[\lambda(a).z_1\,\langle a, \lambda b.\bot\rangle/y_1] \ldots [\lambda(a).z_1\,\langle a, \lambda b.\bot\rangle/y_1]]\!]$, and for any β-normal term of value type $x_1 : \overline{T_1}, \ldots, x_m : \overline{T_m}, y_1 : \neg\overline{S_1}, \ldots, y_n : \neg\overline{S_m} \vdash s : \overline{R}$ there is a term $x_1 : T_1, \ldots, x_m : T_m, y_1 : S_1 \to B, \ldots, y_n : S_n \to B \vdash \widehat{s} : R$ such that $[\![\overline{\widehat{s}}]\!] = [\![\lambda\,\kappa.\kappa\,\langle s[\lambda(a).z_1\,\langle a, \lambda b.\bot\rangle/y_1] \ldots [\lambda(a).z_1\,\langle a, \lambda b.\bot\rangle/y_1]\rangle]\!]$.

For example:

- Suppose $s = x\,\langle t_1, \lambda a.t_2\rangle$, where $x : \overline{S \to T} = \neg(\overline{S} \times \neg\overline{T})$. Then $\widehat{s} = (\lambda a.\widehat{t_2})\,(x_i\,\widehat{t_1})$.
- Suppose $s = y\,\langle t\rangle$, where $y : \neg\overline{S}$. Then $\widehat{s} = y\,\overline{t}$.
- Suppose $s = \lambda(x, y).t : \overline{S \to T} = \neg(\overline{S} \times \neg\overline{T})$. Then $\widehat{s} = \lambda x.\mathtt{call/cc}\,\lambda y.((\lambda k.\widehat{t})\,\bot)$.

□

Corollary 3.7 *CPS translation from the control nu-calculus into CPS-nu-calculus is fully abstract.*

Proof. Given closed terms $s, t : T$ of the control nu-calculus, suppose $\overline{s} \not\approx \overline{t}$. Then there exists a term $\lambda k.r : \neg\overline{T}$ such that $\overline{s}\,\lambda k.r \Downarrow$ and $\overline{t}\,\lambda k.r \Downarrow$. Let $v : T \to o = \lambda k.\mathtt{call/cc}\,\lambda f.\widehat{r}[f\ tt/\top, (f\ ff)/\bot]$ Then $[\![\mathsf{If}\ \overline{v\,s}\ \mathsf{then}\top\ \mathsf{else}\ \bot]\!] = [\![\overline{s}\,\lambda k.r]\!] = \top$ and $[\![\mathsf{If}\ \overline{v\,t}\ \mathsf{then}\top\ \mathsf{else}\ \bot]\!] = [\![\overline{s}\,\lambda k.r]\!] = \bot$ and so $v\,s \Downarrow tt$ and $v\,t \not\Downarrow tt$ as required. □

Having defined an (in)equationally sound semantics for the CPS-nu-calculus in a general class of models, we may ask: for which types are our models complete with respect to observational equivalence? If we consider the model consisting of FM-cpos and bounded continuous functions, then completeness must fail for the fragment of the calculus over just the boolean and continuation types, since the model contains parallel elements. However, the non-sequential character of the model also means that it fails to accurately reflect equivalences specifically related to name generating behaviour. For instance, the only contextual equivalence classes of closed terms of the language at the type $\neg\neg N$ are new, $\lambda\kappa.\top$, $\lambda\kappa.\bot$, and $\lambda\kappa.\kappa\,n$ for each name n. However, in the FM-cpo model, there are many more functions from Σ^N into Σ — e.g. p defined by $p(f) = \top$ if there exists n such that $f(n) = \top$. This is sufficient to break equivalence (1) between $F = \lambda\kappa.\nu\,n.\kappa\,\lambda x.\mathsf{If}\ (x = n)\ \mathsf{then}\top\ \mathsf{else}\ \bot$ and $\lambda x.\bot$ — $F(p) = \nu\,n.p(\lambda x.\mathsf{If}\ (x = n)\ \mathsf{then}\top\ \mathsf{else}\ \bot) = \nu n.\top = \top$ and $\bot(p) = \bot$. The key to defining a model which reflects naming more accurately would therefore seem to be to capture the sequential nature of the calculus fully, and this is what we aim to achieve, using *bistable biorders*, in the remainder of the paper.

4 FM-biorders

Amongst possible equivalent definitions of bistable biorder, we give the following:

Definition 4.1 A (bistable) biorder [2,4] is a tuple $(D, \sqsubseteq, \updownarrow)$, where $(D, \sqsubseteq)$ is a partial order (the *extensional* order), and $\updownarrow$ is an equivalence relation (*bistable coherence*) on D such that each $\updownarrow$-equivalence class is a *distributive lattice* with respect to $\sqsubseteq$, and inclusion into D preserves binary meets and joins.

Definition 4.2 A FM-biorder is a tuple $(D, \sqsubseteq, \updownarrow, \cdot)$ where $(D, \sqsubseteq, \updownarrow)$ is a bistable biorder, $(D, \sqsubseteq, \cdot)$ is a FM-order and for every $d, e \in D$, $d \updownarrow e$ implies $\pi \cdot d \updownarrow \pi \cdot e$.

D is a *FM-bicpo* if $(D, \sqsubseteq, \cdot)$ is a FM-cpo and if $X, Y \subseteq D$ are directed sets with bounded support such that $X \updownarrow Y$ (i.e. for all $x \in X$ and $y \in Y$ there exists $x' \in X, y' \in Y$ such that $x \sqsubseteq x', y \sqsubseteq y'$ and $x' \updownarrow y'$) then $\bigsqcup\{x \wedge y \mid x \in X \wedge y \in Y \wedge x \updownarrow y\} = \bigsqcup X \wedge \bigsqcup Y$.

