Full Title[*]

Subtitle[†]

W. J. Gowers[‡]
PhD Student
Department of Computer Science
University of Bath
Bath, United Kingdom
W.J.Gowers@bath.ac.uk

Abstract

Text of abstract . . . .

Keywords keyword1, keyword2, keyword3

1 Introduction

Text of paper . . .

2 Idealized Algol with Countable Nondeterminism

The language that we will be modelling is the language Idealized Algol [Abramsky and McCusker 1999], extended with an additional constant ? representing countable nondeterminism. This is similar to the approach adopted in [Harmer and McCusker 1999], which extends Idealized Algol with *finite* nondeterminism. The types of the language are defined inductively as follows:

$$T ::= \mathtt{nat} \mid \mathtt{com} \mid \mathtt{Var} \mid T \to T\,.$$

Meanwhile, the terms are those given in [Abramsky and McCusker 1999], together with the nondeterministic choice:

$$\begin{aligned} M ::= {} & x \mid \lambda x.M \mid M\;M \mid \mathbf{Y}_T M \mid \\ & \mathsf{n} \mid \mathsf{skip} \mid \mathsf{suc}\;M \mid \mathsf{pred}\;M \mid \\ & \mathtt{If0}\;M\;M\;M \mid M; M \mid \\ & M := M \mid @M \mid \\ & \mathsf{new}_T\;M \mid \mathsf{mkvar}\;M\;M \mid ?\,. \end{aligned}$$

The typing rule for ? is $\Gamma \vdash ?\colon \mathtt{nat}$. We shall use the letter v to range over variables of type $\mathtt{Var}$.

The value forms in the language are given by

$$V ::= \mathsf{skip} \mid \mathsf{n} \mid \lambda x.M \mid v \mid \mathsf{mkvar}$$

We define predicates $M \Downarrow V$ (M may converge to V) and $M \Downarrow^{\text{must}}$ (M must converge) inductively to give a big-step operational semantics for our language. We give a selection of the appropriate rules in Figure 1; this presentation is almost identical to that given in [Harmer and McCusker 1999], with a slightly different rule for the countable nondeterminism. In each rule, $\langle s, M\rangle$ is a *configuration* of the language, where M is a term, and s is a *store*; i.e., a function from the set of variables free in M to the set of natural numbers. If s is a store and v a variable, we write $\langle s \mid v \mapsto n\rangle$ for the state formed by updating the value of the variable v to n.

Given a term M of ground type $\mathtt{com}$ or $\mathtt{nat}$, a proof π that $\langle s, M\rangle \Downarrow V$ gives rise to a (possibly infinite) sequence of natural numbers corresponding to a bottom-to-top, left-to-right reading of the natural numbers n used in the first rule for ?. We call such a sequence an *evaluation* of the configuration $\langle s, M\rangle$. It is perhaps easier to view this from the perspective of a small-step reduction: the sequence of numbers is formed by listing the different values that the nondeterminism constant ? decays to over the course of the evaluation π.

Let T be an Idealized Algol type, and let $M, N\colon T$ be terms. Then we write $M \sqsubseteq_{m\&m} N$ if for all compatible contexts $C[-]$ of ground type we have

$$\begin{gathered} C[M] \Downarrow V \Rightarrow C[N] \Downarrow V \\ C[M] \Downarrow^{\text{must}} \Rightarrow C[N] \Downarrow^{\text{must}} \end{gathered}$$

We write $M \equiv_{m\&m} N$ if $M \sqsubseteq_{m\&m} N$ and $N \sqsubseteq_{m\&m} M$.

One aspect of the language that makes it particularly difficult to model is that function application is not continuous, either with respect to functions or with respect to arguments.

[*] with title note
[†] with subtitle note
[‡] with author1 note

LICS'18, 9-12 July, 2018, Oxford, UK
2018.

$$\frac{\langle s, N\rangle \Downarrow \langle s', \mathsf{n}\rangle \qquad \langle s', M\rangle \Downarrow \langle s'', v\rangle}{\langle s, M := N\rangle \Downarrow \langle\langle s'' \mid v \mapsto n\rangle, \mathsf{skip}\rangle} \qquad \frac{\langle s, M\rangle \Downarrow \langle s', v\rangle}{\langle s, @M\rangle \Downarrow \langle s', \mathsf{n}\rangle}\ s'(v) = n \qquad \frac{\langle s, M\rangle \Downarrow \langle s', \mathsf{skip}\rangle \qquad \langle s', N\rangle \Downarrow \langle s'', V\rangle}{\langle s, M; N\rangle \Downarrow \langle s'', V\rangle}$$

$$\frac{\langle\langle s \mid v \mapsto 0\rangle, M\rangle \Downarrow \langle\langle s' \mid v \mapsto n\rangle, V\rangle}{\langle s, \mathsf{new}_T\ \lambda v.M\rangle \Downarrow \langle s', V\rangle} \qquad \frac{\langle s, M\mathsf{n}\rangle \Downarrow \langle s', V\rangle}{\langle s, M?\rangle \Downarrow \langle s', V\rangle}\ n \in \mathbb{N} \qquad \frac{\forall n \in \mathbb{N}.\langle s, M\mathsf{n}\rangle \Downarrow^{\mathrm{must}}}{\langle s, M?\rangle \Downarrow^{\mathrm{must}}}$$

$$\frac{\langle s, M\rangle \Downarrow^{\mathrm{must}} \qquad \forall\langle s', \lambda x.M'\rangle\ .\ \langle s, M\rangle \Downarrow \langle s', \lambda x.M'\rangle \Rightarrow \langle s', M'[N/x]\rangle \Downarrow^{\mathrm{must}}}{\langle s, M\ N\rangle \Downarrow^{\mathrm{must}}}$$

Figure 1. Operational semantics of Idealized Algol with Countable Nondeterminism

For example, it is easy to write a sequence of functions $\lessdot n \colon \mathtt{nat} \longrightarrow \mathtt{nat}$ that return 0 if their argument is less than n and diverge otherwise:

$$\begin{aligned} <0 &= \lambda m.\Omega \\ <1 &= \lambda m.\mathtt{If0}\ m\ \mathtt{0}\ \Omega \\ <2 &= \lambda m.\mathtt{If0}\ m\ \mathtt{0}\ (\mathtt{If0}\ (\mathtt{pred}\ m)\ \mathtt{0}\ \Omega) \\ &\dots \end{aligned}$$

It is easy enough to see (for example, by the denotational semantics) that the least upper bound of these functions is the function $\lambda x.\mathtt{0}$. However, it is clear that $<n\ ?\ \not\Downarrow^{\mathrm{must}}$ for each n, while $\lambda x.\mathtt{0}\ ?$ must converge to $\mathtt{0}$. Thus, $\lambda x.\mathtt{0}\ ?$ is not the least upper bound of the $<n\ ?$.

