Full abstraction for nominal exceptions

Nikos Tzevelekos
Oxford University Computing Laboratory

Abstract

We examine the denotational semantics of a language extending the nu-calculus of Pitts and Stark by using names for exceptions and general references. In particular, we examine abstract categorical models capturing nominal computation and construct a concrete fully abstract model in game semantics by using the recently introduced generalisation to nominal games.

1. Introduction

A prevalent feature of programming languages is the use of exceptions for raising and handling eccentric program behaviour, and more generally for manipulating the flow of control. It is a key feature, for example, of ML, Java and C++. The raising of an exception forces a program to escape out of its context and to the nearest exception-handler. In abstract terms, exceptions provide a means (an *effect*) of overriding nested behaviour of pure functional programs. In this paper we examine denotational semantics for exceptions and general references, with our focus being mainly on exceptions (the model of general references has been presented elsewhere [23], but the combination of the two effects has a semantical interest of its own).

In particular, we present the first full-abstraction result for a statically-scoped language with dynamically bound, locally declared (good) exceptions and general references, which faithfully reflects the practice — and reaches the expressivity — of real programming languages such as ML. The language extends the paradigmatic nominal language of Pitts and Stark [21] (ν-calculus) by treating exceptions and references as *names*. Names are constant identifiers with no inner structure which can be "created locally, tested for equality, and passed around via function application" ([21]). Moreover, exception-names can be raised and handled; and reference-names can be dereferenced and updated.

In order to represent names rigorously we found our presentation (of the language and its models) on *nominal sets* [8,20]. These constitute a robust foundational theory of constructions over collections of *atoms*, and are derived from the Fraenkel-Mostowski permutation models of set theory with atoms. Our constructions are built in nominal sets so that names be represented by atoms. Thus, the expressiveness of nominal sets, which goes far beyond our purposes here (see e.g. [7]), provides us with a firm handle on names.

A fully abstract model of exceptions (and ground-type references) has been previously constructed [12] by successfully translating in the semantical universe the override of nested behaviour: in abstract terms, the model allows for jumps in the precedence with which a program answers questions posed by the environment (i.e. it mildens the well-bracketing condition). That description of the exception effect is extremely accurate and intuitive. However, the modelling of exceptions themselves is based on an 'object-oriented' approach which encodes exceptions as products of raise/handle type, in a similar way that non-nominal models of references [4,2] see references as products of read/write type. In order to achieve full-abstraction "bad" constructors need to be included in the syntax, that is, the language examined includes non-exception terms of exception type (*bad exceptions*, and also *bad variables*). These constructs, while solving the full-abstraction problem, distance the language from the programming features it was set out to capture.

The nominal approach resolves this problem in an intuitive way: an exception is simply a name with no inner structure, and a language with exceptions is one equipped with constructs for manipulating (raising/handling) those names. Thus, instead of encapsulating the programming effect within the exception-type, we encapsulate it within *every type*. The same approach is followed for references.

Semantically, the above means that a proper model should include names as an effect and contain appropriate structure for representing other programming effects (exceptions/references) related to names. We represent effects by use of *monads* [17], which are a means of encapsulating an algebra of *computations* within a domain of semantic values. On the other hand, the notion of local state induced by names is described by a family of *comonads*. The monads and comonads of the model are then connected as follows: fresh-name creation is a (monadic) computation which alters the (comonadic) local state. A first contribution of this paper is the formulation of abstract categorical models for exceptions and general references following this approach.

Our main result is the formulation of a specific such model which is, moreover, fully abstract. This is achieved by use of *nominal game semantics* ([1,23,24], and also [13,15,14]), which constitutes a 'nominalised' version of the highly successful denotational paradigm of game semantics (see e.g. [3]). In particular, our nominal games are Honda-Yoshida call-by-value games [11] with local state [19], built inside the universe of nominal sets. This means that computation is modelled as an interaction (game) between two participants, one representing the program (*Player*) and the other the environment (*Opponent*), consisting of sequences of moves which may contain or introduce names. Moreover, each move is equipped with a local state, that is, a history of all names introduced so far in the interaction. These specifications allow for the capturing of the basic nominal effect, that is, the presence and passing around of names and their local, fresh creation. Moreover, the category of nominal games has sufficiently rich structure in order for exception and store monads [17] to be defined. This gives us an adequate model of our language; by restricting the domain of allowed semantic behaviours we are able to also obtain compact definability, and hence full abstraction.

In comparison to previous game models of exceptions and references [12,4,2], we notice that the use of monads allows us to express our computational effects inside a domain which is otherwise too restrictive (i.e. *too pure*). In particular, we are able to express fresh-name creation (a *non-total* effect), exceptions (a *non-well-bracketed* effect) and references (a *non-innocent* and *non-visible* effect) inside a domain of total, well-bracketed, visible, innocent games. In a sense, nominal games provide a fine-gained view of ordinary (non-nominal) games for effectful computation: for example, from a nominal game with exceptions we can obtain an ordinary, non-well-bracketed game by simply hiding the names appearing in the former.

2. Nominal sets

We briefly introduce nominal sets, which will be used at the basis of all our constructions with names. Let us fix a countably infinite family $(\mathbb{A}_i)_{i\in\omega}$ of pairwise disjoint, countably infinite sets, and let us denote by $\mathrm{PERM}(\mathbb{A}_i)$ the group of finite permutations of $\mathbb{A}_i$. The elements of the $\mathbb{A}_i$'s are called ***atoms*** and are denoted by a, b, c and variants. Permutations are denoted by π and variants; `id` is the identity permutation and $(a\ b)$ is the permutation swapping a and b (and fixing all other atoms). We write $\mathbb{A}$ for the union of all the $\mathbb{A}_i$'s. We take $\mathrm{PERM}(\mathbb{A})$ to be the direct sum of the groups $\mathrm{PERM}(\mathbb{A}_i)$, that is, elements of $\mathrm{PERM}(\mathbb{A})$ are those permutations of $\mathbb{A}$ that can be described as finite compositions,

$$\pi = \pi_1 \circ \cdots \circ \pi_n \,,$$

such that each π_i belongs to some $\mathrm{PERM}(\mathbb{A}_j)$. This means, in particular, that for all atoms a and all permutations π,

$$a \in \mathbb{A}_i \implies \pi(a) \in \mathbb{A}_i \,.$$

A ***nominal set*** X is a set $|X|$ (usually written X) equipped with an action from $\mathrm{PERM}(\mathbb{A})$, that is, a function $\_ \circ \_ : \mathrm{PERM}(\mathbb{A}) \times X \to X$ such that, for all $x \in X$ and $\pi, \pi' \in \mathrm{PERM}(\mathbb{A})$,

$$\pi \circ (\pi' \circ x) = (\pi \circ \pi') \circ x \quad \text{and} \quad \mathtt{id} \circ x = x\,.$$

Moreover, each $x \in X$ has ***finite support***, that is there exists a finite set $S \subseteq \mathbb{A}$ such that, for all permutations π,

$$(\forall a \in S.\, \pi(a) = a) \implies \pi \circ x = x\,. \tag{1}$$

Finite support is closed under intersection, and hence each element x of a nominal set ***has a (least) support*** $\mathtt{S}(x)$. This can be concretely expressed as:

$$\mathtt{S}(x) = \{a \in \mathbb{A} \mid \text{for infinitely many } b.\ (a\ b) \circ x \neq x\}\,. \tag{2}$$

We say that a ***is fresh for*** x, written $a \# x$, if $a \notin \mathtt{S}(x)$. x is ***equivariant*** if $\mathtt{S}(x) = \varnothing$.

Clearly, $\mathbb{A}$ is a nominal set by taking $\pi \circ a \triangleq \pi(a)$, for each π and a. More interestingly, the set $\mathbb{A}^{\#}$ of ***finite lists of distinct atoms*** is a nominal set (with permutations acting elementwise). Such lists we denote by $\bar{a}$, $\bar{b}$, $\bar{c}$, etc. If X and Y are nominal sets then so is their cartesian product $X \times Y$, with permutations acting componentwise, and their disjoint union $X + Y$. Moreover, $X' \subseteq X$ is a ***nominal subset*** of X if X' is closed under permutation actions, these acting as on X. A relation $\mathcal{R} \subseteq X \times Y$ is a ***nominal relation*** if it is a nominal subset of $X \times Y$. A ***nominal function*** is a function which is also a nominal relation. Concretely, a relation $\mathcal{R} \subseteq X \times Y$ (resp. a function $f : X \to Y$) is nominal if, for any π and any $(x, y) \in X \times Y$,

$$x \mathcal{R} y \iff (\pi \circ x) \mathcal{R} (\pi \circ y) \quad (\text{resp. } f(\pi \circ x) = \pi \circ f(x))\,. \tag{3}$$

The support of a list $\bar{a} \in \mathbb{A}^{\#}$ is *strong* in a very specific way: any permutation π for which $\pi \circ \bar{a} = \bar{a}$, satisfies $\pi \circ a = a$ for all $a \in \mathtt{S}(\bar{a})$. Accordingly, for any nominal set X, any $x \in X$ and any $S \subseteq \mathbb{A}$, we say that S ***strongly supports*** x if, for all π,

$$(\forall a \in S.\, \pi(a) = a) \iff \pi \circ x = x\,. \tag{4}$$

X is a ***strong nominal set*** if all its elements have strong support. The notion of strong support is stronger than that of support: for example, $\{a, b\} \subseteq \mathbb{A}_i$ does not have strong support. On the other hand, finite lists of atoms have strong support, so $\mathbb{A}^{\#}$ is a strong nominal set. Note that strong support coincides with weak support when the former exists.

Finally, in nominal sets we can *define* atom-abstractions. We will be using a simple such mechanism which abstracts all atoms from a nominal element by orbiting it under all permutations. That is, for a nominal set X and $x \in X$, we define an equivariant $[x]$ by:

$$[x] \triangleq \{y \in X \mid \exists \pi.\, y = \pi \circ x\}\,. \tag{5}$$

3. The $\nu\varepsilon\rho$-calculus

We introduce the $\nu\varepsilon\rho$-calculus, an idealised functional language with nominal exceptions and nominal general references. The calculus includes types for commands, numerals, products, functions, exceptions and references.

$$\mathrm{TY} \ni A, B ::= \ 1 \mid \mathbb{N} \mid A \times B \mid A \to B \mid \mathbb{E} \mid [A]\,. \tag{6}$$

Types of the last two classes are ***nameful***, that is they contain names. Names are denoted by atoms:
– we assume a set $\mathbb{A}_e \in (\mathbb{A}_i)_{i \in \omega}$ with elements denoted by $\dot{a}, \dot{b}, \ldots$,
– and, for each $A \in \mathrm{TY}$, a set $\mathbb{A}_A \in (\mathbb{A}_i)_{i \in \omega}$ with elements denoted by $\ddot{a}, \ddot{b}, \ldots$.
(In general, names are denoted by $a, b, \ldots$.)

We define terms and values as follows.

$$\begin{aligned} \mathrm{TE} \ni M, N ::= \ & x \mid \lambda x.M \mid M\,N \mid \langle M, N\rangle \mid \mathtt{fst}\,M \mid \mathtt{snd}\,N \mid n \mid \mathtt{pred}\,M \mid \mathtt{succ}\,N \\ & \mid \mathtt{if0}\ M\ \mathtt{then}\ N_1\ \mathtt{else}\ N_2 \mid \mathtt{skip} \mid a \mid \nu a.\,M \mid [M = N] \\ & \mid \mathtt{raise}\ M \mid \mathtt{try}\ N_1\ \mathtt{handle}\ M \Rightarrow N_2 \mid\ !M \mid M := N\,, \end{aligned} \tag{7}$$

$$\mathrm{VA} \ni V, W ::= n \mid \mathtt{skip} \mid x \mid \lambda x.M \mid \langle V, W\rangle \mid a\,.$$

We see that the TE and VA are strong nominal sets. A term's support is the set of names it contains, be they free or bound. ν is a name-binder (and λ is a variable-binder), and we follow the usual convention of equating terms up to α-equivalence, for name- and variable-binding. Note that the former is defined 'nominally' [8]:

$$\frac{}{M =_{\alpha_\nu} M}\ M = x, a, n, \texttt{skip} \qquad \frac{M =_{\alpha_\nu} M'}{\lambda x.M =_{\alpha_\nu} \lambda x.M'} \quad \cdots \quad \frac{\text{for cofinitely many } b.\ (a\ b) \circ M =_{\alpha_\nu} (a'\ b) \circ M'}{\nu a.\, M =_{\alpha_\nu} \nu a'.\, M'}$$

Typing in $\nu\varepsilon\rho$ involves environments $S \mid \Gamma$, where S is a finite subset of $\mathbb{A}_e \cup (\mathbb{A}_A)_{A \in \mathrm{TY}}$, and Γ contains variable-type pairs. Using A_ν for nameful types, the typing rules are as follows — plus the standard rules for λ-calculus with products, numerals and if-then-else.

$$\frac{}{S \mid \Gamma \vdash a : A_\nu}\ a \in S \qquad \frac{S, a \mid \Gamma \vdash M : B}{S \mid \Gamma \vdash \nu a.\, M : B} \qquad \frac{S \mid \Gamma \vdash M : A_\nu \quad S \mid \Gamma \vdash N : A_\nu}{S \mid \Gamma \vdash [M = N] : \mathbb{N}}$$

$$\frac{S \mid \Gamma \vdash M : \mathbb{E}}{S \mid \Gamma \vdash \texttt{raise}\ M : A} \qquad \frac{S \mid \Gamma \vdash M : \mathbb{E} \qquad S \mid \Gamma \vdash N_1, N_2 : A}{S \mid \Gamma \vdash \texttt{try}\ N_1\ \texttt{handle}\ M \Rightarrow N_2 : A}$$

$$\frac{S \mid \Gamma \vdash M : [A]}{S \mid \Gamma \vdash\ !M : A} \qquad \frac{S \mid \Gamma \vdash M : [A] \qquad S \mid \Gamma \vdash N : A}{S \mid \Gamma \vdash M := N : 1}$$

Note that, in contrast to the presentation in [23,24], here we are using sets for local state instead of lists. In the semantics, these sets S will have to be implicitly ordered, but this is harmless: in fact, such implicit orderings are regularly used for environments Γ.