We define a FM-category $\mathcal{FB}$ in which objects are FM-bicpos and morphisms from A to B are bounded-continuous functions $f : |A| \to |B|$ which are *bistable*: for each x, $f \restriction [x]_{\updownarrow}$ is a lattice homomorphism into $[f(x)]_{\updownarrow}$ — i.e. for all $x, y \in |D|$ such that $x \updownarrow y$, $f(x) \updownarrow f(y)$, $f(x \wedge y) = f(x) \wedge f(y)$ and $f(x \vee y) = f(x) \vee f(y)$. A function into a biorder with $\top$ and $\bot$ elements is *strict* if $f(\top) = \top$ and $f(\bot) = \bot$.

Cartesian closure is obtained by combining the relevant definitions from the CCCs of bistable bicpos and FM-cpos.

Proposition 4.3 *$\mathcal{FB}$ is bi-Cartesian closed.*

Proof. The product and co-product operations on bistable orders are defined directly (pointwise) from the operations on the underlying sets with units 1 and 0 being the one-point and empty biorders. Given FM-bicpos D, E, we define the exponential $D \Rightarrow E$ to be the set of and bistable functions from D to E, ordered extensionally, with $f \updownarrow g$ if $x \updownarrow y$ implies $f(y) \updownarrow g(x)$ and $f(x) \wedge g(y) = f(y) \wedge g(x)$ and $f(x) \vee g(y) = f(y) \vee g(x)$. The requisite meets and joins are defined pointwise — see [4] for the proof that this is a bicpo. □

$\mathcal{FB}$ has boolean, naming and Σ-objects, and a name generator — the function $\mathsf{new}(f) = f(\epsilon(\nu(f)^c)$ is bistable since for all f, g, we have $\mathsf{new}(f \wedge g) = (f \wedge g)(\epsilon(\mathbb{N} - (\nu(f) \cup \nu(g)))) = f(\epsilon(\mathbb{N} - (\nu(f) \cup \nu(g)))) \wedge g(\epsilon(\mathbb{N} - (\nu(f) \cup \nu(g)))) = \mathsf{new}(f) \wedge \mathsf{new}(g)$ and similarly $\mathsf{new}(f \vee g) = \mathsf{new}(f) \vee \mathsf{new}(g)$.

We may first observe that our model is *bisequential* (i.e. sequential with respect to both $\bot$ and $\top$ elements). For example, it excludes parallel composition $\mathsf{par} : \Sigma \times \Sigma \to \Sigma$ since $\bot = \mathsf{par}(\bot, \bot) \neq \mathsf{par}(\top, \bot) \wedge \mathsf{par}(\bot, \top) = \top \wedge \top = \top$. More generally, the following was proved in [2]:

Lemma 4.4 *Given pointed bistable biorders $A_1, \ldots, A_n$, every strict, monotone and bistable function $f : A_1 \times \ldots \times A_n \to \Sigma$ is i-strict (i.e. $\pi_i(x) = \bot$ implies $f(x) = \bot$ and $\pi_i(x) = \top$ implies $f(x) = \top$) for some $i \leq n$ (unique if the A_i are non-terminal).*

As an example of the force of bistability with respect to naming, we observe that unlike the FM-cpo model there is no "junk" at type $\neg\neg N$. First, define $p_n \in N \Rightarrow \Sigma = \lambda x.\mathsf{If}\ x = n\ \mathsf{then}\ \top\ \mathsf{else}\ \bot$, and $q_n \in N \Rightarrow \Sigma = \lambda x.\mathsf{If}\ x = n\ \mathsf{then}\ \bot\ \mathsf{else}\ \top$

Proposition 4.5 *The only elements of $(N \Rightarrow \Sigma) \Rightarrow \Sigma$ are $\top$, $\bot$, $\{\lambda k.k\, n \mid n \in \mathbb{N}\}$ and* new.

Proof. Suppose $f : (N \Rightarrow \Sigma) \Rightarrow \Sigma$ is not in $\{\top, \bot, \mathsf{new}\}$. Then there exists some $g : N \Rightarrow \Sigma$ such that $f(g) \neq g(\nu(g)^c)$. Suppose $f(g) = \top$, so $g(n) = \top$ if and only if $n \in \nu(g)$. Then $g = \bigvee_{n \in \nu(g)} p_n$ and so by bistability, $f(p_n) = \top$ for some n. Since $f \neq \top$, we have $f(p_n) \wedge f(q_n) = f(p_n \wedge q_n) = f(\bot) = \bot$ — i.e. $f(q_n) = \bot$. Hence $f(g) = \top$ if and only if $p_n \sqsubseteq g$ — i.e. $f = \lambda g.g(n)$. □

Hence the model validates equivalence (1).

Corollary 4.6 $[\![\lambda k.\nu n.k\, \langle \lambda x.\mathsf{If}\ x = n\ \mathsf{then}\ \top\ \mathsf{else}\ \bot \rangle]\!] = [\![\lambda k.k\, \langle \lambda x.\bot \rangle]\!]$.

Proof. $[\![\lambda k.\nu n.k\, \langle \lambda x.\mathsf{If}\ x = n\ \mathsf{then}\ \top\ \mathsf{else}\ \bot \rangle]\!](f) = f(\bot)$ for $f \in \{\top, \bot, \mathsf{new}\} \cup \{\lambda k.k\, n \mid n \in \mathbb{N}\}$. □

5 Full Abstraction to Third Order

In the remainder of the paper we shall consider the completeness properties of our model. Because it is extensional and bounded-continuous, we may prove full

abstraction to order $n+1$ by proving that all elements, or a finite basis of elements, are definable at order n.

A key idea that we shall use is the notion of definable retraction between types — we write $\sigma \trianglelefteq \tau$ if there are terms $x : \sigma \vdash \mathsf{inj} : \tau$ and $x : \tau \vdash \mathsf{proj} : \sigma$ such that $[\![\mathsf{inj}]\!]; [\![\mathsf{proj}]\!]$ is the identity. In this case, if universality holds at type τ then it holds at type σ — if $[\![\mathsf{inj}]\!](e)$ is definable as $M : \tau$ then $e \in [\![\sigma]\!]$ is definable as $\mathsf{proj}[M/x]$. Definable retractions are used to show universality of the name-free fragment of the language as in [4].

