Available at www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

Theoretical Computer Science 308 (2003) 333–366

Theoretical Computer Science

www.elsevier.com/locate/tcs

Adapting innocent game models for the Böhm tree λ-theory

Andrew D. Ker, Hanno Nickau*, C.-H. Luke Ong

Oxford University Computing Laboratory, Parks Road, Oxford. OX1 3QD, UK

Received 20 March 2002; received in revised form 4 November 2002; accepted 12 November 2002
Communicated by G.D. Plotkin

Abstract

We present a game model of the untyped λ-calculus, with equational theory equal to the Böhm tree λ-theory $\mathscr{B}$, which is universal (i.e. every element of the model is definable by some term). This answers a question of Di Gianantonio, Franco and Honsell. We build on our earlier work, which uses the methods of innocent game semantics to develop a universal model inducing the maximal consistent sensible theory $\mathscr{H}^*$. To our knowledge these are the first syntax-independent universal models of the untyped λ-calculus.

Keywords: Game semantics; Effectively almost-everywhere copycat; Innocent strategies; Untyped λ-calculus

1. Introduction

We aim to construct a *universal* model (i.e. every element of the model is the denotation of some term) of the pure untyped λ-calculus which invalidates the η-rule and induces the Böhm tree λ-theory $\mathscr{B}$. We build on the game models presented in [12] which exhibits a reflexive object in a category of innocent games [8,14,16].

A notable feature of game semantics is that the λ-definable strategies are effective methods for copying moves uniformly (from one part of the arena to another). For example, the identity strategy on an arena $A \Rightarrow A$ is *everywhere copycat*, i.e. it always plays back every previous move (but in the opposite copy of A). The key idea is that the innocent strategies definable by untyped λ-terms are, what we call, *effectively almost-everywhere copycat* (EAC), as developed in [12]. EAC strategies give rise to

* Corresponding author.
E-mail address: nickau@comlab.ox.ac.uk (H. Nickau).

0304-3975/03/$ - see front matter
PII: S0304-3975(02)00849-6

a universal λ-model $\mathscr{D}_{\mathrm{EAC}}$ whose theory is the maximal consistent sensible λ-theory $\mathscr{H}^*$. The definition of such strategies uses an efficient encoding of innocent strategies, which we call *economical form*. Section 2 should be regarded as a summary of [12], and this paper is a sequel to that work.

The notion of EAC strategies has a natural extension to *effectively and explicitly almost-everywhere copycat*. However finding an ambient cartesian closed category for these strategies to inhabit proved to be a challenging process as we show in Section 3 – the natural analogues fail to work quite as intended. Once this has been overcome we use a reflexive object to describe a λ-algebra which we call $\mathscr{D}_{\mathrm{XA}}$. In Section 4 we formulate new versions of the powerful *Exact Correspondence Theorem* of the earlier work, to show that $\mathscr{D}_{\mathrm{XA}}$ is a universal λ-model which induces the intended equational theory. To our knowledge, $\mathscr{D}_{\mathrm{EAC}}$ and $\mathscr{D}_{\mathrm{XA}}$ are the first syntax-independent universal λ-models.

In [2] Abramsky and McCusker constructed the first game model for the untyped λ-calculus which was fully abstract for the Lazy Lambda Calculus. More recently in [7], Di Gianantonio *et al* have obtained game models of the untyped λ-calculus using history-free strategies [1]. They show that all their models induce the same λ-theory $\mathscr{H}^*$ and have asked for "a new notion of game to model λ-theories different from $\mathscr{H}^*$." This paper answers that question by constructing a universal game model for the Böhm tree λ-theory.

An extended abstract of this work [11] (in which the model $\mathscr{D}_{\mathrm{XA}}$ was called $\mathscr{M}$) was presented at the European Association for Computer Science Logic conference in Madrid, in September 1999. The work presented in this paper was undertaken as part of the first-named author's EPSRC-funded doctoral research [9] and also under grant GR/L27787.

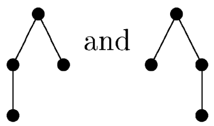
2. Review of Previous Work

In this section we give a brief summary of the results of [12], on which we build in the body of this paper. The definitions in this section are only reviewed in a cursory way; the reader is referred to [12] for a more detailed exposition and proofs of the later results.

We begin with a word on trees. Usually, when we talk about a tree we mean a countably-branching *labelled* tree, presented in a concrete way. Often, the labels will be sequences in $\mathbb{N}^*$; we *do not* include 0 in the set $\mathbb{N}$, and write $\mathbb{N}_0$ for $\mathbb{N} \cup \{0\}$. The set $\mathbb{N}^*$ is the set of finite sequences from $\mathbb{N}$. The root is labelled ε (for the empty sequence) and the descendants of the node labelled $\vec{s}$ are labelled $\vec{s}\cdot 1, \ldots, \vec{s}\cdot n$, if there are n such descendants (we use the notation $\vec{s}$ for sequences and $\cdot$ for concatenation, prefix, or extension). Thus we can talk about the "m^{th} descendant of the node $\vec{s}$" — it is the node labelled $\vec{s}\cdot m$. We can describe a tree by the set of labels of its nodes. In what follows, we will only consider non-empty trees; i.e. they always contain the root node.

When we draw trees they are illustrated "upside-down" with the root at the top, rather more like family trees than the botanical kind (following this analogy, we can refer to a *child* of a node, and say that one node *inherits from* another, with the obvious meanings). Because of the labelling by sequences of natural numbers, the children come with an order, also like a family tree. When we draw a tree without labels, we intend that the numbering of children goes from left to right.

This means that we do not consider the trees drawn



to be the same. The first is described by the set $\{\varepsilon, \langle 1\rangle, \langle 1,1\rangle, \langle 2\rangle\}$ and the second by $\{\varepsilon, \langle 1\rangle, \langle 2\rangle, \langle 2,1\rangle\}$. In the graph-theoretic sense, we would expect these to be different pictures of the same tree, but for our purposes they are different trees.

Definition 1. We define an *arena* to be a finite tuple of nonempty trees of *moves*.

For example, the empty tuple $\langle\rangle$ is an arena, which we call the *empty arena*; $\langle\{\varepsilon\}\rangle$ is the minimal one-tree arena consisting of a root node; the maximal one-tree arena, consisting of an infinitely deep, infinitely branching tree, is $\langle\mathbb{N}^*\rangle$. As the empty arena, the minimal and maximal one-tree arenas are important, we shall name them E, M and U respectively.

The root of each tree is called an *initial move*. We refer to the *depth* of an element of a tree, which is the length of the sequence which encodes it. Thus the root of a tree is at depth zero. We say that moves at an even depth of the trees (including the roots at depth 0) are *O-moves*, and moves at an odd depth are *P-moves*. O-moves are often represented in diagrams by $\bullet$ and P-moves by $\circ$.

We assume that the reader is familiar with the construction of *product arena* (notation $A\times B$) and *function space arena* ($A \Rightarrow B$) of two arenas A and B. We also assume familiarity with the notions *justified sequence*, *P-view* and *O-view*, *legal position*, *P-strategy*, usually referred to just as strategy, *O-strategy* and *innocence* which are all defined in the standard way. If we have strategies σ and τ on arenas $A \Rightarrow B$ and $B \Rightarrow C$ respectively then we denote their *composite strategy* on $A \Rightarrow C$ by $\sigma;\tau$. Please refer to [12] for motivation and formal definitions of these notions.

In this paper we will be dealing only with strategies that are innocent. An innocent strategy is completely determined by its innocent function, a partial function from odd-length P-views to justified P-moves. And we will identify strategies with their innocent functions as in [16]. An innocent strategy is said to be *recursive* if the innocent function representing it is recursive. It is easy to see that the composition of two recursive innocent strategies is itself recursive.

Definition 2. Objects of the *Category of Arenas and Innocent Strategies*, $\mathbb{A}$, are arenas (in tuple-subset form); morphisms $f : A \to B$ are innocent strategies on the function space arena $A \Rightarrow B$. Composition of morphisms is composition as strategies. The *Category of Arenas and Recursive Innocent Strategies*, $\mathbb{A}_{\mathrm{REC}}$, has recursively enumerable arenas as objects and recursive innocent strategies as morphisms.

Theorem 3. *$\mathbb{A}$ and $\mathbb{A}_{REC}$ both cartesian closed*

The terminal object **1** of both $\mathbb{A}$ and $\mathbb{A}_{REC}$ is the empty arena E, and the categorical constructions of product and function space are exactly the respective arena constructs. The category $\mathbb{A}$ is enriched over dI-domains. (One cannot say the same of $\mathbb{A}_{REC}$, because the computable partial functions do not form a cpo. For example, one can "approximate" the Halting Problem by computable functions.)

Scott has observed that every λ-algebra arises from a reflexive object R in some cartesian closed category $\mathbb{C}$. We give the construction explicitly as it is needed for analysis later (although the reader is referred to [4] for a comprehensive treatment).

Given a cartesian closed $\mathbb{C}$ with reflexive object R, together with morphisms Fun : $R \to [R \Rightarrow R]$ and *Graph* : $[R \Rightarrow R] \to R$ such that $Graph; Fun = \mathrm{id}_{[R\Rightarrow R]}$, define a λ-algebra $\langle \mathscr{A}, \cdot, [\![-]\!]^{-} \rangle$ as follows:

(1) $\mathscr{A}$ is the homset $\mathrm{Hom}_{\mathbb{C}}(1, R)$.
(2) For any object A with $f, g : A \to R$ define $f \cdot g = \langle f; Fun, g\rangle; \mathrm{eval}_{R,R}$. In particular this defines a binary operation on $\mathscr{A}$.
(3) If $\{x_1, \ldots, x_n\} \supseteq \mathrm{FV}(s)$ define inductively the morphism $[\![s]\!]_\Delta : R^n \to R$, where $\Delta = \langle x_1, \ldots x_n \rangle$, as follows:

$$\begin{aligned} [\![x]\!]_\Delta &= \Pi^\Delta_x \text{ (the obvious projection morphism)},\\ [\![st]\!]_\Delta &= [\![s]\!]_\Delta \cdot [\![t]\!]_\Delta,\\ [\![\lambda x.s]\!]_\Delta &= \Lambda([\![s]\!]_{\Delta\cdot x}); Graph. \end{aligned}$$

In the last clause we may assume that x does not appear in Δ (by renaming if necessary).

(4) If ρ is a valuation mapping variables to elements of $\mathscr{A}$, and Δ is as above, define the morphism $\rho^\Delta : 1 \to R^n$ by $\rho^\Delta = \langle \rho(x_1), \ldots, \rho(x_n)\rangle$. Then set

$$[\![s]\!]_\rho = \rho^\Delta; [\![s]\!]_\Delta.$$

Thus we may specify a λ-algebra by a 4-tuple which we shall write as $\mathscr{M}(\mathbb{C}, R, Fun, Graph)$ (it is in fact a $\lambda\eta$-algebra if $Fun; Graph = \mathrm{id}_R$). If the reflexive object R *has enough points* (i.e. $\forall f, g : R \to R.\ [\forall r : 1 \to R.\ r; f = r; g] \Rightarrow f = g$) then $\mathscr{M}(\mathbb{C}, R, Fun, Graph)$ is a *λ-model* (i.e. a weakly extensional λ-algebra).

Note that the arena U has the key property that $U = U \Rightarrow U$ so that in this case the morphisms *Fun* and *Graph* are both the identity on U. We can now define the first two of our game λ-algebras (which are both $\lambda\eta$-algebras): $\mathscr{M}(\mathbb{A}, U, \mathrm{id}_U, \mathrm{id}_U)$ which we shall write simply as $\mathscr{D}$, and $\mathscr{M}(\mathbb{A}, U, \mathrm{id}_U, \mathrm{id}_U)$ which we shall write as $\mathscr{D}_{REC}$. By abuse of notation, we shall use $\mathscr{D}$ and $\mathscr{D}_{REC}$ to denote the respective underlying sets. Clearly $\mathscr{D}_{REC} \subset \mathscr{D}$. By a method of approximation introduced in [12] we can show that both the $\lambda\eta$-algebras are *sensible*, i.e. all unsolvable λ-terms have the same denotation which in this case is given by the everywhere undefined innocent function.

We can encode innocent strategies σ, over any single-tree arena, as (partial) maps from $\mathbb{N}^*$ to $\mathbb{N} \times \mathbb{N}_0$. We call this encoding the *economical form* of σ and denote it

f_σ. It is defined as follows:

$$f_\sigma : \langle v_1, \ldots, v_n \rangle \mapsto (i, p) \text{ if and only if}$$

$$\sigma : \overset{\bullet}{\varepsilon} \quad \overset{\circ}{\vec{s}_1} \quad \overset{\bullet}{\vec{s}_1 v_1} \quad \overset{\circ}{\vec{s}_2} \quad \overset{\bullet}{\vec{s}_2 v_2} \quad \cdots \quad \overset{\circ}{\vec{s}_{n-p}} \quad \overset{\bullet}{\vec{s}_{n-p} v_{n-p}} \quad \cdots \quad \overset{\circ}{\vec{s}_n} \quad \overset{\bullet}{\vec{s}_n v_n} \quad \mapsto \quad \overset{\circ}{\vec{s}_{n-p}(v_{n-p} i)}$$

(Here the line drawn indicates that $\vec{s}_{n-p}(v_{n-p}i)$ is explicitly justified by $\vec{s}_{n-p}v_{n-p}$. When the resulting move from a clause of the innocent function is a child of the root node we have $n = p$, and the intention is that the (non-existent) move $\vec{s_0}v_0$ should mean the initial move ε.)

Justification pointers within the P-view can be deduced from the behaviour of σ on shorter P-views, and they do not affect the value of $f_\sigma\langle v_1, \ldots, v_n \rangle$, and so have been omitted. Note that each $\vec{s_i}$ is a sequence of natural numbers coding a move of the arena.