The operational semantics is defined via a small-step reduction relation in environments P. These are nominal sets enlisting all names appearing in a computation and storing the values of those names that are references. Formally, P is a finite partial function from names to values, that is,

$$P ::= \varnothing \mid a, P \mid \ddot{a} :: V, P \tag{8}$$

where a and $\ddot{a}$ do not appear in $\mathtt{dom}(P)$. We stipulate that valid environments satisfy $\mathtt{dom}(P) = \mathtt{S}(P)$. Observe that a reference name $\ddot{a}$ may appear uninitialised inside an environment; in fact, this is what happens when a fresh reference is created. The reduction rules are as follows.

$$\text{UPD}\ \frac{}{P, \ddot{a}(:: W), P' \models \ddot{a} := V \longrightarrow P, \ddot{a} :: V, P' \models \texttt{skip}} \qquad \text{DRF}\ \frac{}{P, \ddot{a} :: V, P' \models\ !\ddot{a} \longrightarrow P, \ddot{a} :: V, P' \models V}$$

$$\text{VHL}\ \frac{}{P \models \texttt{try}\ V\ \texttt{handle}\ \dot{a} \Rightarrow N \longrightarrow P \models V} \qquad \text{XPN}\ \frac{}{P \models \mathrm{Z}[\texttt{raise}\ \dot{a}] \longrightarrow P \models \texttt{raise}\ \dot{a}}$$

$$\text{HL}\ \frac{}{P \models \texttt{try}\ (\texttt{raise}\ \dot{a})\ \texttt{handle}\ \dot{a} \Rightarrow N \longrightarrow P \models N} \qquad \text{CHK}\ \frac{}{P \models [a = b] \longrightarrow P \models n}\ {\scriptstyle n=1 \text{ if } a \neq b \atop n=0 \text{ if } a=b}$$

$$\text{NHL}\ \frac{}{P \models \texttt{try}\ (\texttt{raise}\ \dot{b})\ \texttt{handle}\ \dot{a} \Rightarrow N \longrightarrow P \models \texttt{raise}\ \dot{b}}\ \dot{a} \neq \dot{b} \qquad \text{NEW}\ \frac{}{P \models \nu a.\, M \longrightarrow P, a \models M}\ a \# P$$

$$\text{LAM}\ \frac{}{P \models (\lambda x.M)\, V \longrightarrow P \models M\{V/x\}} \qquad \text{SUC}\ \frac{}{P \models \texttt{succ}\, n \longrightarrow P \models n{+}1}$$

$$\text{PRD}\ \frac{}{P \models \texttt{pred}\,(n{+}1) \longrightarrow P \models n} \qquad \text{PRD}\ \frac{}{P \models \texttt{pred}\, 0 \longrightarrow P \models 0}$$

$$\text{FST}\ \frac{}{P \models \texttt{fst}\, \langle V, W \rangle \longrightarrow P \models V} \qquad \text{SND}\ \frac{}{P \models \texttt{snd}\, \langle V, W \rangle \longrightarrow P \models W}$$

$$\text{IF0}\ \frac{}{P \models \texttt{if0}\ n\ \texttt{then}\ N_0\ \texttt{else}\ N_1 \longrightarrow P \models N_j}\ {\scriptstyle j=0 \text{ if } n=0 \atop j=1 \text{ if } n>0} \qquad \text{CTX}\ \frac{P \models M \longrightarrow P' \models M'}{P \models \mathrm{E}[M] \longrightarrow P' \models \mathrm{E}[M']}$$

Unhandled evaluation contexts are of the forms:

$$\begin{aligned}\mathrm{Z} \ ::= \ & (\lambda x.N)\,\_ \mid \_\,N \mid \langle V,\_\rangle \mid \langle \_,N\rangle \mid \mathtt{fst}\,\_ \mid \mathtt{snd}\,\_ \mid \mathtt{pred}\,\_ \mid \mathtt{succ}\,\_\ ,\ \mathtt{if0}\ \_\ \mathtt{then}\ N\ \mathtt{else}\ N' \\ & \mid [\_ = N] \mid [a = \_] \mid\ !\,\_ \mid \_ := N \mid \ddot{a} := \_ \mid \mathtt{raise}\ \_ \mid \mathtt{try}\ N_1\ \mathtt{handle}\ \_\ \texttt{=>}\ N_2\end{aligned}$$

and *general evaluation contexts* are of the forms:

$$\mathrm{E} \ ::= \ \mathrm{Z} \mid \mathtt{try}\ \_\ \mathtt{handle}\ \dot{a}\ \texttt{=>}\ N\,.$$

Apart from evaluation contexts, we also have single-holed, variable-capturing contexts C defined as usually. For any $S \mid \Gamma \models M, N : A$, we say that M ***observationally approximates*** N, written $S \mid \Gamma \models M \lessapprox N$, if, for any variable- and name-closing context $\mathrm{C} : 1$,

$$\exists P'.(\models \mathrm{C}[M] \twoheadrightarrow P' \models \mathtt{skip}) \implies \exists P''.(\models \mathrm{C}[N] \twoheadrightarrow P'' \models \mathtt{skip})\,. \tag{9}$$

We usually write simply $M \lessapprox N$. Moreover, we set $\approxeq \triangleq \lessapprox \cap \gtrapprox$.

Sub-calculi and expressiveness

Let us briefly compare the expressiveness of $\nu\varepsilon\rho$ with that of the following three sub-calculi.

(i) The ν-calculus [21] is the restriction of $\nu\varepsilon\rho$ with no raising, handling, updating or dereferencing constructs, and a single nameful type. [1]

(ii) The $\nu\rho$-calculus [23] is the restriction containing general references but no exceptions.

(iii) The $\nu\varepsilon$-calculus is taken to be the restriction containing exceptions but no references, so the only nameful type is $\mathbb{E}$.

The syntax, static and operational semantics of these languages are defined by selecting the relevant clauses from $\nu\varepsilon\rho$'s specifications.

These languages are separated observationally by the following terms. First, for any type A, we define the terms:

$$\begin{aligned}\mathtt{stop}_A &\triangleq \nu b.\,(b := \lambda x.(!\,b)\mathtt{skip})\,;(!\,b)\mathtt{skip} : A\,,\\ [M \Leftrightarrow N] &\triangleq \mathtt{if0}\ M\ \mathtt{then}\ N\ \mathtt{else}\ (\mathtt{if0}\ N\ \mathtt{then}\ 1\ \mathtt{else}\ 0) : \mathbb{N}\,,\end{aligned} \tag{10}$$

where composition $M\,;N$ is given by $(\lambda x.N)M$, some x not in N. $\mathtt{stop}$ is the divergent term (Ω), while $[M \Leftrightarrow N]$ compares M and N as booleans. Take A_ν to be some nameful type of minimal size, according to the calculus at hand. Define:

$$\begin{aligned}M_1 &\triangleq \lambda f.\,0 : (A_\nu \to \mathbb{N}) \to \mathbb{N}\,,\\ M_2 &\triangleq \nu a.\,\nu b.\,\lambda f.\,[fa \Leftrightarrow fb] : (A_\nu \to \mathbb{N}) \to \mathbb{N}\,,\\ M_3 &\triangleq \nu a.\,\lambda f.\,[fa \Leftrightarrow fa] : (A_\nu \to \mathbb{N}) \to \mathbb{N}\,,\\ M_4 &\triangleq \lambda f.\,\mathtt{stop}_1 : (1 \to 1) \to 1\,,\\ M_5 &\triangleq \lambda f.\,f\,\mathtt{skip}\,;\mathtt{stop}_1 : (1 \to 1) \to 1\,.\end{aligned} \tag{11}$$

Note that M_4 and M_5 are meaningful only in the presence of references (i.e. in $\nu\rho$, $\nu\varepsilon\rho$), because of $\mathtt{stop}$. Our nominal calculi exhibit the following behaviour.

	$M_1 \approxeq M_2$	$M_2 \approxeq M_3$	$M_4 \approxeq M_5$
ν	✓ (1.1)	✓ (1.2)	–
$\nu\rho$	✗ (2.1)	✗ (2.2)	✓ (2.3)
$\nu\varepsilon$	✗ (3.1)	✓ (3.2)	–
$\nu\varepsilon\rho$	✗ (4.1)	✗ (4.2)	✗ (4.3)

Table 1: Equivalences separating our nominal calculi

(1.1) is shown in [22], and (1.2) follows from (3.2). The latter is shown semantically in the last section. (2.3) can also be shown semantically, see [24]. Inequivalences are left as exercise.

[1] In fact, the ν-calculus of [21] contains booleans and not numerals.

4. $\nu\varepsilon\rho$-models

We now formulate conditions for a correct categorical semantics of $\nu\varepsilon\rho$. Assuming an underlying category with finite products, the semantics we formulate is ***monadic*** over a computational monad T (v. [17]) and ***comonadic*** over a family of local-state comonads $Q = (Q^{\bar{a}})_{\bar{a}\in\mathbb{A}^{\#}}$ (v. [6]). Thus, the morphism related to each $S \mid \varGamma \vdash M : A$ is of the form

$$[\![M]\!] : Q^{\bar{a}}[\![\varGamma]\!] \to T[\![A]\!]$$

where $\bar{a}$ is an ordering of S, i.e. $\mathtt{S}(\bar{a}) = S$.

Recall that a ***strong monad*** (T, η, μ, τ) on a category $\mathcal{C}$ with finite products comprises of a functor $T : \mathcal{C} \to \mathcal{C}$ and natural transformations

$$\eta : \mathtt{Id} \to T\,, \quad \mu : T^2 \to T\,, \quad \tau : (\_ \times T\_) \to T(\_ \times \_)$$

such that the following diagrams commute.

$$\begin{tikzcd}
T^3A \arrow[r, "\mu_{TA}"] \arrow[d, "T\mu_A"'] & T^2A \arrow[d, "\mu_A"] \\
T^2A \arrow[r, "\mu_A"'] & TA
\end{tikzcd}
\qquad
\begin{tikzcd}
TA \arrow[r, "\eta_{TA}"] \arrow[d, "T\eta_A"'] \arrow[dr, "\mathtt{id}_{TA}"] & T^2A \arrow[d, "\mu_A"] \\
T^2A \arrow[r, "\mu_A"'] & TA
\end{tikzcd}
\qquad
\begin{tikzcd}
A \times T^2B \arrow[r, "\mathtt{id}_A \times \mu_B"] \arrow[d, "\tau_{A,TB}"'] & A \times TB \arrow[dr, "\tau_{A,B}"] & A \times B \arrow[l, "\mathtt{id}_A \times \eta_B"'] \arrow[d, "\eta_{A\times B}"] \\
T(A \times TB) \arrow[r, "T\tau_{A,B}"'] & T^2(A \times B) \arrow[r, "\mu_{A\times B}"'] & T(A \times B)
\end{tikzcd}$$

$$\begin{tikzcd}
1 \times TA \arrow[r, "\tau_{1,A}"] \arrow[dr, "\cong"'] & T(1 \times A) \arrow[d, "T\cong"] \\
& TA
\end{tikzcd}
\qquad
\begin{tikzcd}
(A \times B) \times TC \arrow[r, "\tau_{A\times B,C}"] \arrow[d, "\cong"'] & T((A \times B) \times C) \arrow[d, "T\cong"] \\
A \times (B \times TC) \arrow[r, "\mathtt{id}_A \times \tau_{B,C}\,;\,\tau_{A,B\times C}"'] & T(A \times (B \times C))
\end{tikzcd}$$

$\mathcal{C}$ has ***T-exponentials*** if, moreover, for each pair of objects B, C, there is an object TC^B and an arrow $\mathsf{ev}^T : TC^B \times B \to TC$ such that for each arrow $f : A \times B \to TC$ there exists a unique $\Lambda^T(f) : A \to TC^B$ satisfying:

$$f = \Lambda^T(f) \times \mathtt{id}_B\,;\mathsf{ev}^T\,.$$

Let us write $\tau' : T\_ \times \_ \to T(\_ \times \_)$ for the transformation derived from τ and product symmetries, and take

$$\begin{aligned}
\psi_{A,B} &\triangleq TA \times TB \xrightarrow{\tau'} T(A \times TB) \xrightarrow{T\tau} T^2(A \times B) \xrightarrow{\mu} TB\,, \\
\psi'_{A,B} &\triangleq TA \times TB \xrightarrow{\tau} T(TA \times B) \xrightarrow{T\tau'} T^2(A \times B) \xrightarrow{\mu} TB\,.
\end{aligned} \tag{12}$$

In general, $\psi \neq \psi'$ represents the non-commutativity of consecutive effects.

A monad may encapsulate several effects consecutively. One way of formalising this is by stipulating that the monad be *compound*, i.e. of the form $T = T_1 \circ T_2$ (plus a distributivity law [5]). Such a description presupposes knowledge of the constituent sub-monads. However, in our case it suffices to know that the consecutive effects inside T are separable in the following sense.

Definition 1 Let T be a strong monad on a category $\mathcal{C}$. We say that T is ***precompound*** if there exists a category $\mathcal{C}'$ such that:
(i) $\mathcal{C}$ is a lluf subcategory of $\mathcal{C}'$ and T extends to a strong monad on $\mathcal{C}'$;
(ii) there is a natural transformation $\theta : T \to T^2$ in $\mathcal{C}'$ such that the following diagrams commute.

$$\begin{tikzcd}
TA \arrow[r, "\theta_A"] \arrow[dr, "\mathtt{id}"'] & T^2A \arrow[d, "\mu_A"] \\
& TA
\end{tikzcd}
\qquad
\begin{tikzcd}
T^3A \arrow[d, "\mu_{TA}\,;\,\theta_{TA}"'] & T^2A \arrow[l, "T\theta_A"'] \arrow[r, "\theta_{TA}"] \arrow[d, "\mu_A\,;\,\theta_A"] & T^3A \arrow[d, "T(\mu_A\,;\,\theta_A)"] \\
T^3A \arrow[r, "T\mu_A"'] & T^2A & T^3A \arrow[l, "\mu_{TA}"]
\end{tikzcd}
\qquad
\begin{tikzcd}
A \times TB \arrow[r, "\tau_{A,B}"] \arrow[d, "\mathtt{id}\times\theta_B"'] & T(A \times B) \arrow[d, "\theta_{A\times B}"] \\
A \times T^2B \arrow[r, "\tau_{A,TB}\,;\,T\tau_{A,B}"'] & T^2(A \times B)
\end{tikzcd}$$

Moreover, each η_A is an inner- and outer-component arrow, where an arrow $f : A \to TB$ is said to be
– an *inner-component arrow* if $f\,;\theta_B = f\,;\eta_{TB}$,
– an *outer-component arrow* if $f\,;\theta_B = f\,;T\eta_B$.

We write T as $(T_{\mathcal{C}'}, \theta)$. ▲

Thus, θ separates the two components of T in the following sense. Each $\theta_A : TA \to T^2A$ sends the outer T-component of TA to the outer T of T^2A, and similarly for the inner one. In general, though, the two components of T may not be separable within $\mathcal{C}$ (i.e. the separating arrows θ_A may not live in $\mathcal{C}$) but in a supcategory $\mathcal{C}'$. Note that a compound monad is easily shown to be precompound, by taking:

$$\theta \triangleq T_1T_2 \xrightarrow{T_1\eta_2} T_1T_2T_2 \xrightarrow{T_1T_2\eta_1} T_1T_2T_1T_2 \,. \tag{13}$$

A ***comonad*** (Q, ε, δ) in $\mathcal{C}$ is a monad in $\mathcal{C}^{\mathrm{op}}$, that is

$$Q : \mathcal{C} \to \mathcal{C}\,, \quad \varepsilon : Q \to \mathtt{Id}\,, \quad \delta : Q \to Q^2,$$

and the first two monadic diagrams are satisfied (when reversed). We say that Q is a ***product comonad*** if the canonical natural transformation

$$\tilde{\zeta} \triangleq \langle Q\pi_1, Q\pi_2; \varepsilon_B \rangle : Q(A \times B) \to QA \times B \tag{14}$$

has an inverse ζ. We write Q as $(Q, \varepsilon, \delta, \zeta)$, and denote the symmetric counterparts of $\zeta, \tilde{\zeta}$ by $\zeta', \tilde{\zeta}'$. Note that if Q is a product comonad then it can be expressed as $Q \cong Q1 \times \_$.

In the nominal setting, comonads will be used for the modelling of (constant) local state. This will be accomplished by the following construction.

Definition 2 Let $\mathcal{C}$ be a category with finite products and a booleans-object $1{+}1$. A ***comonadic nominal setting on*** $\mathcal{C}$ is given by a family of product comonads $(Q^{\bar{a}}, \varepsilon, \delta, \zeta)_{\bar{a}\in\mathbb{A}^{\#}}$ on $\mathcal{C}$ such that:[2]

(i) $Q^{\epsilon} \cong \mathtt{Id}_{\mathcal{C}}$ and $Q^{\bar{a}} = Q^{\bar{a}'}$ whenever $[\bar{a}] = [\bar{a}']$. For each $\bar{a} \in \mathbb{A}^{\#}$, we set

$$\mathrm{A}^{\bar{a}} \triangleq Q^{\bar{a}}1$$

(and therefore $Q^{\bar{a}} \cong \mathrm{A}^{\bar{a}} \times \_$) and write A_i for A^a with $a \in \mathbb{A}_i$. These represent *names-objects* within $\mathcal{C}$.