This is a bit of a problem, since continuity often plays an important role in full abstraction proofs – firstly, for proving Computational Adequacy of the recursion combinator, and, secondly, for deducing full abstraction from compact definability. Semantically, we shall deal with the first problem by adopting a novel approach to Computational Adequacy, and with the second problem by appealing to the stronger *universality* result: namely, that every *recursive* strategy is definable.

3 Game Semantics

We will use the Hyland-Ong version of Game Semantics, as in [Abramsky and McCusker 1999].

3.1 Arenas

An *arena* is given by a triple $A = (M_A, \lambda_A, \vdash_A)$, where

- M_A is a countable set of moves,
- $\lambda_A \colon m_A \longrightarrow \{O, P\} \times \{Q, A\}$ designates each move as either an O-*move* or a P-*move*, and as either a *question* or an *answer*. We define $\lambda_A^{OP} = \mathrm{pr}_1 \circ \lambda_A$ and $\lambda_A^{QA} = \mathrm{pr}_2 \circ \lambda_A$. We also define $\neg \colon \{O, P\} \times \{Q, A\} \to \{O, P\} \times \{Q, A\}$ to be the function that reverses the values of O and P while leaving $\{Q, A\}$ unchanged.
- $\vdash_A$ is an *enabling relation* between $M_A + \{*\}$ and M_A satisfying the following rules:
  - If $a \vdash_A b$ and $a \neq b$, then $\lambda_A^{OP}(a) \neq \lambda_A^{OP}(b)$.
  - If $* \vdash_A a$, then $\lambda_A(a) = OQ$ and $b \not\vdash_A a$ for all $b \in M_A$.
  - If $a \vdash_A b$ and b is an answer, then a is a question.

  We say that a move $a \in M_A$ is *initial* in A if $* \vdash_A a$.

Our base arenas will be the *flat arenas* for the types $\mathtt{nat}$ and $\mathtt{com}$. Given a set X, the flat arena on X is the arena with a single O-question q and a P-answer x for each $x \in X$, where $* \vdash q$ and $q \vdash x$ for each x. The denotation of the type $\mathtt{nat}$ will be the flat arena $\mathbb{N}$ on the set of natural numbers, while the denotation of the type $\mathtt{com}$ will be the flat arena $\mathbb{C}$ on the singleton set $\{a\}$.

Given an arena A, a *justified string* in A is a sequence s of moves in A, together with *justification pointers* that go from move to move in the sequence in such a way that every non-initial move m in s has exactly one justification pointer going back to an earlier move n in s such that $n \vdash_A m$. We say that n *justifies* m. It is easy to see that every justified string must begin with an initial move, an hence with an O-question.

A *legal play* s is a justified string in A that strictly alternates between O-moves and P-moves and is such that the corresponding QA-sequence formed by applying λ_A^{QA} to terms is well-bracketed. We write L_A for the set of legal plays in A.

3.2 Games and strategies

We follow the approach taken by Abramsky and McCusker [Abramsky and McCusker 1999] – a middle road between the *arenas* of Hyland and Ong and the *games* of [Abramsky et al. 2000] that makes the linear structure more apparent.

Let s be a legal play in some arena A. If m and n are moves in s such that there is a chain of justification pointers leading from m back to n, we say that n *hereditarily justifies* m. Given some set S of initial moves in s, we write $s|_S$ for the subsequence of s made up of all those moves that are hereditarily justified by some move in S.

A *game* is a tuple $A = (M_A, \lambda_A, \vdash_A, P_A)$, where $(M_A, \lambda_A, \vdash_A)$ is an arena and P_A is a non-empty prefix-closed set of legal plays in that arena such that if $s \in P_A$ and I is a non-empty set of initial moves in s, then $s|_I \in P_A$.

3.2.1 Multiplicatives

Let A, B be games. We define games $A \otimes B$ and $A \multimap B$ as follows.

$$\begin{array}{rcl}
M_{A\otimes B} & = & M_A + M_B\,. \\
\lambda_{A\otimes B} & = & [\lambda_A, \lambda_B]\,. \\
* \vdash_{A\otimes B} n & \Leftrightarrow & * \vdash_A n \text{ or } * \vdash_B n\,. \\
m \vdash_{A\otimes B} n & \Leftrightarrow & m \vdash_A n \text{ or } m \vdash_B n\,. \\
P_{A\otimes B} & = & \{s \in L_{A\otimes B} \ :\ s|_A \in P_A \text{ and } s|_B \in P_B\}\,.
\end{array}$$

$$\begin{array}{rcl}
M_{A\multimap B} & = & M_A + M_B\,. \\
\lambda_{A\multimap B} & = & [\neg \circ \lambda_A, \lambda_B]\,. \\
* \vdash_{A\multimap B} n & \Leftrightarrow & * \vdash_B m\,. \\
m \vdash_{A\multimap B} n & \Leftrightarrow & \begin{array}{l} m \vdash_A n \text{ or } m \vdash_B n \\ \text{or (for } m \neq *) * \vdash_B m \text{ and } * \vdash_A n\,. \end{array} \\
P_{A\multimap B} & = & \{s \in L_{A\multimap B} \ :\ s|_A \in P_A \text{ and } s|_B \in P_B\}\,.
\end{array}$$

3.2.2 Modelling countable nondeterminism

Our definition of a strategy will be modelled upon that given in [Harmer and McCusker 1999]. We model nondeterministic computations by relaxing the determinism constraint on strategies – so player P may have multiple replies to any given O-move.

In addition, we have to keep track of any possible divergence in the computation; this is so we can distinguish terms such as

$$\texttt{If0}\ ?\ \Omega\ \texttt{0} \qquad\qquad \texttt{0}\,,$$

where the term on the right must converge (to $\texttt{0}$), while the term on the left has a possible divergence. The traditional way of representing divergences in game semantics is by a partiality in the strategy; i.e., an O-move to which P has no reply, but this partiality will be obscured by the alternative behaviour in the denotation of the strategy on the left.

We follow [Harmer and McCusker 1999] by modelling a strategy as a pair (T_σ, D_σ), where T_σ is a nondeterministic strategy in the usual sense and D_σ is a set of O-plays after which there is a possibility of divergence.

Tracking divergences explicitly in this way requires some care when we compose strategies. Specifically, we need to be able to add new divergences into strategies when they arise through 'infinite chattering' or *livelock*. For example, the denotation of the term

$$M = \mathbf{Y}_{\mathsf{nat}\to\mathsf{nat}}(\lambda f.\lambda n.n; (f n))\,,$$

where $n; P$ is a shorthand for $\texttt{If0}\ n\ P\ P$, is given by a total strategy, without divergences: namely the strategy μ with plays of the form

$$\begin{array}{cc}
\mathbb{N} & \mathbb{N} \\
 & q \\
q & \\
n_1 & \\
q & \\
n_2 & \\
\vdots &
\end{array}\,.$$

However, when we compose this strategy with any total strategy for $\mathbb{N}$ on the left, we expect the resulting strategy to contain divergences, since the term $M\mathsf{n}$ diverges for any n.