Example 4. The following is the innocent function of the "copycat" strategy

$$\overset{\bullet}{\varepsilon} \;\mapsto\; \overset{\circ}{1} \qquad\qquad \overset{\bullet}{\varepsilon} \;\cdots\; \overset{\bullet}{(a{+}1)\vec{s}} \;\; \overset{\circ}{1a\vec{s}} \;\; \overset{\bullet}{1a\vec{s}b} \;\mapsto\; \overset{\circ}{(a{+}1)\vec{s}b}$$

$$\overset{\bullet}{\varepsilon} \;\; \overset{\circ}{1} \;\; \overset{\bullet}{1a} \;\mapsto\; \overset{\circ}{(a{+}1)} \qquad\qquad \overset{\bullet}{\varepsilon} \;\cdots\; \overset{\bullet}{1a\vec{s}} \;\; \overset{\circ}{(a{+}1)\vec{s}} \;\; \overset{\bullet}{(a{+}1)\vec{s}b} \;\mapsto\; \overset{\circ}{1a\vec{s}b}$$

Here $\vec{s}$ range over sequences of appropriate parity, a and b over positive natural numbers. The reader is invited to check that the economical form of this strategy is given by: $\varepsilon \mapsto (1, 0), i \mapsto (i + 1, 1)$ and for nonempty sequences $\vec{v}$, $\vec{v}i \mapsto (i, 1)$.

A principle of the λ-calculus is that a term can be applied successively to any other term. So the term $\lambda x.x$ (say) is really more like "$\lambda x z_0 z_1 z_2 \ldots \bullet x z_0 z_1 z_2 \ldots$" (we use a large dot $\bullet$ to make the "end" of the infinite chain of abstractions really clear). Thus there is some notion of infinite η-expansion. If we think about the denotation of $\lambda x.x$ in the game models, it is similarly expanded — it copies the whole of the first subtree to the rest of the arena, as if copying not only the x variable but also all of its arguments. This correspondence turns out to be general, and can be made precise by relating innocent strategies in economical form to a kind of (infinitely) η-expanded Böhm trees first studied by Nakajima in [15]. We call a formal connexion of this form an *Exact Correspondence Theorem.*

For a λ-term s the *Nakajima tree* of s, written NT(s), is (informally) the countably branching, countably deep tree labelled as follows. If s is unsolvable then $\mathrm{NT}(s) = \bot$. If s has head normal form $\lambda x_1 \ldots x_n . y s_1 \ldots s_m$ then

$$\mathrm{NT}(s) \;=\; \begin{array}{c} \lambda x_1 \ldots x_n z_0 z_1 \ldots \bullet y \\ \swarrow \quad \swarrow \quad \downarrow \quad \searrow \\ \mathrm{NT}(s_1) \;\; \cdots \;\; \mathrm{NT}(s_m) \;\; \mathrm{NT}(z_0) \;\; \mathrm{NT}(z_1) \;\; \cdots \end{array}$$

where $z_0, z_1, \dots$ are countably many fresh variables. The process of finding such fresh variables given in [15] is quite complicated.

In [12] we propose a *variable-free* representation of Nakajima trees so that for a closed term s, $\mathrm{NT}(s)$ is represented as $\mathrm{VFF}(s)$, a partial function from $\mathbb{N}^*$ to $\mathbb{N} \times \mathbb{N}_0$. Note that the "infinitely nested" λ-abstractions of the form $\lambda z_1 z_2 \dots \cdot y$, which label the nodes of a Nakajima tree (of a closed term), can be coded as a pair (i, r) whereby the head variable y is the i^{th} in the infinite list of variables bound by the λ-abstraction situated r levels up in the tree. The map $\mathrm{VFF}(s)$ is just a function that maps occurrences (of nodes) to such labels encoded as pairs of numbers.

We formalize this as follows:

Definition 5. For a partially $(\mathbb{N} \times \mathbb{N}_0)$-labelled tree p the tree $\{p\}^*$ is the same tree labelled identically, except that nodes at depth d labelled $(i, d+1)$ are relabelled $(i, d+2)$.

Similarly the tree $\{p\}^n$, for $n \in \mathbb{N}_0$, is labelled identically except for nodes of depth d as follows:

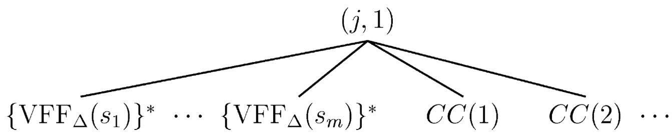
(1) those labelled (i, d) are relabelled $(i+n, d)$;
(2) those labelled $(i, d+1)$ for $i \leqslant n$ are relabelled $(n-i+1, d)$;
(3) those labelled $(i, d+1)$ for $i > n$ are relabelled $(i-n, d+1)$.

For a term s with free variables within Δ the *variable-free form* of the Nakajima tree of s, $\mathrm{VFF}_\Delta(s)$, is the following partially $(\mathbb{N} \times \mathbb{N}_0)$-labelled tree:

$$\mathrm{VFF}_\Delta(s) = \bot, \text{for unsolvable } s.$$

$$\mathrm{VFF}_\Delta(\lambda x_1 \dots x_n \cdot s) = \{\mathrm{VFF}_{\Delta \cdot \langle x_1, \dots, x_n \rangle}(s)\}^n, \text{if } s \text{ is of the form } v_j s_1 \dots s_m.$$

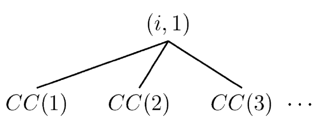
$$\mathrm{VFF}_\Delta(v_j s_1 \dots s_m) =$$



where $\Delta = \langle v_k, \dots, v_1 \rangle$ (note the reverse order).

Here $CC(i)$ is the infinite tree defined by

$CC(i) =$



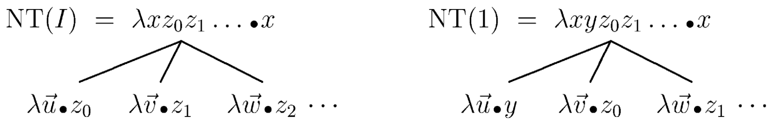
To see $\mathrm{VFF}_\Delta(s)$ as a partial function from $\mathbb{N}^*$ to $\mathbb{N} \times \mathbb{N}_0$, note that we can identify the labelled tree with the labelling function, which maps the encoding of each node in $\mathbb{N}^*$ to the pair labelling that node.

Lemma 6. *This definition coincides with the informal notion of variable-free form described earlier.*

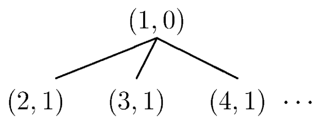
The following theorem, a key result in [12], connects the economical form of an innocent strategy denoting a term, with the variable free form of the Nakajima tree of that term.

Theorem 7 (Exact Correspondence). *For every closed λ-term s, the innocent strategy denotation $[\![s]\!]$ in both $\mathscr{D}$* and $\mathscr{D}_{\text{REC}}$, *given in economical form, is exactly* VFF(s), *the Nakajima tree of s in variable free form.*

Example 8. We now introduce example terms and strategies which we will use repeatedly to illustrate many of the concepts in the rest of the paper. Consider the terms $I = \lambda x.x$ and $1 = \lambda xy.xy$. The reader may wish to verify that the following represents the first two levels of the Nakajima trees of those terms:



After renaming of bound variables, these are the same. Since I and 1 differ only by η-conversion, this should be no surprise. Thus we can calculate their common variable-free form, the first two levels of which is:



For example, the node labelled $(2,1)$ means that the head variable of the corresponding node in the Nakajima tree is found as the second in the list of variables abstracted at the node one level above. The Exact Correspondence Theorem tells us that $[\![I]\!] = [\![1]\!]$ has the economical form which is given (in part) by $\varepsilon \mapsto (1,0), \langle 1\rangle \mapsto (2,1), \langle 2\rangle \mapsto (3,1)$ and so on.

We say that a λ-algebra is *universal* if every element is the denotation of some λ-term. By the Exact Correspondence Theorem, it is easy to see that neither $\mathscr{D}$ nor $\mathscr{D}_{\text{REC}}$ is universal, since no *non-trivial* compact innocent strategy can be the denotation of any λ-term (note that the only finite Nakajima tree is the trivial tree $\bot$).

Our aim in the rest of this section is to characterise the definable parts of $\mathscr{D}_{\text{REC}}$, and we shall do so by capturing the right ambient CCC.

Notation. For tree-like $A \subseteq \mathbb{N}^*$ (i.e. those subsets which are prefix-closed and satisfy $\vec{s}\cdot n \in A \Rightarrow \vec{s}\cdot m \in A$ for all $m < n$) and for any $\vec{s} \in A$ we define $A@\vec{s}$ to be the subtree of A rooted at $\vec{s}$ and $A^{>m}$ to be the tree obtained from A by deleting the first m branches. For example, for the maximal single-tree arena U, we have $U@\vec{s} = U = U^{>n}$ for all sequences $\vec{s}$ and numbers n. Next fix an innocent strategy in economical form f and let $\vec{v} \in \text{dom}(f)$. We shall use the following shorthand: $\mathbf{m}_{\mathbf{o}}^{f}(\vec{v})$ denotes the last move of the P-view encoded by $\vec{v}$ and $\mathbf{m_p}^f(\vec{v})$ denotes the response of f at the P-view. Note that the former is by definition an O-move and the latter a P-move. We omit the superscript f wherever it is clear which strategy is intended. For example, for any innocent strategy f the O-move $\mathbf{m_o}(\varepsilon)$ is the initial move ε and $\mathbf{m_p}(\varepsilon)$ is the first P-move made by f in response. Now we can define a new property of strategies:

Definition 9. Consider an innocent strategy in economical form $f : \mathbb{N}^* \to \mathbb{N} \times \mathbb{N}_0$, over some single-tree arena A. We say that f is *everywhere copycat* (EC) at $\vec{v} \in \mathbb{N}^*$ if f is undefined at $\vec{v}$ or the following hold:

(EC1) The arenas $A@\mathbf{m_o}(\vec{v})$ and $A@\mathbf{m_p}(\vec{v})$ are isomorphic.

(EC2) Whenever $\vec{w} \geqslant \vec{v}$ we have that for all $i \in \mathbb{N}$ such that the move coded by $\vec{w} \cdot i$ exists, $f(\vec{w} \cdot i) = (i, 1)$.

(EC3) If $f(\vec{v}) = (i, p)$ then $p > 0$.

We say that f is *almost-everywhere copycat* (AC) at $\vec{v}$ if f is undefined at $\vec{v}$ or there exist numbers $t_v \in \mathbb{N}_0$ and $o_v \in \mathbb{Z}$ with $o_v \leqslant t_v$ called the *copycat threshold* and *offset* respectively, such that

(AC1) The arenas $(A@\mathbf{m_o}(\vec{v}))^{>(t_v - o_v)}$ and $(A@\mathbf{m_p}(\vec{v}))^{>t_v}$ are isomorphic.

(AC2) For all $i > t_v$ such that the move coded by $\vec{v} \cdot i$ exists, $f(\vec{v} \cdot i) = (i - o_v, 1)$ and f is everywhere copycat at $\vec{v} \cdot i$.

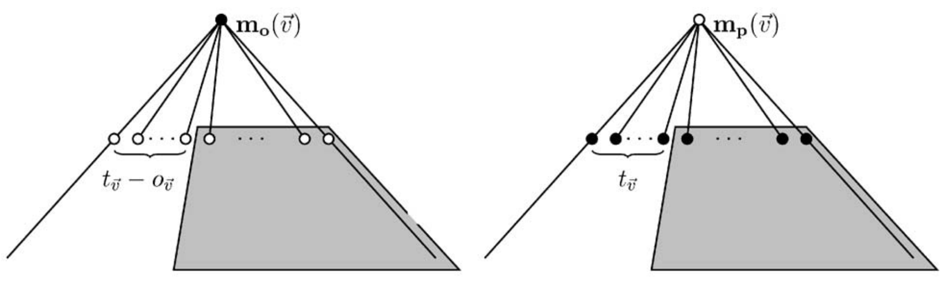
(AC3) For all $\vec{w} \geqslant (\vec{v} \cdot k)$ with $k \leqslant t_v$, if $f(\vec{w}) = (i, |\vec{w}| - |\vec{v}|)$ then $i \leqslant t_v - o_v$.

(AC4) If $f(\vec{v}) = (i, 0)$ then $i \leqslant t_v - o_v$.

Note that f is EC at $\vec{v}$ if and only if f is AC at $\vec{v}$ with $t_v = o_v = 0$.

Finally, we say that f is *effectively almost-everywhere copycat* (EAC) if f is computable, almost-everywhere copycat at every sequence on which it is defined and there are computable functions $\vec{v} \mapsto t_v$ and $\vec{v} \mapsto o_v$ giving valid thresholds and offsets respectively. A strategy σ over an arena A is EAC if its innocent function is EAC, and we can generalise to multiple-tree arenas in the usual way.

We illustrate the effect of (AC1) and (AC2) as follows. Suppose P plays a strategy which is almost-everywhere copycat at $\vec{v}$. The two arenas $A@\mathbf{m_o}(\vec{v})$ and $A@\mathbf{m_p}(\vec{v})$ are shown below.



The idea is that except for finitely many subtrees of the moves in question, P's behaviour is "everywhere copycat" at $\mathbf{m_o}(\vec{v})$ i.e. P simply copies O's move between two isomorphic subarenas (which are shaded in the figure). Conditions (AC3) and (AC4) ensure that P never plays in the shaded area unless forced by (AC2).

Since the notion of EAC is only defined for innocent strategies, we will just say "EAC strategy" instead of "EAC innocent strategy".

For a specific P-view $\vec{v}$ of such a function f, we will say that t_v and o_v are *valid* copycat threshold and offset, respectively, if f satisfies the conditions (AC1)–(AC4) at that P-view with those particular values. Valid copycat thresholds are not unique, as the following lemma shows.

Lemma 10. If $f : \mathbb{N}^* \to \mathbb{N} \times \mathbb{N}_0$ is an innocent strategy in economical form, and f is defined and AC at some P-view $\vec{v}$ with copycat threshold and offset t_v and o_v respectively, then for any $t' \geqslant t_v$, f is also AC at the P-view $\vec{v}$ with threshold and offset t' and o_v respectively. That is, any value larger than a valid copycat threshold is still a valid threshold for a specific P-view (with the same offset).