(ii) If $\mathtt{S}(\bar{a}') \subseteq \mathtt{S}(\bar{a})$ then there exists a comonad morphism $\frac{\bar{a}}{\bar{a}'} : Q^{\bar{a}} \to Q^{\bar{a}'}$ such that $\frac{\bar{a}}{\epsilon} = \varepsilon$ and $\frac{\bar{a}}{\bar{a}} = \mathtt{id}$. Moreover, whenever $\mathtt{S}(\bar{a}') \subseteq \mathtt{S}(\bar{a}'') \subseteq \mathtt{S}(\bar{a})$,

$$\frac{\bar{a}}{\bar{a}''} ; \frac{\bar{a}''}{\bar{a}'} = \frac{\bar{a}}{\bar{a}'} \,. \tag{CR}$$

(iii) For each $i \in \omega$ there is a name-equality arrow $\mathtt{eq}_i : \mathrm{A}_i \times \mathrm{A}_i \to 1{+}1$ such that, for any distinct $a, b \in \mathbb{A}_i$, the following diagram commutes.

$$\begin{tikzcd}
Q^a1 \arrow[r, "\Delta"] \arrow[d, "!"'] & \mathrm{A}_i \times \mathrm{A}_i \arrow[d, "\mathtt{eq}_i"] & Q^{ab}1 \arrow[l, "\langle \frac{ab}{a}, \frac{ab}{b} \rangle"'] \arrow[d, "!"] \\
1 \arrow[r, "\mathtt{in}_1"] & 1+1 & 1 \arrow[l, "\mathtt{in}_2"']
\end{tikzcd} \tag{N1}$$

▲

The above specifications describe local state by means of comonads, and change of local state by means of monad transformations ($\frac{\bar{a}}{\bar{a}'}$). The latter, however, is restricted to the case where no fresh names are involved ($\mathtt{S}(\bar{a}') \subseteq \mathtt{S}(\bar{a})$); for fresh-name creation we also need a monad.

Definition 3 A ***monadic-comonadic nominal setting*** on a category $\mathcal{C}$ comprises of a strong monad (T, η, μ, τ) with T-exponentials and of a family of product comonads $Q = (Q^{\bar{a}}, \varepsilon, \delta, \zeta)_{\bar{a}\in\mathbb{A}^{\#}}$ on $\mathcal{C}$ such that:

(i) Q is a comonadic nominal setting on $\mathcal{C}$,

[2] In fact, $(Q^{\bar{a}}, \varepsilon, \delta, \zeta)$ stands for the more cumbersome $(Q^{\bar{a}}, \varepsilon^{\bar{a}}, \delta^{\bar{a}}, \zeta^{\bar{a}})$.

(ii) for each $\bar{a}a \in \mathbb{A}^{\#}$ there exists a natural transformation $\mathtt{nu}^{\bar{a}a} : Q^{\bar{a}} \to TQ^{\bar{a}a}$ such that, for any $\bar{a}'a$ with $\mathtt{S}(\bar{a}a) \subseteq \mathtt{S}(\bar{a}'a)$, the following diagrams commute.

$$\begin{tikzcd}
A \times Q^{\bar{a}}B \arrow[r, "\zeta"] \arrow[d, "\mathtt{id}\times\mathtt{nu}^{\bar{a}a}_{B}"'] & Q^{\bar{a}}(A \times B) \arrow[d, "\mathtt{nu}^{\bar{a}a}_{A\times B}"] \\
A \times TQ^{\bar{a}a}B \arrow[r, "\tau\,;\,T\zeta"'] & TQ^{\bar{a}a}(A \times B)
\end{tikzcd}
\quad
\begin{tikzcd}
Q^{\bar{a}}A \arrow[r, "\langle \mathtt{id},\mathtt{nu}^{\bar{a}a}\rangle"] \arrow[d, "\mathtt{nu}^{\bar{a}a}"'] & Q^{\bar{a}}A \times TQ^{\bar{a}a}A \arrow[d, "\tau"] \\
TQ^{\bar{a}a}A \arrow[r, "T\langle \frac{\bar{a}a}{\bar{a}},\mathtt{id}\rangle"'] & T(Q^{\bar{a}}A \times Q^{\bar{a}a}A)
\end{tikzcd}
\quad
\begin{tikzcd}
Q^{\bar{a}'}A \arrow[r, "\mathtt{nu}^{\bar{a}'a}"] \arrow[d, "\frac{\bar{a}'}{\bar{a}}"'] & TQ^{\bar{a}'a}A \arrow[d, "T\frac{\bar{a}'a}{\bar{a}a}"] \\
Q^{\bar{a}}A \arrow[r, "\mathtt{nu}^{\bar{a}a}"'] & TQ^{\bar{a}a}A
\end{tikzcd}
\tag{N2}$$

▲

This completes the specifications of a *basic nominal model*, which is an abstract categorical model of the ν-calculus. From that, we obtain a model of the $\nu\varepsilon\rho$-calculus as follows. Note that we write A_e for $\mathrm{A}^{\dot{a}}$ (any $\dot{a} \in \mathbb{A}_e$), and A_A for $\mathrm{A}^{\ddot{a}}$ with $\ddot{a} \in \mathbb{A}_A$.

Definition 4 A ***νερ-model*** $\mathcal{M}$ is a monadic-comonadic nominal setting $(\mathcal{M}, T, Q)$ satisfying the following conditions.

I. $\mathcal{M}$ contains an object N along with arrows $\tilde{n} : 1 \to \mathrm{N}$, each $n \in \mathbb{N}$, and successor/predeccessor arrows. Moreover, there is an appropriate natural transformation with components $\mathtt{cnd}_A : \mathrm{N} \times TA \times TA \to TA$ for zero-equality conditionals.

II. $\mathcal{M}$ contains a natural transformation $\mathtt{inx} : K_{\mathrm{A}_e} \to T$ for exception-inclusion, where K_{A_e} is the constant-A_e functor, such that the following diagrams commute.

$$\begin{tikzcd}
A \times \mathrm{A}_e \arrow[r, "\mathtt{id}\times\mathtt{inx}_B"] \arrow[d, "\pi_2"'] & A \times TB \arrow[d, "\tau"] \\
\mathrm{A}_e \arrow[r, "\mathtt{inx}_{A\times B}"'] & T(A \times B)
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathrm{A}_e \arrow[r, "\mathtt{inx}_{TB}"] \arrow[dr, "\mathtt{inx}_B"'] & T^2B \arrow[d, "\mu"] \\
 & TB
\end{tikzcd}
\tag{NE1}$$

Moreover, for each object A, an arrow $\mathtt{hdl}_A : \mathrm{A}_e \times TA \times TA \to TA$ for exception-handling such that the following diagram commutes.

$$\begin{tikzcd}
Q^{\dot{a}\dot{b}}1 \times TA \arrow[r, "\langle \frac{\dot{a}\dot{b}}{\dot{a}},\, \frac{\dot{a}\dot{b}}{\dot{b}};\mathtt{inx}_A\rangle\times\mathtt{id}"] \arrow[d, "\pi_1;\frac{\dot{a}\dot{b}}{\dot{a}}"'] & \mathrm{A}_e \times TA \times TA \arrow[d, "\mathtt{hdl}_A"] & \mathrm{A}_e \times TA \arrow[l, "\langle \mathtt{id},\mathtt{inx}_A\rangle\times\mathtt{id}"'] \arrow[dl, "\pi_2" near start] \\
\mathrm{A}_e \arrow[r, "\mathtt{inx}_A"'] & TA & \mathrm{A}_e \times A \times TA \arrow[l, "\pi_{12}\,;\,\eta"] \arrow[ul, crossing over, "\mathtt{id}\times\eta\times\mathtt{id}"' near start]
\end{tikzcd}
\tag{NE2}$$

III. Setting

$$[\![1]\!] \triangleq 1\,,\ [\![\mathbb{N}]\!] \triangleq \mathrm{N}\,,\ [\![[A]]\!] \triangleq \mathrm{A}_A\,,\ [\![\mathbb{E}]\!] \triangleq \mathrm{A}_e\,,\ [\![A \to B]\!] \triangleq T[\![B]\!]^{[\![A]\!]}\,,\ [\![A \times B]\!] \triangleq [\![A]\!] \times [\![B]\!]\,,$$

$\mathcal{M}$ contains, for each $A \in \mathrm{TY}$, arrows $\mathtt{drf}_A : \mathrm{A}_A \to T[\![A]\!]$ and $\mathtt{upd}_A : \mathrm{A}_A \times [\![A]\!] \to T1$ such that the following diagrams commute,

$$\begin{tikzcd}
\mathrm{A}_A \times [\![A]\!] \arrow[r, "\langle \mathtt{id},\mathtt{upd}_A\rangle\,;\,\tau\,;\,\cong"] & T(\mathrm{A}_A \times [\![A]\!]) \arrow[r, bend left, "T\pi_1\,;\,T\mathtt{drf}\,;\,\mu"] \arrow[r, bend right, "T\pi_2"'] & T[\![A]\!]
\end{tikzcd}$$

$$\begin{tikzcd}
\mathrm{A}_A \times [\![A]\!] \times [\![A]\!] \arrow[r, "\langle \mathtt{id}\times\pi_1;\mathtt{upd}_A,\,\mathtt{id}\times\pi_2;\mathtt{upd}_A\rangle"] & T1 \times T1 \arrow[r, bend left, "\psi\,;\,\cong"] \arrow[r, bend right, "\pi_2"'] & T1
\end{tikzcd}
\tag{NR}$$

$$\begin{tikzcd}
Q^{\ddot{a}\ddot{b}}1 \times [\![A]\!] \times [\![B]\!] \arrow[r, "\langle \frac{\ddot{a}\ddot{b}}{\ddot{a}}\times\pi_1;\mathtt{upd}_A,\,\frac{\ddot{a}\ddot{b}}{\ddot{b}}\times\pi_2;\mathtt{upd}_B\rangle"] & T1 \times T1 \arrow[r, bend left, "\psi\,;\,\cong"] \arrow[r, bend right, "\psi'\,;\,\cong"'] & T1
\end{tikzcd}$$

and, moreover, updates and fresh-name creation are independent effects, that is:

$$(\mathtt{nu}^{\bar{a}a}_A \times \mathtt{upd}_B)\,;\,\psi = (\mathtt{nu}^{\bar{a}a}_A \times \mathtt{upd}_B)\,;\,\psi'\,. \tag{SNR}$$

IV. T is precompound, $(T_{\mathcal{M}'}, \theta)$, with `nu`, `upd` being in the outer component and `inx` in the inner one. Moreover, for each object A:

$$\mathtt{hdl}_A = \mathrm{A}_e{\times}TA{\times}TA \xrightarrow{\mathtt{id}\times\theta_A\times\mathtt{id}} \mathrm{A}_e{\times}TA^2{\times}TA \xrightarrow{\tau\times\mathtt{id}\,;\,\tau'} T(\mathrm{A}_e{\times}TA{\times}TA) \xrightarrow{T\mathtt{hdl}_A} T^2A \xrightarrow{\mu} TA \quad \text{(NE3)}$$

The translation of the $\nu\varepsilon\rho$-calculus in such a model is given in figure 1. ▲

Regarding the diagrams used in the definition, (NE1) attaches coproduct inclusion properties on `inx` while (NE2) is a plain categorical translation of the reduction rules NHL, HL, VHL. On the other hand, (NR) represents the following $\nu\varepsilon\rho$-equivalences, for $\ddot{a} \neq \ddot{b}$.

$$\begin{aligned} \ddot{a} := V\,;\,!\,\ddot{a} \;&\cong\; \ddot{a} := V\,;V \\ \ddot{a} := V\,;\ddot{a} := W \;&\cong\; \ddot{a} := W \\ \ddot{a} := V\,;\ddot{b} := W \;&\cong\; \ddot{b} := W\,;\ddot{a} := V \end{aligned} \tag{15}$$

We now explain the role of precompoundness. Although θ does not appear explicitly in the semantic translation, it does so implicitly: because of (NE), the translation $\llbracket \mathtt{try}\ N_1\ \mathtt{handle}\ M \Rightarrow N_2 \rrbracket$ is, in fact,

$$Q^{\bar{a}}\Gamma \xrightarrow{\langle \llbracket M \rrbracket,\, \llbracket N_1 \rrbracket\,;\,\theta,\, \llbracket N_2 \rrbracket \rangle} T\mathrm{A}_e \times T^2A \times TA \xrightarrow{\psi\times\mathtt{id}\,;\,\tau'} T(\mathrm{A}_e \times TA \times TA) \xrightarrow{\mathtt{hdl}\,;\,\mu} TA\,. \tag{16}$$

The purpose of θ above is to separate the two components of TA yielded by $\llbracket N_1 \rrbracket$, so that the inner component be passed on to the exception-handler and the outer component to the output of the computation. Thus, fresh-names and name-updates of N_1 are not lost, and `try _ handle` $M \Rightarrow N_2$ behaves like a proper (handled) evaluation context. Finally, note the utility of allowing θ not to be part of our model $\mathcal{M}$ (but of

$$\llbracket n \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{Q^{\bar{a}}!} Q^{\bar{a}}1 \xrightarrow{\frac{\bar{a}}{\epsilon}} 1 \xrightarrow{\tilde{n}} \mathrm{N} \xrightarrow{\eta} T\mathrm{N}$$

$$\llbracket x \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{Q^{\bar{a}}\pi} Q^{\bar{a}}A \xrightarrow{\frac{\bar{a}}{\epsilon}} A \xrightarrow{\eta} TA$$

$$\llbracket a \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{Q^{\bar{a}}!} Q^{\bar{a}}1 \xrightarrow{\frac{\bar{a}}{a}} \mathrm{A}_x \xrightarrow{\eta} T\mathrm{A}_x$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}(\Gamma \times A) \to TB}{\llbracket \lambda x.M \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\Lambda^T(\zeta\,;\,\llbracket M \rrbracket)} TB^A \xrightarrow{\eta} T(TB^A)}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T(TB^A) \qquad \llbracket N \rrbracket : Q^{\bar{a}}\Gamma \to TA}{\llbracket M\,N \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\langle \llbracket M \rrbracket, \llbracket N \rrbracket \rangle\,;\,\psi} T(TB^A \times A) \xrightarrow{T\mathtt{ev}^T;\,\mu} TB}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{A}_A \qquad \llbracket N \rrbracket : Q^{\bar{a}}\Gamma \to TA}{\llbracket M := N \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\langle \llbracket M \rrbracket, \llbracket N \rrbracket \rangle\,;\,\psi} T(\mathrm{A}_A \times A) \xrightarrow{T\mathtt{upd}_A;\,\mu} T1}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{A}_A}{\llbracket !\,M \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\llbracket M \rrbracket} T\mathrm{A}_A \xrightarrow{T\mathtt{drf}_A\,;\,\mu} TA}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{N} \qquad \llbracket N_i \rrbracket : Q^{\bar{a}}\Gamma \to TA}{\begin{array}{l}\llbracket \mathtt{if0}\ M\ \mathtt{then}\ N_1\ \mathtt{else}\ N_2 \rrbracket : \\ Q^{\bar{a}}\Gamma \xrightarrow{\langle \llbracket M \rrbracket, \llbracket N_1 \rrbracket, \llbracket N_2 \rrbracket \rangle\,;\,\tau'} T(\mathrm{N} \times TA \times TA) \xrightarrow{T\mathtt{cnd}_A\,;\,\mu} TA\end{array}}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}a}\Gamma \to TA}{\llbracket \nu a.\,M \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\mathtt{nu}} TQ^{\bar{a}a}\Gamma \xrightarrow{T\llbracket M \rrbracket\,;\,\mu} TA}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{A}_x \qquad \llbracket N \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{A}_x}{\llbracket [M = N] \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\langle \llbracket M \rrbracket, \llbracket N \rrbracket \rangle\,;\,\psi} T(\mathrm{A}_x \times \mathrm{A}_x) \xrightarrow{T(\mathtt{eq}\,;[\tilde{0},\tilde{1}])} T\mathrm{N}}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{A}_e}{\llbracket \mathtt{raise}\ M \rrbracket : Q^{\bar{a}}\Gamma \xrightarrow{\llbracket M \rrbracket} T\mathrm{A}_e \xrightarrow{T\mathtt{inx}_A\,;\,\mu} TA}$$

$$\frac{\llbracket M \rrbracket : Q^{\bar{a}}\Gamma \to T\mathrm{A}_e \qquad \llbracket N_i \rrbracket : Q^{\bar{a}}\Gamma \to TA}{\begin{array}{l}\llbracket \mathtt{try}\ N_1\ \mathtt{handle}\ M \Rightarrow N_2 \rrbracket : \\ Q^{\bar{a}}\Gamma \xrightarrow{\langle \llbracket M \rrbracket, \llbracket N_1 \rrbracket, \llbracket N_2 \rrbracket \rangle\,;\,\tau'} T(\mathrm{A}_e \times TA \times TA) \xrightarrow{T\mathtt{hdl}_A\,;\,\mu} TA\end{array}}$$

Figure 1. The translation of $\nu\varepsilon\rho$ inside a $\nu\varepsilon\rho$-model.

the supcategory $\mathcal{M}'$): in the $\nu\varepsilon\rho$-calculus, when a function is called it is not possible to separate its outer from its inner effects — and e.g. discard the inner ones — so θ is not definable. [3]

We now proceed to demonstrate that the above construction yields indeed a model of $\nu\varepsilon\rho$. Let us define for each environment P a term $\widehat{P}$ by:

$$\widehat{\epsilon} \triangleq \texttt{skip}\,, \quad \widehat{\ddot{a} :: V, P} \triangleq \ddot{a} := V\,; \widehat{P}\,, \quad \widehat{a, P} \triangleq \widehat{P}\,. \tag{17}$$

Moreover, let us use labelled arrows, $\xrightarrow{\mathrm{r}}$, to denote the last non-CTX rule used to derive a reduction. We can show the following.