The approach adopted in [Harmer and McCusker 1999] is to check specifically for infinite chattering between strategies $\sigma\colon A \multimap B$ and $\tau\colon B \multimap C$ by checking whether the set $\sigma \| \tau$ contains any infinite increasing sequence of plays ending with moves in B. If there is such a sequence, then it restricts to some O-position in $\sigma; \tau$ and we add in a divergence at that position.

This works very satisfactorily for finite nondeterminism, but not at all for countable nondeterminism. To see why, consider the term

$$N = \mathbf{Y}_{\mathsf{nat}\to\mathsf{nat}\to\mathsf{nat}}(\lambda g.\lambda mn.\texttt{If0}\ m\ \texttt{0}\ n; (g\ (\texttt{pred}\ m)\ n))?$$

This term first chooses a natural number m, and then reads from its input n for a total of m times before eventually returning $\texttt{0}$. Thus, its denotation is the strategy ν with maximal plays of the form:

$$\begin{array}{cc}
\mathbb{N} & \mathbb{N} \\
 & q \\
q & \\
n_1 & \\
\vdots & \\
q & \\
n_m & \\
 & 0
\end{array}\,.$$

Note that this strategy strictly contains the one we considered before, and therefore that the denotation of

$$\texttt{If0}\ ?MN$$

has the same denotation as N, even though it has all the divergent evaluations of M, while $N\mathsf{n} \Downarrow^{\mathrm{must}}$ for all n. Moreover, if we try to compose $[\![N]\!]$ with the strategy on $\mathbb{N}$ that always returns 1, then we end up with an infinite increasing sequence of positions, which triggers the introduction of a divergence into the composite – even though no divergence occurs in the evaluation of N.

Aside from making violating soundness for the model, this example actually leads to composition not being associative if we naively extend the Harmer-McCusker model from finite to infinite nondeterminism (e.g., see [Harmer 1999, 4.4.1]).

Somehow, the crucial point is that we need to distinguish between terms like M, which contain infinite sequences of moves, and terms like N, which contain arbitrarily large finite sequences of oves. The way that we do this is by making the infinite sequences of moves explicit in our strategies, in the style of [Roscoe 1993] and [Levy 2008]. Then the denotation of M will contain an infinite sequence, while the denotation of N will contain arbitarily long finite sequences, but no infinite sequences.

The games in our model will be the same as those that we considered in the last section, but our definition of a strategy will change.

3.2.3 Strategies

Given an arena A, we define an *infinite justified string* in the obvious way. We define $\bar{P_A}$ to be P_A together with the set of all those infinite justified sequences that have all finite prefixes in P_A.

Let A be a game. A *strategy* σ for A is a pair (T_σ, D_σ), where:

- T_σ is a non-empty prefix-closed subset of $\bar{P_A}$ such that if $s \in T_\sigma$ is a P-position and $sa \in P_A$ then $sa \in T_\sigma$.
- $D_\sigma \subseteq \bar{P_A}$ is a postfix-closed set of plays in $\bar{P_A}$ that either end with an O-move or are infinite. We require D_σ to obey the following rules:

  Divergences come from plays If $d \in D_\sigma$ then there exists some $s \sqsubseteq d$ such that $s \in T_\sigma \cap D_\sigma$.

  Diverge-or-reply If $s \in T_\sigma$ is an O-position, then either $s \in D_\sigma$ or $sa \in T_\sigma$ for some legal play sa.

  Infinite positions are divergent If $s \in T_\sigma$ is infinite, then $s \in D_\sigma$.

3.2.4 Composition of strategies

Given games A, B, C, we define a justified string over A, B, C to be a sequence $\mathfrak{s}$ of moves with justification pointers from all moves except the initial moves in C. Given such a string, we may form the restrictions $\mathfrak{s}|_{A,B}$ and $\mathfrak{s}|_{B,C}$ by removing all moves in either C or A, together with all justification pointers pointing into these games. We define $\mathfrak{s}|_{A,C}$ to be the sequence formed by removing all moves from B from $\mathfrak{s}$ and all pointers to moves in B, *unless* we have a sequence of pointers $a \longrightarrow b \longrightarrow c$, in which case we replace them with a pointer $a \longrightarrow c$.

We call such a sequence $\mathfrak{s}$ a *legal interaction* if $\mathfrak{s}|_{A,B} \in P_{A \multimap B}$, $\mathfrak{s}|_{B,C} \in P_{B \multimap C}$ and $\mathfrak{s}|_{A,C} \in P_{A \multimap C}$. We write $\mathrm{int}_\infty(A, B, C)$ for the set of (possibly infinite) legal interactions between A, B and C.

Now, given strategies $\sigma \colon A \multimap B$ and $\tau \colon B \multimap C$, we define

$$T_\sigma \| T_\tau = \{\mathfrak{s} \in \mathrm{int}_\infty(A, B, C) \,:\, \mathfrak{s}|_{A,B} \in T_\sigma,\ \mathfrak{s}|_{B,C} \in T_\tau\}\,,$$

and then set

$$T_{\sigma;\tau} = \{\mathfrak{s}|_{A,C} \,:\, \mathfrak{s} \in T_\sigma \| T_\tau\}\,.$$

As for divergences in $\sigma; \tau$, our approach is actually simpler than that in [Harmer and McCusker 1999]; we set

$$D_\sigma \lightning D_\tau = \left\{ \mathfrak{s} \in \mathrm{int}_\infty(A, B, C) \;\middle|\; \begin{array}{l} \textbf{either } \mathfrak{s}|_{A,B} \in D_\sigma \\ \quad \text{and } s|_{B,C} \in T_\tau \\ \textbf{or } \mathfrak{s}|_{A,B} \in T_\sigma \\ \quad \text{and } s|_{B,C} \in D_\tau \end{array} \right\}.$$

We then set

$$D_{\sigma;\tau} = \mathrm{pocl}_{A \multimap C}\{\mathfrak{s}|_{A,C} \,:\, \mathfrak{s} \in D_\sigma \lightning D_\tau\}\,,$$

where $\mathrm{pocl}\, X$ denotes the *postfix closure* of X; i.e., the set of all O-plays in $P_{A \multimap C}$ that have some prefix in X.

Note that there is no need to consider separately, as Harmer and McCusker do, divergences that arise through 'infinite chattering': in our model, a case of infinite chattering between strategies σ and τ is itself a legal interaction between the two strategies, which is necessarily divergent (because it is infinite) and therefore gives rise to some divergence in $\sigma; \tau$.

We need to impose one more condition on strategies:

Definition 3.1. Let σ be a strategy for a game A. We say that σ is *complete* if $T_\sigma = \bar{T_\sigma}$; i.e., T_σ contains an infinite position s if it contains every finite prefix of s.