Thus at each P-view of an EAC strategy there will be a *least copycat threshold*, the least value for t_v which is still a valid threshold. However, the existence of a computable function giving valid thresholds does not imply the computability of the function giving least thresholds.

Definition 11. The *category of arenas and EAC strategies*, $\mathbb{A}_{\text{EAC}}$, has recursively enumerable arenas as objects and EAC strategies on $A \Rightarrow B$ as morphisms from A to B.

A main result in [12] is that the category $\mathbb{A}_{\text{EAC}}$ is well-defined; the proof that EAC strategies compose is omitted from this reference, but is in fact a simple consequence of Theorem 20 which appears later in this paper. Additionally,

Theorem 12. $\mathbb{A}_{\text{EAC}}$ *is cartesian closed.*

The arena U is still an object of $\mathbb{A}_{\text{EAC}}$ and still equal to its function space. Thus, in the same way as before, we can define a $\lambda\eta$-algebra $\mathscr{M}(\mathbb{A}_{\text{EAC}}, U, \text{id}_U, \text{id}_U)$ which we shall denote by $\mathscr{D}_{\text{EAC}}$. Since $\mathbb{A}_{\text{EAC}}$ is a subcategory of $\mathbb{A}_{\text{REC}}$, with the same class of objects and the same cartesian closed structure, $\mathscr{D}_{\text{EAC}} \subset \mathscr{D}_{\text{REC}}$.

Other properties of $\mathscr{D}, \mathscr{D}_{\text{REC}}$ and $\mathscr{D}_{\text{EAC}}$ are explored in [12]; one which is immediate from the Exact Correspondence Theorem is that two terms of the λ-calculus have the same denotation precisely when they have the same Nakajima tree. This equality is captured by the maximal consistent sensible theory $\mathscr{H}^*$ (see [4]). Also, almost by construction, every EAC strategy on U is the denotation of some closed term of the λ-calculus. Furthermore, $\mathscr{D}_{\text{EAC}}$ inherits the order-extensionality property which the syntax of the λ-calculus (modulo the theory $\mathscr{H}^*$) enjoys. Thus:

Theorem 13. *The equational theory of the models* $\mathscr{D}, \mathscr{D}_{\text{REC}}$ *and* $\mathscr{D}_{\text{EAC}}$ *is* $\mathscr{H}^*$, *the maximal consistent sensible theory. Additionally*, $\mathscr{D}_{\text{EAC}}$ *is a universal and order-extensional* λ*-model.*

3. Effectively and Explicitly Almost-Everywhere Copycat Strategies

Our aim in this sequel paper is to find a game model which does not validate η-conversion. To do so, we will need to augment EAC strategies with some extra information. By examining the parts of Nakajima trees which use fresh variables, in the light of the Exact Correspondence Result, it becomes clear that the additional information we require is that which specifies copycat thresholds at each P-view. We are thus lead to the definition of an *EXAC strategy*, and we describe the problems involved in the search for a CCC of such strategies, and the solution $\mathbb{XA}$.

3.1. Specifying Copycat Thresholds

To find a model in which η-conversion is not validated, we require the terms I and 1 to be denoted differently. They have the same variable-free form of Nakajima tree, so it is not apparent how this might be achieved. The key is to make use of the fact that the copycat thresholds are not unique — any number greater than a given valid copycat threshold is also a valid copycat threshold (Lemma 10). Different thresholds (at some P-view) may be used to distinguish I and 1.

This idea is prompted by the observation that when one compares a term with its denotation, the part of the EAC strategy which is specified by the rules of copycat, the part which is not specified *explicitly*, corresponds precisely to the part of the Nakajima tree which has been generated by η-expansion (i.e. the part of the tree with the fresh variables as the head variables). Recall the Nakajima trees of I and 1 — the former has fresh variables appearing at every node except the root, whereas the latter is similar except that there is not a fresh variable at the first child of the root. Therefore we aim to find a model where I and 1 are represented by the same strategy, but the copycat threshold of $[\![I]\!]$ at the first P-view is 0, whereas that of $[\![1]\!]$ is 1.

However, the definition of an EAC strategy is stated in terms of the existence of some computable function which associates a pair of numbers to each P-view of the strategy and this function is *not* specified along with the strategy. (A consequence of this is that there is no computable procedure for finding valid thresholds for an EAC strategy, nor for deciding whether a given innocent strategy is EAC.)

Remark 14. It is really the thresholds (rather than the offsets) which are important because, for a certain P-view $\vec{v}$ of an EAC strategy σ, the copycat threshold t usually gives enough information to compute the offset o directly. This is clear since $f_\sigma(\vec{v}\cdot(t+1)) = (t+1-o, 1)$, as long as the move coded by $\vec{v}\cdot(t+1)$ actually exists in the arena in question. When this move does not exist, any value would do for the offset. However, in this case the value of the offset is irrelevant and we will not take this technicality into account. Moreover when none of the moves coded by $\vec{v}\cdot i$ exist for any i, the threshold is irrelevant as well. We will not distinguish between strategies which only differ in such circumstances. In practice, we only consider strategies over the arena U, in which all such moves always exist, so this technicality can be ignored.

This motivates the following definition:

Definition 15. An *effectively and explicitly almost-everywhere copycat strategy* (EXAC strategy) is given by a pair $\langle \sigma, t_\sigma \rangle$, where σ is an EAC strategy and t_σ is an effective function mapping the P-views where σ is defined to valid copycat thresholds. We sometimes write the EXAC strategy $\langle \sigma, t_\sigma \rangle$ just as σ.

We will usually refer to the first and second part of an EXAC strategy as the "(underlying) EAC strategy (part)" and the "threshold function (part)", respectively. In view of our remark above, however, we will sometimes speak of the offsets as if they too are specified by the threshold function.

This definition allows us to make the intended finer distinction between strategies: two strategies with the same moves must be equal as EAC strategies, but may have different copycat thresholds and so can be distinguished as EXAC strategies. There is an obvious forgetful map from EXAC strategies to EAC strategies, which takes only the strategy part (i.e. erasing the threshold information).

In a similar vein to the economical form of innocent strategies, using the same encoding of a P-view as a sequence of natural numbers, we can give an economical form of EXAC strategies over single-tree arenas. We can also take advantage of the fact that parts of the strategy are completely dictated by its copycat nature. Let us say that a P-view is *entirely explicit* if none of the O-moves in it exceed the given copycat threshold of the P-view at which they are made. Thus if a P-view is not entirely explicit the ensuing move can be deduced from the threshold and offset of the P-view preceding the first O-move in it which did exceed the copycat threshold.

Definition 16. The *economical form* of an EXAC strategy is a map from $\mathbb{N}^*$ to $\mathbb{N} \times \mathbb{N}_0 \times \mathbb{N}_0 \times \mathbb{Z}$. The domain is the encoding of P-views in the same way as economical form of an EAC strategy. The map is defined at a sequence $\vec{v}$ only if the P-view encoded by $\vec{v}$ is entirely explicit, in which case

$$\vec{v} \mapsto (i, r, t, o)$$

where the resulting P-move is encoded as before — it is the i^{th} child of the move $2r$ from last of the P-view — and the copycat threshold and offset at this P-view are t and o, respectively.

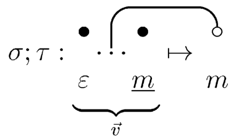
Example 17. We take the EXAC strategies η_0 and η_1 to be $\langle [\![I]\!], t_0 \rangle$ and $\langle [\![1]\!], t_1 \rangle$, where t_0 maps every P-view to the threshold 0 and t_1 does likewise except that the minimal P-view is mapped to the threshold 1. Since $[\![I]\!] = [\![1]\!]$, they have the same EAC strategy part, but different threshold functions. These are the suggestions we made for the denotations of I and 1 in a model not supporting η-conversion. Nearly every P-view of either is not entirely explicit, and the respective economical forms are given by:

$$\varepsilon \mapsto (1,0,0,-1) \text{ and } \varepsilon \mapsto (1,0,1,-1)$$
$$\langle 1 \rangle \mapsto (2,1,0,0)$$

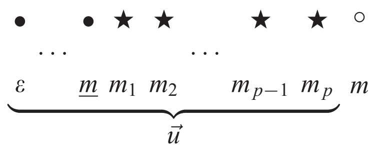
3.2. Composition of EXAC Strategies

We now need a method to compose EXAC strategies. Of course the EAC strategy part will just be the standard composition of innocent strategies, and we give below an algorithm for computing the composition of the threshold functions.

Algorithm 18 (The Composition Algorithm). Let $\langle \sigma, t_\sigma \rangle$ be an EXAC strategy over $A \Rightarrow B$, and $\langle \tau, t_\tau \rangle$ be an EXAC strategy over $B \Rightarrow C$. Take a P-view $\vec{v}$ on which the strategy $\sigma;\tau$ (which is given by the usual composition of innocent strategies) is defined and suppose that the last move of the P-view is $\underline{m}$ and the resulting move is m:



In order to be able to calculate a valid threshold and offset for m we have to look at the moves m_i that are the intermediate interactions which might have taken place between σ and τ starting at the P-view $\vec{v}$ before the move m became the visible outcome.



The moves m_i are all in the arena B and they are all both O- and P-moves, depending on which strategy is looking at them which is why they are written ★. Possibly there are no such intermediate moves, in which case $p = 0$. We do not care about justification pointers, and for tidiness set $m_0 = \underline{m}$ and $m_{p+1} = m$. We call this sequence $\vec{u} = \mathbf{u}(\vec{v}, \sigma, \tau)$ the *uncovering* of the composition up to the move m.

For $1 \leqslant i \leqslant p+1$ we consider the P-view that the strategy σ (respectively τ) is faced with when making the move m_i, and denote this P-view by $\vec{u_i}$.

Define t_i and o_i to be the copycat threshold and offset of σ, or τ as appropriate, at the P-view $\vec{u_i}$. These are specified by t_σ or t_τ. Then set:

$$T_i = \begin{cases} t_i + |A| & \text{if } m_i \text{ is a root of } B \\ t_i, & \text{otherwise,} \end{cases} \qquad O_i = \begin{cases} o_i + |A| & \text{if } m_i \text{ is a root of } B, \\ o_i, & \text{otherwise,} \end{cases}$$

$$\hat{T}_1 = T_1 \qquad\qquad \hat{O}_1 = O_1,$$

$$\hat{T}_{i+1} = \max(\hat{T}_i + O_{i+1}, T_{i+1}), \qquad\qquad \hat{O}_{i+1} = \hat{O}_i + O_{i+1},$$

$$t = \hat{T}_{p+1} \qquad o = \begin{cases} \hat{O}_{p+1} - |A| + |B| & \text{if } \underline{m} \text{ is a root of } C, \\ \hat{O}_{p+1} & \text{otherwise.} \end{cases}$$

Then t and o are the copycat threshold and offset of the composition $\langle \sigma, t_\sigma \rangle; \langle \tau, t_\tau \rangle$ at the P-view $\vec{v}$.

Now we must show that this method does indeed produce an EXAC strategy, i.e. that the composite threshold function specifies valid thresholds and offsets for the composite strategy. In fact it does so only under some restrictions, for which we need an additional definition.

Definition 19. Let σ be an EAC strategy over a single-tree arena. If σ has a first move, then it has a copycat threshold and offset, say t and o, at the P-view consisting only of the root O-move. The *l-number* of σ is the value $t - o$, and we write it $\mathbf{1}(\sigma)$. If σ is undefined we set $\mathbf{1}(\sigma) = \infty$.

If $\sigma = \langle \sigma_1, \ldots, \sigma_n \rangle$ is an EAC strategy over an arena with n trees, then we define $\mathbf{1}(\sigma) = \min_{i=1}^{n} \{\mathbf{1}(\sigma_i)\}$.

We also use the notation $|A|$ for the number of trees in the arena A.

Theorem 20. *If $\sigma : A \Rightarrow B$ and $\tau : B \Rightarrow C$ are EAC strategies satisfying $\mathbf{1}(\sigma) \geqslant |A|$ and $\mathbf{1}(\tau) \geqslant |B|$ then Algorithm 18 produces valid copycat thresholds and offsets for $\sigma; \tau$.*

The proof of this theorem may be found in the appendix. The condition on the l-number for σ (respectively for τ) ensures that none of the copy-cat areas is intersecting both parts A and B (respectively B and C).

We will also need the following:

Lemma 21. *Composition of EXAC strategies is associative.*

The proof is straightforward and we omit it. One proceeds by examining the uncovering of three strategies together (that is, the sequence including intermediate moves in both hidden arenas) and relating this to the uncoverings of the two different bracketings of the triple composition.

3.3. The Category $\mathbb{A}_{\text{EXAC}}$

We first define the "obvious" category, which derives directly from the conditions required for the composition algorithm to work correctly. Perhaps surprisingly we can show that this does not give rise to a CCC.

Definition 22. The *category of arenas and EXAC strategies*, written $\mathbb{A}_{\text{EXAC}}$, is defined as follows:

(1) Objects are recursively enumerable arenas.
(2) The morphisms from A to B are the EXAC strategies on the arena $A \Rightarrow B$ which have l-number greater than or equal to $|A|$, or are everywhere undefined.

(3) The identity morphism on A is the EXAC strategy $\langle \mathrm{id}_A, 0\rangle$, i.e. the copycat threshold is zero everywhere. The offset at the minimal P-view is $-|A|$, and 0 everywhere else.
(4) Composition of morphisms is given by composition of EXAC strategies via Algorithm 18.

One can show that this does indeed specify a category. Also, $\mathbb{A}_{\mathrm{EXAC}}$ has the obvious terminal object E (the arena consisting of no trees defined in Section 2) and products given in the usual way. However, $\mathbb{A}_{\mathrm{EXAC}}$ does not form a CCC with the usual constructions.

Suppose that $\sigma : A \times B \to C$. Then we know that $\mathbf{1}(\sigma) \geqslant |A| + |B|$. We need a morphism $\Lambda(\sigma) : A \to B \Rightarrow C$, which must have l-number at least $|A|$, so we could take $\Lambda(\sigma)$ to be the same EXAC strategy as σ. However there can be more than one strategy satisfying the required universal property for $\Lambda(\sigma)$ as the following example illustrates.