Proposition 5 *For any typed term $S \mid \Gamma \vdash M : A$, any environment P and any reduction rule* r*,*

1. *if* $\mathrm{r} \notin \{\text{NEW, UPD, DRF}\}$ *then* $P \models M \xrightarrow{\mathrm{r}} P \models M' \implies [\![M]\!] = [\![M']\!]\,,$

2. *if* $\mathrm{r} \in \{\text{UPD, DRF}\}$ *then* $P \models M \xrightarrow{\mathrm{r}} P' \models M' \implies [\![\widehat{P}\,; M]\!] = [\![\widehat{P'}\,; M']\!]\,,$

3. $P \models M \xrightarrow{\text{NEW}} P, a \models M' \implies [\![\widehat{P}\,; M]\!] = [\![\nu a.\ \widehat{P}\,; M']\!]\,.$

Therefore, $P \models M \twoheadrightarrow P' \models M'$ *implies* $[\![\widehat{P}\,; M]\!] = [\![\nu \bar{a}.\,(\widehat{P'}\,; M')]\!]$ *, with* $\mathtt{dom}(P') \setminus \mathtt{dom}(P) = \mathtt{S}(\bar{a})$*.* ■

Proof: (*sketch*) The last clause follows from 1-3. For those, we do induction on the reduction's derivation. The base cases follow relatively easily from the specifications. Note that (SNR) is needed in the case of NEW; (NE1) is used for XPN; and condition IV of the previous definition is used for HL, NHL.
The inductive step of 1 follows from compositionality of the semantics. For 2, using standard semantical methods and employing again condition IV along with (NE3), we can show that for any term M, environment P and evaluation context E,

$$[\![\mathrm{E}[\widehat{P}; M]]\!] = [\![\widehat{P}; \mathrm{E}[M]]\!]\,. \tag{18}$$

With the aid of conditions (N2), IV and (NE3) we can extend the above to:

$$[\![\mathrm{E}[\nu a.\ \widehat{P}; M]]\!] = [\![\nu a.\ \widehat{P}; \mathrm{E}[M]]\!]\,, \tag{19}$$

for any name a. This solves 3. ■

This is, in fact, the furthest we can go with $\nu\varepsilon\rho$-models — correctness. For soundness we need to stipulate that our models satisfy *computational adequacy.*

Definition 6 Let $\mathcal{M}$ be a $\nu\varepsilon\rho$-model and $[\![\_]\!]$ the respective translation of $\nu\varepsilon\rho$. $\mathcal{M}$ is ***adequate*** if, for any closed (wrt. variables and names) term $M : 1$, if $[\![M]\!] = [\![\nu\bar{a}.\,\widehat{P}]\!]$ for some $P, \bar{a}$ then there exists P' such that $\models M \twoheadrightarrow P' \models \texttt{skip}$. ▲

Proposition 7 (Soundness) *Translating $\nu\varepsilon\rho$ into an adequate $\nu\varepsilon\rho$-model $\mathcal{M}$ we obtain:*

$$[\![M]\!] = [\![N]\!] \implies M \lessapprox N\,.$$

■

5. The nominal games model

We proceed to build a fully abstract model of $\nu\varepsilon\rho$ in nominal games. The basic construction is the category $\mathcal{V}_{\mathtt{t}}$, which provides also the backbone for the fully abstract models of the ν-calculus [1] and the $\nu\rho$-calculus [23], and in effect of any nominal calculus with computational effects definable in $\mathcal{V}_{\mathtt{t}}$ with monads. The following definition gives the objects of $\mathcal{V}_{\mathtt{t}}$.

Definition 8 A ***nominal arena*** $A \triangleq (M_A, I_A, \vdash_A, \lambda_A)$ is given by:

[3] An opposite approach is followed in [24]. There, θ is taken as part of the $\nu\varepsilon\rho$-model and definability is proven for *$\nu\varepsilon\rho$-submodels*, that is, appropriate lluf subcategories where problematic arrows like θ are excluded.

- a strong nominal set M_A of moves,
- a nominal subset $I_A \subseteq M_A$ of initial moves,
- a nominal justification relation $\vdash_A \subseteq M_A \times (M_A \setminus I_A)$,
- a nominal labelling function $\lambda_A : M_A \to \{O, P\} \times \{A, Q\}$.

λ_A labels moves as *Opponent* or *Player* moves and as *Questions* or *Answers*. Initial moves must be P-Answers, Answers may only justify Questions, and if $m_1 \vdash_A m_2$ then λ_A assigns them complementary OP-labels. Moreover, for each $m \in M_A$ there exists unique $k \geq 0$ such that, for some m_l's in M_A,

$$I_A \ni m_1 \vdash_A \cdots \vdash_A m_k \vdash_A m \,.$$

k is called the *level* of m (so initial moves have level 0).

A ***prearena*** is an arena with its initial moves labelled OQ. Given arenas A, B, and writing $\bar{\lambda}_A$ for the OP-complement of λ_A, we construct the prearena $A \to B$ by:

$$M_{A\to B} \triangleq M_A + M_B \qquad \lambda_{A\to B} \triangleq [(i_A \mapsto OQ\ ,\ m_A \mapsto \bar{\lambda}_A(m_A))\ ,\ \lambda_B]$$
$$I_{A\to B} \triangleq I_A \qquad \vdash_{A\to B} \triangleq \{(i_A, i_B)\} \cup \{(m,n) \mid m \vdash_{A,B} n\}\,.$$

▲

Moves of an arena A are denoted by m_A and variants, and initial moves by i_A and variants. We set

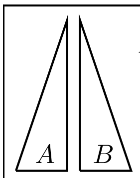
$$J_A \triangleq \{m \in M_A \mid \mathit{level}(m) = 1\} \tag{20}$$

and denote such moves by j_A and variants. By $\bar{I}_A$ we denote $M_A \setminus I_A$ and by $\bar{J}_A$ we denote $M_A \setminus J_A$. We say that an arena A is ***pointed*** if $|I_A| = 1$.

The simplest arena is $0 \triangleq (\varnothing, \varnothing, \varnothing, \varnothing)$. Other flat arenas are 1, N and $\mathrm{A}^{\bar{a}}$, each $\bar{a} \in \mathbb{A}^{\#}$, defined by:

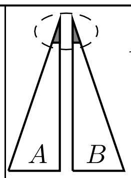
$$M_{\mathrm{N}} = I_{\mathrm{N}} \triangleq \mathbb{N}\,, \qquad M_1 = I_1 \triangleq \{*\}\,, \qquad M_{\mathrm{A}^{\bar{a}}} = I_{\mathrm{A}^{\bar{a}}} \triangleq \mathbb{A}^{\bar{a}}\,. \tag{21}$$

Note that for $\bar{a}$ empty we get $\mathrm{A}^{\epsilon} = 1$. We write A_i for A^a with $a \in \mathbb{A}_i$, $i \in \omega$ (and similarly do we write A_x, each $x \in \mathrm{TY} \cup \{e\}$). Moreover, from arenas A, B we construct the following compound arenas. Note that, because each move has a unique level, arenas can be seen as *levelled labelled graphs* with vertices labelled by λ.



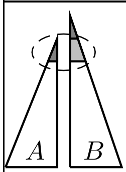
$A + B$

$$M_{A+B} \triangleq M_A + M_B \quad I_{A+B} \triangleq I_A + I_B \quad \lambda_{A+B} \triangleq [\lambda_A\ ,\ \lambda_B] \quad \vdash_{A+B} \triangleq\ \vdash_A \cup \vdash_B$$



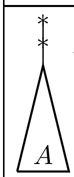
$A \otimes B$

$$M_{A\otimes B} \triangleq I_A \times I_B + \bar{I}_A + \bar{I}_B$$
$$I_{A\otimes B} \triangleq I_A \times I_B$$
$$\lambda_{A\otimes B} \triangleq [((i_A, i_B), PA), \lambda_A \upharpoonright \bar{I}_A, \lambda_B \upharpoonright \bar{I}_B]$$
$$\vdash_{A\otimes B} \triangleq \{((i_A, i_B), m) \mid m \in J_A \cup J_B\} \cup (\vdash_A \upharpoonright \bar{I}_A{}^2) \cup (\vdash_B \upharpoonright \bar{I}_B{}^2)$$



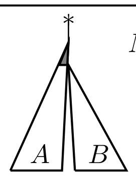
$A \Rightarrow\!\!\otimes B$

$$M_{A\Rightarrow\otimes B} \triangleq I_B + I_A \times J_B + \bar{I}_A + \bar{I}_B \cap \bar{J}_B$$
$$I_{A\Rightarrow\otimes B} \triangleq I_B$$
$$\lambda_{A\Rightarrow\otimes B} \triangleq [(i_B, PA), ((i_A, j_B), OQ), \bar{\lambda}_A \upharpoonright \bar{I}_A, \lambda_B \upharpoonright (\bar{I}_B \cap \bar{J}_B)]$$
$$\vdash_{A\Rightarrow\otimes B} \triangleq \{(i_B, (i_A, j_B))\} \cup \{((i_A, j_B), m) \mid i_A \vdash_A m \vee j_B \vdash_B m\} \cup (\vdash_A \upharpoonright \bar{I}_A{}^2) \cup (\vdash_B \upharpoonright (\bar{I}_B \cap \bar{J}_B)^2)$$



$A_\perp$

$$M_{A_\perp} \triangleq \{*_1, *_2\} + M_A$$
$$I_{A_\perp} \triangleq \{*_1\}$$
$$\lambda_{A_\perp} \triangleq [((*_1, PA), (*_2, OQ)), \lambda_A]$$
$$\vdash_{A_\perp} \triangleq \{(*_1, *_2), (*_2, i_A)\} \cup \vdash_A$$



$A \Rightarrow B$

$$M_{A\Rightarrow B} \triangleq \{*\} + I_A + \bar{I}_A + M_B$$
$$I_{A\Rightarrow B} \triangleq \{*\}$$
$$\lambda_{A\Rightarrow B} \triangleq [(*, PA), (i_A, OQ), \bar{\lambda}_A \upharpoonright \bar{I}_A, \lambda_B]$$
$$\vdash_{A\Rightarrow B} \triangleq \{(*, i_A), (i_A, j_B)\} \cup \vdash_A \cup \vdash_B$$

Figure 2. Basic arena constructions

We will usually identify graph-isomorphic arenas related by isomorphisms which simply manipulate $*$'s; for example, for any A, B,

$$0 + A = A + 0 = A\,, \quad 1 \multimap\!\!\otimes A = A\,, \quad A \Rightarrow B = A \multimap\!\!\otimes B_\perp\,.$$

Most of the previous constructors are familiar from [11]: $\otimes$ and $+$ yield products and coproducts, $_\perp$ is a lifting, and $\Rightarrow$ is a function-space constructor. On the other hand, $\multimap\!\!\otimes$ can be seen as a function-space constructor merging the contravariant part of its RHS with its LHS. For example, for any A, B, C, we have:

$$A \multimap\!\!\otimes (B \Rightarrow C) = (A \otimes B) \Rightarrow C\,. \tag{22}$$

Reading the equality from left to right, the contravariant part of $B \Rightarrow C$, i.e. B, is merged with A. Read from right to left, (22) corresponds to *implication introduction* in Logic; in call-by-name arenas [16] it has the form:

$$A \to (B \to C) = (A \times B) \to C\,.$$

A nominal game is an interaction between Player and Opponent on a certain prearena. The interaction is given by a sequence of ***moves-with-names***. Each such is written $m^{\bar{a}}$ and consists of a move $m \in M_A$ attached with a name-list $\bar{a} \in \mathbb{A}^{\#}$ (hence $m^{\bar{a}}$ has strong support). For a move-with-names x we write:

$$x = \underline{x}^{\mathsf{nlist}(x)} \tag{23}$$

At this point let us introduce some notation for sequences (of names, moves, etc.). If $s = s_1 \cdots s_n$ is a sequence then:
– s^- denotes $s_1 \cdots s_{n-1}$,
– $s.i$ denotes s_i , and $s.\text{-}i$ denotes s_{n+1-i} ,
– $s_{\le s_i}$ denotes $s_1 \cdots s_i$, and so does $s_{< s_{i+1}}$.
For example, $s.\text{-}1$ denotes the last element of s, and hence $s = s^-(s.\text{-}1)$.

A ***justified sequence*** over a prearena A is a finite sequence s of OP-alternating moves-with-names such that, except for $s.1$ which is initial, every move $s.i$ has a *justification pointer* to some $s.j$ such that $j < i$ and $s.j \vdash_A s.i$; we say that $s.j$ *(explicitly) justifies* $s.i$. The *view* $\ulcorner s \urcorner$ of s is a subsequence of s computed by:

$$\begin{aligned} \ulcorner \epsilon \urcorner &\triangleq \epsilon \\ \ulcorner x \urcorner &\triangleq x \\ \ulcorner s\, y\, t\, x \urcorner &\triangleq \ulcorner s \urcorner\, y\, x\,, \ x \text{ explicitly justified by } y. \end{aligned} \tag{24}$$

This definition incorporates those of P-view and O-view [16].

A ***legal sequence*** s on A is a justified sequence of moves-with-names that satisfies *Visibility* and *Well-Bracketing*. The former condition states that, for any x in s, x is explicitly justified by a move in $\ulcorner s_{<x} \urcorner$. The latter stipulates that any Answer x in s be justified by the last open Question in $s_{<x}$ (the *pending Question*).

Definition 9 A legal sequence s is a ***play*** if $s.1$ has empty name-list and s also satisfies:
(NC1) The name-list of any P-move x in s contains as a prefix that of its preceding move, that is, $\mathsf{nlist}(s_{<x}.\text{-}1) \le \mathsf{nlist}(x)$. It possibly contains other names, all of which are fresh for $s_{<x}$.
(NC2) Any name in the support of a P-move x in s that is fresh for $\ulcorner s_{<x} \urcorner$ is contained in $\mathsf{nlist}(x)$.
(NC3) The name-list of any non-initial O-move in s is that of the move explicitly justifying it.
The set of plays on a prearena A is denoted by P_A. ▲

Thus, we take plays to be *innocent ϵ-plays* in terms of [23]. A name a ***is introduced*** (by Player) in a play s if there exists a P-move x in s such that $a \in \mathsf{S}(x)$ and $a \# s_{<x}$. From the definition, this is equivalent to stating:
– $a \in \mathsf{S}(x)$ and $a \# \ulcorner s_{<x} \urcorner$,
– $a \in \mathsf{S}(\mathsf{nlist}(x))$ and $a \# \mathsf{nlist}(y)$,
– $\mathsf{nlist}(x) = \mathsf{nlist}(y)\, \bar{a}_1 a\, \bar{a}_2$.