Any finite-nondeterminism strategy in the sense of [Harmer and McCusker 1999] may be interpreted as a complete strategy by enlarging it with all its infinite limiting plays. However, when we introduce countable nondeterminism, we introduce the possibility of strategies that are not complete. For example, the strategy ν that we mentioned above has an infinite increasing sequence of plays $q0 \sqsubseteq q0q0 \sqsubseteq \cdots$, but has no infinite play corresponding to its limit. Nonetheless, we do not want to allow arbitrary strategies: for example, the strategy μ above should include the infinite play $qq0q0\ldots$; the strategy μ° formed by removing this infinite play has no meaning in our language. Indeed, if we compose μ° with the strategy $\mathtt{0}$ for $\mathbb{N}$ on the left, then the resulting strategy does not satisfy diverge-or-reply.

Definition 3.2. Let σ be a strategy for a game A. We say that σ is *locally complete* if it may be written as the union of countably many complete strategies; i.e., there exist σ_n such that $T_\sigma = \bigcup T_{\sigma_n}$ and $D_\sigma = \bigcup D_{\sigma_n}$.

From now on, we will use 'strategy' to mean *locally complete strategy.*

We need to show that the composition of locally complete strategies is locally complete. Note that this does not hold for *complete* strategies: for example, our term N above can be written as N' ?, where N' is a deterministic term with complete denotation ν'. Then we have $\nu = \top_{\mathbb{N}}; \nu'$, but ν is not complete. However, we can show that the composition of *deterministic* complete strategies is complete; since a locally complete strategy may always be written as the union of complete deterministic strategies, this is sufficient to show that the composition of locally complete strategies is locally complete.

Definition 3.3. We say that a strategy σ for a game A is *deterministic* if

- it is complete;
- whenever sab, sac are P-plays in T_σ we have $b = c$ and the justifier of b is the justifier of c;
- If $s \in D_\sigma$ then either s is infinite or there is no a such that $sa \in T_\sigma$.

Lemma 3.4. *Let A, B, C be games and let $\sigma\colon A \multimap B, \tau\colon B \multimap C$ be deterministic complete strategies. Then $\sigma; \tau$ is complete.*

Proof. The proof relies on a lemma from [Hyland and Ong 2000] that states (in our language) that if σ and τ are deterministic strategies and $s \in T_{\sigma;\tau}$ then there is a unique minimal $\mathfrak{s} \in T_\sigma \| T_\tau$ such that $\mathfrak{s}|_{A,C} = s$. That means that if $s_1 \sqsubseteq s_2 \sqsubseteq \cdots$ is an infinite increasing sequence of plays in $T_{\sigma;\tau}$, with infinite limit s, then there is a corresponding infinite increasing sequence of legal interactions $\mathfrak{s}_1 \sqsubseteq \mathfrak{s}_2 \sqsubseteq \cdots$. Then the limit of this sequence is an infinite legal interaction $\mathfrak{s}$ and we must have $\mathfrak{s}|_{A,B} \in \sigma$, $\mathfrak{s}|_{B,C} \in \tau$ by completeness of σ and τ. Therefore, $s = \mathfrak{s}|_{A,C} \in T_{\sigma;\tau}$. □

Corollary 3.5. *The composition of strategies $\sigma\colon A \multimap B$ and $\tau\colon B \multimap C$ is a well-formed strategy for $A \multimap C$.*

Proof. The only tricky point is eastablishing that diverge-or-reply holds for $\sigma; \tau$. Again, it is sufficient to prove this in the case that σ and τ are deterministic and complete. Then it essentially follows from the argument used in [Abramsky and Jagadeesan 1994] that shows that a partiality at an O-position $s \in T_{\sigma;\tau}$ must arise either from a partiality in T_σ or T_τ or from 'infinite chattering' between σ and τ. In the first case, the diverge-or-reply rule for σ and τ gives us a divergence at s in $\sigma; \tau$. In the second case, an infinite chattering between σ and τ corresponds to an infinite interaction $\mathfrak{s} \in \int(A, B, C)$ ending with infinitely many moves in B such that $\mathfrak{s}|_{A,C} = s$. Completeness for σ and τ tells us that $\mathfrak{s}|_{A,B} \in D_\sigma$ and $\mathfrak{s}|_{B,C} \in D_\tau$ and therefore that $\mathfrak{s}|_{A,C} \in D_{\sigma;\tau}$. □

3.2.5 Associativity of composition

In fact, the proof of associativity of composition is pretty much the same in our model as it is in any other model of game semantics. However, it is worth saying a few words about it, since the model obtained by naively extending the Harmer-McCusker model to unbounded nondeterminism does not have an associative composition. The point is that this is not really a problem with associativity, but rather that this naive model gives the wrong result for the composition of strategies. For example, if ν is the strategy we defined above, and $\mathtt{0}$ is the 'constant 0' strategy on $\mathbb{N}$, then $\mathtt{0}; \nu$ has a divergence in the naive model, because the strategies $\mathtt{0}$ and ν appear to be engaged in infinite chattering. In our model, we have fixed that problem, because the strategy ν contains no infinite plays, and so no divergences arise in the composition.

3.3 A symmetric monoidal closed category

Given a game A, we define a strategy id_A on $A \multimap A$, where T_{id_A} is given by

$$\{s \in P_{A_1 \multimap A_2} \;:\; \text{for all even-length } t \sqsubseteq s,\; t|_{A_1} = t|_{A_2}\}\,,$$

where we distinguish between the two copies of A by calling them A_1 and A_2, and where D_σ is the set of all infinite plays in T_σ. This is an identity for the composition we have defined, and so we get a category $\mathcal{G}_{ND}$ of games and nondeterministic strategies. Moreover, the connectives $\otimes$ and $\multimap$ exhibit $\mathcal{G}_{ND}$ as a symmetric monoidal closed category.

$\mathcal{G}_{ND}$ has an important subcategory $\mathcal{G}_D$ of deterministic complete strategies; this category is isomorphic to the category considered in [Abramsky and McCusker 1999].

3.4 Products and Exponentials

Let A, B be games We define a new game $A \times B$. The arena for $A \times B$ is the same as the arena for $A \otimes B$, but now $P_{A \times B}$ is the disjoint union of P_A and P_B inside that arena:

$$\begin{aligned} P_{A \times B} = {} & \{s \in L_{A \otimes B} \;:\; s|_A \in P_A \text{ and } s|_B = \epsilon\} \\ & \cup \{s \in L_{A \otimes B} \;:\; s|_A = \epsilon \text{ and } s|_B \in P_B\}\,. \end{aligned}$$

Then $A \times B$ is the category-theoretic product of A and B in $\mathcal{G}_{ND}$.

We define a game $!A$ as follows. The arena for $!A$ is the same as the arena for A, but the set of plays is given by

$$P_{!A} = \{s \in L_A \;:\; s|_m \in P_A \text{ for each initial move } m\}\,.$$

There are strategies $\texttt{mult}_A \colon !A \longrightarrow !A \otimes !A$, $\supset \colon !(A \times B) \xrightarrow{\cong} !A \otimes !B$ and $\texttt{der}_A \colon !A \longrightarrow A$ that can be used to create a Cartesian closed category $\mathcal{G}^{!}_{ND}$, in which morphisms from A to B are strategies for the game $!A \multimap B$. This construction is essentially the co-Kleisli category on a comonad given by !, but certain technical issues prevent us from presenting it in this way. See [Abramsky and McCusker 1999] for full details.