Proposition 5.1 *Universality holds at every name-free type.*

Proof. We show that every name-free type is a retract of $(\neg\neg B^k)^l$ for some k, l.□

We also have a definable retraction $B \trianglelefteq N$, via the terms $x \vdash x = 0$ and $y \vdash \mathsf{If}\ y\ \mathsf{then}\ 0\ \mathsf{else}\ 1$.

Lemma 5.2 *For any k, l, $\neg N^k \times \neg N^l \trianglelefteq \neg N^{\max\{k,l\}+1}$*

Proof. We have $\neg N^k \times \neg N^l \trianglelefteq (\neg N^{\max\{k,l\}})^2 \cong \neg(B \times N^{\max\{k,l\}}) \trianglelefteq \neg(N \times N^{\max\{k,l\}}) = \neg N^{\max\{k,l\}+1}$. □

By repeated application of this lemma we obtain the following:

Lemma 5.3 *For any second-order σ there exist k, l such that $\sigma \trianglelefteq \neg(\neg N^k \times N^l)$.*

Lemma 5.4 *If universality holds at $\neg T$ then universality holds at $\neg(N \times T)$.*

Proof. Given $f \in \neg(N \times T)$, suppose $m \notin \nu(f) = i_1, \ldots, i_n$. Let $M_f = \lambda(x, \vec{y}).$ $\mathsf{If}\ x = i_1\ \mathsf{then}\ M_{\lambda e.f(i_1,e)}\ \langle\vec{y}\rangle \ldots \mathsf{If}\ x = i_n\ \mathsf{then}\ M_{\lambda e.f(i_n,e)}\ \langle\vec{y}\rangle\ \mathsf{else}\ M_{\lambda e.f(m,e)}[x/m]\ \langle\vec{y}\rangle$.

Then if $j \in \nu(f)$, $[\![M]\!](j, e) = [\![M_{f(j)}]\!](e) = f(j, e)$, and if $j \notin \nu(f)$, $[\![M]\!](j, e) = [\![M_{f(m)}[j/m]]\!](e) = [m \leftrightarrow j]([\![M_{f(m)}]\!](e)) = [m \leftrightarrow j](f(m, e)) = f(j, e)$ as required. □

Corollary 5.5 *Universality holds at $\neg N^k$ for any k.*

Note that $g \updownarrow h$ for every $g, h \in N^k \Rightarrow \Sigma$ and so $N^k \Rightarrow \Sigma$ is a lattice. Moreover, it is a boolean algebra: every element $g \in N^k \Rightarrow \Sigma$ has a complement $g^\perp$ (defined $g^\perp(i) = \perp$ iff $g(i) = \top$) such that $g \wedge g^\perp = \perp$ and $g \vee g^\perp = \top$. Note that every strict map $f : (N^k \Rightarrow \Sigma) \to \Sigma$ is a boolean homomorphism: if $f(g) = \perp$ then $f(g^\perp) = \top$, and vice-versa, since $f(g) \vee f(g^\perp) = f(g \vee g^\perp) = f(\top) = \top$.

Definition 5.6 An element $p \in N^k \Rightarrow \Sigma$ is a *quasi-atom* if it is a true atom — i.e. $p \sqsubseteq g \vee h$ implies $p \sqsubseteq g$ or $p \sqsubseteq h$ — or an 'invariant atom" — i.e. p is invariant and for any invariants g, h, $p \sqsubseteq g \vee h$ implies $p \sqsubseteq g$ or $p \sqsubseteq h$. A *literal* is an element e such that e or $e^\perp$ is a quasi-atom.

Given $i < k$ and $n \in \mathbb{N}$, let $p_n(i) : N^k \Rightarrow \Sigma = \lambda(\vec{x}).\mathsf{If}\ x_i = n\ \mathsf{then}\ \top\ \mathsf{else}\ \perp$. Given $i, j < k$, let $p(i, j) : N^k \Rightarrow \Sigma = \lambda(\vec{x}).\mathsf{If}\ x_i = x_j\ \mathsf{then}\ \top\ \mathsf{else}\ \perp$.

It is straightforward to see that p is a true atom iff $p = \perp$ or $p = p_n(i)$ for some i, n, and p is an invariant atom iff $p = \perp$ or $p = p(i, j)$ for some i, j. We have the following "literal completeness" property for $N^k \Rightarrow \Sigma$

Lemma 5.7 *For any element $g \in N^k \Rightarrow \Sigma$ there exists a finite family of finite families of literals $\{\{h_{ij} \mid j \in J_i\} \mid i \in I\}$ such that $g = \bigvee_{i\in I} \bigwedge_{j\in J_i} h_{ij}$.*

Proof. By Corollary 5.5 it suffices to show that for any (β-normal) term $\lambda(\vec{x}).t : N^k \Rightarrow \Sigma$, $[\![t]\!]$ has a disjunctive normal form as claimed. Supposing $t \notin \{\top, \bot\}$. Then $t = \mathsf{If}\ B\ \mathsf{then} s_1\ \mathsf{else}\ s_2$ for some s_1, s_2, and $[\![t]\!] = ([\![\mathsf{If}\ B\ \mathsf{then}\ \top\ \mathsf{else}\ \bot]\!] \wedge [\![s_1]\!]) \vee ([\![\mathsf{If}\ B\ \mathsf{then}\ \bot\ \mathsf{else}\ \top]\!] \wedge [\![s_2]\!])$. Since B has a disjunctive normal form, so do $[\![\mathsf{If}\ B\ \mathsf{then}\ \top\ \mathsf{else}\ \bot]\!]$ and $[\![\mathsf{If}\ B\ \mathsf{then}\ \bot\ \mathsf{else}\ \top]\!]$, and thus applying the inductive hypothesis and distributivity laws we are done. □

Hence to show that $f : (N^k \Rightarrow \Sigma) \to \Sigma = g$, it suffices to show that $f(p) = g(p)$ for all quasi-atoms p.