Note that (NC1,2) imply that Player cannot play a name that does not appear in his view:[4] if x is a P-move in s and a a name appearing in $s_{<x}$ but not in $\ulcorner s_{<x} \urcorner$, then $a \in \mathtt{S}(x)$ would imply $a \in \mathtt{S}(\mathsf{nlist}(x))$ by NC2 and therefore $a \# s_{<x}$ by NC1 (and the fact that $s_{<x}$.-1 appears in $\ulcorner s_{<x} \urcorner$ and so $a \# \mathsf{nlist}(s_{<x}.\text{-}1)$).

Plays on $A \to B$ and $B \to C$ yield plays on $A \to C$ via *parallel composition and hiding*. Firstly, $s \in P_{A \to B}$ and $t \in P_{B \to C}$ are ***composable*** if

$$\underline{s} \upharpoonright B = \underline{t} \upharpoonright B$$

and, for any $s' \leq s$, $t' \leq t$ with $\underline{s'} \upharpoonright B = \underline{t'} \upharpoonright B$:

(C1) If s' ends in a P-move in A introducing some name a then $a \# t'$; dually, if t' ends in a P-move in C introducing some name a then $a \# s'$.

(C2) If both s', t' end in B and s' ends in a P-move introducing some name a then $a \# t'^{-}$; dually, if t' ends in a P-move introducing some name a then $a \# s'^{-}$.

The ***parallel interaction*** $s \,\|\, t$ of composable plays s, t is a sequence of moves-with-names from A, B, C computed as follows. Writing $\bar{a}_s$ for $\mathsf{nlist}(s.\text{-}1)$, $\bar{a}_{s \,\|\, t}$ for $\mathsf{nlist}((s \,\|\, t).\text{-}1)$, and so on, we define recursively:

$$\begin{aligned} s m_{A(P)}^{\bar{a}_s \bar{a}} \,\|\, t &\triangleq (s \,\|\, t) m_A^{\bar{a}_{s \,\|\, t} \bar{a}} , & s m_{A(O)}^{\bar{b}} \,\|\, t &\triangleq (s \,\|\, t) m_A^{\bar{b}'} , \\ s \,\|\, t m_{C(P)}^{\bar{a}_t \bar{a}} &\triangleq (s \,\|\, t) m_C^{\bar{a}_{s \,\|\, t} \bar{a}} , & s \,\|\, t m_{C(O)}^{\bar{b}} &\triangleq (s \,\|\, t) m_C^{\bar{b}'} , \\ s m_{B(P)}^{\bar{a}_s \bar{a}} \,\|\, t m_{B(O)}^{\bar{b}} &\triangleq (s \,\|\, t) m_B^{\bar{a}_{s \,\|\, t} \bar{a}} , & s m_{B(O)}^{\bar{b}} \,\|\, t m_{B(P)}^{\bar{a}_t \bar{a}} &\triangleq (s \,\|\, t) m_B^{\bar{a}_{s \,\|\, t} \bar{a}} , \end{aligned} \tag{25}$$

and $\epsilon \,\|\, \epsilon \triangleq \epsilon$, where we write $\bar{b}'$ for the name-list of m_A's (m_C's) justifier in $s \,\|\, t$. Take then,

$$(s\,;t) \triangleq s \,\|\, t \upharpoonright A, C \,. \tag{26}$$

We can show [24] that $s\,;t \in P_{A \to C}$.

Definition 10 A ***strategy*** σ on a prearena A is a set of equivalence classes $[s]$ of plays on A satisfying *prefix closure*, *contingency completeness*, *determinacy*, *innocence* and *totality*:

- If $[su] \in \sigma$ then $[s] \in \sigma$.
- If even-length $[s] \in \sigma$ and sx is a play then $[sx] \in \sigma$.
- If even-length $[s_1 x_1], [s_2 x_2] \in \sigma$ and $[s_1] = [s_2]$ then $[s_1 x_1] = [s_2 x_2]$.
- If $[s_1 x_1], [s_2] \in \sigma$ and odd-length $[\ulcorner s_1 \urcorner] = [\ulcorner s_2 \urcorner]$ then there exists $[s_2 x_2] \in \sigma$ such that $[\ulcorner s_1 x_1 \urcorner] = [\ulcorner s_2 x_2 \urcorner]$.
- If $[i_A] \in \sigma$ then there exists an Answer $m \in M_A$ such that $[i_A\, m] \in \sigma$.

We write $\sigma : A$ if σ is a strategy on A. ▲

Strategies are the arrows of $\mathcal{V}_{\mathtt{t}}$. For example, for any $\mathtt{S}(\bar{a}') \subseteq \mathtt{S}(\bar{a})$, any $n \in \mathbb{N}$, any $i \in \omega$ and any arena B, we have the strategies:

$$\begin{aligned} \frac{\bar{a}}{\bar{a}'} &: \mathrm{A}^{\bar{a}} \to \mathrm{A}^{\bar{a}'} \triangleq \{[\bar{a}\,\bar{a}']\} , \\ \tilde{n} &: 1 \to \mathrm{N} \triangleq \{[*\, n\,]\} , \\ !_B &: B \to 1 \triangleq \{[i_B\, *]\} , \\ \mathtt{eq}_i &: \mathrm{A}_i \otimes \mathrm{A}_i \to \mathrm{N} \triangleq \{[(a,a)\, 0], [(a,b)\, 1] \mid a \# b\} , \\ \mathtt{id}_B &: B \to B \triangleq \{[s] \mid s \in P_{B_{(l)} \to B_{(r)}} \wedge \forall t \leq^{even} s.\ t \upharpoonright B_{(l)} = t \upharpoonright B_{(r)}\} . \end{aligned} \tag{27}$$

Note that in strategy definitions as the ones above we tend to be frugal; we usually omit plays that are obviously in a strategy because of totality, prefix closure, etc. For example $\tilde{n}$ is formally given by the set $\{[\epsilon], [*], [*\, n\,]\}$.

If $\sigma : A \to B$ and $\tau : B \to C$ are strategies then we define the composite strategy:

$$\sigma\,;\tau : A \to C \triangleq \{[s\,;t] \mid [s] \in \sigma \wedge [t] \in \tau \wedge s,t \text{ composable}\} . \tag{28}$$

Strategy-composition is well-defined, associative and has `id` as unit (see [24]). Hence, we have a category.

[4] even if it is the case that it was Player who introduced it in the play — this is innocence.

Definition 11 $\mathcal{V}_{\mathtt{t}}$ is the category having nominal arenas as objects and strategies as arrows. ▲

There is rich structure in $\mathcal{V}_{\mathtt{t}}$: $\otimes$ and $+$ of figure 2 yield products and coproducts respectively, and there are also *partial exponentials* given by the $\multimap$ constructor: for any triple A, B, C of arenas with C pointed there is a bijection

$$\Lambda : \mathcal{V}_{\mathtt{t}}(A \otimes B, C) \xrightarrow{\cong} \mathcal{V}_{\mathtt{t}}(A, B \multimap C) \tag{29}$$

natural in A. Moreover, 1 is terminal, 0 is initial, and the constructor $\_{}_\perp$ yields a strong monad with exponentials. Finally, $\otimes$ can be generalised to an infinite tensor $\bigotimes$ applicable to pointed arenas.

We proceed to the construction of a $\nu\varepsilon\rho$-model in $\mathcal{V}_{\mathtt{t}}$. References are modelled by a store-monad, built on a store-arena $\xi \triangleq \bigotimes_{A\in\mathrm{TY}}(\mathrm{A}_A \Rightarrow [\![A]\!])$, while for exceptions we use the coproduct monad $\_ + \mathrm{A}_e$. Thus, the computational monad to use is (obtained from the functor):

$$T : \mathcal{V}_{\mathtt{t}} \to \mathcal{V}_{\mathtt{t}} \triangleq \xi \Rightarrow ((\_ + \mathrm{A}_e) \otimes \xi) \tag{30}$$

Note that T is compound (as $T = T_1 T_2$, $T_1 \triangleq \xi \Rightarrow (\_ \otimes \xi)$, $T_2 \triangleq \mathrm{A}_e + \_$ and a standard distributive law) and hence precompound. Given ξ, and using the fact that $\_{}_\perp$ is a strong monad with exponentials, we can show that T is a strong monad with exponentials, with TB^A being $A \multimap TB$. Thus, the definitions of ξ and $[\![A]\!]$ are interrelated by the following domain equation.

$$\begin{gathered}[\![1]\!] = 1\,, \quad [\![\mathbb{N}]\!] = \mathrm{N}\,, \quad [\![\mathbb{E}]\!] = \mathrm{A}_e\,, \quad [\![[A]]\!] = \mathrm{A}_A\,, \quad [\![A \times B]\!] = [\![A]\!] \otimes [\![B]\!]\,, \\ [\![A \to B]\!] = [\![A]\!] \multimap (\xi \Rightarrow ([\![B]\!] + \mathrm{A}_e) \otimes \xi)\,, \qquad \xi = \textstyle\bigotimes_A(\mathrm{A}_A \Rightarrow [\![A]\!])\,.\end{gathered} \tag{SE}$$

(SE) is solved by expressing it as a fixpoint functorial equation and finding its minimal invariant [24]. The computations are almost identical to those followed in [23].

Explicitly, the solution is depicted below. For example, the arena ξ contains an initial move $\circledast$ wich justifies Questions $\ddot{a}$, all $\ddot{a} \in \mathbb{A}_A$ and all $A \in \mathrm{TY}$, and each such $\ddot{a}$ justifies a subarena $[\![A]\!]$ where the value of $\ddot{a}$ is stored. On the other hand, $TA = \xi_1 \Rightarrow (A \otimes \xi_2)$ contains an initial move $*$ which justifies a move $\circledast$ opening the store ξ_1. The latter justifies the rest of ξ_1 and it also justifies some Answers opening the store ξ_2. These can be either of the form $(i_A, \circledast)$ (i.e. values) or of the form $(\dot{a}, \circledast)$ (i.e. exceptions).

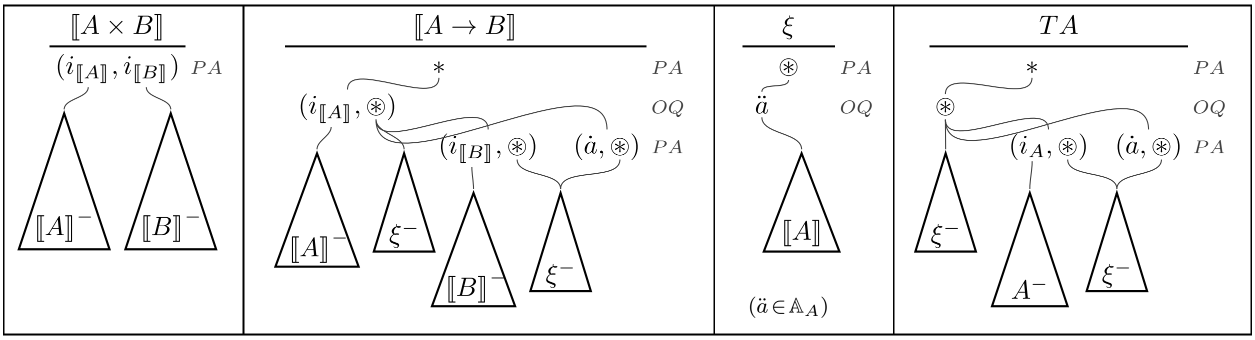


Figure 3. The translation of $\nu\varepsilon\rho$-types in nominal arenas.

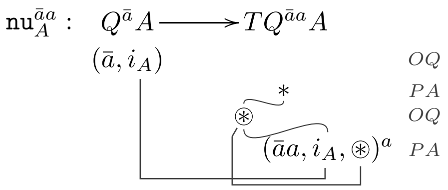
Notice that we reserve $\circledast$ for the initial move of ξ. The monadic natural transformations of T are obtained from those of its components (see [24]).

Having defined arenas $\mathrm{A}_{\bar{a}}$ for each $\bar{a} \in \mathbb{A}^{\#}$ we construct the product comonads $Q^{\bar{a}}$ by:

$$Q^{\bar{a}} : \mathcal{V}_{\mathtt{t}} \to \mathcal{V}_{\mathtt{t}} \triangleq \mathrm{A}_{\bar{a}} \times \_ \tag{31}$$

For each $\bar{a}' \subseteq \bar{a}$, we have already defined a natural transformation $\frac{\bar{a}}{\bar{a}'} : Q^{\bar{a}} \to Q^{\bar{a}'}$. Moreover, we can define a transformation $\mathtt{nu}^{\bar{a}a} : Q^{\bar{a}} \to TQ^{\bar{a}a}$ by using the following strategy.

Opponent starts by supplying the initial local state $\bar{a}$ and the initial A-value i_A; Player answers with a dummy $*$ (dictated by totality); Opponent then plays $\circledast$ opening thus the initial store; Player introduces a fresh-name a, copies i_A and opens a new store, playing $(\bar{a}a, i_A, \circledast)^a$. From that point on, Player copycats between the two copies of i_A and $\circledast$: the latter means that Player has made no store-update when playing $(\bar{a}a, i_A, \circledast)^a$.

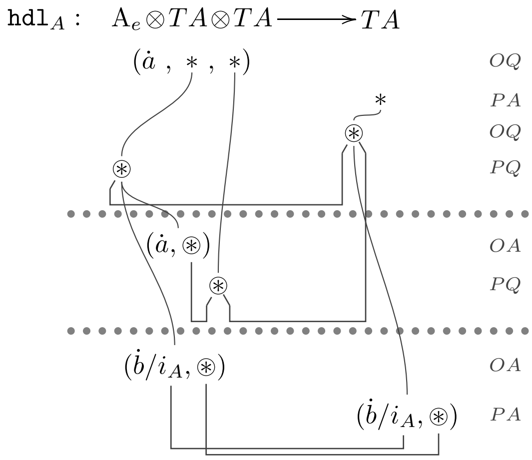


(32)

Note that in diagrams for strategies like the above we depict the strategy's behaviour on P-views, that is, sequences s such that $\ulcorner s' \urcorner = s'$, for each $s' \leq^{odd} s$. The curved lines are justification pointers. The polygonal lines stand for *copycat links*, that is, the strategy copycats (i.e. it plays like `id`) between (the relevant components of) the two linked moves.

The last pieces of structure we need for a $\nu\varepsilon\rho$-model are arrows `upd`, `drf` and `inx`, `hdl`. The former two are essentially the same as those used in [23] and are given in figure 4. On the other hand, `inx` is easily defined by means of coproduct injections while `hdl` is given as follows.

Opponent starts by opening the two TA's and supplying the name $\dot{a}$ to be handled; Player answers $*$; Opponent supplies the initial store $\circledast$; Player copies it under the TA to be tried. If Opponent now asks a name under $\circledast$, Player will copycat it (under the previous $\circledast$).
If instead Opponent answers with $(\dot{a}, \circledast)$ (which means that the TA which was tried resulted to an exception $\dot{a}$) then Player *catches* $\dot{a}$: he plays under the second TA and copycats between that and the TA at the output.
Otherwise, if Opponent answers with another exception name $\dot{b}$ or with a value i_A then there is nothing to catch: Player simply copycats between the tried TA and the output.



(33)

It is not difficult then to obtain the following.