3.5 Constraining strategies

Given a non-empty justified string s in an arena A, we define the *P-view* $\ulcorner s \urcorner$ of s inductively as follows.

$$\begin{aligned} \ulcorner sm \urcorner &= m\,, && \text{if } m \text{ is initial;} \\ \ulcorner sntm \urcorner &= \ulcorner s \urcorner nm\,, && \text{if } m \text{ is an } O\text{-move and} \\ &&& n \text{ justifies } m; \\ \ulcorner sm \urcorner &= \ulcorner s \urcorner m\,, && \text{if } m \text{ is a } P\text{-move.} \end{aligned}$$

We say that a play sm ending in a P-move is *P-visible* if the justifier of m is contained in $\ulcorner m \urcorner$. We say that a strategy σ for a game A is *visible* if every P-position $s \in T_\sigma$ is P-visible. It can be shown that the composition of visible strategies is visible, and that we can build a Cartesian closed category using our exponential.

The resulting category $\mathcal{G}^{!}_{D,vis}$ of games and deterministic visible strategies is a fully abstract model of Idealized Algol [Abramsky and McCusker 1999].

3.6 Recursive games and strategies

Most full abstraction results go via a definability result that says that all *compact* strategies are definable [Curien 2007]. However, deducing full abstraction from compact definability makes essential use of continuity properties that are absent when we deal with countable nondeterminism. We will therefore need to appeal to a stronger result – that of *universality*, which states that *every* strategy is definable. Clearly, universality does not hold for any of our categories of games, since there are many non-computable functions $\mathbb{N} \longrightarrow \mathbb{N}$. However, Hyland and Ong proved in [Hyland and Ong 2000] that every *recursively presentable* innocent strategy is PCF-definable.

In order to define recursively presentable strategies, we need to work with *enumerated games*; i.e., games where the set of moves comes with an enumeration to the natural numbers. Clearly our base games $\mathbb{N}$ and $\mathbb{C}$ can be enumerated, as can the tensor product, linear implication, exponential and product of games.

Proposition 3.6. *Let $\mathcal{G}_{D,vis,rec}$ be the category of games and recursive visible strategies. Then $\mathcal{G}_{D,vis,rec}$ is a fully abstract model of Idealized Algol in which every morphism is definable.*

Proof. This follows from the corresponding results for PCF, together with the *innocent factorization* result of [Abramsky and McCusker 1999]. See also [Murawski and Tzevelekos 2013]. □

3.7 Deterministic Factorization

Our definability results will hinge on a *factorization theorem*, showing that every nondeterministic strategy may be written as the composition of a deterministic strategy with the nondeterministic 'oracle' $\top_{\mathbb{N}}$. We can then deduce definability from definability in the model of deterministic Idealized Algol.

Note that our result is a bit simpler than the corresponding result in [Harmer and McCusker 1999]; this is because it is easier to model a countable source of nondeterminism than a 'finite but arbitrarily large' source.

Proposition 3.7. *Let $\sigma \colon I \longrightarrow A$ be a strategy for a game A in $\mathcal{G}_{ND}$. Then we may write σ as $\top_{\mathbb{N}}; \mathrm{Det}(\sigma)$, where $\mathrm{Det}(\sigma) \colon !\mathbb{N} \longrightarrow A$ is a deterministic strategy and $\top_{\mathbb{N}}$ is the strategy for $!\mathbb{N}$ given by*

- *$T_{\top_{\mathbb{N}}} = P_{!\mathbb{N}}$.*
- *$D_{\top_{\mathbb{N}}}$ is the set of infinite positions in $T_{\top_{\mathbb{N}}}$.*

Proof. We begin by fixing an injection code_A from the set of P-moves in A into the natural numbers. In the enumerated case, this is given to us already.

We first assume that the strategy σ is complete. Then the strategy $\mathrm{Det}(\sigma)$ is very easy to describe. For each O-position $sa \in T_\sigma$, we have some set B of possible replies to sa, which we order as $b_1, b_2, \cdots$, where $\mathrm{code}_A(b_1) < \mathrm{code}_A(b_2) < \cdots$. We insert a request to the oracle for a natural number; then, depending on her answer j, we play the next move as follows:

- If $0 < j \leq \mathrm{code}_A(b_1)$, then play b_1.
- If $\mathrm{code}_A(b_n) < j \leq \mathrm{code}_A(b_{n+1})$ then play b_{n+1}.
- If $j = 0$ and $sa \in D_\sigma$, then play nothing, and put the resulting play inside $D_{\mathrm{Det}(\sigma)}$. Otherwise, play b_1.

Lastly, we close under limits to make the strategy $\mathrm{Det}(\sigma)$ complete. $\mathrm{Det}(\sigma)$ is clearly nondeterministic. Checking that $\top_{\mathbb{N}}; \mathrm{Det}(\sigma) = \sigma$ is easy for finite plays; for infinite plays, it follows by completeness of σ.

Lastly, if σ is the union of complete strategies $\sigma_1, \sigma_2, \cdots$, we insert an additional request to the oracle immediately after the very first move by player O; after receiving a reply k, we play according to σ_k. □

Note that in the recursive case, $\mathrm{Det}(\sigma)$ is clearly recursively presentable if σ is. Furthermore, if σ is visible, then so is $\mathrm{Det}(\sigma)$.

4 Cones on Monoidal Functors

4.1 Surjective and winning strategies

A useful fact about the category $\mathcal{G}$ of games and deterministic strategies, proved in [Hyland and Schalk 2002], is that it admits a faithful, monoidal and linearly distributive functor into the category **Rel** of sets and relations, with monoidal structure given by the Cartesian product and exponentials given by the multiset comonad W. The functor sends a game A to the set P_A and sends a strategy $\sigma\colon A \multimap B$ to the relation

$$\underline{\sigma} = \{(s|_A, s|_B)\ :\ s \in \sigma\} \subseteq P_A \times P_B$$

Given a strategy $\sigma\colon A \multimap B$, we say that σ is *surjective* if $\underline{\sigma}$ is a surjective relation; i.e., if for any $b \in P_B$ there exists some $s \in \sigma$ such that $s|_B = b$. Monoidality and linear distributivity of the Hyland-Schalk functor $\mathcal{G} \to$ **Rel** show that the composition of surjective strategies is surjective, as is the tensor product of surjective strategies and the bang of a surjective strategy.

Given a strategy $\sigma\colon A \multimap B$, we say that σ is *winning* if σ is total and if for all positions $s \in \sigma$, if $sb \in \sigma$, where b is an O-move in B, then there is some $t \in \sigma$ such that $sb \sqsubseteq t$ and t ends with a move in B.