Lemma 5.8 *Every invariant element $f \in (N^k \Rightarrow \Sigma) \Rightarrow \Sigma$ is definable.*

Proof. If $f \in \{\top, \bot\}$ then it is definable, so assume that f is strict. Define $m : \{1, \ldots k\} \to \{1, \ldots k\}$ by $m(i) = \min\{j \leq k \mid f(p(i,j)) = \top\}$, and let $M_f = \lambda g.\nu x_1 \ldots \nu x_k.g\, x_{m(1)} \ldots x_{m(k)}$. Then $[\![M_f]\!] = f$: for any atom $p_n(i)$, we have $f(p_n(i)) = [\![M_f]\!](p_n(i)) = \bot$ since for $m \neq n$, $f(p_m(i)) = [m \leftrightarrow n]f(p_n(i)) = f(p_n(i))$ and so $f(p_n(i)) = f(p_n(i)) \wedge f(p_m(i)) = f(p_n(i) \wedge p_m(i)) = f(\bot) = \bot$.

For an atom $p(i,j)$, if $f(p(i,j)) = \top$ then $m(i) = m(j)$ and so $[\![M_f]\!](p(i,j)) = \mathsf{new}([\![\lambda x.\mathsf{If}\ x = x\ \mathsf{then}\ \top\ \mathsf{else}\ \bot]\!]) = \top$. If $f(p(i,j)) = \bot$ then $m(i) \neq m(j)$ and so $[\![M_f]\!](p(i,j)) = [\![\nu x.\nu y.\mathsf{If}\ x = y\ \mathsf{then}\ \top\ \mathsf{else}\ \bot]\!] = \bot$.

□

Lemma 5.9 *Universality holds at $\neg\neg N^k$ for all k.*

Proof. By induction on k. If $f \in (N^{k+1} \Rightarrow \Sigma) \Rightarrow \Sigma$ is invariant then it is definable by Lemma 5.8. So suppose f is not invariant. Then there exists $g \in N^{k+1} \Rightarrow \Sigma$, and $\pi \in G$ such that $f(g) = \top$ and $f(\pi \cdot g) = \bot$, and since the meet, join and complement operations are π-invariant, we may assume by Lemma 5.7 that g is atomic — i.e. $g = p_n(i)$ for some $n \in N$ and $i \leq k+1$.

Let $f^i_n : (N^k \Rightarrow \Sigma) \Rightarrow \Sigma$ be defined $f^i_n(g) = f(\lambda\vec{x}.g\, x_1 \ldots x_{i-1} n x_i \ldots x_k)$. Then we claim that for any $h : N^{k+1} \Rightarrow \Sigma$, $f(h) = f^i_n(\lambda\vec{x}.h(x_1 \ldots x_{i-1} n x_i \ldots x_k)$. Suppose $f(h) = \top$, then $\top = f(p_n(i) \wedge h) = f(\lambda\vec{x}.\mathsf{If}\ x_i = n\ \mathsf{then}\ h(\vec{x})\ \mathsf{else}\ \bot) \sqsubseteq f(\lambda\vec{x}.h(x_1 \ldots x_{i-1} n x_i \ldots x_k))$. Similarly, if $f(h) = \bot$, then $\bot = f(p_n(i)^\bot \vee h) \sqsupseteq f(\lambda\vec{x}.h(x_1 \ldots x_{i-1} n x_i \ldots x_k))$.

By hypothesis, f^i_n is definable as a term $M_{f^n_i}$ and so f is definable as $\lambda g.M_{f^i_n}\, (\lambda\vec{x}.g\, x_1 \ldots x_{i-1} n x_i \ldots x_k)$. □

Proposition 5.10 *Universality holds at all second-order types.*

Proof. By Lemma 5.3 it suffices to show this for types of the form $\neg(N^k \times \neg N^l)$. But this follows by induction on k for which the base case is Lemma 5.9 and the induction case is Lemma 5.4. □

6 Full Abstraction to Fourth Order

We shall now show sketch the proof of full abstraction up to fourth order. Universality does not hold at third order — consider the least upper bound of the chain of approximants $g_i \in [\![\neg(N \times \neg(N \times \neg N))]\!]$ given by $g_0 = \lambda(x,f).\bot$, $g_{i+1} = \lambda(x,f).f\,\langle x, \lambda y.g_i\,\langle y, f\rangle\rangle$: this is definable only if we include recursion in the language. However, we may show that every element at third-order type is the limit of a (finitely supported) chain of definable approximants, and thus prove full abstraction up to fourth order. We establish definability of finite chains of approximants for types of the form $\neg\neg(N^k \times \neg N^l)$ and extend it to all third-order types using definable retractions and Lemma 5.4. A (non-constant) function f of this type first tests its argument with a k-tuple of new and explicit names and receives a l-tuple of new and explicit names in return, from which it can construct a second k-tuple of names, and so on. Thus to construct a defining chain of approximating terms for f we successively extract representations of these k and l-tuples, whilst keeping track of introduced names.

By our results in the previous section, we may think of a strict function $f \in (N^k \Rightarrow \Sigma) \Rightarrow \Sigma$ as representing a "generalised" k-tuple of names, each of which may be either concrete — i.e. $f(p_n(i)) = \top$ if the ith name is n — or fresh names, which can be tested for equality with one another. So given a set of "known names" S, $f \in (N^k \Rightarrow \Sigma) \Rightarrow \Sigma$ and a k-tuple of names $\vec{a} \in N^k$, we may define a corresponding test "equivalence up to S" which holds essentially when f cannot be distinguished from $\lambda\kappa.\kappa\,\vec{a}$ by an observer who only knows the names in S.