Proposition 12 *$(\mathcal{V}_{\mathtt{t}}, T, Q)$ is a $\nu\varepsilon\rho$-model.* ■

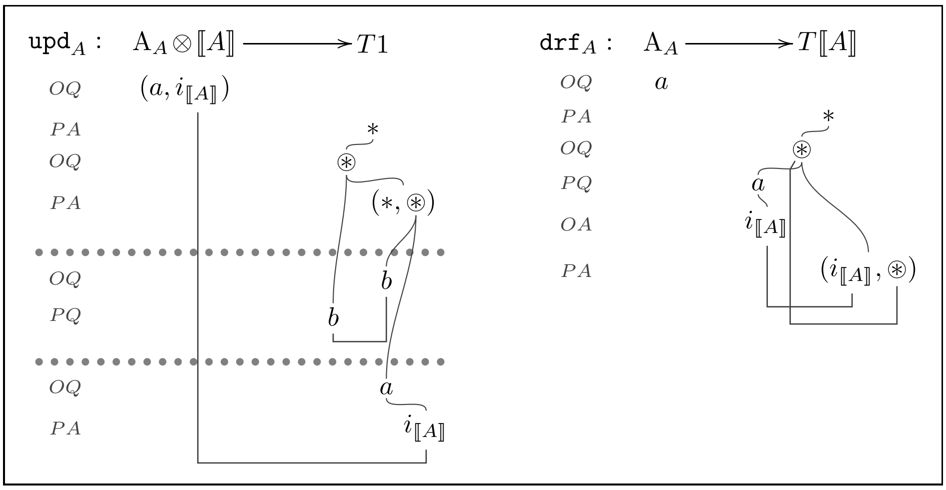


Figure 4. Reference-update and dereferencing in $\mathcal{V}_{\mathtt{t}}$

Adequacy Our last task for this section is to show adequacy for $\mathcal{V}_{\mathtt{t}}$ as a $\nu\varepsilon\rho$-model. First, we note that if a term is non-reducing and behaves like a value (resp. a raised exception) then it is indeed a value (an exception).

Lemma 13 *Let $\mathtt{S}(\bar{a}) \mid \varnothing \vdash M : A$ be a typed term. For any environment P, if $P \models M$ is non-reducing then*
(i) if M is not a value then for no $\bar{b}, i_A$ do we have $[(\bar{a},*) * \circledast (i_A,\circledast)^{\bar{b}}] \in [\![\widehat{P}\,; M]\!]$,
(ii) if M is not a raised exception then for no $\bar{b}, \dot{a}$ do we have $[(\bar{a},*) * \circledast (\dot{a},\circledast)^{\bar{b}}] \in [\![\widehat{P}\,; M]\!]$. ■

Now, for each term M, define $(M)^\circ$ recursively as follows.

$$\begin{aligned}&(a)^\circ \triangleq a\,, \quad (x)^\circ \triangleq x\,, \quad \dots \quad (\lambda x.M)^\circ \triangleq \lambda x.(M)^\circ\,, \quad (M\,N)^\circ \triangleq (M)^\circ(N)^\circ\,, \quad \dots\\ &(\mathtt{try}\ N_1\ \mathtt{handle}\ M \Rightarrow N_2)^\circ \triangleq \mathtt{try}\ (N_1)^\circ\ \mathtt{handle}\ (M)^\circ \Rightarrow \nu a.(N_2)^\circ,\ \text{some } a \text{ not free in } N_2\,.\end{aligned} \tag{34}$$

The main technical result is the following lemma (see [24] for a proof).

Lemma 14 *For any $\mathtt{S}(\bar{a}) \mid \Gamma \vdash M : A$ and any initial move i_Γ of $[\![\Gamma]\!]$, if there is a pair $\bar{b}, i_A$ such that $[(\bar{a},i_\Gamma)*\circledast(i_A,\circledast)^{\bar{b}}] \in [\![M]\!]$, then there is some $\bar{b}'$ such that $\mathtt{S}(\bar{b}) \subseteq \mathtt{S}(\bar{b}')$ and $[(\bar{a},i_\Gamma)*\circledast(i_A,\circledast)^{\bar{b}'}] \in [\![(M)^\circ]\!]$.* ■

Proposition 15 (Adequacy) *$\mathcal{V}_{\mathtt{t}}$ is adequate: for any closed term $M : 1$, if $[\![M]\!] = [\![\nu\bar{a}.\,\widehat{P}]\!]$ for some P then there exists P' such that $\models M \longrightarrow\!\!\!\!\!\rightarrow P' \models \mathtt{skip}$.*

Proof: By lemma 13 it suffices to show that, for any such M, there is a non-reducing sequent $P' \models N$ such that $\models M \longrightarrow\!\!\!\!\!\rightarrow P' \models N$. For sake of contradiction suppose the opposite, that is, there exists an infinite reduction sequence starting from $\models M$.
The sequence must contain infinitely many reductions from the set $\{\mathrm{HL}, \mathrm{NHL}, \mathrm{VHL}, \mathrm{XPN}\}$, or otherwise it would end in an infinite reduction sequence in $\nu\rho$, contradicting the latter's adequacy (see [23,24]). Moreover, if it contained infinitely many reductions from $\{\mathrm{NHL}, \mathrm{XPN}, \mathrm{VHL}\}$ but finitely many HL reductions, then it would have either to terminate at some raised exception or to end in an infinite sequence of reductions in $\nu\rho + \mathrm{VHL}$. The latter would then produce an infinite reduction sequence in $\nu\rho$. We therefore have that $\models M$ has a reduction sequence containing infinitely many HL reductions. Clearly then, $\models (M)^\circ$ diverges using infinitely many NEW reduction steps.
Now, $[\![M]\!] = [\![\nu\bar{a}.\,\widehat{P}]\!]$ implies $[* * \circledast(*,\circledast)^{\bar{a}}] \in [\![M]\!]$ and hence $[* * \circledast(*,\circledast)^{\bar{a}'}] \in [\![(M)^\circ]\!]$ for some $\bar{a}'$, by previous lemma. But we have that $\models (M)^\circ$ diverges creating infinitely many fresh names, so in particular $\models (M)^\circ \longrightarrow\!\!\!\!\!\rightarrow P' \models M'$ with $|\mathtt{dom}(P')| = |\bar{a}'|+1$. By correctness, $[\![(M)^\circ]\!] = [\![\nu\bar{a}''.\,\widehat{P}'; M]\!]$ with $\mathtt{S}(\bar{a}'') = \mathtt{dom}(P')$ and therefore $[* * \circledast(*,\circledast)^{\bar{a}'}] \in [\![(M)^\circ]\!]$ implies that $\bar{a}'$ contains at least $\bar{a}''$, contradicting $|\bar{a}''| = |\bar{a}'| + 1$. ■

6. Full abstraction

In the previous section we showed that $\mathcal{V}_{\mathtt{t}}$ is a sound model for $\nu\varepsilon\rho$. However, in our games we have included store- and exception-related behaviours that are disallowed in the operational semantics. The problems with store-discipline in $\mathcal{V}_{\mathtt{t}}$ are explained in [23]. Regarding exceptions, the problem is that strategies *may well handle fresh (unknown) exceptions*, whereas in the operational semantics a fresh exception always escapes out of its context.

In order to obtain a fully abstract semantics we will have to constrain strategies by disallowing such behaviours. Specifically, we constrain arenas to type-denotations and strategies to *x-tidy* ones.

Definition 16 Consider $\mathcal{V}_{\nu\varepsilon\rho}$, the full subcategory of $\mathcal{V}_{\mathtt{t}}$ with the following set of objects. [5]

$$Ob(\mathcal{V}_{\nu\varepsilon\rho}) \ni A, B ::= 1 \mid \mathrm{N} \mid \mathrm{A}^{\bar{a}} \mid A \otimes B \mid A \Rightarrow\!\!\!\otimes\, TB$$

For each object A define its set of ***store-Handles***, H_A, and its set of ***exception-raisers***, X_A, as follows. Setting $A \Rightarrow\!\!\!\otimes\, TB \triangleq A \Rightarrow\!\!\!\otimes\, (\xi_A \Rightarrow (B + \mathrm{A}_e) \otimes \xi_B)$ and $\xi \triangleq \bigotimes_C (\mathrm{A}_C \Rightarrow [\![C]\!])$, we take (recall also fig. 3):

[5] Note in particular that $[\![A]\!], Q^{\bar{a}}[\![A]\!], T[\![A]\!] \in Ob(\mathcal{V}_{\nu\varepsilon\rho})$, for each type A, by taking $T[\![A]\!] = 1 \Rightarrow\!\!\!\otimes\, T[\![A]\!]$.

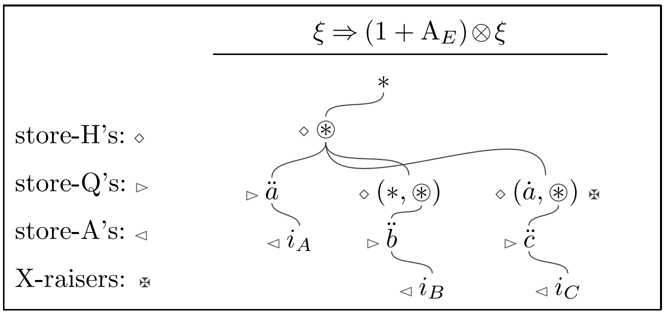


Figure 5. Move-classes in the arena $T1$.

$$
\begin{aligned}
H_1 = H_{\mathrm{N}} = H_{\mathrm{A}^{\bar{a}}} &\triangleq \varnothing\,,\\
H_{A\otimes B} &\triangleq H_A \cup H_B\,,\\
H_{A \Rightarrow_{\otimes} TB} &\triangleq \{(i_A,\circledast_A),(i_B,\circledast_B),(\dot{a},\circledast_B)\}\\
&\quad\cup H_A \cup H_B \cup H_{\xi_A} \cup H_{\xi_B}\,,\\
H_\xi &\triangleq \bigcup\nolimits_C H_{[\![C]\!]}\,;
\end{aligned}
\qquad
\begin{aligned}
X_1 = X_{\mathrm{N}} = X_{\mathrm{A}^{\bar{a}}} &\triangleq \varnothing\,,\\
X_{A\otimes B} &\triangleq X_A \cup X_B\,,\\
X_{A \Rightarrow_{\otimes} TB} &\triangleq \{(\dot{a},\circledast_B)\} \cup X_A \cup X_B \cup X_{\xi_A} \cup X_{\xi_B}\,,\\
X_\xi &\triangleq \bigcup\nolimits_C X_{[\![C]\!]}\,.
\end{aligned}
$$

In an object A, a store-Handle justifies Questions of the form $\ddot{a}$, which we call ***store-Questions***. Answers to store-Questions are called ***store-Answers***. ▲

The classification of moves relatively to the store is familiar from [23]: store-H's are moves opening new stores, where a store consists of combinations of store-Q's and store-A's. Regarding exceptions, X-raisers are moves raising an exception — note that exceptions (i.e. exception-names) may also appear in a game unraised, as values (compare $[\![\dot{a}]\!]$ with $[\![\texttt{raise}\ \dot{a}]\!]$). Note that X-raisers are A-store-H's (i.e. store-H's that are Answers) justified by Q-store-H's, and that every Q-store-H justifies X-raisers. We can show that a move in $A \in Ob(\mathcal{V}_{\nu\varepsilon\rho})$ is exclusively either initial or a store-H or a store-Q or a store-A. An example of how these classes of moves are related is given in the diagram of $T1$ in figure 5.

These notions can be straightforwardly extended to prearenas by setting

$$H_{A\to B} \triangleq H_A \cup H_B \quad\text{and}\quad X_{A\to B} \triangleq X_A \cup X_B\,. \tag{35}$$

Around them we define x-tidy strategies. Note that since store-H's may occur in several places in a game we may use tags to distinguish identical moves from different stores. For example, the same store-Q q may be denoted $q_{(O)}$ or $q_{(P)}$, the particular notation denoting also the OP-polarity of the move.

Definition 17 A strategy σ is ***x-tidy*** if whenever odd-length $[s] \in \sigma$ then:

(TD1) If s ends in a store-Q q then $[sx] \in \sigma$, with x being either a store-A to q introducing no new names, or a copy of q. In particular, if $q = \ddot{a}^{\bar{a}}$ with $\ddot{a} \,\#\, \ulcorner s \urcorner^{-}$ then the latter case is the case.

(TD2) If $[sq] \in \sigma$ with q a store-Q then q is justified by last O-store-H in $\ulcorner s \urcorner$.

(TD3) If $\ulcorner s \urcorner = s' q_{(O)} q_{(P)} t\, y_{(O)}$ with q a store-Q then $[s\, y_{(P)}] \in \sigma$ with $y_{(P)}$ justified by $\ulcorner s \urcorner$.-3.

(xTD1) If s ends in an X-raiser $(\dot{a},\circledast)^{\bar{a}}$ with $\dot{a} \,\#\, \ulcorner s \urcorner^{-}$ then $[s(\dot{a},\circledast)^{\bar{a}}] \in \sigma$.

(xTD3) If $\ulcorner s \urcorner = s'(\dot{a},\circledast)^{\bar{a}}_{(O)}(\dot{a},\circledast)^{\bar{a}}_{(P)} q_{(O)}$ with $q_{(O)}$ a store-Q, $(\dot{a},\circledast)_{(O)}$ an X-raiser and $\dot{a} \# s'$, then $[s\, q_{(P)}] \in \sigma$.

Let $\chi\mathcal{T}$ be the lluf subcategory of $\mathcal{V}_{\nu\varepsilon\rho}$ of x-tidy strategies. ▲

The (TD) conditions define tidy strategies of [23] imposing a certain store-discipline:

- (TD1) states that, whenever O plays a store-Q, say $\ddot{a}^{\bar{a}}$, Player must either answer it (providing thus the stored value of $\ddot{a}$) or copycat it (expressing thus the fact that he has not updated $\ddot{a}$ since the last store-H played by O).
- (TD2) states that Player may ask store-Q's only at the last store-H played by O in the view.
- (TD3) ensures that whenever Player decides to copycat a store-Q he must preserve that copycat link.

The tidiness conditions describe the interactive nature of our nominal store: when encountered with a store-Q, each participant either answers with an updated value or asks the same store-Q himself and establishes a copycat link between the two store-Q's. Thus, the whole of the store can be accessed without breaking innocence! — see also [24, *innocent store*].
On the other hand, the (xTD) conditions provide a straightforward fresh-exception-discipline:
- (xTD1) states that when a fresh raised exception is encountered, it must be copycatted (i.e. it must escape).
- (xTD3) ensures that a fresh exception is copycatted without any store-updates taking place in the process.

In fact, behind (xTD1) there is a hidden lemma: the move to be played by P is an Answer, so it should be an Answer to the pending Question.

Lemma 18 *If odd-length* $[s] \in \sigma$ *ends in an X-raiser* $(\dot{a}, \circledast)^{\bar{a}}$ *then* s *has a pending-Q which is an O-store-H, and* $s(\dot{a}, \circledast)^{\bar{a}}$ *is a play.*

Proof: s being odd-length implies that it has a pending Question, say q. If q were a P-move then $s = s_1 q s_2$ with s_1, s_2 being odd-length, so an A in s_2 should be justified by q, contradiction. Hence, q an O-move. Moreover, q cannot be initial, by totality, and neither a store-Q: q being unanswered would mean that P copycats after it, so the move following q would be a copy of it answered by an O-store-A y, say. When y is played, P must answer q with a copy of y, thus y can only be the last move in s, i.e. the X-raiser $(\dot{a}, \circledast)^{\bar{a}}$, conrtadiction as y a store-A. Hence, q an O-store-H. Thus, $s(\dot{a}, \circledast)^{\bar{a}}$ is a justified sequence satisfying well-bracketing, and it clearly satisfies NC's. Finally, it also satisfies visibility since s and $\ulcorner s \urcorner$ have the same pending-Q (see e.g. [16]). ■

It is easy to see that identity arrows are x-tidy. Moreover, x-tidy strategies compose and thus we have a subcategory of nominal arenas and x-tidy strategies.

Proposition 19 *If* $\sigma : A \to B$ *and* $\tau : B \to C$ *are x-tidy strategies then so is* $\sigma ; \tau$.