Consider η_0 and η_1 as defined earlier in this section. Note that the only difference between the two strategies is their threshold for the initial P-view. One can verify that both η_0 and η_1 can be considered as morphisms $U \to U \Rightarrow U$ and that in this case $\eta_0 \times \mathrm{id}_U; \mathrm{eval}_{U,U} = \eta_1 \times \mathrm{id}_U; \mathrm{eval}_{U,U} : U \times U \to U$, and that this is the same as the morphism $U \times U \to U$ described by η_1. Hence there are two candidates for $\Lambda(\eta_1)$.

It is not clear that $\mathbb{A}_{\mathrm{EXAC}}$ forms a CCC with any unusual constructions either.

In order to fix this, we made another attempt. The problem with $\mathbb{A}_{\mathrm{EXAC}}$ is that the conditions for an EXAC strategy to be a morphism $A \to B \Rightarrow C$ are weaker than those to be a morphism on $A \times B \to C$. One solution might be the following:
(1) An object is a pair (A, n) where $n \in \mathbb{N}_0$.
(2) A morphism $\sigma : (A, n) \to (B, m)$ is an EXAC strategies on $A \Rightarrow B$ such that $\mathbf{1}(\sigma) \geqslant m + |A|$.
(3) Composition is composition of EXAC strategies.
(4) The identity on (A, n) is $\langle \mathrm{id}_A, t\rangle$, where t is the function mapping the minimal P-view to n and the others to zero.

Then we can set $(A, n) \Rightarrow (B, m) = (A \Rightarrow B, m + |A|)$, which gives the same set of morphisms $A \times B \to C$ and $A \to B \Rightarrow C$.

However, in this case, the identity will not work correctly! Sometimes the thresholds of $\sigma; \mathrm{id}$ come out greater than σ. To fix this, we find we must include information in the objects specifying minimal thresholds for the morphisms. But this breaks the function spaces again, and we have to add information specifying the minimum l-number for some other P-views, whereupon there are again problems with identities...

What we believe to be the least fixed point of this fixing process is presented in the next section.

3.4. The Category $\mathbb{X}\mathbb{A}$

We now present a new category based on EXAC strategies, which does form a CCC. Although it is more complicated than the "almost-CCC" $\mathbb{A}_{\mathrm{EXAC}}$, it seems to be the simplest way to construct a CCC.

Firstly we say that an arena $A = \langle A_1, \ldots, A_m \rangle$ is a *subarena* of $B = \langle B_1, \ldots, B_n \rangle$ if $m = n$ and for each i, A_i is a subset of B_i. We say that an arena is *finitely-branching* if every tree in it is finitely branching.

Then, for any move m of a finitely-branching arena A, let us write $\text{br}(A@m)$, to mean the number of direct children of m in A.

We make the following definition:

Definition 23. Let A be an arena and X a finitely branching recursively enumerable subarena of A. We say that an EXAC strategy σ over A is X-*explicit* if the following holds:

Let $\sigma : \vec{v} \mapsto (i, r, t, o)$ be the economical form of any clause of the innocent function. Suppose that the sequence $\vec{v}$ codes a P-view ending in the O-move $\underline{m}$, and that the consequent P-move encoded by this clause is m. Then

(EX1) if $\underline{m}$ is in the subarena X then $t - o \geqslant \text{br}(X@\underline{m})$,

(EX2) if m is in the subarena X then $t \geqslant \text{br}(X@m)$.

An intuitive description of this definition is the following: The subarena X determines a part of the arena A where the strategy is known to be explicitly defined, i.e. no move from X is in the domain or in the range of automatic copycat forced by the threshold information of σ. This means that given a strategy σ over A which is X-explicit, any P-view of σ, such that all of its moves are in X, is entirely explicit.

We are now in a position to define a category of EXAC strategies that is cartesian closed.

Definition 24. The category $\mathbb{XA}_{\text{EXAC}}$, or simply $\mathbb{XA}$, is given by the following:

(1) Objects are pairs (A, X) consisting of a r.e. arena A and a finitely-branching r.e. subarena X.
(2) A morphism $\sigma : (A, X) \to (B, Y)$ is an EXAC morphism on $A \Rightarrow B$ which is $(X \Rightarrow Y)$-explicit.
(3) Composition of morphisms is composition of EXAC strategies via Algorithm 18.
(4) The identity strategy on (A, X), $\text{id}_{(A,X)}$, is the EXAC strategy $\langle \text{id}_A, t \rangle$, where id_A is the EAC identity strategy on A, and t is the function that takes the least value on every P-view which still leaves the EXAC strategy $\langle \text{id}_A, t \rangle$ as $(X \Rightarrow X)$-explicit.

Theorem 25. $\mathbb{XA}$ is indeed a category.

Proof. We already know that composition is associative so it remains to show that:

(i) Composition of morphisms is well-defined: If $\sigma : (A, X) \to (B, Y)$ and $\tau : (B, Y) \to (C, Z)$ then $\sigma; \tau$ is $(X \Rightarrow Z)$-explicit.
(ii) Identities work as required: If $\sigma : (A, X) \to (B, Y)$ then $\sigma; \text{id}_{(B,Y)} = \sigma = \text{id}_{(A,X)}; \sigma$.

(i) Take any P-view $\vec{v}$ on which the composite strategy $\sigma; \tau$ is defined. Suppose that the last move of this P-view is $\underline{m}$ and the resulting P-move is m.

Firstly suppose that $\underline{m} \in X \Rightarrow Z$. We will have to take the special case when $\underline{m}$ is a root of C separately, so first assume that this is not the case. Then we know that either $\underline{m} \in X \Rightarrow Y$ or $\underline{m} \in Y \Rightarrow Z$, depending on whether σ or τ makes the next (possibly hidden) move after $\underline{m}$. In either case, in the notation of the Composition Algorithm, we have that $t_1 - o_1 \geqslant \text{br}(X \Rightarrow Z@\underline{m})$ by hypothesis. But then if we examine t and o, the threshold and offset of the composition at this P-view, we see that:

$$t - o = \hat{T}_{p+1} - \hat{O}_{p+1} \geqslant (T_1 + \sum_{i=2}^{p+1} O_i) - (\sum_{i=1}^{p+1} O_i)$$
$$= T_1 - O_1 = t_1 - o_1 \geqslant \text{br}(X \Rightarrow Z@\underline{m})$$

The first inequality holds by Lemma 41. In the special case when $\underline{m}$ is the root of C, we have that $o = \hat{O}_{p+1} - |A| + |B|$, but also we know that the strategy making the first move after $\underline{m}$ is τ, and that $\text{br}(X \Rightarrow Z@\underline{m}) = \text{br}(Y \Rightarrow Z@\underline{m}) + |A| - |B|$, so the above reasoning is still sound.

Secondly, suppose that $m \in X \Rightarrow Z$. There are no special cases; we always have that $T_{p+1} = t_{p+1} \geqslant \text{br}(X \Rightarrow Z@m)$ by hypothesis (this is because m cannot be a root of B or C). But $t = \hat{T}_{p+1} \geqslant T_{p+1}$ so that $t \geqslant \text{br}(X \Rightarrow Z@m)$.

Hence $\sigma;\tau$ is $X \Rightarrow Z$-explicit.

(ii) It is simple to verify that the identity on (A, X) has the following economical form. Suppose that $A = \langle A_1, \ldots, A_n \rangle$, so that $\text{id}_{(A,X)} = \langle \text{id}_1, \ldots, \text{id}_n \rangle$, say. Then:

$$\begin{aligned} \text{id}_i : \varepsilon &\mapsto (i, 0, \text{br}(X_i@\varepsilon), -n) \\ t &\mapsto (t+n, 1, \text{br}(X_i@t), 0) \\ \vec{s}\cdot t &\mapsto (t, 1, \text{br}(X_i@(\vec{s}\cdot t)), 0) \text{ for all nonempty sequences } \vec{s} \end{aligned}$$

Here we have extended the definition of br slightly, to have that $\text{br}(A@m) = 0$ when m is not in A.

That is, the copycat threshold of $\text{id}_{(A,X)}$ at a P-view ending in the O-move coded by $\underline{m}$ in either of the two copies of the arena A (which copy it will be depends on the parity of the length of $\underline{m}$) is the number of children of $\underline{m}$ in X, or zero if $\underline{m}$ is not in X.

Now we know that for an EAC strategy σ on $A \Rightarrow B$, $\text{id}_A;\sigma = \sigma = \sigma;\text{id}_B$, where id_A and id_B are the EAC identity strategies. It remains to show that this still holds when one also includes copycat thresholds and offsets. We will only consider $\text{id}_A;\sigma$, as the other proof is very similar and uses no additional techniques.

Let us select some P-view $\vec{v}$ on which $\text{id}_A;\sigma$ is defined, and suppose that it ends in the O-move $\underline{m}$, with m the resulting P-move. There are four cases:

(i) Both $\underline{m}$ and m are in B.
(ii) $\underline{m}$ is in B, m is in some component (A, a).
(iii) m is in B, $\underline{m}$ is in some component (A, a).
(iv) $\underline{m}$ is in (A, a_1), m is in (A, a_2).

Let us write t_σ and o_σ for the copycat threshold and offset of σ at the same P-view $\vec{v}$. In each case we want to show that these match the copycat threshold and offset of the composition. In what follows a superscript $(EX1)$ or $(EX2)$ on top of an equality sign refers to the fact that the equality is justified by σ being $X \Rightarrow Y$-explicit, at the P-view $\vec{v}$.

(i) Is completely trivial.

(ii) Examining the way the EAC identity strategies work (simply copycat) we see that the uncovering must be of the form

$$\begin{array}{llll} & \bullet & \star & \circ \\ \cdots & & & \\ & \underline{m} & m_1 & m \end{array}$$

where m is the same move in A as m_1, since it was arrived at by copycat. σ made the move m_1 in response to the P-view $\vec{v}$.

There are three cases. Let us first assume that $\underline{m}$ is not a root of B and m (equivalently m_1) not a root of A. Then, in the notation of the Composition Algorithm, we have

$$\begin{aligned} T_1 &= t_\sigma, & O_1 &= o_\sigma, \\ T_2 &= \mathrm{br}(X@m_1), & O_2 &= 0, \\ \hat{T}_1 &= t_\sigma, & \hat{O}_1 &= o_\sigma, \\ \hat{T}_2 &= \max(t_\sigma, \mathrm{br}(X@m_1)) \overset{(EX2)}{=} t_\sigma, & \hat{O}_2 &= o_\sigma. \end{aligned}$$

Hence $t = t_\sigma$ and $o = o_\sigma$ as required.

In the case when $\underline{m}$ is not a root of B but m is a root of A, we have similarly:

$$\begin{aligned} T_1 &= t_\sigma + |A|, & O_1 &= o_\sigma + |A|, \\ T_2 &= \mathrm{br}(X@m_1), & O_2 &= -|A|, \\ \hat{T}_1 &= t_\sigma + |A|, & \hat{O}_1 &= o_\sigma + |A|, \\ \hat{T}_2 &= \max(t_\sigma, \mathrm{br}(X@m_1)) \overset{(EX2)}{=} t_\sigma, & \hat{O}_2 &= o_\sigma. \end{aligned}$$

In the case when $\underline{m}$ is a root of B, we must have that m is a root of A, and the calculation above applies.

(iii) Very similar to (ii), with no special cases.

(iv) As before, we see that the uncovering must be of the form

$$\begin{array}{lllll} & \bullet & \star & \star & \circ \\ \cdots & & & & \\ & \underline{m} & m_1 & m_2 & m \end{array}$$

where $\underline{m}$ is the same move in A as m_1, and m_2 is the same move in A as m. The move m_2 is made by σ in response to the P-view $\vec{v}$.

There are two cases this time. First suppose that m_2 is not a root of A. Then

$$\begin{aligned} T_1 &= \mathrm{br}(X@\underline{m}) = \mathrm{br}(X@m_1), & O_1 &= 0, \\ T_2 &= t_\sigma, & O_2 &= o_\sigma, \end{aligned}$$

$$T_3 = \mathrm{br}(X@m_2), \qquad O_3 = 0,$$
$$\hat{T}_1 = \mathrm{br}(X@m_1), \qquad \hat{O}_1 = 0,$$
$$\hat{T}_2 = \max(\mathrm{br}(X@m_1) + o_\sigma, t_\sigma) \stackrel{(EX1)}{=} t_\sigma, \qquad \hat{O}_2 = o_\sigma,$$
$$\hat{T}_3 = \max(t_\sigma, \mathrm{br}(X@m_2)) \stackrel{(EX2)}{=} t_\sigma, \qquad \hat{O}_3 = o_\sigma,$$

If m_2 is a root of A then

$$T_1 = \mathrm{br}(X@\underline{m}) = \mathrm{br}(X@m_1), \qquad O_1 = 0,$$
$$T_2 = t_\sigma + |A|, \qquad O_2 = o_\sigma + |A|,$$
$$T_3 = \mathrm{br}(X@m_2), \qquad O_3 = -|A|,$$
$$\hat{T}_1 = \mathrm{br}(X@m_1), \qquad \hat{O}_1 = 0,$$
$$\hat{T}_2 = \max(\mathrm{br}(X@m_1) + o_\sigma + |A|, t_\sigma + |A|) \stackrel{(EX1)}{=} t_\sigma + |A|, \; \hat{O}_2 = o_\sigma + |A|,$$
$$\hat{T}_3 = \max(t_\sigma, \mathrm{br}(X@m_2)) \stackrel{(EX2)}{=} t_\sigma, \qquad \hat{O}_3 = o_\sigma.$$

So either way the result holds. □

Theorem 26. *The following constructions make* $\mathbb{XA}$ *into a CCC*:

The terminal object is (E,E).