Define the predicate $\mathrm{Eq}_S(f,\vec{a})$ to hold iff for all $i \leq k$:

- (i) $f\,p_n(i) = \top$ implies $a_i = n$
- (ii) if $f\,p_n(i) = \bot$ for all n then $a_i \notin S$
- (iii) for all $j \leq k$, $f\,p(i,j) = \top$ if and only if $a_i = a_j$

Lemma 6.1 *If $\nu(d) \subseteq S$ and $\mathrm{Eq}_S(f,\vec{a})$ then $f\,d = d\,\vec{a}$.*

Proof. It is sufficient to prove this for quasi-atomic d. If $d = p_n(i)$ then $f(d) = \top$ implies $a_i = n$ and hence $d(\vec{a}) = \top$ by definition (i). If $f(d) = \bot$, then either $f(p_m(i)) = \top$ for some $m \neq n$ — in which case $a_i = m$ and so $d(\vec{a}) = \bot$ — or else $f(p_m(i)) = \bot$ for all $m \in \mathbb{N}$, and so by (ii), $a_i \notin S$ thus $a_i \notin \nu(d)$ and so $a_i \neq n$ and $d(\vec{a}) = \bot$ as required. If $d = p(x,y)$ then $f(d) = d(\vec{a})$ by definition (iii). □

Say that an element $e \in (N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma$ is *parametric* if $e \notin \{\top, \bot\}$ and for each prime $p_n(i)$ either $e\langle\vec{a},d\rangle = e(\pi\cdot\vec{a})p_n(i)$ for all $\vec{a} \in N_k$ and $\pi \in G$, or $e(\vec{a})p_n(i) = e(\pi\cdot\vec{a})p_{\pi(n)}(i)$ for all $\vec{a} \in N_k$ and $\pi \in G$. Thus a parametric element represents a generalised l-tuple of names each of which is either a concrete name, a new name, or some one of the a_i.

Given $f \in ((N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma) \Rightarrow \Sigma$ and $e \in (N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma$, we now define a series of tuples of names passed between f and e, and thus a series of approximants to the "revealed fragment" of e.

Given boolean values $b_1,\ldots,b_k$, and names $n_1,\ldots,n_{k+1}$, let $\mathtt{case}\,[b_1 \mapsto$

$n_1, \ldots, b_k \mapsto n_k, n_{k+1}] = n_i$, where i is the least value such that $b_i = tt$, or $\mathtt{case}\,[b_1 \mapsto n_1, \ldots b_k \mapsto n_k, n_{k+1}] = n_{k+1}$ if each $b_i = ff$.

For each $i \in \omega$ we define:

- a set of "revealed names" $U_f^i(e) \subseteq \mathbb{N}$ consisting of all of the names known to f together with all names revealed by e so far during interaction.
- a generalised k-tuple of names supplied by f at the ith step of interaction; $\Phi_f^i(e) : (N^k \Rightarrow \Sigma) \Rightarrow \Sigma$
- a (parametrised) generalised l-tuple of names returned by e, $\Psi_f^i(e) : (N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma$.
- a "reconstruction" $\Gamma_f^i(e, d, S) : (((N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma)$ of e with respect to a set of names S, which tests its input for equality (up to S) with the tuples already supplied by f, and returns the corresponding l-tuple observed to have been supplied by e, and otherwise behaves as d.

These are defined:
$U_f^0(e) = \nu(f)$ and $\Gamma_f^0(e, d, S) = d$
$\Phi_f^{i+1}(e) = \lambda h.f(\Gamma_f^i(e, (\lambda(a, b).h(a)), U_f^i(e)))$.
$\Psi_f^{i+1}(e) = \lambda(\vec{x}, y).\Phi_{i+1}^f(e)\ \lambda\vec{z}.e\ \langle\vec{z}, \lambda\vec{w}.y\ \langle t_1 \ldots t_l\rangle\rangle$, where $t_j = \mathtt{case}\,[(z_1 = w_j \wedge w_j \notin S \mapsto x_1), \ldots (z_k = w_j \wedge w_j \notin S \mapsto x_k), w_j]$.
$U_f^{i+1}(e) = U_f^i(e) \cup \nu(\Gamma_f^{i+1}(e, U_f^i(e))$
$\Gamma_f^{i+1}(e, d, S) = \lambda(\vec{x}, h).\mathsf{If}\ \mathrm{Eq}_S(\Phi_f^i(e, S), \vec{x})\ \mathsf{then}\ \Psi_f^i(e)\langle\vec{x}, h\rangle\ \mathsf{else}\ \Gamma_f^i(e, d, S)\langle\vec{x}, h\rangle$.

We now prove the following facts using Lemma 6.1.

Lemma 6.2 *If $\nu(e) \cup \nu(f) \subseteq S$, then the following hold for all $i \in \omega$:*

- *If $\mathrm{Eq}_S(\Phi_f^{i+1}(e), a_1, \ldots a_k)$ then $\Psi_f^{i+1}(e)\langle\vec{a}, q\rangle = e\,\langle\vec{a}, q\rangle$.*
- *If $\Gamma_f^i(e, \bot, S) \sqsubseteq e \sqsubseteq \Gamma_f^i(e, \top, S)$.*
- *$\Phi_f^{n+1}(e) \notin \{\top, \bot\}$ implies $\Phi_f^i(e) \neq \Phi_f^{n+1}(e)$.*
- *For each n, either $\Psi_f^{n+1}(e) \in \{\top, \bot\}$ or $\Gamma_f^n(e, \bot, S) \sqsubset \Gamma_f^{n+1}(e, \bot, S)$.*

Lemma 6.3 *Given an infinite chain with bounded support $e_1 \sqsubseteq e_2 \sqsubseteq \ldots \in (N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma$ there exists m such that $e_m = e_n$ for all $n \geq m$.*

Proof. By induction on k. For the base case, we note that if $e \sqsubseteq e' \in (N^l \Rightarrow \Sigma) \Rightarrow \Sigma$ then either $e = \bot$ or $e = e'$ or $e' = \top$. For the induction case, suppose we have a chain $e_1 \sqsubseteq e_2 \sqsubseteq \ldots \in N \Rightarrow (N^k \Rightarrow (N^l \Rightarrow \Sigma) \Rightarrow \Sigma)$, with bounded support S. Then for each $i \in S$ there exists m_i with $e_{m_i}(i) = e_n(i)$ for all $n \geq m_i$. Also, there exists m' such that for all $j \notin S$, we have $e_{m'}(j) = e_n(j)$ for all $n \geq m'$. So we may take $m = \max(\{m'\} \cup \{m_i \mid i \in S\})$. □