Proof: We know from [23,24] that the (TD) conditions are preserved under composition, so we need only focus on the (xTD) ones — as the proof is not particulary involved, we only show (xTD1). So let odd-length $[s ; t] \in \sigma ; \tau$ be ending in an X-raiser $(\dot{a}, \circledast)^{\bar{a}}$ with $\dot{a} \# \ulcorner s ; t \urcorner^{-}$. Assume, wlog, that $s ; t$ ends in A, so $s.\text{-}1 = (\dot{a}, \circledast)^{\bar{a}_1}$, some sublist $\bar{a}_1$ of $\bar{a}$. By a standard nominal-games argument we then have $\dot{a} \# \ulcorner s \urcorner^{-}$, so $[s(\dot{a}, \circledast)^{\bar{a}_1}] \in \sigma$. If $(\dot{a}, \circledast)^{\bar{a}_1}$ is in A then we are done. Otherwise, we have that $[t(\dot{a}, \circledast)^{\bar{a}_2}] \in \tau$ some sublist $\bar{a}_2$ of $\bar{a}$. Applying the same reasoning consecutively, some $(\dot{a}, \circledast)^{\bar{a}_n}$ is played in AC, giving the required copy of $(\dot{a}, \circledast)^{\bar{a}}$. ■

Definition 20 $\chi\mathcal{T}$ is the lluf subcategory of $\mathcal{V}_{\nu\varepsilon\rho}$ of x-tidy strategies. ▲

We can check that all of the structure of $\mathcal{V}_{\mathtt{t}}$ required for modelling $\nu\varepsilon\rho$ is x-tidy — but θ is not,[6] and this was the reason for ostracising it to a supcategory in definition 1 — so $\chi\mathcal{T}$ is an adequate $\nu\varepsilon\rho$-model. Our remaining task is to show definability, and from that full-abstraction.

We henceforth consider solely x-tidy strategies. Because of innocence, each strategy σ is defined by its ***viewfunction***,

$$\mathtt{viewf}(\sigma) \triangleq \{\ [s] \in \sigma \mid |s| \text{ even } \wedge s = \ulcorner s \urcorner\ \}. \tag{36}$$

`viewf` has an inverse function `strat`, which goes from viewfunctions to strategies.

The viewfunction of a strategy still contains a lot of extra information, in the sense of behaviours which are anyway common to all x-tidy strategies. In fact, each strategy σ is defined by $\mathtt{trunc}(\sigma)$, which is the subset of $\mathtt{viewf}(\sigma)$ excluding:
- all default initial Answers (dictated by totality),
- all the store-copycats (dictated by (TD) conditions),

[6] The interested reader may indulge himself verifying this fact, and also that the loss of x-tidiness is hidden by the compositions in $\mathtt{id} \times \theta \times \mathtt{id} ; \tau \times \mathtt{id} ; \tau' ; T\mathtt{hdl} ; \mu$.

- and all fresh-exception copycats (dictated by (xTD) conditions).

We say a strategy σ is ***finitary*** if $\mathtt{trunc}(\sigma)$ is finite. Intuitively, a strategy is finitary if its non-default behaviour is finite.

Proposition 21 (Definability) *Let A, B be types and $\sigma : Q^{\bar{a}}[\![A]\!] \to T[\![B]\!]$ be finitary. Then σ is definable, i.e. there exists a term $\mathtt{S}(\bar{a}) \mid \Gamma \vdash M : B$ such that $\sigma = [\![M]\!]$.*

From the above, by a more-or-less standard game-semantical argument we can obtain full-abstraction with relation to the following semantical preorder. For each $f, g : Q^{\bar{a}}A \to TB$, set

$$f \lesssim^{\bar{a}} g \stackrel{\Delta}{\iff} \forall \rho : Q^{\bar{a}}(A \Rightarrow\!\!\otimes\, TB) \to T1.\ (\Lambda^{\bar{a}}(f)\,;\rho\downarrow \implies \Lambda^{\bar{a}}(g)\,;\rho\downarrow)\,, \tag{37}$$

where $\Lambda^{\bar{a}}(f) \triangleq Q^{\bar{a}}1 \xrightarrow{\delta} Q^{\bar{a}}Q^{\bar{a}}1 \xrightarrow{Q^{\bar{a}}\Lambda^T(\zeta\,;f)} Q^{\bar{a}}(A \Rightarrow\!\!\otimes\, TB)$ and, for each $\sigma : Q^{\bar{a}}1 \to T1$, $\sigma\downarrow$ iff there is some $\bar{b}$ such that $[(\bar{a},*) * \circledast\, (*,\circledast)^{\bar{b}}] \in \sigma$.

Theorem 22 (FA) *For any $S \mid \Gamma \vdash M, N : A$, $\ M \lessapprox N \iff [\![M]\!] \lesssim [\![N]\!]$.* ■

Proof of Definability: Assume $A = A_1 \times \cdots \times A_n$ and $B = B_1 \times \cdots \times B_m$, with A_i's and B_i's non-products, and fix a context $\Gamma = z_1 : A_1, ..., z_n : A_n$. We do induction on $(|\mathtt{trunc}(\sigma)|, \|\sigma\|)$, where we let $\|\sigma\|$ be the maximum number of names introduced in any play of $\mathtt{trunc}(\sigma)$. If $|\mathtt{trunc}(\sigma)| = 0$ then $\sigma = [\![\mathtt{stop}_B]\!]$; otherwise, there exist $x_0, i_{A(0)}$ such that $[(\bar{a}, i_{A(0)}) * \circledast\, x_0] \in \sigma$. If we set $\sigma_0 \triangleq \sigma\restriction(\bar{a}, i_{A(0)})$ and $\sigma' = \sigma \setminus \sigma_0$ [7] then

$$\sigma = \langle [x \stackrel{\bar{a}}{=} i_{A(0)}], \langle \sigma_0, \sigma' \rangle\rangle; \mathtt{cnd}\,.$$

$[x \stackrel{\bar{a}}{=} i_{A(0)}] : Q^{\bar{a}}[\![A]\!] \to \mathrm{N}$ is the strategy which returns 0 if the initial move is (a permutation of) $(\bar{a}, i_{A(0)})$ and otherwise 1. It is not difficult to construct a term $\mathtt{S}(\bar{a}) \mid \Gamma \vdash N_0 : \mathbb{N}$ such that $[\![N_0]\!] = [x \stackrel{\bar{a}}{=} i_{A(0)}];\eta$. Moreover, $|\mathtt{trunc}(\sigma')| < |\mathtt{trunc}(\sigma)|$ and $(0,0) < (|\mathtt{trunc}(\sigma_0)|, \|\sigma_0\|) \leq (|\mathtt{trunc}(\sigma)|, \|\sigma\|)$, so by IH there exists term M' such that $[\![M']\!] = \sigma'$. Hence, if there exists a term M_0 with $[\![M_0]\!]\restriction(\bar{a}, i_{A(0)}) = \sigma_0$ then

$$\sigma = [\![\mathtt{if0}\ N_0\ \mathtt{then}\ M_0\ \mathtt{else}\ M']\!]\,.$$

We proceed to find M_0 . If the move x_0 introduces fresh names $\bar{b}$, say, then we can use the IH (on $\|\sigma\|$) and obtain a term $M_{\bar{b}}$ such that $\sigma_0 = \mathtt{nu}^{\bar{a}\bar{b}}; T[\![M_{\bar{b}}]\!]; \mu$ and hence we can take $M_0 \triangleq \nu\bar{b}.\, M_{\bar{b}}$.

Assume now $x_0 = m_0$. If m_0 is a store-Q $\ddot{a}$ of type C, say, then define the strategy

$$\sigma_{\ddot{a}} : Q^{\bar{a}}([\![A]\!] \otimes [\![C]\!]) \to T[\![B]\!] \triangleq \mathtt{strat}\{\ [(\bar{a}, i_{A(0)}, i_C) * \circledast\, s] \mid [(\bar{a}, i_{A(0)}) * \circledast\, \ddot{a}\, i_C s] \in \mathtt{viewf}(\sigma_0)\ \}\,.$$

We have $\mathtt{trunc}(\sigma_{\ddot{a}}) < \mathtt{trunc}(\sigma_0)$, and therefore there exists $\mathtt{S}(\bar{a}) \mid \Gamma, y : C \vdash M_{\ddot{a}} : B$ such that $\sigma_{\ddot{a}} = [\![M_{\ddot{a}}]\!]$, and taking

$$M_0 \triangleq \begin{cases} (\lambda y.M_{\ddot{a}})(!\,\ddot{a}) & \text{, if } \ddot{a} \in \mathtt{S}(\bar{a}) \\ (\lambda y.M_{\ddot{a}})(!\,z_j) & \text{, if } \ddot{a} \# \bar{a} \wedge j = \min\{j \mid \ddot{a} = (i_{A(0)})_j\} \end{cases}$$

we have $\sigma_0 = [\![M_0]\!]\restriction(\bar{a}, i_{A(0)})$.

Otherwise, $m_0 = j_A \vee m_0 = (i_B/\dot{a}, \circledast)$, a store-H. If there exists a store-Q $\ddot{a} \in \mathbb{A}_C$ such that σ_0 answers to $[i_{A(0)} * \circledast\, m_0\, \ddot{a}]$ then define the strategy

$$\sigma_{\ddot{a}} : Q^{\bar{a}}[\![A]\!] \to T[\![C]\!] \triangleq \mathtt{strat}\{\ [(\bar{a}, i_{A(0)}) * \circledast\, (i_C, \circledast)s] \mid [(\bar{a}, i_{A(0)}) * \circledast\, m_0\, \ddot{a}\, i_C s] \in \mathtt{viewf}(\sigma_0)\ \}\,.$$

$\sigma_{\ddot{a}}$ denotes the value stored for $\ddot{a}$. Taking $\sigma' \triangleq \sigma_0 \setminus \sigma_{\ddot{a}}$ [8] we have $|\mathtt{trunc}(\sigma_{\ddot{a}})|\,, |\mathtt{trunc}(\sigma')| < |\mathtt{trunc}(\sigma_0)|$. By IH, there exist $\mathtt{S}(\bar{a}) \mid \Gamma \vdash M_{\ddot{a}} : C$ and $\mathtt{S}(\bar{a}) \mid \Gamma \vdash M' : B$ such that $\sigma_{\ddot{a}} = [\![M_{\ddot{a}}]\!]$ and $\sigma' = [\![M']\!]$. Taking

$$M_0 \triangleq \begin{cases} (\ddot{a} := M_{\ddot{a}}); M' & \text{, if } \ddot{a} \in \mathtt{S}(\bar{a}) \\ (z_j := M_{\ddot{a}}); M' & \text{, if } \ddot{a} \# \bar{a} \wedge j = \min\{j \mid \ddot{a} = (i_{A(0)})_j\} \end{cases}$$

[7] The notation here is slightly abusive: by $\sigma \setminus \sigma_0$ we do not mean exactly the set-theoretic difference, but rather the latter extended in a default way to a total strategy.

[8] Again, the notation is abusive: σ' plays exactly like σ_0 except for the play $[(\bar{a}, i_{A(0)}) * \circledast\, m_0\, \ddot{a}]$ to which it replies by opening a store-copycat.

we obtain $\sigma_0 = [\![M_0]\!]$.
We are left with the case of m_0 being as above and σ_0 not answering to any store-Q, which corresponds to the case of Player not updating any names before playing m_0. If $m_0 = (\dot{a}, \circledast)$ then $\sigma_0 = [\![M_0]\!] \upharpoonright (\bar{a}, i_{A(0)})$ by taking

$$M_0 \triangleq \begin{cases} \texttt{raise}\ \dot{a} & , \text{if } \dot{a} \in \mathtt{S}(\bar{a}) \\ \texttt{raise}\ z_j & , \text{if } \dot{a} \mathrel{\#} \bar{a} \wedge j = \min\{j \mid \dot{a} = (i_{A(0)})_j\} \,. \end{cases}$$

If $m_0 = (i_B, \circledast)$ then we need to derive a value term $\langle V_1, ..., V_m \rangle$ (as $B = B_1 \times \cdots \times B_m$). For each p, if B_p is a base or reference type then we can choose V_p canonically so that its denotation be i_{B_p} . Otherwise, $B_p = B'_p \to B''_p$ and from σ_0 we obtain the strategy $\sigma' : Q^{\bar{a}}([\![A]\!] \otimes [\![B'_p]\!]) \to T[\![B''_p]\!]$ by:

$$\sigma' \triangleq \mathtt{strat}\{[(\bar{a}, i_{A(0)}, i_{B'_p}) * \circledast\, s] \mid [(\bar{a}, i_{A(0)}) * \circledast\, (i_B, \circledast)\, (i_{B'_p}, \circledast)\, s] \in \mathtt{viewf}(\sigma_0)\}.$$

It is not difficult to see that σ' fully describes σ_0 after $(i_{B'_p}, \circledast)$. By IH, there exists $\mathtt{S}(\bar{a}) \mid \mathit{\Gamma}, y : B'_p \models N : B''_p$ such that $[\![N]\!] = \sigma'$; take then $V_p \triangleq \lambda y.N$. Hence, taking

$$M_0 \triangleq \langle V_1, ..., V_m \rangle$$

we obtain $\sigma_0 = [\![M_0]\!] \upharpoonright (\bar{a}, i_{A(0)})$.
If $m_0 = j_A$, played in some $A_i = A'_i \to A''_i$, then $m_0 = (i_{A'_i}, \circledast)$. Assume that $A'_i = A'_{i,1} \times \cdots \times A'_{i,n_i}$ with $A'_{i,p}$'s being non-products. Now, Opponent can either ask some name $\ddot{a}$ (which would lead to a store-CC), or answer at A''_i, or raise a known exception $\dot{b}$, or raise some fresh exception $\dot{a}$ (which would lead to an exception-CC), or play at some $A'_{i,p}$ of arrow type, say $A'_{i,p} = C_{i,p} \to C'_{i,p}$. Hence, taking $S \triangleq \mathtt{S}(\bar{a}, i_{A(0)})$ we have:

$$\mathtt{viewf}(\sigma_0) = f_A \cup \bigcup\nolimits_{\dot{b} \in S} f_{\dot{b}} \cup \bigcup\nolimits_{p=1}^{n_i} f_p$$

where:

$$\begin{aligned} f_A &\triangleq f_0 \cup \{\ [(\bar{a}, i_{A(0)}) * \circledast\, (i_{A'_i}, \circledast)\, (i_{A''_i}, \circledast)\, s] \in \mathtt{viewf}(\sigma_0)\ \} \\ f_{\dot{b}} &\triangleq f_0 \cup \{\ [(\bar{a}, i_{A(0)}) * \circledast\, (i_{A'_i}, \circledast)\, (\dot{b}, \circledast)\, s] \in \mathtt{viewf}(\sigma_0)\ \} \\ f_p &\triangleq f_0 \cup \{\ [(\bar{a}, i_{A(0)}) * \circledast\, (i_{A'_i}, \circledast)\, (i_{C_{i,p}}, \circledast)\, s] \in \mathtt{viewf}(\sigma_0)\ \} \\ f_0 &\triangleq \{\ [(\bar{a}, i_{A(0)}) * \circledast\, (i_{A'_i}, \circledast)\, s] \mid [\circledast\ \circledast\ s] \in \mathtt{viewf}(\mathtt{id}_\xi) \\ &\qquad\qquad \vee (s.1 = (\dot{a}, \circledast) \wedge \dot{a} \notin S \wedge [s] \in \mathtt{viewf}(\mathtt{id}_{\mathrm{A}_e \otimes \xi}))\ \} \end{aligned}$$

and where we assume $f_p \triangleq f_0$ if $A'_{i,p}$ is not an arrow type. It is not difficult to see that $f_A, f_{\dot{b}}, f_p$ are viewfunctions. Now, from f_A we obtain the strategy

$$\sigma_A : Q^{\bar{a}}([\![A]\!] \otimes [\![A''_i]\!]) \to T[\![B]\!] \triangleq \mathtt{strat}\{\ [(\bar{a}, i_{A(0)}, i_{A''_i}) * \circledast\, s] \mid [(\bar{a}, i_{A(0)}) * \circledast\, (i_{A'_i}, \circledast)\, (i_{A''_i}, \circledast)\, s] \in f_A\ \}\,.$$