The product $(A,X) \times (B,Y)$ *is* $(A \times B, X \times Y)$. *The projection strategy* $\pi^{(A,X)\times(B,Y)}_{(A,X)}$ *is given by the EXAC strategy* $\langle \pi^{A\times B}_A, t\rangle$, *where* $\pi^{A\times B}_A$ *is the EAC projection strategy and* t *is the threshold function specifying the least value at each P-view to make this EXAC strategy* $((X \times Y) \Rightarrow X)$*-explicit. Similarly for the other projection.*

The exponential object $(A,X) \Rightarrow (B,Y)$ is $(A \Rightarrow B, X \Rightarrow Y)$. *The evaluation map* $\mathrm{eval}_{(B,Y),(C,Z)}$ *is the same EXAC strategy as* $\mathrm{id}_{(B,Y)\Rightarrow(C,Z)}$.

The proof follows a similar case analysis as in the proof of Theorem 25 (see [9]).

4. A Universal Model of Böhm Tree Equality

Now that we have described an ambient Cartesian Closed Category based on EXAC strategies, we show how this does give rise to a λ-algebra $\mathscr{D}_{\mathrm{XA}}$ with precisely the equational theory we intended. Many properties of the model $\mathscr{D}_{\mathrm{EAC}}$ also hold for $\mathscr{D}_{\mathrm{XA}}$, but $\mathscr{D}_{\mathrm{XA}}$ is not a λ-model, i.e. it is not weakly extensional (Lemma 39).

4.1. The Model $\mathbb{D}_{\mathrm{XA}}$

Recall the single-tree arenas U and M defined in Section 2. The former is "maximal", in that it is (countably) infinitely branching and infinitely deep. The latter is "minimal", consisting of a single node.

Let us write U_0 for the object (U, M) of $\mathbb{XA}$, and U_1 for the object $U_0 \Rightarrow U_0$. The reader may wish to verify that, for example, the concrete representation of U_1 is given by $(U, \langle \{\varepsilon, \langle 1 \rangle\} \rangle)$.

Recall also the EXAC strategy η_1 from Section 3.1. We repeat the definition for convenience: the EAC strategy part is the same as the strategy id_U (since $U = U \Rightarrow U$ this does define an EAC strategy over U). The copycat threshold is zero at every P-view except the initial P-view, when it is 1. Thus the economical form is given by:

$$\varepsilon \mapsto (1, 0, 1, -1)$$

$$\langle 1 \rangle \mapsto (2, 1, 0, 0)$$

It is routine to check that η_1 specifies two morphisms of $\mathbb{XA}$, $f : U_0 \to U_1$ and $g : U_1 \to U_0$. (In fact it is the case that these morphisms could equally be specified as the EXAC strategies which have EAC strategy part id_U, and the least copycat thresholds to make them explicit in the necessary subarenas to be morphisms of that type. The definition of morphisms by EAC strategy and "least threshold function" to make them explicit in the appropriate subarenas seems to be a recurring theme.)

We can now show that f and g form a retraction from U_1 into U_0: we know that EAC strategy part of f, g, id_{U_0} and id_{U_1} are all the same as id_U, so that same holds for $f; g$ and $g; f$. Thus it remains only to check threshold functions. Simple applications of the Composition Algorithm show that the thresholds of both $f; g$ and $g; f$ are 1 at the minimal P-view, and zero elsewhere. The same holds for id_{U_1}, whereas id_{U_0} has threshold which is zero at every P-view. Hence, as EXAC strategies and thus as morphisms in $\mathbb{XA}$

$$g; f = \mathrm{id}_{U_1}, f; g \neq \mathrm{id}_{U_0}.$$

This allows us to identify a λ-algebra $\mathscr{M}(\mathbb{XA}, U_0, f, g)$, which we shall write as $\mathscr{D}_{\mathrm{XA}}$. To distinguish the denotation of a term as an EAC strategy in $\mathscr{D}_{\mathrm{EAC}}$ (which is the same as the denotation in $\mathscr{D}$ and $\mathscr{D}_{\mathrm{REC}}$), from the denotation as an EXAC strategy in this model we write it $[\![s]\!]^{\mathrm{XA}}$.

Remark 27. Following Scott, we regard λ-algebras as reflexive objects in CCCs (precisely, any reflexive object in an arbitrary CCC defines a λ-algebra, and every λ-algebra can be obtained in this way), and we think it important to identify the ambient CCC when defining a λ-algebra. It is in principle possible to construct the λ-algebra first, and then obtain an ambient CCC by the so-called *Karoubi envelope* construction (see e.g. [4,13]). Our preference is to give priority to the ambient CCC, and to identify it directly, as it seems to us the logical thing to do from a conceptual point of view. Incidentally we do not believe that it would be any simpler technically to construct the λ-algebra from scratch.

We will want to lift some results about EAC strategies and the model $\mathscr{D}_{\mathrm{EAC}}$ to EXAC strategies and the model $\mathscr{D}_{\mathrm{XA}}$. We make that connection precise with the following.

Theorem 28. *Let $E : \mathbb{XA} \to \mathbb{A}_{\mathrm{EAC}}$ be given on objects by $E((A,X)) = A$ and on morphisms by $E(\langle \sigma, t_\sigma \rangle) = \sigma$. Then*
(i) *E is a full, strict (i.e. on-the-nose) cartesian closed, functor, and*
(ii) *E preserves the reflexive object and the retraction morphisms.*
Hence for any term s and valuation $\rho, [\![s]\!]_\rho = E([\![s]\!]_\rho^{\mathrm{XA}})$.

Proof. (i) To show that E is full suppose that $\sigma : A \to B$ is an arrow of $\mathbb{A}_{\mathrm{EAC}}$. (Suppose that neither A nor B are equal to the empty arena E, these are special cases dealt with below). Then σ is an EAC strategy on $A \Rightarrow B$ and hence has some recursive valid threshold function t_σ, so $\langle \sigma, t_\sigma \rangle$ is an EXAC strategy over $A \Rightarrow B$. It is easy to show that this EXAC strategy is $(M \Rightarrow M)$-explicit hence it is a morphism $(A,M) \to (B,M)$ of $\mathbb{XA}$. (The special cases are as follows: if A is the empty arena E then the EXAC strategy above will vacuously be $(E \Rightarrow M)$-explicit, symmetrically for $B = E$, and if $A = B = E$ then $\sigma = \bot$, which is vacuously $(E \Rightarrow E)$-explicit.)

The other properties have a straightforward verification (for example the projections in $\mathbb{XA}$ are defined to be precisely the projections in $\mathbb{A}_{\mathrm{EAC}}$, along with the least copycat threshold to make them explicit in the appropriate subarena).

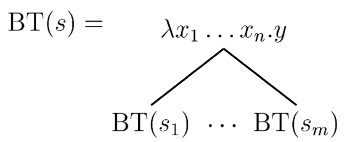
For (ii), clearly $E(U_0) = U$, $E(f) = Fun$ and $E(g) = Graph$, and we know that the λ-algebra $\mathscr{M}(\mathbb{C}, R, Fun, Graph)$ (where R is a reflexive object via *Fun* and *Graph* in the CCC $\mathbb{C}$) is completely determined by cartesian closed structure of $\mathbb{C}$, along with R, *Fun* and *Graph*. □

Corollary 29. *$\mathscr{D}_{\mathrm{XA}}$ is sensible. $[\![s]\!]^{\mathrm{XA}} = \bot$ (the everywhere undefined EXAC strategy) if and only if s is unsolvable.*

4.2. Böhm Trees in Variable-Free Form and Exact Correspondence

The Exact Correspondence Result proved in [12] showed that the denotation of a term, in the models $\mathscr{D}, \mathscr{D}_{\mathrm{REC}}$ and $\mathscr{D}_{\mathrm{EAC}}$, has a very close connexion with the term's Nakajima tree. Here we will show that the same connexion exists between the denotation of a term in $\mathscr{D}_{\mathrm{XA}}$ and the Böhm tree of the term.

Recall that the *Böhm tree* of term s, written $\mathrm{BT}(s)$ is given (informally) by the following: if s is unsolvable then $\mathrm{BT}(s) = \bot$. If s has head normal form $\lambda x_1 \ldots x_n . y s_1 \ldots s_m$ then



Thus each node of the Böhm tree has a finite number of abstractions, a head variable, and a finite number of children. We consider Böhm tree modulo α-conversion, so that the following information is sufficient to describe a Böhm tree of a closed term: for each node we have numbers specifying the number of abstractions and children, and information to describe where the head variable was abstracted in the tree in the same way as we did for Nakajima trees.

We encode this information into a *variable-free form* in the following manner:

Definition 30. For a $(\mathbb{N} \times \mathbb{N}_0 \times \mathbb{N}_0 \times \mathbb{Z})$-labelled tree p the tree p^* is the same tree labelled identically, except that nodes at depth d labelled $(i, d+1, t, o)$ are relabelled $(i, d+2, t, o)$.

Similarly the tree $\{p\}^n$, for $n \in \mathbb{N}_0$, is labelled identically except that firstly the node at the root (i, r, t, o) is first relabelled to $(i, r, t, o-n)$, and then nodes of depth d are relabelled as follows:
(1) those labelled (i, d, t, o) are relabelled $(i+n, d, t, o)$;
(2) those labelled $(i, d+1, t, o)$ for $i \leqslant n$ are relabelled $(n-i+1, d, t, o)$;
(3) those labelled $(i, d+1, t, o)$ for $i > n$ are relabelled $(i-n, d+1, t, o)$.

For a term s with free variables within Δ the *variable-free form* of the Böhm tree of s, $\mathrm{VFBT}_\Delta(s)$, is the following $(\mathbb{N} \times \mathbb{N}_0 \times \mathbb{N}_0 \times \mathbb{Z})$-labelled tree:

$$\mathrm{VFBT}_\Delta(s) = \bot, \qquad \text{the empty tree, for unsolvable } s.$$

$$\mathrm{VFBT}_\Delta(\lambda x_1 \dots x_n.s) = \{\mathrm{VFBT}_{\Delta\cdot\langle x_1,\dots,x_n\rangle}(s)\}^n, \quad \text{if } s \text{ is of the form } v_j s_1 \dots s_m.$$

$$\mathrm{VFBT}_\Delta(v_j s_1 \dots s_m) = \begin{array}{c} (j, 1, m, m) \\ \diagup \qquad \diagdown \\ \mathrm{VFBT}_\Delta(s_1)^* \qquad \mathrm{VFBT}_\Delta(s_m)^* \end{array}$$

where $\Delta = \langle v_k, \dots, v_1\rangle$ (note the reverse order).

Lemma 31. *The encoding of head variables in VFBT matches that in VFF. Precisely, for any term s with free variables within* Δ *and any sequence* $\vec{v} \in \mathbb{N}^*$ *at which (the labelling function of)* $VFBT_\Delta(s)$ *is defined we have* $VFBT_\Delta(s)(\vec{v}) = (i, r, t, o) \Rightarrow VFF_\Delta(s)(\vec{v}) = (i, r)$.

Furthermore, for any k, $\{VFBT_\Delta(s)(\vec{v})\}^k = (i, r, t, o) \Rightarrow \{VFF_\Delta(s)(\vec{v})\}^k = (i, r)$.

This statement is obvious when one compares the definitions of VFBT and VFF.

This definition is at first sight rather opaque, and indeed it could have been stated in a clearer fashion but that would have complicated Theorem 35 below. The effect of the definition is illustrated by the following lemma, which is analogous to Lemma 4.2.1 of [12].

Lemma 32. *Let s be a term with all free variables occurring within* $\Delta = \langle v_k, \dots, v_1\rangle$. *Construct the Böhm tree of s, and rename all the bound variables so that if* $\vec{a} \in \mathbb{N}^*$ *codes a node of BT(s) then the ith abstracted variable at this node is* $x_i^{\vec{a}}$. *Let this renamed Böhm tree have labelling function A, and consider* $VFBT_\Delta(s)$ *also as a labelling function.*

Then for any sequence $\vec{a} = \langle a_1, \dots, a_p\rangle$ *there are three possibilities for* $A(\vec{a})$:
(1) *If* $\vec{a} \notin dom(A)$ *then* $VFBT_\Delta(s)$ *is unlabelled or undefined at* $\vec{a}$,
(2) *If* $A(\vec{a}) = \lambda x_1^{\vec{a}} \dots x_n^{\vec{a}}.v_j$, *and the node coded by* $\vec{a}$ *has m children, then* $VFBT_\Delta(s)(\vec{a}) = (j, p+1, m, m-n)$.
(3) *If* $A(\vec{a}) = \lambda x_1^{\vec{a}} \dots x_n^{\vec{a}}.x_j^{\langle a_1,\dots,a_{p-r}\rangle}$, *and the node coded by* $\vec{a}$ *has m children, then* $VFBT_\Delta(s)(\vec{a}) = (j, r, m, m-n)$.

The proof for this lemma is straightforward, and entirely similar to the proof of Lemma 4.2.1 in [12].

Example 33. One may check that VFBT(I) and VFBT(1) are:

$$(1,0,0,-1) \text{ and } \begin{array}{c}(1,0,1,-1)\\ |\\ (2,1,0,0)\end{array}$$

The node $\lambda xy.x$, in the Böhm tree of 1, corresponds to the node of VFBT(1) labelled $(1,0,1,-1)$, which is so labelled because the head variable is the first abstracted variable zero levels up the tree (namely x), the node has one child, and the number of abstractions at this level is $1-(-1)=2$.

The following result will be useful in what follows.

Lemma 34. *If $\sigma : (A,X) \to U_1$ is a morphism in $\mathbb{XA}$, for any object (A,X), then $\sigma; g : (A,X) \to U_0$ is the same EXAC strategy as σ.*

If $\tau : (A,X) \to U_0$ is a morphism in $\mathbb{XA}$, for any object (A,X), then $\tau; f : (A,X) \to U_1$ is the same EXAC strategy as τ, unless the threshold and offset of τ at the minimal P-view, t and o, satisfy $t-o=|A|$. In this case $\tau; f$ is the same EXAC strategy as τ except that the threshold at the minimal P-view is $t+1$.

Proof. Recall that both f and g are the EXAC strategy η_1, which is the same as the EXAC strategy id_{U_0}, except that the threshold at the minimal P-view is 1 rather than 0. Hence we can use the proof that identities work as required in Theorem 25 to show that $\sigma; f$ and $\tau; g$ are the same EXAC strategies as σ and τ respectively, except at the minimal P-view.