Proposition 6.4 *For all f, e, there exists n such that $\Phi_f^n(e) \in \{\top, \bot\}$.*

Proof. Let $S = \nu(f) \cup \nu(e)$. Then if $\Phi_f^n(e) \notin \{\top, \bot\}$ for all n, we have $\Gamma_f^n(e, \bot, S) \sqsubset \Gamma_f^{n+1}(e, \bot, S)$ for all n — i.e. $\Gamma_f^n(e, \bot, S)(\bot)$ forms an infinite, strictly increasing

chain with support bounded by S, contradicting Lemma 6.3. □

For any element $e \in (N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma$, and $d \in \{\top, \bot\}$, let $e[d]^i_n \in N^k \Rightarrow (N^l \Rightarrow \Sigma) \Rightarrow \Sigma$ be $\lambda(\vec{x}, g).\mathsf{If}\ x_i = n\ \mathsf{then}\ d\ \mathsf{else}\ e\ \langle \vec{x}, g\rangle$.

Lemma 6.5 *If $n \notin \nu(f) \cup \nu(e)$ then $f(e[\bot]^i_n) = f(e[\top]^i_n) = f(e)$ for each i.*

Proof. Suppose $f(e) = \top$. Choose m such that $m \notin \nu(f) \cup \nu(e)$. Then $f(e[\top]^i_n[\top]^i_m) = \top$, and since $e[\top]^i_n[\bot]^i_m \updownarrow e[\top]^i_m[\bot]^i_n$ and $e[\top]^i_n[\bot]^i_m \vee e[\top]^i_m[\bot]^i_n = e[\top]^i_n[\top]^i_m$, by bistability we have $f(e[\top]^i_n[\bot]^i_m) = \top$ or $f(e[\top]^i_m[\bot]^i_n) = \top$. But we also have $f(e[\top]^i_m[\bot]^i_n) = f([n \leftrightarrow m] \cdot e[\top]^i_n[\bot]^i_m) = [n \leftrightarrow m] \cdot f(e[\top]^i_n[\bot]^i_m) = f(e[\top]^i_n[\bot]^i_m) = \top$. Hence by bistability, $f([\bot]^i_m[\bot^i_m]) = f(e[\top]^i_n[\bot]^i_m \wedge e[\top]^i_m[\bot]^i_n) = f(e[\top]^i_n[\bot]^i_m) \wedge f(e[\top]^i_m[\bot]^i_n) = \top$. So $f(e[\bot]^i_n) =\geq f([\bot]^i_m[\bot^i_m]) = \top$ as required. □

For a set of names $X = \{a_1, \ldots, a_n\}$, let $e[d]_X = e[d]^1_{a_1} \ldots {}^k_{a_1} \ldots {}^1_{a_n} \ldots {}^k_{a_n}$.

Lemma 6.6 *For all f, e, there exists n such that $\Psi^n_f(e) = \lambda x.f(e)$.*

Proof. By Proposition 6.4, there exists n such that $\Psi^{n+1}_f \in \{\top, \bot\}$. Suppose w.l.og. that $\Psi^n_f(e) = \top$ — then $f(\Gamma^n_f(e, U^n_f(e))(\bot)) = \top$. Let $X = S - U^n_f(e)$. Then $\Gamma^n_f(e, U^n)(\bot)[\bot]_X \sqsubseteq \Gamma^n_f(e, S)(\bot)$, since for any $\vec{a} \in N^k$, if $\nu(\vec{a}) \cap X \neq \varnothing$ then $\Gamma^n_f(e, U^n_f(e)(\bot)[\bot]_X(\vec{a}) = \bot$, and if $\nu(\vec{a}) \cap X = \varnothing$ then $\mathrm{Eq}_{U^n_f(e)}(g, \vec{a}, g)$ if and only if $\mathrm{Eq}_S(g, \vec{a})$ and so $\Gamma^n_f(e, \bot, U^n_f(e)\langle\vec{a}\rangle = \Gamma^n_f(e, \bot, S)\langle\vec{a}\rangle$. By repeated application of Lemma 6.5, $f(\Gamma^n_f(e, \bot, U(e)^n)[\bot]_X) = f(\Gamma^n_f(e, \bot, U(e)^n)) = \top$ and so $f(\Gamma^n_f(e, \bot, S)) = \top$. Since $\Gamma^n_f(e, \bot, S) \sqsubseteq e$ we have $f(e) = \top$ as required. □

For each $f \in ((N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma) \Rightarrow \Sigma$ we have $\Psi^n_f(e)(\bot) = \top$ implies $\Psi^{n+1}_f(e)(\bot) = \top$ by definition — i.e. $\{\lambda x.\Psi^i_f(x)(\bot) \mid i \in \omega\}$ is an ω-chain.

Proposition 6.7 *For any $f \in (N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma) \Rightarrow \Sigma$, $f = \bigsqcup \lambda x.\Psi^i_f(x)(\bot)$.*

Proof. For any e, if $f(e) = \top$ then by Lemma 6.6 there exists n such that $\Psi^n_f(e)(\bot) = \top$ and so $(\bigsqcup \lambda x.\Psi^i_f(x)(\bot)) = \top$. Conversely, if $(\bigsqcup \lambda x.\Psi^i_f(x)(\bot))(e) = \top$ then there exists n such that $\Psi^n_f(e)(\bot) = \top$ and so by Lemma 6.6, $f(e) = \top$. □

6.1 Definability

We now need to show that for each $f \in ((N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma) \Rightarrow \Sigma$, and $i \in \omega$, the function sending e to $\Psi^i_f(e)(\bot)$ is definable. Note that the test Eq_S is definable for any S — more precisely, for any k, m there is a definable function $\mathtt{test} \in (N^m \times ((N^k \Rightarrow \Sigma) \Rightarrow \Sigma) \times N^k \times (B \Rightarrow \Sigma)) \Rightarrow \Sigma$ such that $\mathtt{test}(n_1, \ldots n_m, f, \vec{a}, g) = f$, if $f \in \{\top, \bot\}$, $\mathtt{test}(n_1, \ldots n_m, f, \vec{a}, g) = g\ tt$ if $\mathrm{Eq}_{n_1,\ldots,n_m}(f, \vec{a})$ and $\mathtt{test}(n_1, \ldots n_m, f, \vec{a}, g) = g\ ff$ otherwise.