Once again, the composition, tensor product and bang of winning strategies is still winning. One way of seeing this is by relating our definition to the definition of a winning strategy given in [Abramsky and Jagadeesan 1994], in which a game comes attached with a set of winning infinite positions. Then we say that a strategy σ is winning if it is total and if every infinite position arising as the limit of plays in σ is a winning position. If A and B are such that every infinite position is winning, then our notion of a winning strategy coincides with that notion.

The reason we are interested in surjective winning strategies is that they preserve top strategies:

Lemma 4.1. *We say that a game A is P-*winning *if for all O-positions $s \in P_A$ there exists some a such that $sa \in P_A$. Let A, B be P-winning games, and let $\sigma\colon A \multimap B$ be a surjective winning strategy. Write $\top_A$ for the (nondeterministic) strategy given by $T_{\top_A} = P_A$ and $D_{\top_A} = P_A^\omega$ (i.e., the only divergent plays are the infinite ones), and do the same for B. Then $\top_A;\sigma = \top_B$.*

Proof. We can show that $T_{\top_A;\sigma} = T_{\top_B}$ from surjectivity of σ, using the Hyland-Schalk functor, since the equivalent result clearly holds in **Rel**. Now suppose that s is some divergent position in $\top_A;\sigma$. Then $s = \mathfrak{s}|_B$, where $\mathfrak{s}$ is a position in σ and $\mathfrak{s}|_A$ is a divergent position in A. Then $\mathfrak{s}|_A$ must be infinite; since σ is winning, this means that $\mathfrak{s}|_B$ must be infinite as well, and therefore that $s \in D_{\top_B}$. □

$$\begin{tikzcd}
\mathcal{D}' \arrow[r, "F"] \arrow[d, "g"'] \arrow[dr, phantom, "\Downarrow \phi"] & \mathcal{C}' \arrow[d, "f"] \\
\mathcal{D} \arrow[r, "G"'] & \mathcal{C}
\end{tikzcd}$$

Figure 2. A generalized monoidal natural transformation ϕ is a particular sort of natural transformation between composites of lax and oplax monoidal functors.

We now define a category $\mathcal{G}_{sw}$ whose objects are P-winning games and where the morphisms from a game A to a game B are deterministic surjective winning strategies from A to B. We write $\mathcal{G}_{sw,rec}$ for the category where the morphisms are *recursive* surjective, winning strategies.

Write j for the inclusion $\mathcal{G}_{sw,rec} \hookrightarrow \mathcal{G}_{rec}$, and write J for the inclusion functor $\mathcal{G} \hookrightarrow \mathcal{G}_{ND}$ that sends a deterministic strategy σ to the strategy given by $T = \sigma$ and $D = \mathrm{pocl}\{s \in \sigma^-\ :\ sa \notin \sigma \forall a\}$ Then the lemma we have just proves means that

$$\top\colon I \Rightarrow J \circ j$$

is a natural transformation, where I is the constant I functor.

Moreover, we would expect Lemma 4.1 to hold in *any* sensible game-semantics model of nondeterministic PCF, so the following makes sense as a definition

Definition 4.2. A *game semantics category for nondeterministic PCF* is a category $\mathcal{D}$ together with a functor $J\colon \mathcal{G}_{rec} \to \mathcal{D}$ and a natural transformation $\top\colon I \Rightarrow J \circ j\colon \mathcal{G}_{sw,rec} \to \mathcal{D}$.

4.2 Generalized monoidal natural transformations

We will need one more technical definition.

Let C',C',$\mathcal{D}'$,$\mathcal{D}$ be monoidal categories, let $f\colon C' \to C$, $g\colon \mathcal{D}' \to \mathcal{D}$ be oplax monoidal functors and let $F\colon \mathcal{D}' \to C'$, $G\colon \mathcal{D} \to C$ be lax monoidal functors, as in Figure 2.

We say that a natural transformation $\phi\colon fF \Rightarrow Gg$ is a *generalized monoidal natural transformation* if the following two diagrams commute for all objects X, Y of $\mathcal{D}'$:

$$\begin{tikzcd}
fFX \otimes fFY \arrow[r, "\phi_X \otimes \phi_Y"] & GgX \otimes GgY \arrow[d, "m_G"] \\
f(FX \otimes FY) \arrow[u, "m_f"] \arrow[d, "fm_F"'] & G(gX \otimes gY) \\
fF(X \otimes Y) \arrow[r, "\phi_{X \otimes Y}"] & Gg(X \otimes Y) \arrow[u, "Gm_g"']
\end{tikzcd}$$

$$\begin{tikzcd}
 & fFI_{\mathcal{D}'} \arrow[r, "\phi_I"] & GgI_{\mathcal{D}'} \arrow[dr, "G\epsilon_g"] & \\
fI_{C'} \arrow[ur, "f\epsilon_F"] \arrow[drr, "\epsilon_f"] & & & GI_{\mathcal{D}} \\
 & & I_C \arrow[ur, "\epsilon_G"] &
\end{tikzcd} .$$

The particular structure that generalized natural transformations fit into is that of a *double category* (see [Grandis and Paré 1999] for the definition); in particular, the double category **SymmMonCat** introduced in [Grandis 2004], in which the objects are symmetric monoidal categories, the horizontal morphisms are lax symmetric monoidal functors, the vertical morphisms are oplax symmetric monoidal functors and the 2-cells are generalized monoidal natural transformations as defined above.

Generalized monoidal natural transformations may thus be composed either horizontally or vertically in the usual way; i.e., given a pair of squares as in Figure 2 that may be pasted together along an edge, then we may define a generalized monoidal natural transformation across the pasted square. We write $(\phi|\psi)$ for the horizontal composition of natural transformations ϕ, ψ pasted together along an oplax monoidal functor and $\left(\frac{\phi}{\psi}\right)$ for the vertical composition of natural transformations ϕ, ψ pasted together along a lax monoidal natural functor.

4.3 The cone of a monoidal functor

Let $j \colon C' \to C$ be an oplax monoidal functor. We define a *j-category* to be a category $\mathcal{D}$ together with a lax monoidal functor $J \colon C \to \mathcal{D}$ and a generalized natural transformation $\phi \colon I \to Jj$:

$$\begin{tikzcd}
C' \arrow[r, "()"] \arrow[d, "j"'] \arrow[dr, phantom, "\Downarrow \phi"] & I \arrow[d, "I"] \\
C \arrow[r, "J"'] & \mathcal{D}
\end{tikzcd} .$$

Example 4.3. For example, our previous definition of a game semantics category for nondeterministic PCF can be thought of as a j-category, where j is the inclusion functor $\mathcal{G}_{sw,rec} \hookrightarrow \mathcal{G}_{rec}$.