It remains to examine the minimal P-view. Suppose that the first move of σ is in the arena A rather than U, with copycat threshold and offset t_σ and o_σ. Then, in the notation of the Composition Algorithm,

$$\begin{aligned} T_1 &= 1+|A|, & O_1 &= -1+|A|,\\ T_2 &= t_\sigma, & O_2 &= o_\sigma,\\ \hat{T}_1 &= 1+|A|, & \hat{O}_1 &= -1+|A|,\\ \hat{T}_2 &= \max(1+|A|+o_\sigma, t_\sigma) \overset{(*)}{=} t_\sigma, & \hat{O}_2 &= o_\sigma - 1 + |A|, \end{aligned}$$

Recall that $U_1 = (U, M \Rightarrow M)$. Then $(*)$ holds because σ must be $(X \Rightarrow (M \Rightarrow M))$-explicit, hence $t_\sigma - o_\sigma \geqslant |A|+1$.

Thus $t = t_\sigma$ and $o = o_\sigma$, i.e. the threshold and offset of $\sigma; f$ at the minimal P-view match that of σ. A similar calculation applies when the first move of σ is in the arena U.

The same figures occur in the calculation of $\tau; g$ at the minimal P-view, except that in this case we only know that τ, as a morphism $(A,X) \to U_0$, must be $(X \Rightarrow M)$-explicit, so that $t_\tau - o_\tau \geqslant |A|$. When equality holds, $t = t_\tau + 1$; when it is a strict inequality, $t = t_\tau$ as in the first part of the proof for σ. □

We use this to show a powerful connexion between the denotations of terms in $\mathscr{D}_{\mathrm{XA}}$ and Böhm trees.

Theorem 35 (Exact correspondence for $\mathscr{D}_{\mathrm{XA}}$). *If $s \in \Lambda$ with free variables in $\Delta = \langle v_k, \ldots, v_1 \rangle$ then $[\![s]\!]^{\mathrm{XA}}_{\Delta} = \{VFBT_{\Delta}(s)\}^k$ when the former is considered as an EXAC strategy in economical form and the latter as a labelling function.*

In particular for closed terms s, $[\![s]\!]^{\mathrm{XA}}_{\varepsilon} = VFBT_{\varepsilon}(s)$.

Proof. We show by induction on the length of $\vec{\alpha}$, for all terms s and contexts Δ simultaneously, that

(i) $\vec{\alpha} \cdot i \in \mathrm{dom}([\![s]\!]^{\mathrm{XA}}_{\Delta})$ if and only if $\vec{\alpha} \cdot i \in \mathrm{dom}(\mathrm{VFBT}_{\Delta}(s))$. The latter is trivially equal to $\mathrm{dom}(\{\mathrm{VFBT}_{\Delta}(s)\}^k)$.

(ii) If $[\![s]\!]^{\mathrm{XA}}_{\Delta}(\vec{\alpha}) = (i, r, t, o)$ and $\{\mathrm{VFBT}_{\Delta}(s)\}^k(\vec{\alpha}) = (i', r', t', o')$ then $i' = i, r' = r$ and $t' = t$.

We will be able to use the Exact Correspondence Theorem for $\mathscr{D}$, together with Lemma 31 and Theorem 28 to prove that $i' = i$ and $r' = r$, and then show $t' = t$ by considering the composition algorithm. In view of Remark 14 this will ensure that $o' = o$ too (in the base case $o' = o$ comes for free, but the proof of the inductive step is easier without having to consider offsets).

Base case: If s is unsolvable then both $[\![s]\!]^{\mathrm{XA}}_{\Delta}$ and $\mathrm{VFBT}_{\Delta}(s)$ are everywhere undefined.

Otherwise s has a head normal form $\lambda x_1 \ldots x_n . y s_1 \ldots s_m$. Then in the notation above $t' = m$ and $o' = m - n - k$.

We must return to the definition of $[\![-]\!]^{\mathrm{XA}}$ to discover the copycat thresholds and offsets.

$$[\![s]\!]^{\mathrm{XA}}_{\Delta} = \underbrace{\Lambda(\cdots \Lambda(\Lambda}_{n\ \Lambda'\mathrm{s}}([\![y s_1 \ldots s_m]\!]_{\Gamma}); g); g \cdots); g,$$

where $\Gamma = \Delta \cdot \langle x_1, \ldots, x_n \rangle$. Since $\Lambda(\sigma)$ is the same EXAC strategy as σ, and by Lemma 34 so is $\sigma; g$, this has the same thresholds and offsets as $[\![y s_1 \ldots s_m]\!]^{\mathrm{XA}}_{\Gamma}$. Let

$$V = \underbrace{U_0 \times \cdots \times U_0}_{k+n\ \text{times}}.$$

$$[\![y s_1 \ldots s_m]\!]^{\mathrm{XA}}_{\Gamma} = (\Pi^{\Gamma}_{y} \cdot [\![s_1]\!]^{\mathrm{XA}}_{\Gamma}) \cdots \cdot [\![s_m]\!]^{\mathrm{XA}}_{\Gamma}$$

We will show, by induction on m, that the threshold of this strategy at the minimal P-view is m and the offset is $m - n - k$. For convenience we also include the fact that the first P-move of the composite strategy is played in the arena V in the induction hypothesis.

Case $m = 0$: The strategy is just Π^{Γ}_{y}. Depending on y, this is just one of the projections $V \to U_0$, which we know has threshold 0 and offset $-n - k$ at the minimal P-view. We also know that the first P-move is played in the arena V.

Inductive case: Suppose that we have a strategy $\sigma : V \to U_0$ with threshold m and offset $m - n - k$ at the minimal P-view, and which makes the first P-move in the arena V. Then for any strategy $\tau : V \to U_0$,

$$\sigma \cdot \tau = \langle \sigma; f, \tau \rangle; \mathrm{eval}_{U_0, U_0}.$$

First examine $\sigma; f$ — we are in the special case of Lemma 34, so that the threshold of $\sigma; f$ is $m+1$ and the offset $m-n-k$.

It is now simple to use the Composition Algorithm to examine the threshold and offset of $\langle \sigma; f, \tau \rangle; \mathrm{eval}_{U_0,U_0}$ at the minimal P-view. Since we know that the first P-move of σ is in the arena V, there is one intermediate move, which is the root of $V \Rightarrow U_1$ at which σ is to play. Thus the strategy τ is irrelevant for this calculation and in the notation of the Composition Algorithm the resulting threshold and offset is calculated as follows:

$$\begin{aligned} T_1 &= 1+k+n, & O_1 &= -1+k+n,\\ T_2 &= m+1, & O_2 &= m-n-k,\\ \hat{T}_1 &= 1+k+n, & \hat{O}_1 &= -1+k+n,\\ \hat{T}_2 &= \max(m+1, 1+k+n+m-n-k), & \hat{O}_2 &= -1+k+n+m-n-k,\\ &= m+1, & &= m-1. \end{aligned}$$

Hence $t = m+1$ and $o = m-1-(n+k)+2 = m+1-n-k$. One can also see that the first move of the composition is in the arena V, which completes the inductive step of this claim.

This completes the proof that the threshold and offset of $[\![s]\!]^{\mathrm{XA}}_{\Delta}$ at the minimal P-view are m and $m-n-k$ respectively.

This shows that $\langle i \rangle \in \mathrm{dom}([\![s]\!]^{\mathrm{XA}}_{\Delta})$ if and only if $1 \leqslant i \leqslant m$ and s_i is solvable. On the other hand, $\langle i \rangle \in \mathrm{dom}(\mathrm{VFBT}_{\Delta}(s))$ if and only if $1 \leqslant i \leqslant m$ and s_i is solvable. This completes the proof of (i).

Suppose that $[\![s]\!]^{\mathrm{XA}}_{\Delta}(\varepsilon) = (i,r,t,o)$ and $\{\mathrm{VFBT}_{\Delta}(s)\}^k(\varepsilon) = (i',r',t',o')$. Now Lemma 31 means that $\{\mathrm{VFF}_{\Delta}(\varepsilon)\}^k(\varepsilon) = (i',r')$. On the other hand, Theorem 28 means that $[\![s]\!]^{\mathrm{XA}}_{\Delta}(\varepsilon) = (i,r)$, and the Exact Correspondence Theorem for $\mathscr{D}_{\mathrm{EAC}}$ gives that $i' = i$ and $r' = r$.

Finally, by the definition of VFBT, $t' = m = t$ and $o' = m-n-k = o$, completing the base case of the outer induction.

Inductive step: Again if s is unsolvable then both functions are everywhere undefined, so assume that s has head normal form $\lambda x_1 \ldots x_n.ys_1 \ldots s_m$ and again write $\Gamma = \Delta \cdot \langle x_1,\ldots,x_n \rangle$. We assume the results (i) and (ii) for each of the terms $s_1,\ldots,s_m$, each with the context Γ, for all sequences $\vec{\alpha}$ of length up to l.

Let $\vec{\alpha}$ be any sequence of length l. Take $1 \leqslant j \leqslant m$. We show that (i) and (ii) hold for the term s and context Δ, for the sequence $j \cdot \vec{\alpha}$. We already know it holds for the sequence ε, and (also by the base case) that the domain of both functions is contained in the set $\{j \cdot \vec{\alpha} | 1 \leqslant j \leqslant m, \vec{\alpha} \in \mathbb{N}^*\}$. This will therefore establish the inductive step.

Suppose that $[\![s]\!]^{\mathrm{XA}}_{\Delta}(j \cdot \vec{\alpha}) = (i,r,t,o)$ and $\{\mathrm{VFBT}_{\Delta}(s)\}^k(j \cdot \vec{\alpha}) = (i',r',t',o')$. We know by result (i) of the inductive hypothesis that one is defined if and only if the other is. Lemma 31 means that $\{\mathrm{VFF}_{\Delta}(s)\}^k(j \cdot \vec{\alpha}) = (i',r')$. On the other hand, Theorem 28 means that $[\![s]\!]^{\mathrm{XA}}_{\Delta}(j \cdot \vec{\alpha}) = (i,r)$, and the Exact Correspondence Theorem for $\mathscr{D}_{\mathrm{EAC}}$ gives that $i' = i$ and $r' = r$. We next show that $t' = t$, completing the proof of (ii).

Now by the definition of VFBT,

$$\{\text{VFBT}_\Delta(s)\}^k(j\cdot\vec{\alpha}) = (i',r',t',o')$$
$$\text{if and only if } \{\text{VFBT}_\Delta(s)\}^{(k+n)}(s_j)(\vec{\alpha}) = (i'',r'',t',o'')$$

for some irrelevant numbers i'', r'', o'' (this is simple to verify). Then by the inductive hypothesis $[\![s_j]\!]^{\text{XA}}_\Gamma(\vec{\alpha}) = (i'',r'',t',o'')$

With this fact in hand we examine $[\![s]\!]^{\text{XA}}_\Delta$, aiming to calculate $[\![s]\!]^{\text{XA}}_\Delta(j\cdot\vec{\alpha})$.

As we found in the base case, $[\![s]\!]^{\text{XA}}_\Delta = [\![ys_1\dots s_m]\!]^{\text{XA}}_\Gamma = (\Pi^\Gamma_y\cdot[\![s_1]\!]^{\text{XA}}_\Gamma)\cdot\dots\cdot[\![s_m]\!]^{\text{XA}}_\Gamma$, which with the ·'s decoded is

$$\langle\cdots\langle\langle\Pi^\Gamma_y; f, [\![s_1]\!]^{\text{XA}}_\Gamma\rangle; \text{eval}; f, [\![s_2]\!]^{\text{XA}}_\Gamma\rangle; \text{eval}; f\cdots, [\![s_m]\!]^{\text{XA}}_\Gamma\rangle; \text{eval}$$

where eval is eval_{U_0,U_0}. What follows is only an outline analysis, as a completely formal proof would be extremely tedious.

We already know, from the proof of the Exact Correspondence Theorem for $\mathscr{D}_{\text{EAC}}$, that if the first O-move made against this strategy is j then the result of this multiple composition is to copy moves made by and against σ_j from the components where they are hidden into ones where they are not; this composite strategy makes moves which are (a small translation of) those of σ_j. What is important is that between visible moves, except between the initial move and the first P-move, all of the intermediate moves are *not* roots of the arenas they occur in. The reason this is important is because for Π^Γ_y and eval_{U_0,U_0} the copycat thresholds and offsets are always zero except at the initial P-view (this is very simple to check).

Thus when we work out the threshold and offset of $[\![s]\!]^{\text{XA}}_\Delta$ at the P-view coded by $j\cdot\vec{\alpha}$ using the composition algorithm and the fact that the threshold and offset of $[\![s_j]\!]^{\text{XA}}_\Gamma$ at the P-view coded by $\vec{\alpha}$ are t' and o'', the calculation will be either of the form $T_1 = T_2 = \dots T_p = 0$, $O_1 = O_2 = \dots O_p = 0$, $T_{p+1} = t'$ and $O_{p+1} = o''$ or $T_1 = t'$, $O_1 = o''$, $T_2 = T_3 = \dots T_{p+1} = 0$ and $O_2 = O_3 = \dots O_{p+1} = 0$, depending on which component the visible moves appear in. In the first case $\hat{T}_1 = \hat{T}_2 = \dots = \hat{T}_p = 0$ and $t = \hat{T}_{p+1} = \max(t', o'') = t'$. In the second case, $\hat{T}_1 = t'$ so $t' = \hat{T}_2 = \dots = \hat{T}_{p+1} = t$.

Thus in either case we have shown that the copycat threshold of $[\![s]\!]^{\text{XA}}_\Delta$ at the P-view coded by $j\cdot\vec{\alpha}$ is t', but by assumption it is also t. Hence $t = t'$.