A key element of our definability proof is to establish that we may encode $\Psi^i_f(e)$ as a tuple of names; since it is a parametric element of $(N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma$ it may be represented as a tuples $(\vec{b}, \vec{c})$ in $N^l \times N^l$. The basic idea is that if $e(\vec{a})(p_n(i)) = \bot$ for all n (i.e. $\lambda g.e\langle\vec{a}, \lambda\vec{m}.g(m_i)\rangle = \mathsf{new}$) then we record this by setting $b_i = 0$. We

then use c_i to record the least j such that $e\langle \vec{a}, p(i,j)\rangle = \top$ for all $\vec{a}$. If $e\langle \vec{a}, p_n(i)\rangle = \top$ for all $\vec{a}$, then we record this by setting $b_i = k+1$ and $c_i = n$. Otherwise, there exists $j \leq k$ such that $e\langle \vec{a}, p_{a_j}(i)\rangle = \top$ for all $\vec{a}$, and we may record this by setting b_i to be the least such j.

In order to define Φ^i_f for each i we also need a definable function to keep track of the revealed names $U^i_f(e)$: for each i we define $\chi^i_f : ((N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma) \Rightarrow (N^{(l\times i)} \Rightarrow \Sigma) \Rightarrow \Sigma$ such that $\nu(\chi^i_f(e)) \cup \nu(f) = U^f_i(e)$ for all i by:
$\chi^{i+1}_f = \lambda(x,g).\chi^i_f(x)\, \lambda\vec{y}.\nu\vec{a}.\Phi^{i+1}_f\, \vec{a}\, \lambda z.g\, \vec{y}\vec{z}$. Using second-order definability we may now show:

Proposition 6.8 *For each $f \in ((N^k \times (N^l \Rightarrow \Sigma)) \Rightarrow \Sigma) \Rightarrow \Sigma$, and $i \in \omega$, Ψ^i_f, Φ^i_f and χ^i_f are all definable.*

Using definable retractions and Lemma 5.4 (modified) we may now prove:

Theorem 6.9 *Full abstraction holds at every fourth-order type.*

7 Failure of Full Abstraction at Fifth Order

We shall now give an example of a non-definable element at fourth order, and show that it leads to a failure of full abstraction at fifth order. Although the possible behaviours at this order are complicated, the counterexample itself is relatively simple: equivalence 2 between $\lambda k.\nu n.k\, \lambda f.\nu\, m.f\, \langle m, \lambda a.f\, \langle n, \bot\rangle\rangle$ and $\lambda k.k\, \lambda f.\nu\, m.f\, \langle m, \bot\rangle$ does not hold in our model. First we shall prove that these terms are indeed observationally equivalent (using the FM-biorder semantics). Fix a name n, and define $F_n : \neg\neg(N \times \neg U) = \lambda f.\nu\, m.f\, \langle m, \lambda a.f\, \langle n, \bot\rangle\rangle$ and $F_\bot : \neg\neg(N \times \neg U) = \lambda f.\nu\, m.f\, \langle m, \bot\rangle$.

By the Context Lemma, the equivalence holds if for any term $M : \neg\neg\neg(N \times (\neg U)$ not containing the name n, $M\, F_n \Downarrow$ implies $M\, F_\bot \Downarrow$.

Lemma 7.1 *For any term $x_1 : N, \ldots, x_k : N, y_1 : \neg U, \ldots, y_l : \neg U, g : \neg\neg(N \times \neg U) \vdash M : \Sigma$ not containing the name n, any $\vec{m} \in N^k$ not including n, and $e \in \Sigma^l$: $[\![M]\!](\vec{m}, e, F_n) = [\![M]\!](\vec{m}, e, F_\bot)$.*

Proof. By induction on the length of the β-normal form of M, with the additional inductive hypothesis $(*)$: for any $\vec{m} \in N^k$ (possibly including occurrences of n), $e \in \Sigma^l$, and a name $a \notin \nu([\![M]\!]) \cup \{n\}$, if $[\![M]\!](\vec{m}[a/n], e[\bot]_i, F_n) = \bot$ and $[\![M]\!](\vec{m}[a/n], e[\top]_i, F_n) = \top$ then $[\![M]\!](\vec{m}, e[\bot]_i, F_n) = \bot$. (Where $\vec{m}[a/n]$ is $\vec{m}$ with all occurrences of n replaced with a.)

If $M \equiv \bot$, $M \equiv \top$, or $M \equiv y_i\, \langle\rangle$ then the result is immediate. If $M = \nu x.M'$ then we may apply the inductive hypothesis to M'. If $M \equiv \mathsf{If}\ N_1\ \mathsf{then}\ N_2\ \mathsf{else}\ N_3$ then we may show $[\![N_1(\vec{m})]\!] = [\![N_1(\vec{m}[a/n])]\!]$ and so we may apply the induction hypotheses to each of N_2, N_3.

So suppose $M \equiv g\, \lambda(x_{k+1}, y_{l+1}).N$. We first show that $(*)$ holds: suppose $[\![M]\!](\vec{m}[a/n], e[\bot]_i, F_n) = \bot$ and $[\![M]\!](\vec{m}[a/n], e[\top]_i, F_n) = \top$. Let b be a fresh name. By strong sequentiality of the model (Lemma 4.4), we have either:

- $[\![N]\!](\vec{m}[a/n], b, \top[\bot]_i, \top, F_n) = \bot$ and $[\![N]\!](\vec{m}[a/n], b, \bot[\top]_i, \bot, F_n) = \top$ — in which case $[\![N]\!](\vec{m}, b, \top[\bot]_i, \top, F_n) = \bot$ by hypothesis and so $[\![M]\!](\vec{m}, e, F_n) = \bot$ as required — or else:
- $[\![N]\!](\vec{m}[a/n], b, \top, \bot, F_n) = \bot$ and $[\![N]\!](\vec{m}[a/n], b, \top, \top, F_n) = \top$. But then by hypothesis, $[\![N]\!](\vec{m}[a/n], n, \top, \bot, F_n) = \bot$ and hence $[\![M]\!](\vec{m}[a/n], e[\top]_i, F_n) = \bot$, contradicting the assumption above.