By IH, there exists some $\mathtt{S}(\bar{a}) \mid \mathit{\Gamma}, y : A''_i \models M_A : B$ such that $[\![M_A]\!] = \sigma_A$.
From each $f_p \neq f_0$ we obtain a strategy

$$\sigma_p : Q^{\bar{a}}([\![A]\!] \otimes [\![C_{i,p}]\!]) \to T[\![C'_{i,p}]\!] \triangleq \mathtt{strat}\{[(\bar{a}, i_{A(0)}, i_{C_{i,p}}) * \circledast\, s] \mid [(\bar{a}, i_{A(0)}) * \circledast\, (i_{A'_i}, \circledast)\, (i_{C_{i,p}}, \circledast)\, s] \in f_p\}\,.$$

By IH, there exists some $\mathtt{S}(\bar{a}) \mid \mathit{\Gamma}, y' : C_{i,p} \models M_p : C'_{i,p}$ such that $[\![M_p]\!] = \sigma_p$, so take $V_p \triangleq \lambda y'.M_p$. For each $A'_{i,p}$ of non-arrow type, the behaviour of σ_0 at $A'_{i,p}$ is fully described by $(i_{A'_i})_p$, so we take V_p to be the denotation of $(i_{A'_i})_p$. $\langle V_1, ..., V_{n_i} \rangle$ is now of type A'_i and describes σ_0's behaviour in A'_i.
Finally, from each $f_{\dot{b}}$ we obtain a strategy

$$\sigma_{\dot{b}} : Q^{\bar{a}}[\![A]\!] \to T[\![B]\!] \triangleq \mathtt{strat}\{[(\bar{a}, i_{A(0)})\ *\ \circledast\, s] \mid [(\bar{a}, i_{A(0)})\ *\ \circledast\, (i_{A'_i}, \circledast)\, (\dot{b}, \circledast)\, s] \in f_{\dot{b}}\}\,.$$

By IH, there exists some $\mathtt{S}(\bar{a}) \mid \mathit{\Gamma} \models M_{\dot{b}} : B$ such that $[\![M_{\dot{b}}]\!] = \sigma_{\dot{b}}$.
Now, taking for each known exception-name $\dot{b}$

$$N_{\dot{b}} \triangleq \begin{cases} \dot{b} & , \text{if } \dot{b} \in \mathtt{S}(\bar{a}) \\ z_j & , \text{if } \dot{b} \mathrel{\#} \bar{a} \wedge j = \min\{j \mid \dot{b} = (i_{A(0)})_j\}\,, \end{cases}$$

and (note below that the vector-notation stands for nesting of handlers for each $\dot{b} \in S$)

$$M_0 \triangleq (\texttt{try}\ (\lambda x'.\lambda x.(\lambda y.M_A)\, x')(z_i\langle V_1, ..., V_{n_i}\rangle)\ \texttt{handle}\ \overrightarrow{N_{\dot{b}}\ \texttt{=>}\ \lambda x.M_{\dot{b}}})\ \texttt{skip}\,,$$

for some x, x' not free in $M_A, M_{\dot{b}}$'s, we obtain $\sigma_0 = [\![M_0]\!]\restriction(\bar{a}, i_{A(0)})$. ■

7. An equivalence established semantically

We prove the equivalence $M_2 \cong M_3$ of page 5 in the $\nu\varepsilon$-calculus using the fully abstract model for $\nu\varepsilon\rho$. By soundness, it suffices to show that, for any x-tidy strategy $\rho : T((\mathrm{A}_e \Rightarrow_{\otimes} T\mathrm{N}) \Rightarrow_{\otimes} \mathrm{N}) \to T1$ which *does not use the store*,

$$[\![M_2]\!]\,;\rho\downarrow \iff [\![M_3]\!]\,;\rho\downarrow\ . \tag{38}$$

In fact, it suffices to assume ρ does not ask store-Q's unless in a copycat. The denotations $[\![M_2]\!]$ and $[\![M_3]\!]$ are given in the following figure. Note that we have omitted store-copycat links and also the exception-copycat that occurs if Opponent plays an exception under $(\dot{b}, \circledast)^{\dot{a}\dot{b}}_{(4)}/(\dot{a}, \circledast)^{\dot{a}}_{(4)}$.

We show only one direction of the equivalence (other similar). Let $[* * \circledast (*, \circledast)^{\bar{a}}] \in [\![M_2]\!]\,;\rho$, some $\rho, \bar{a}$ with ρ not asking store-Q's. Then, the interaction witnessing this sequence starts with $* * * \circledast\ \circledast^{\bar{b}}$, some $\bar{b}$ introduced by ρ,[9] to which $[\![M_2]\!]$ plays $(*, \circledast)^{\bar{b}\dot{a}\dot{b}}_{(1)}$. At this point, ρ can either play $(*, \circledast)^{\bar{a}}$ or ask $(*, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}}_{(2)}$. In the latter case, $[\![M_2]\!]$ plays $(\dot{a}, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}}_{(3)}$ and now ρ has two choices: either play some $(n, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}'}$ or some exception $(\dot{c}, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}'}$. In the latter case, $[\![M_2]\!]$ responds by also playing $(\dot{c}, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}'}$. Note that $\dot{c}$ cannot be one of $\dot{a}, \dot{b}$ as then x-tidiness of ρ would copycat $(\dot{c}, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}'}$ to the output giving $[* * \circledast (\dot{c}, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}'}] \in [\![M_2]\!]\,;\rho$. At this point, ρ can play either (again) $(*, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}''}_{(2)}$ or $(*, \circledast)^{\bar{a}}$. In the former case, $[\![M_2]\!]$ will play $(*, \circledast)^{\bar{b}\dot{a}\dot{b}\bar{c}''}_{(2)}$. In all cases and up to now, the interaction can be played (modulo $\dot{b}$) by $[\![M_3]\!]\,;\rho$.

So suppose that, after some rounds of Opponent answering with exceptions to $(\dot{a}, \circledast)^{\bar{b}\dot{a}\dot{b}\cdots}_{(3)}$, Opponent plays some $(n, \circledast)^{\bar{b}\dot{a}\dot{b}\cdots}$. At this point, $[\![M_2]\!]$ plays $(\dot{b}, \circledast)^{\bar{b}\dot{a}\dot{b}\cdots}_{(4)}$ and the play continues. But note that $(\dot{b}, \circledast)^{\bar{b}\dot{a}\dot{b}\cdots}_{(4)}$ has now hidden $\dot{a}$ from the P-view of ρ and therefore, because of innocence and the fact that ρ does not use the store, the latter will play in the same way as if $(\dot{a}, \circledast)^{\bar{b}\dot{a}\dot{b}\cdots}_{(4)}$ had been played. Hence, $[\![M_3]\!]\,;\rho$ can simulate the whole play.

---

[9] We may assume that ρ plays a level-1 move of $T((\mathrm{A}_e \Rightarrow_{\otimes} T\mathrm{N}) \Rightarrow_{\otimes} \mathrm{N})$ (such as $\circledast^{\bar{b}}$) *exactly once* in the interaction (*tl4 tests suffice* [24]).

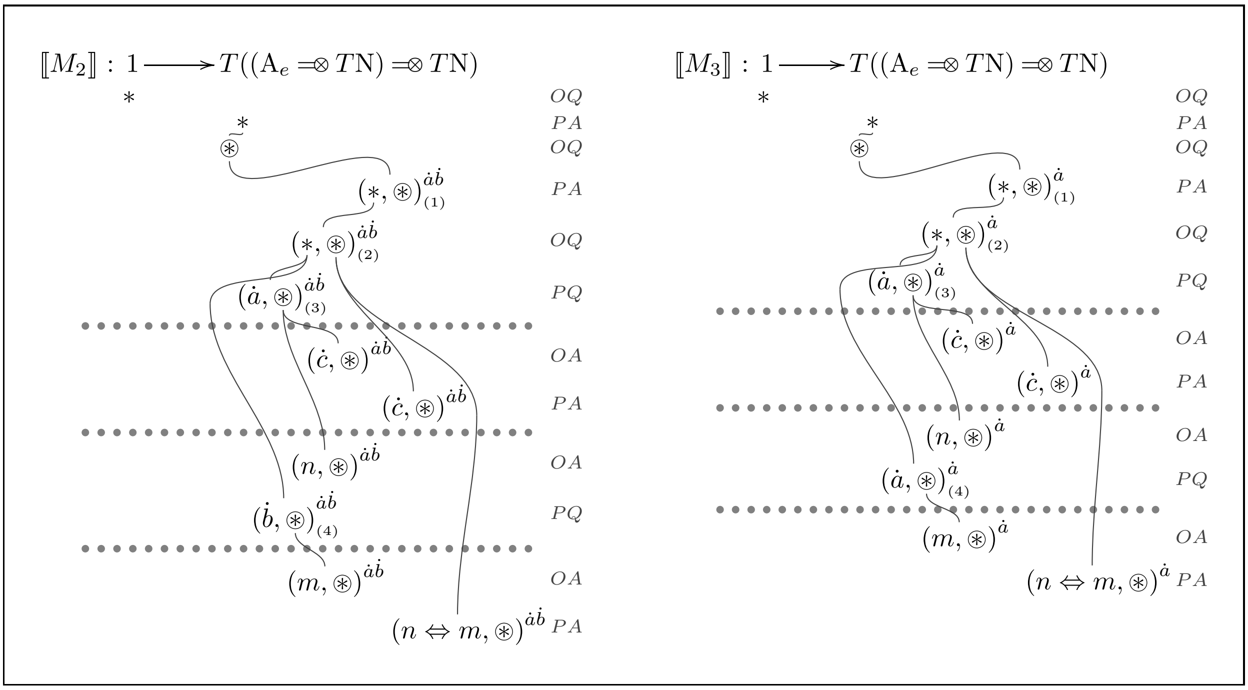


Figure 6: The denotations of the terms M_2 and M_3 of page 5.

8. Further directions

In this paper we have used the nominal games formalism which has evolved from [1], in order to describe a language with nominal exceptions and general references. A defect of our approach is the use of games with local state where names are enlisted in state at the point of their introduction. This makes the semantics too fined grained since it distinguishes, for example, strategies which introduce dummy names. We now think that this approach is somehow outdated and that a precise *name-availability* analysis, in the sense of [15], would allow us to have a stateless formulation of nominal games which would overcome such shortcomings.

What is clearly manifested in this paper and other work on nominal games [1,23,24,13,15,14] is their applicability as a generic denotational framework for nominal computation. Hence, their adaptation to languages with other nominal effects is a further step to consider. Moreover, and as is the case with any semantical framework, nominal games should be used for attacking open issues in nominal programming behaviour, the first such candidate being decidability of program equivalence — in the spirit of [10,9,19,18].

Acknowledgements

References

[1] Abramsky, S., Ghica, D., Murawski, A., Ong, L., and Stark, I. Nominal games and full abstraction for the nu-calculus. In *LICS '04: Proceedings of the 19th Annual IEEE Symposium on Logic in Computer Science* (Turku, Finland, 2004), IEEE Computer Society Press, pp. 150–159.

[2] Abramsky, S., Honda, K., and McCusker, G. A fully abstract game semantics for general references. In *LICS '98: Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science* (Indianapolis, USA, 1998), IEEE Computer Society Press, p. 334.

[3] Abramsky, S., and McCusker, G. Game semantics. In *Computational Logic: Proceedings of the 1997 Marktoberdorf Summer School*, H. Schwichtenberg and U. Berger, Eds., Springer-Verlag, pp. 1–56.

[4] Abramsky, S., and McCusker, G. Linearity, Sharing and State: a fully abstract game semantics for Idealized Algol. In *Algol-like languages*, P. O'Hearn and R. D. Tennent, Eds., vol. 2. Birkhäuser, Boston, 1997, pp. 297–329.

[5] Beck, J. Distributive laws. In *Seminar on Triples and Categorical Homology Theory, ETH, Zürich, 1966/67*, B. Eckmann, Ed., vol. 80 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1969, pp. 119–140.

[6] Brookes, S., and Geva, S. Computational comonads and intensional semantics. In *Applications of Categories in Computer Science: Proceedings LMS Symposium*, vol. 177. Cambridge University Press, Durham, UK, 1991, pp. 1–44.

[7] Cheney, J. Nominal logic and abstract syntax. *SIGACT News 36*, 4 (2005), 47–69.

[8] Gabbay, M. J., and Pitts, A. M. A new approach to abstract syntax with variable binding. *Formal Aspects of Computing 13* (2002), 341–363.

[9] Ghica, D. R. Regular-language semantics for a call-by-value programming language. In *MFPS '01: 17th Annual Conference on Mathematical Foundations of Programming Semantics* (Aarhus, Denmark, 2001), vol. 45, pp. 106–118.

[10] Ghica, D. R., and McCusker, G. Reasoning about Idealized Algol using regular languages. In *ICALP '00: Proceedings of 27th International Colloquium on Automata, Languages and Programming* (Geneva, Switzerland, 2000), vol. 1853 of *LNCS*, Springer-Verlag, pp. 103–116.

[11] Honda, K., and Yoshida, N. Game-theoretic analysis of call-by-value computation. *Theoretical Computer Science 221*, 1–2 (1999), 393–456.

[12] Laird, J. A fully abstract game semantics of local exceptions. In *LICS '01: Proceedings of the 16th Annual IEEE Symposium on Logic in Computer Science* (Boston, MA, USA, 2001), IEEE Computer Society Press, p. 105.

[13] Laird, J. A game semantics of local names and good variables. In *FoSSaCS '04: Proceedings of the 7th International Conference on Foundations of Software Science and Computation Structures* (Barcelona, Spain, 2004), vol. 2987 of *Lecture Notes in Computer Science*, Springer, pp. 289–303.

[14] Laird, J. Game semantics for higher-order concurrency. In *FSTTCS '06: Proceedings of the 26th International Conference on Foundations of Software Technology and Theoretical Computer Science* (Kolkata, India, 2006), vol. 4337 of *Lecture Notes in Computer Science*, Springer, pp. 417–428.

[15] Laird, J. A game semantics of names and pointers. *Annals of Pure and Applied Logic 151* (2008), 151–169. GaLoP '05: First Games for Logic and Programming Languages Workshop (post-proceedings).

[16] McCusker, G. *Games and Full Abstraction for a Functional Metalanguage with Recursive Types.* Distinguished Dissertations. Springer-Verlag, London, 1998.

[17] Moggi, E. Computational lambda-calculus and monads. In *LICS '89: Proceedings of 4th Annual IEEE Symposium on Logic in Computer Science* (Pacific Grove, CA, USA, 1989), IEEE Computer Society Press, pp. 14–23.

[18] Murawski, A. S. On program equivalence in languages with ground-type references. In *LICS '03: Proceedings od the 18th IEEE Symposium on Logic in Computer Science* (Ottawa, ON, Canada, 2003), IEEE Computer Society Press, pp. 108–117.

[19] Ong, C.-H. L. Observational equivalence of third-order Idealized Algol is decidable. In *LICS '02: Proceedings of the 17th Annual IEEE Symposium on Logic in Computer Science* (Copenhagen, Denmark, 2002), IEEE Computer Society Press, pp. 245–256.

[20] Pitts, A. M. Nominal logic, a first order theory of names and binding. *Information and Computation 186* (2003), 165–193.

[21] Pitts, A. M., and Stark, I. D. B. Observable properties of higher order functions that dynamically create local names, or: What's new? In *MFCS '93: Proceedings of 18th International Symposium on Mathematical Foundations of Computer Science* (Gdańsk, Poland, 1993), vol. 711 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, pp. 122–141.

[22] Stark, I. D. B. *Names and Higher-Order Functions*. PhD thesis, University of Cambridge, Dec. 1994. Also available as Technical Report 363, University of Cambridge Computer Laboratory.

[23] Tzevelekos, N. Full abstraction for nominal general references. In *LICS '07: Proceedings of the 22nd Annual IEEE Symposium on Logic in Computer Science* (Wroclaw, Poland, 2007), IEEE Computer Society Press, pp. 399–410.

[24] Tzevelekos, N. *Nominal game semantics*. DPhil thesis, Oxford University, 2008. Submitted. Available at: `http://web.comlab.ox.ac.uk/people/Nikos.Tzevelekos/Thesis.pdf`.