We define a *morphism of j-categories* going from a j-category $(\mathcal{D}', J', \phi')$ to a j-category $(\mathcal{D}, J, \phi)$ to be a lax monoidal functor $\mathcal{F} \colon \mathcal{D}' \to \mathcal{D}$, sending I to I, such that $J = \mathcal{F} \circ J'$ and ϕ is the vertical composition of ϕ' with the 'identity transformation' given by this identity; i.e.:

$$\begin{tikzcd}
C' \arrow[r, "()"] \arrow[d, "j"'] \arrow[dr, phantom, "\Downarrow \phi'"] \arrow[ddr, phantom, "\Downarrow \phi"'] & I \arrow[d, "I"] \\
C \arrow[r, "J'"'] \arrow[d, "\mathrm{id}"'] \arrow[dr, phantom, "\Downarrow \mathrm{id}"] & \mathcal{D}' \arrow[d, "\mathcal{F}"] \\
C \arrow[r, "J"'] & \mathcal{D}
\end{tikzcd} .$$

Our main technical result in this section will be the following:

Proposition 4.4. *The category j-***Cat** *of j-categories and j-category morphisms has an initial object C_j*

By analogy with homotopy theory, we call this initial object the *mapping cone* or *cone* on j.

To see why this is useful, consider another important topic from the theory of semantics. We say that a j-category $(\mathcal{D}, J, \phi)$ satisfies *ϕ-factorization* if for any pair of objects A and B of C and for every morphism $f \colon JA \to JB$ in $\mathcal{D}$, we may write f as a composite

$$JA \xrightarrow{\mathrm{lunit};\phi_X \otimes JA} JjX \otimes JA \xrightarrow{m_J} J(jX \otimes A) \xrightarrow{Jf'} JB,$$

where X is some object of C' and $f' \colon jX \otimes A \to B$ is some morphism in C.

Example 4.5. In the case that j is the inclusion $\mathcal{G}_{sw,rec} \hookrightarrow \mathcal{G}_{rec}$, and $\phi = \top$, ϕ-factorization is the same as *deterministic factorization*: the idea that every morphism can be written as the composition of a deterministic recursive strategy with $\top_X$ for some X.

Then the following result is easy to prove:

Proposition 4.6. *Let $(\mathcal{D}, J, \phi)$ be a j-category satisfying ϕ-factorization. Suppose $\mathcal{F} \colon \mathcal{D}' \to \mathcal{D}$ is a morphism of j-categories. Then $\mathcal{F}$ is a full functor.*

In particular, the unique j-category morphism $C_j \to \mathcal{D}$ is a full functor. This is useful, because we are viewing game semantic models of nondeterministic PCF as j-categories (for a particular j), and therefore we can see that C_j admits a full functor into every such model that satisfies deterministic factorization. Since we typically require that our models satisfy such a factorization result, we can form any such model by taking C_j and imposing some equivalence relation on the morphisms. This suggests that we might consider C_j itself as our model of nondeterministic PCF.

4.4 Constructing the cone

We will now define what the cone C_j is and give an outline of why it is initial in the category of j-categories.

Recall that we start with an oplax symmetric monoidal functor $j\colon C' \to C$. The objects of C_j are the objects of C. Given objects A, B of C, the class of morphisms $A \to B$ is given by the colimit of the functor

$$C'^{\text{op}} \xrightarrow{j^{\text{op}}} C^{\text{op}} \xrightarrow{C[\,\_\_ \otimes A, B]} \mathbf{Set}\,.$$

That is, a morphism from A to B is a pair (X, f), where X is an object of C' and $f\colon jX \otimes A \to B$ is a morphism in C. Furthermore, (X, f) and (X', f') are considered to be equivalent if there is some morphism $h\colon X' \to X$ in C' making the following diagram commute:

$$\begin{tikzcd}
jX' \otimes A \arrow[r, "f'"] \arrow[d, "jh\otimes A"'] & B \\
jX \otimes A \arrow[ur, "f"'] &
\end{tikzcd} .$$

Given morphisms $(X, f)\colon A \to B$, $(Y, g)\colon B \to C$, the composite is given by $(Y \otimes X, f;g)$, where $f;g$ is the following composite in C:

$$\begin{aligned} j(Y \otimes X) \otimes A &\xrightarrow{m_j \otimes A} (jY \otimes jX) \otimes A \\ &\xrightarrow{\text{assoc}} jY \otimes (jX \otimes A) \\ &\xrightarrow{jY \otimes f} jY \otimes B \xrightarrow{g} C\,. \end{aligned}$$

We can prove that this composition is associative using the usual coherence theorems for monoidal categories. Similarly, the symmetric monoidal structure from C induces a symmetric monoidal structure on C_j. If C is symmetric monoidal *closed*, then so is C_j.

There is a functor $J\colon C \to C_j$ that is the identity on objects and which sends a strategy $f\colon A \to B$ to the strategy

$$jI \otimes A \xrightarrow{\epsilon_j \otimes A} I \otimes A \xrightarrow{\text{lunit}^{-1}} A \xrightarrow{f} B\,.$$

This functor is (strong) symmetric monoidal with respect to the monoidal structure on C_j. Lastly, the natural transformation $\phi\colon I \Rightarrow J \circ j$ is given by the composite

$$\phi_X = jX \otimes I \xrightarrow{\text{runit}^{-1}} jX\,,$$

considered as a morphism from I to jX in C_j. Proving that this is indeed a natural transformation makes essential use of the equivalence relation on morphisms in C_j. We can also prove that ϕ is a generalized monoidal natural transformation. Therefore, C_j is a j-category.

Now suppose that $(\mathcal{D}, F, \psi)$ is another j-category. We define a functor $H\colon C_j \to \mathcal{D}$ that sends an object $A = JA$ in C_j to the object FA in $\mathcal{D}$. Moreover, we send a morphism $f\colon jX \otimes A \to B$ from JA to JB in C_j to the following composite in $\mathcal{D}$:

$$FA \xrightarrow{\text{lunit};\psi_X \otimes FA} FjX \otimes FA \xrightarrow{m_F} F(jX \otimes A) \xrightarrow{Ff} FB\,.$$

Proving that this is indeed a functor is a little tricky, and relies on ψ being a generalized monoidal natural transformation. We can then go through and check that H is indeed a morphism of j-categories.

In order to prove uniqueness, note that, as a morphism of j-categories, H is determined on morphisms of the form Jf by the equation $F = H \circ J$ and on morphisms of the form ϕ_X by the equation $\psi = \left(\frac{\phi}{H}\right)$. But now, if $f\colon jX \otimes A \to B$ is any morphism from A to B in C_j, we may write f as the composite

$$JA \xrightarrow{\text{lunit};\phi_X \otimes A} JjX \otimes JA \xrightarrow{m_J} J(jX \otimes A) \xrightarrow{Jf} JB\,;$$

i.e., C_j satisfies ϕ-factorization. Since the functor H is required to be monoidal, this implies that it is determined everywhere on C_j by the requirement that it be a morphism of j-categories. Therefore, C_j is initial in the category of j-categories.