We may now prove the main induction hypothesis. Suppose $[\![M]\!](\vec{m}, e, F_\bot) = \bot$ and $[\![M]\!](\vec{m}, e, F_n) = \top$. Then $[\![N]\!](\vec{m}, b, e, F_\bot) = [\![N]\!](\vec{m}, b, e, \bot, F_n) = \bot$ by hypothesis on N, and so we must have $[\![N]\!](\vec{m}, b, e, \top, F_n) = \top$ and thus $[\![N]\!](\vec{m}, b, \vec{\top}, \bot, F_n) = \bot$ and $[\![N]\!](\vec{m}, b, \vec{\bot}, \top, F_n) = \top$. By $(*)$, $[\![N]\!](\vec{m}, n, \vec{\top}, \bot, F_n)$
$= \bot$, so $[\![M]\!](\vec{m}, e, F_n) = [\![N]\!](\vec{m}, b, e, [\![N]\!](\vec{m}, n, e, \bot, F_n), F_n) = [\![N]\!](\vec{m}, b, e, \bot, F_n)$
$= \bot$ which is a contradiction. □

Corollary 7.2 *Equivalence* (2) *holds in the CPS-nu-calculus.*

To show that it does not hold in the FM-bicpo model, for each $i \in \mathbb{N}$ define $f_i : (N \times \Sigma) \Rightarrow \Sigma$: $f_i(n, e) = \top$ iff $i = n$ or $e = \top$. Consider the function $G : (((N \times \Sigma) \Rightarrow \Sigma) \Rightarrow \Sigma) \Rightarrow \Sigma$ defined $G(h) = \top$ iff $h(f_i) = \top$ for some $i \in \mathbb{N}$. This is a bistable function — it clearly preserves all binary joins and to show that it preserves bistable meets, suppose $h \updownarrow k$ and $G(h) = G(k) = \top$. Then there exists i such that $h(f_i) = \top$ and j such that $k(f_j) = \top$. Define $g_i : (N \times \Sigma) \Rightarrow \Sigma$ by $g_i(n, e) = \bot$ iff $i = n$ or $e = \bot$. Then $g_i \updownarrow f_i$ and $g_i \sqsubseteq f_j$ for all j. So by definition of $\updownarrow$, $\top = h(f_i) = h(f_i) \vee k(g_i) = h(g_i) \vee k(f_i)$ and so either $k(f_i) = \top$ (and so $(h \wedge k)(f_i) = \top$) or else $h(f_j) \sqsupseteq h(g_i) = \top$ (and so $(h \wedge k)(f_j) = \top$). Thus $G(h \wedge k) = \top$ as required.

Proposition 7.3 $[\![\lambda k.\nu n.k\, \langle \lambda f.\nu m.f\, \langle m, \lambda a.f\, \langle n, \bot\rangle\rangle\rangle]\!] \neq [\![\lambda k.k\, \langle \lambda f.\nu m.f\, \langle m, \bot\rangle\rangle]\!]$.

Proof. $G([\![\lambda f.\nu m.f\, \langle m, \lambda a.f\, \langle n, \bot\rangle\rangle]\!]) = [\![\lambda f.\nu m.f\, \langle m, \lambda a.f\, \langle n, \bot\rangle\rangle]\!](f_n) = \top$. Hence $[\![\lambda k.\nu n.k\, \lambda f.\nu m.f\, \langle m, \lambda a.f\, \langle n, \bot\rangle\rangle]\!](G) = \top$. But $[\![\lambda f.\nu m.(f\, \langle m, \bot\rangle)]\!](f_i) = \bot$ for all i and hence $[\![\lambda k.k\, \lambda f.\nu m.\, \langle m, \bot\rangle]\!](G) = \bot$. □

So we may deduce that G is not definable in the CPS-nu-calculus.

8 Conclusions

We have given a general notion of CPS model with fresh names, of which many notions of functional (stable, strongly stable, etc.) will yield an instance. We have shown that "bistable coherence" structure, by imposing sequentiality, increases the fragment of the language which can be interpreted fully abstractly to all fourth-order terms. (This fourth order fragment is quite expressive; it may be used, for example, to describe protocols involving the generation and exchange of fresh names as in the cryptographic λ-calcuus [8].

The most obvious further questions concern the failure of full abstraction in the bistable model. On the one hand, the functional G can be implemented sequentially

(using state, for example), and it can be added to the language. Is the bistable model fully abstract for the resulting language? On the other hand, is there a stronger constraint on functionals which will eliminate G from the language? Is this sufficient to obtain full abstraction?

References

[1] S. Abramsky, D. R. Ghica, A. S. Murawski, C.-H. L. Ong, and I. Stark. Nominal games and full abstracion for the nu-calculus. In *Proceedings of LICS '04*. IEEE Press, 2004.

[2] J. Laird. Bistability: an extensional characterization of sequentiality. In *Proceedings of CSL '03*, number 2803 in LNCS. Springer, 2003.

[3] J. Laird. A game semantics of local names and good variables. In *Proceedings of FOSSACS '04*, number 2987 in LNCS, pages 289–303. Springer, 2004.

[4] J. Laird. Bistability: A sequential domain theory. To appear in Logical Methods in Computer Science, 2007.

[5] R. Milner. Fully abstract models of typed lambda-calculi. *Theoretical Computer Science*, 4:1–22, 1977.

[6] M. R. Shinwell and A. M. Pitts. On a monadic semantics for freshness. *Theoretical Computer Science*, 342:28–55, 2005.

[7] I. Stark. *Names and Higher-Order Functions*. PhD thesis, University of Cambridge Computer Laboratory, 1995.

[8] E. Sumii and B.C. Pierce. Logical relations for encryption. *Journal of Computer Security*, 11:521–554, 2002.