Finally, we show (i) as follows: $j\cdot\vec{\alpha}\cdot i \in \text{dom}([\![s]\!]^{\text{XA}}_\Delta)$ if and only if $\vec{\alpha}\cdot i \in \text{dom}([\![s_j]\!]^{\text{XA}}_\Gamma)$, this is because of the way the multiple composition which defines $[\![s]\!]^{\text{XA}}_\Delta$ copies the moves of s_j after the first P-move j. But $\vec{\alpha}\cdot i \in \text{dom}([\![s_j]\!]^{\text{XA}}_\Gamma)$ if and only if $\vec{\alpha}\cdot i \in \text{dom}(\text{VFBT}_\Gamma(s_j))$ (by (i) of the inductive hypothesis) if and only if $j\cdot\vec{\alpha}\cdot i \in \text{dom}(\text{VFBT}_\Delta(s))$ by the definition of VFBT.

This completes the inductive step of the outermost induction. □

Corollary 36. *For closed terms s and t,*

$$[\![s]\!]^{\text{XA}} \subseteq [\![t]\!]^{\text{XA}} \Leftrightarrow BT(s) \subseteq BT(t).$$

The order on $\mathscr{D}_{\text{XA}}$ is inclusion of EXAC strategies, namely inclusion of both EAC strategy part and threshold function. The order on Böhm trees is inclusion of labelling *function, modulo renaming of bound variables, which amounts to inclusion of variable-free form. Thus the local structure of $\mathscr{D}_{\mathrm{XA}}$ is the λ-theory $\mathscr{B}$, which equates terms with the same Böhm tree.*

Example 37. Applying the Exact Correspondence Theorem to the variable-free forms of the Böhm trees of the terms I and 1, which we looked at earlier in the section, we can deduce that the economical forms of $[\![I]\!]^{\mathrm{XA}}$ and $[\![1]\!]^{\mathrm{XA}}$ are, as we hoped, the EXAC strategies η_0 and η_1 described in Section 3.1.

As with the model $\mathscr{D}_{\mathrm{EAC}}$, the Exact Correspondence Theorem allows us to prove the powerful result of universality holds for $\mathscr{D}_{\mathrm{XA}}$.

Theorem 38. *$\mathscr{D}_{\mathrm{XA}}$ is a universal λ-algebra.*

The proof for this theorem follows exactly the same reasoning as was presented in the corresponding proof for $\mathscr{D}_{\mathrm{EAC}}$ in [12].

4.3. Non-extensionality of $\mathscr{D}_{\mathrm{XA}}$

Recall that a λ-algebra $\mathscr{A}$ (more generally, any applicative structure) is called *extensional* if for all $s, t \in \mathscr{A}$,

$$(\forall a \in \mathscr{A}.s \cdot a = t \cdot a) \Rightarrow s = t.$$

In [12] we showed that $\mathscr{D}_{\mathrm{EAC}}$ was extensional (in fact it was order-extensional, a stronger property). However, since $[\![I]\!]^{\mathrm{XA}} \neq [\![1]\!]^{\mathrm{XA}}$, but for all terms s and t, $Ist = 1st$ in the $\lambda\beta$-theory, we can be sure that $\mathscr{D}_{\mathrm{XA}}$ is not extensional.

In fact we will show that $\mathscr{D}_{\mathrm{XA}}$ is not even *weakly extensional*. A λ-algebra $\mathscr{A}$ (more generally, any combinatory algebra) is weakly extensional if the first-order statement

$$\mathscr{A} \models (\forall x.\ s = t) \Rightarrow \lambda x.s = \lambda x.t$$

is true. A weakly extensional λ-algebra is called a *λ-model*.

Lemma 39. *$\mathscr{D}_{\mathrm{XA}}$ is not weakly extensional.*

Proof. Recall that $\mathscr{D}_{\mathrm{XA}}$ is the model $\mathscr{M}(\mathbb{XA}, U_0, f, g)$. Define $s = \mathrm{id}_{U_0}$ and $t = f; g$. Both are morphisms $U_0 \to U_0$ and $s \neq t$, as demonstrated in Section 4.1.

Now take any $r : 1 \to U_0$, then certainly $r; s = r$. Moreover, it is easy to see that we must have $r = \bot$ or $\mathbf{1}(r) \geqslant 1$. But then $r; t = r$, trivially in the first case and by Lemma 34 (twice) in the second.

So U_0 does not have enough points in $\mathbb{XA}$, hence $\mathscr{D}_{\mathrm{XA}}$ is not weakly extensional. $\square$

Conclusion and Further Work

We have presented a game model of the untyped λ-calculus, with equational theory equal to the Böhm tree λ-theory $\mathscr{B}$. A noteworthy feature of the construction is its *universality* i.e. every element of the model is definable by some term; in other words, there is "no junk" in the model. Indeed the correspondence between the model and the theory of Böhm trees is so exact (see Theorem 34) that the former may be regarded as a mathematical reformulation of the latter. In our view, our model does tell us something new and important, over and above the existing theory of Böhm trees. Böhm trees (as presented e.g. in [4]) are defined in terms of λ-calculus syntax, and the Böhm tree λ-model (see e.g. [4, 18.3]) is essentially a term model. That our game models are syntax-independent is perhaps best illustrated by reference to the composition algorithm, which can be seen as an analysis of how computation with Böhm trees works.

The work reported here builds on and extends our recent characterization of the maximal consistent sensible λ-theory $\mathscr{H}^*$ (see [12]) in terms of the *effectively almost-everywhere copycat* (EAC) strategies. The very concrete nature of the EAC definition has enabled us to prove the Universality and the Exact Correspondence results. However it would be good to find a more algebraic description of the underlying uniformity constraints captured by the EAC definition. In the simply-typed or cartesian-closed setting, families that are uniformly defined, in the sense that a strategy is obtainable from some representative member of the same family purely by *copycat expansion* (see [12,9]), are the dinatural ones. We seek a similarly abstract description of the effectively almost-everywhere copycat strategies, which may be thought of as the corresponding notion of uniformity in the reflexive or untyped case.

Recently Di Gianantonio [6] has sketched the construction of a fully abstract model for the pure Lazy Lambda Calculus, settling an open problem identified in [18,3]. The key innovation is the introduction of an ordering on moves of an AJM game [1]. The denotable elements of the model are the history-free strategies that respect the ordering. Di Gianantonio's monotonicity condition has the effect of constraining the strategies to be copycat in a similar [1] way to ours. We intend to situate the monotonicity property in the innocent setting by relating Di Gianantonio's ordering to the justification relation. This was the motivation behind the work in [17], which gives a universal model of the Lazy Lambda Calculus that seems simpler than the game models in this paper. We hope a similar construction will lead to a more abstract description of EAC strategies.

Initial study of the nature of EAC suggests that two orthogonal conditions are being imposed on the strategies – not only are they being constrained to act in a copycat fashion but they are also forced to be pairwise observationally inequivalent (with respect to a straightforward notion of observables [8]). The EAC condition will likely be better understood if these two constraints can be separated. A conjecture is that the observational quotient of the innocent game category (which is *not* constrained by EAC) leads to a universal model of an infinitary λ-calculus [10,5], in which the Nakajima trees are graphical representations of head normal forms with respect to a

[1] There is however an important difference: the corresponding η-law is conditional upon a notion of convergence: $\vdash s\downarrow \rightarrow \lambda x.sx = s$ provided x does not occur free in s.

(possibly transfinite) head reduction. While this may lead to interesting investigations of infinitary λ-calculus, it may also suggest a succinct syntax for the representation of (appropriate classes of) innocent strategies, which would be of interest to the game semantics community. We may try to find game models of other (perhaps nonsensible) λ-theories. If we hope for an Exact Correspondence result, we will need the λ-theory to have some sort of normal form for terms, and this may be a sticking point. It would appear that a new idea is needed.

Finally we would like to mention an intriguing problem raised by Barendregt: Is there a model (game or otherwise) whose theory is exactly $\lambda\beta$ (or $\lambda\beta\eta$)?

Appendix A. Proof of Theorem 20

The appendix is devoted to the proof of Theorem 20. Throughout we use the notation of the Composition Algorithm for composing EAC strategies σ on $A \Rightarrow B$ and τ on $B \Rightarrow C$, including references to $\underline{m}$, m, t_i, T_i, $\hat{T}_i$, $\vec{u}_i$, and so on.

We begin by restricting our attention to the case when C is a single-tree arena. For if not, say $C = \langle C_1, \dots, C_n \rangle$, then we know that τ is of the form $\langle \tau_1, \dots, \tau_n \rangle$, and $\sigma;\tau$ means $\langle \sigma;\tau_1, \dots, \sigma;\tau_n \rangle$. And since $\mathbf{1}(\tau) = \min_{i=1}^{n}\{\mathbf{1}(\tau_i)\}$ we can be sure that the condition $\mathbf{1}(\tau_i) \geqslant |B|$ holds for each i where τ_i has a first move. So it is valid to consider the τ_i individually instead.

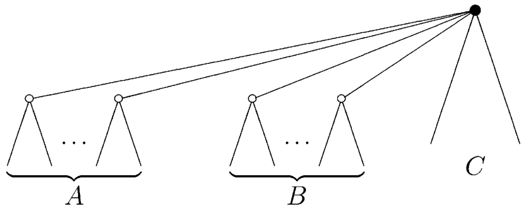
We also discard the possibility that either σ or τ are everywhere undefined; they are easy special cases.

Suppose, then, that the economical form of the composite strategy $\sigma;\tau$ is f, and choose a P-view $\vec{v}$ at which it is defined. We overload notation so that $\vec{v}$ refers both to the P-view and the sequence of natural numbers encoding it; likewise for moves (but see below). We must show that the arenas and f satisfy conditions (AC1)–(AC4). In a succession of lemmas below, we will prove:

(1) The arenas $((A \Rightarrow C)@\underline{m})^{>(t-o)}$ and $((A \Rightarrow C)@m)^{>t}$ are isomorphic;
(2) If $i > t$ then $f(\vec{v} \cdot i) = (i - o, 1)$ and further $f(\vec{v} \cdot i \cdot \vec{w} \cdot j) = (j, 1)$ for all sequences $\vec{w}$ and numbers $j \geqslant 1$;
(3) For all $\vec{w} \geqslant (\vec{v} \cdot k)$ with $k \leqslant t$, if $f(\vec{w}) = (i, |\vec{w}| - |\vec{v}|)$ then $i \leqslant t - o$;
(4) If $f(\vec{v}) = (i, 0)$ then $i \leqslant t - o$.

Note that the second half of property (2) is sufficient for the composition to be EC where necessary because condition (EC3) will hold vacuously.

In fact we will prove a slight modification of the above properties. There is a potential pitfall when dealing in moves coded by sequences of natural numbers, because "the *ith*" child of a move m may be ambiguous — in the special case where m is the root of C, the *ith* child of the move m in the arena $A \Rightarrow C$ is the $(i - |A|)$*th* child of the same move in the arena C. We avoid this complication by pretending that the interaction sequences in the composition of $\sigma;\tau$ take place in a special arena we call D, which looks like $A \Rightarrow (B \Rightarrow C)$ and is pictured below. The arena is special because in the composition the strategy σ is playing in A and B, so the normal switching condition looks like it is being violated. But really all we are doing is pretending for the purposes of coding that the moves appear as in D; the composition still happens in the normal way and the parts of the interaction in B are hidden in the result.



Observe that the conversion from t_i to T_i and o_i to O_i is simply that which corrects the thresholds and offsets of σ and τ to be right when counting children of moves in the arena D rather than $A \Rightarrow B$ or $B \Rightarrow C$. Then observe that the conversion from $\hat{T}_{p+1}$ to t and $\hat{O}_{p+1}$ to o corrects thresholds and offsets for the arena $A \Rightarrow C$ rather than D. So the composition algorithm is really taking place in D anyway, translating thresholds and offsets as required at the beginning and end.

So we will prove the analogues of properties (1)–(4) as if the moves took place in D, which will reference T_i, $\hat{T}_{p+1}$, and so on, rather than t_i, t, etc. But before that we need a couple of lemmas.

Lemma 40. *If A and B are arenas satisfying $A^{>m} \cong B^{>n}$ then for all $l \geqslant 0$ we have $A^{m+l} \cong B^{n+l}$.*

Proof. Obvious, since we are cutting down the two isomorphic arenas in the same way.

Lemma 41. *For $1 \leqslant i < j \leqslant p+1$,*

$$\hat{T}_j - \hat{O}_j \geqslant T_i - \sum_{k=1}^{i} O_k$$

Proof. Directly from the definition we have that $\hat{T}_j \geqslant \hat{T}_{j-1} + O_j$, so inductively $\hat{T}_j \geqslant \hat{T}_i + \sum_{k=i+1}^{j} O_k$. But also $\hat{O}_j = \sum_{k=1}^{j} O_k$, and $\hat{T}_i \geqslant T_i$. Hence the result.

Lemma 42. *For $1 \leqslant i \leqslant p+1$,*

$$(D@\underline{m})^{>(\hat{T}_i - \hat{O}_i)} \cong (D@m_i)^{>\hat{T}_i}.$$

Proof. By induction on i.

The base case $i = 1$ follows directly from property of (AC1) of σ or τ at $\vec{u_1}$.

For the inductive step, suppose the result holds for $\hat{T}_i$ and $\hat{O}_i$.

Then **either** $\hat{T}_{i+1} = \hat{T}_i + O_{i+1}$, so we must have $\hat{T}_i + O_{i+1} \geqslant T_{i+1}$, say $\hat{T}_i = T_{i+1} - O_{i+1} + l$. Then

$$(D@\underline{m})^{>(\hat{T}_{i+1} - \hat{O}_{i+1})} = (D@\underline{m})^{>(\hat{T}_i - \hat{O}_i)} \overset{(1)}{\cong} (D@m_i)^{>\hat{T}_i} = (D@m_i)^{>(T_{i+1} - O_{i+1} + l)}$$

$$\overset{(2)}{\cong} (D@m_{i+1})^{>T_{i+1}+l} = (D@m_{i+1})^{>\hat{T}_{i+1}}$$

(each equality follows from equality of the superscripted indices, and (1) is the inductive hypothesis; (2) comes from property (AC1) of σ or τ at $\vec{u}_{i+1}$, which gives $(D@m_i)^{>(T_{i+1}-O_{i+1})} \cong (D@m_{i+1})^{>T_{i+1}}$, and applying Lemma 40.)