A consequence of the initiality of the cone construction is that it is *functorial*; that is, given a square of the following form:

$$\begin{tikzcd}
C' \arrow[r, "F"] \arrow[d, "j"'] & \mathcal{D}' \arrow[d, "j'"] \\
C \arrow[r, "G"'] \arrow[ur, phantom, "\Downarrow \psi"] & \mathcal{D}
\end{tikzcd} ,$$

where horizontal arrows correspond to lax monoidal functors and vertical arrows to monoidal functors, we get an induced lax monoidal functor $C_j \to C_j$, in a way that respects horizontal pasting of squares. In particular, if C and C' have linear structures, compatible with j, then C_j inherits a linear structure.

5 Denotational Semantics

We now restrict our attention to the inclusion functor $j\colon G_{sw,rec} \hookrightarrow \mathcal{G}_{rec}$. Then C_j inherits a linear structure from $\mathcal{G}_{rec}$.

5.1 Extensional order

Let $\sigma, \tau\colon I \to \mathbb{N}$ be morphisms in C_j, corrseponding to morphisms $\sigma'\colon jX \to \mathbb{N}$, $\tau'\colon jY \to \mathbb{N}$ in $\mathcal{G}_{rec}$. We say that $\sigma \lesssim \tau$ if the following two conditions hold:

- If there exists $s \in \sigma'$ such that $s|_{\mathbb{N}} = qn$, then there exists $t \in \tau'$ such that $t|_{\mathbb{N}} = qn$.
- If σ' is winning then τ' is winning.

In general, given strategies $\sigma, \tau\colon I \longrightarrow A$ in C_j, we say that $\sigma \lesssim \tau$ if for all morphisms $\rho\colon A \longrightarrow \mathbb{N}$ we have $\sigma;\rho \lesssim \tau;\rho$.

We can show that this is well-defined; i.e., that it respects the equivalence relation we have already defined on morphisms. This is a good exercise to show the importance of surjective and winning strategies to our model.

5.2 Countable nondeterminism

In order to model *countable* nondeterminism, rather than arbitrary nondeterminism, we want to constrain our model so that morphisms are strategies for $!\mathbb{N} \otimes A \longrightarrow B$, rather than $X \otimes A \longrightarrow B$ for an arbitrary game X.

Since $!\mathbb{N}$ has the structure of a comonoid, it gives rise to an oplax monoidal functor

$$n\colon I \longrightarrow \mathcal{G}_{rec}\,,$$

where I is the one-object monoidal category. Moreover, this functor is linear distributive from the trivial linear structure on I on to $\mathcal{G}$. The cone C_n is therefore a linear category.

Since $\mathcal{G}_{sw,rec}$ is a sub-linear category of $\mathcal{G}_{rec}$, the functor n factors through a functor $n'\colon I \longrightarrow \mathcal{G}_{rec}$. By initiality of the cone construction, this gives rise to a functor $C_n \longrightarrow C_j$. Our model of nondeterministic PCF will be the category C that is the image of this functor. A morphism in C from a game A to a game B will be a strategy for $!\mathbb{N} \otimes A \longrightarrow B$, but the equivalence relation on morphisms is that from C_j. For example, two morphisms $f, g\colon !\mathbb{N} \longrightarrow A$ are equivalent in C if there is a surjective winning strategy $!\mathbb{N} \longrightarrow !\mathbb{N}$ commuting with f and g, but they are not equivalent in C_n.

In fact, once we have applied the extensional quotient, there is no difference between working in C and working in C_n. We will use C, because it is closer to being a fully abstract model for nondeterministic PCF before we apply the extensional quotient.

5.3 The denotation of a term

Let $M = C[?]\colon T$ be a term of nondeterministic PCF, where $C[-]$ is a PCF context (i.e., a context not mentioning the constant ?). Then the denotation of M in C is given by the following composite:

$$I \xrightarrow{\phi_{!\mathbb{N}}} !\mathbb{N} \xrightarrow{J(\llbracket \lambda n.C[n] \rrbracket)} \llbracket T \rrbracket$$

In other words, when considered as a morphism in $\mathcal{G}_{rec}$, $\llbracket M \rrbracket$ is precisely the denotation of the term $\lambda n.C[n]$ in $\mathcal{G}_{rec}$.

5.4 Relation to our previous model

Since our Harmer-McCusker-style model $\mathcal{G}_{ND}$

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. nnnnnnn and Grant No. mmmmmmm. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

References

Samson Abramsky and Radha Jagadeesan. 1994. Games and Full Completeness for Multiplicative Linear Logic. *The Journal of Symbolic Logic* 59, 2 (1994), 543–574. http://arxiv.org/abs/1311.6057

Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. 2000. Full Abstraction for PCF. *Information and Computation* 163, 2 (2000), 409 – 470. https://doi.org/10.1006/inco.2000.2930

Samson Abramsky and Guy McCusker. 1999. Full Abstraction for Idealized Algol with Passive Expressions. *Theor. Comput. Sci.* 227, 1-2 (Sept. 1999), 3–42. https://doi.org/10.1016/S0304-3975(99)00047-X

Pierre-Louis Curien. 2007. Definability and Full Abstraction. *Electronic Notes in Theoretical Computer Science* 172 (2007), 301 – 310. https://doi.org/10.1016/j.entcs.2007.02.011 Computation, Meaning, and Logic: Articles dedicated to Gordon Plotkin.

Marco Grandis and Robert Paré. 1999. Limits in double categories. *Cah. Topologie Géom. Différ. Catégoriques* 40, 3 (1999), 162–220.

Pare Robert Grandis, Marco. 2004. Adjoint for double categories. *Cahiers de Topologie et Géométrie Différentielle Catégoriques* 45, 3 (2004), 193–240. http://eudml.org/doc/91684

R. Harmer and G. McCusker. 1999. A fully abstract game semantics for finite nondeterminism. In *Proceedings. 14th Symposium on Logic in Computer Science (Cat. No. PR00158)*. 422–430. https://doi.org/10.1109/LICS.1999.782637

Russell S. Harmer. 1999. *Games and full abstraction for nondeterministic languages.* Technical Report.

J.M.E. Hyland and C.-H.L. Ong. 2000. On Full Abstraction for PCF: I, II, and III. *Information and Computation* 163, 2 (2000), 285 – 408. https://doi.org/10.1006/inco.2000.2917

Martin Hyland and Andrea Schalk. 2002. Games on graphs and sequentially realizable functionals. Extended abstract. In *Logic in Computer Science, 2002. Proceedings. 17th Annual IEEE Symposium on*. IEEE, 257–264.

Paul Blain Levy. 2008. Infinite trace equivalence. *Annals of Pure and Applied Logic* 151, 2 (2008), 170 – 198. https://doi.org/10.1016/j.apal.2007.10.007

Andrzej S. Murawski and Nikos Tzevelekos. 2013. Deconstructing General References via Game Semantics. In *Foundations of Software Science and Computation Structures*, Frank Pfenning (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 241–256.

A. W. Roscoe. 1993. Unbounded Non-determinism in CSP. *Journal of Logic and Computation* 3, 2 (1993), 131. https://doi.org/10.1093/logcom/3.2.131

A Appendix

Text of appendix ...