Or $\hat{T}_{i+1} = T_{i+1}$, so we must have $\hat{T}_i + O_{i+1} \leqslant T_{i+1}$, say $\hat{T}_i = T_{i+1} - O_{i+1} - l$. Then

$$
\begin{aligned}
(D@\underline{m})^{>(\hat{T}_{i+1}-\hat{O}_{i+1})} = (D@\underline{m})^{>(\hat{T}_i-\hat{O}_i+l)} &\overset{(1)}{\cong} (D@m_i)^{>\hat{T}_i+l} = (D@m_i)^{>(T_{i+1}-O_{i+1})} \\
&\overset{(2)}{\cong} (D@m_{i+1})^{>T_{i+1}} = (D@m_{i+1})^{>\hat{T}_{i+1}}
\end{aligned}
$$

(again, each equality follows from equality of the superscripted indices, (1) is the inductive hypothesis combined with an application of Lemma 40, and (2) is property (AC1) of σ or τ at $\vec{u}_{i+1}$.)

That completes the inductive step.

Lemma 43. *The following holds for each* $1 \leqslant i \leqslant p+1$, *where all moves are coded as in the arena* D.

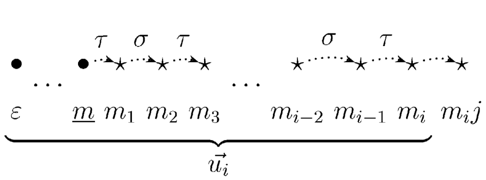
If $j > \hat{T}_i$ *and the jth child of the last move in the interaction* $\vec{u_i}$ *(i.e.* $m_i j$*) is played the resulting continuation is forced by the copycat properties of* σ *and* τ, *and has as the next move in A or C the P-move* $\underline{m}(j - \hat{O}_i)$, *justified by* $\underline{m}$;

Furthermore a subsequent move $\underline{m}(j - O_i)w_1$ *will result in an interaction ending in* $m_i j w_1$, *and in general for any odd-length sequence* $\vec{w}$ *the move* $\underline{m}(j - \hat{O}_i)\vec{w}$ *will result in an interaction ending in* $m_i j \vec{w}$ *and for even-length sequences* $\vec{w}$ *the move* $m_i j \vec{w}$ *will result in an interaction ending in* $\underline{m}(j - \hat{O}_i)\vec{w}$.

Proof. By induction on i. When $i \leqslant p$ all the moves m_i and their descendants will be in the B part of D i.e. invisible in the composition. For $i = p+1$ the moves $m_i = m$ and $m_i j$ and their descendants are in the A or C part i.e. visible.

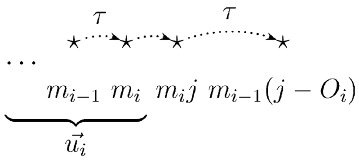
The base case is inherited from the (AC2) property of σ or τ at $\vec{u}_1$.

For the inductive step, suppose that the strategy dictating the first move of the interaction is τ, i.e. $\underline{m}$ is in the arena C (the other case is similar and will require some τ's to be replaced by σ's and one other difference which we note below). Suppose that the move $m_i j$ is made after the move m_i, with $j > \hat{T}_i$. Then the interaction sequence looks like:



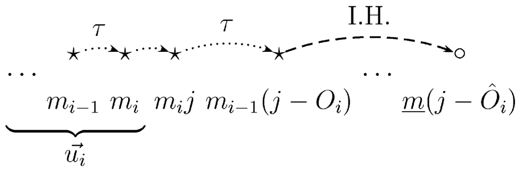
(Here and hereafter we are omitting the justification pointers, because where they should point to are determined by the properties of AC.) We can predict how this interaction would continue: since $j > \hat{T}_i \geqslant T_i$ we know by the EAC property of τ

that we must continue:



Remember that m_ij refers to the jth child of m_i relative to the arena D, and $m_{i-1}(j-O_i)$ is the $(j-O_i)th$ child relative to the arena D.

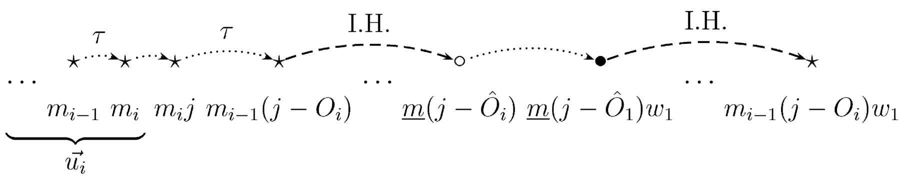
But $j-O_i > \hat{T}_i - O_i \geqslant \hat{T}_{i-1}$ so by the induction hypothesis we know that this interaction must continue until it reaches the visible P-move $\underline{m}(j-O_i-\hat{O}_{i-1}) = \underline{m}(j-\hat{O}_i)$:



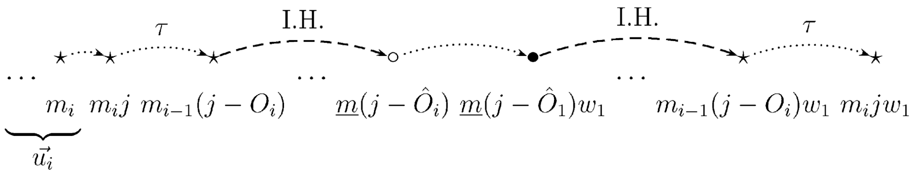
We have to be a little careful, because the move $\underline{m}(j-\hat{O}_i)$ has to be visible, i.e. it is in the arena C. The only problem could be if $\underline{m}$ is a root of C, in which case we need $j-\hat{O}_i > |B|$. Thankfully we have $j-\hat{O}_i > \hat{T}_i - \hat{O}_i \geqslant T_1 - O_1 = t_1 - o_1 \geqslant \mathbf{1}(\tau) \geqslant |B|$. The second inequality in this chain is an application of Lemma 41, the third is because $\underline{m}$ is a root of C and using the definition of l-number, and the final inequality is by hypothesis.

There could be no such problem in the case that σ was the strategy to dictate the move immediately after $\underline{m}$, because $\underline{m}$ would have to be a A-move. However, in this case we have to show that the move $m_{i-1}(j-O_i)$ is not visible, i.e. it is in the arena B. There is no problem unless m_{i-1} is a root of B. Then, similarly to above, we have $j-O_i > \hat{T}_i - O_i \geqslant T_i - O_i = t_i - o_i \geqslant \mathbf{1}(\sigma) \geqslant |A|$.

If there is a further P-move $\underline{m}(j-\hat{O}_i)w_1$, by the inductive hypothesis, the following must result:



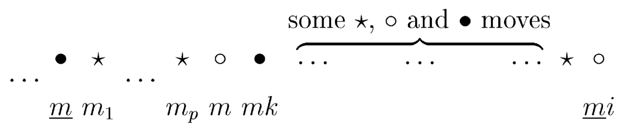
and because τ must be EC at this view, we have



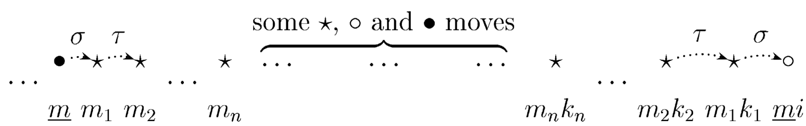
And it is clear how subsequent moves will be copied by τ, and the induction hypothesis will always apply, giving the desired result.

Lemma 44. *Suppose that $\underline{m}$ and m are as before, so $\sigma;\tau$ makes the move m in response to the P-view coded by $\vec{v}$, and $\underline{m}$ was the final move of that P-view. Suppose also that $\sigma;\tau$ makes the move mi (where the i^{th} child is counted in the arena D) in response to the P-view coded by $\vec{v} \cdot k \cdot \vec{w}$, where $k \leqslant \hat{T}_{p+1}$. We claim that $i \leqslant \hat{T}_{p+1} - \hat{O}_{p+1}$.*

Proof. All we know about the interaction which produced the move mi is that it must be of the form



Working backwards from the end of the interaction sequence, pick out the first ★ move which has not been forced by the AC properties of σ or τ. The interaction sequence must be of this form



where each pictured move $m_i k_i$ is justified by the corresponding m_i, the move $m_n k_n$ is *not* forced by the AC properties, but all the moves $m_i k_i$ for $i < n$ *are* forced by copycat.

We know that the interaction sequence must have been of that form because, reasoning backwards from the end, if $\underline{mi}$ was forced by copycat then the move before it must have been justified by the move after $\underline{m}$, and so on along the sequence until we do find a move not forced by copycat. Working backwards, the forced-by-copycat moves way must run out no further back than $m_p k_p$, because the hypothesis that $k \leqslant \hat{T}_{p+1}$ means that the move justified by m was not forced by copycat.

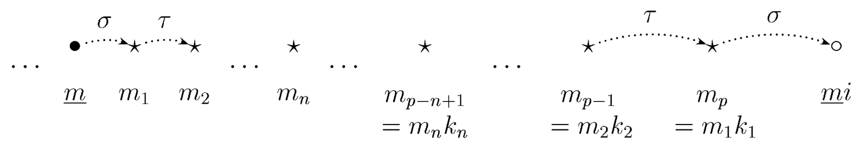
Because of the AC properties of the moves forced by copycat, we know that $i = k_1 - O_1$, $k_1 = k_2 - O_2$, $\dots$, $k_{n-1} = k_n - O_n$, and $k_n \leqslant T_n$. Thus

$$i = k_n - \sum_{j=1}^{n} O_j \leqslant T_n - \sum_{j=1}^{n} O_j \leqslant \hat{T}_{p+1} - \hat{O}_{p+1},$$

the final inequality by Lemma 41.

Lemma 45. *Suppose that the move $m = \underline{m}i$ is the reaction by $\sigma;\tau$ to a P-view $\vec{v}$ whose final move is $\underline{m}$ (where the i^{th} child is counted in the arena D). Then $i \leqslant \hat{T}_{p+1} - \hat{O}_{p+1}$*

Proof. The proof is very similar to that of Lemma 44. This time the interaction sequence must be of the form



There can only be up to $p/2$ moves before $\underline{m}i$ forced by copycat (note that p must be even because $\underline{m}$ and $\underline{m}i$ are in the same component); at some point one of the moves must be not forced by copycat, and exactly the same calculations as in Lemma 44 apply.

References

[1] S. Abramsky, R. Jagadeesan, P. Malacaria, Full abstraction for PCF, Information and Computation 163 (2000) 409–470.

[2] S. Abramsky, G.A. McCusker, Games and full abstraction for the lazy λ-calculus, In: Proceedings, Tenth Ann. IEEE Sym. on Logic in Computer Science, IEEE Computer Society Press, 1995, 234–243.

[3] S. Abramsky, C.-H.L. Ong, Full abstraction in the lazy lambda calculus, Information and Computation 105 (1993) 159–267.

[4] H.P. Barendregt, The Lambda Calculus, Its Syntax and Semantics, 103, Studies in Logic and the Foundations of Mathematics, of North-Holland, Amsterdam, 1984.

[5] A. Berarducci, M. Dezani-Ciancaglini, Infinite λ-calculus and types, Theore. Comput. Sci. 212 (1999) 29–75.

[6] P. Di Gianantonio, Game semantics of the pure lazy λ-calculus, In: Proc. Typed Lambda Calculus and Applications, 5th Internat. Conf. TLCA 2001, Krakow, Poland, May 2–5, 2001, Vol. 2044, Lecture Notes in Computer Science, Springer, Berlin, 2001, pp. 106–120.

[7] P. Di Gianantonio, G. Franco, F. Honsell, Game semantics for untyped λβη-calculus, Vol. 1581, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1999, pp. 114–128.

[8] J.M.E. Hyland, C.-H.L. Ong, On full abstraction for PCF: I, II and III, Inform. Comput. 163 (2000) 285–408.

[9] A.D. Ker, Innocent Game Models of Untyped Lambda Calculus. PhD Thesis, University of Oxford, 2001.

[10] J.R. Kennaway, J.W. Klop, M.R. Sleep, F.J. de Vries, Infinitary lambda calculus, Theore. Comput. Sci. 175 (1997) 93–125.

[11] A.D. Ker, H. Nickau, C.-H.L. Ong, A universal innocent game model of the Böhm tree lambda theory, In: Computer Science Logic: Pro. 8th Ann. Conf. of the EACSL, Madrid, Spain, September 1999, Vol. 1683, Lecture Notes in Computer Science, Springer, Berlin, 1999, pp. 405–419.

[12] A.D. Ker, H. Nickau, C.-H.L. Ong, Innocent game models of untyped λ-calculus, Theoret. Comput. Sci. 272 (2002) 247–292.
[13] J. Lambek, P.J. Scott, Introduction to higher order categorical logic, Vol. 7, Cambridge Studies in Advanced Mathematics, of Cambridge University Press, Cambridge, 1986.
[14] G.A. McCusker, Games and full abstraction for a functional metalanguage with recursive types. Distinguished Dissertations in Computer Science. Springer, Berlin, 1998.
[15] R. Nakajima, Infinite normal forms for the λ-calculus, In: C. Böhm, (edi.), Proc. Symp. on λ-calculus and Computer Science Theory, Vol. 37, Lecture Notes in Computer Science, Springer, Berlin, 1975, pp. 62–82.
[16] Hanno Nickau, Hereditarily sequential functionals: a game-theoretic approach to sequentiality. Shaker, 1996. Dissertation, Universität Gesamthochschule Siegen.
[17] C.-H.L. Ong, P. Di Gianantonio, Games Characterizing Levy-Longo Trees, in: Widmayer et al. (Ed.), Proc. of 29th Internat. Colloq. on Automata, Languages and Programming (ICALP 2002), Málaga, Spain, July 2002, volume 2380 of Lecture Notes in Computer Science, Springer, Berlin, 2002, pp. 476–487.
[18] C.-H.L. Ong, The lazy lambda calculus: an investigation into the foundations of functional programming, Ph.D. Thesis, University of London, 1988.