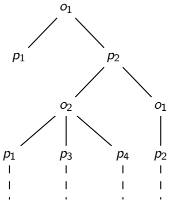
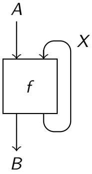
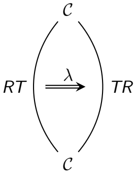
Games, traces and distributive laws







Tamara von Glehn

j.w.w. Martin Hyland
University of Cambridge

BLC, 9 September 2017

Motivation

- Game semantics is a way of modelling computation as a two-player game between a program and its environment.
- Crucially, games and strategies are compositional. But the mathematical details are quite complicated.
- Is there a more natural way to understand strategy composition?

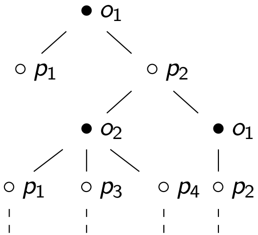
Talk outline

- Simple categories of games
- From simple games to traces
- More complex categories of games
- Categorical tools - comonads and distributive laws
- From games to traces

Simple games

A game A consists of:

- Two sets P_A, O_A of Player moves and Opponent moves
- A tree T_A of moves from P_A and O_A specifying the allowable plays, e.g.



Opponent starts and then play alternates.

Simple games

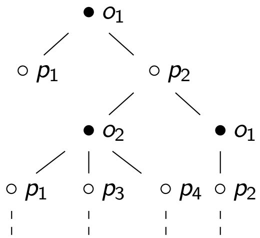
A history-free strategy σ for Player in game A is a choice of move at each point in the game, based on the previous move.

σ is a partial function

$$O_A \rightharpoonup P_A$$

compatible with the game structure T_A.

e.g. $\sigma(o_1) = p_2, \sigma(o_2) = p_3, \ldots$



Simple games - constructions on games

- The cogame $A^{\perp}$ is A with the roles of Player and Opponent reversed.
  Opponent moves $= P_A$, Player moves $= O_A$.

- In the game $A \otimes B$, the games A and B are played in parallel. Opponent chooses which game to start in.
  Opponent moves $= O_A + O_B$, Player moves $= P_A + P_B$.

- In the game $A \multimap B$, the games $A^{\perp}$ and B are played in parallel. Opponent starts in B, and then Player can choose to switch.
  Opponent moves $= P_A + O_B$, Player moves $= O_A + P_B$.

Simple games - strategies

A strategy σ from game A to game B is a strategy in the game $A \multimap B$, i.e. σ is a partial function

$$P_A + O_B \rightharpoonup O_A + P_B.$$

For example:
'copycat strategy' $A \to A$ is
$id : P_A + O_A \rightharpoonup O_A + P_A$.

$\mathbf{A}^\perp$	$\mathbf{A}$
	$\bullet o_1$

Simple games - strategies

A strategy σ from game A to game B is a strategy in the game $A \multimap B$, i.e. σ is a partial function

$$P_A + O_B \rightharpoonup O_A + P_B.$$

For example:
'copycat strategy' $A \to A$ is
$id : P_A + O_A \rightharpoonup O_A + P_A$.

$$\begin{array}{ccc} \mathbf{A}^{\perp} & & \mathbf{A} \\ & & \\ \circ o_1 & \longleftarrow & \bullet o_1 \end{array}$$

Simple games - strategies

A strategy σ from game A to game B is a strategy in the game $A \multimap B$, i.e. σ is a partial function

$$P_A + O_B \rightharpoonup O_A + P_B.$$

For example:
'copycat strategy' $A \to A$ is
$id : P_A + O_A \rightharpoonup O_A + P_A$.

$$\begin{tikzcd} \mathbf{A}^{\perp} & \mathbf{A} \\ \circ o_1 \arrow[d] & \bullet o_1 \arrow[l] \\ \bullet p_1 & \end{tikzcd}$$

Simple games - strategies

A strategy σ from game A to game B is a strategy in the game $A \multimap B$, i.e. σ is a partial function

$$P_A + O_B \rightharpoonup O_A + P_B.$$

For example:
'copycat strategy' $A \to A$ is
$id : P_A + O_A \rightharpoonup O_A + P_A$.

$$\begin{tikzcd}
\mathbf{A}^{\perp} & & \mathbf{A} \\
\circ o_1 \arrow[d] & & \bullet o_1 \arrow[ll] \\
\bullet p_1 \arrow[rr] & & \circ p_1 \arrow[d, dashed, no head] \\
 & & {}
\end{tikzcd}$$

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$\begin{array}{cccc} \mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\ & & & \bullet c_1 \end{array}$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$
\begin{array}{cccc}
\mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\
 & & \circ b_1 & \xleftarrow{\ \tau\ } \bullet c_1
\end{array}
$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$\begin{array}{cccc} \mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\ & \bullet b_1 & \circ b_1 \xleftarrow{\ \tau\ } & \bullet c_1 \end{array}$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$\begin{tikzcd}
\mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\
 & \bullet b_1 \arrow[d, "\sigma"] & \circ b_1 & \bullet c_1 \arrow[l, "\tau"'] \\
 & \circ b_2 & &
\end{tikzcd}$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$
\begin{tikzcd}
\mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\
 & \bullet b_1 \arrow[d, "\sigma"] & \circ b_1 & \bullet c_1 \arrow[l, "\tau"'] \\
 & \circ b_2 & \bullet b_2 &
\end{tikzcd}
$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$
\begin{array}{ccccc}
\mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & & \mathbf{C} \\
 & & & & \\
 & \bullet b_1 & \circ b_1 & \xleftarrow{\quad\tau\quad} & \bullet c_1 \\
 & \downarrow \sigma & & & \\
 & \circ b_2 & \bullet b_2 & & \\
 & & \downarrow \tau & & \\
 & & \circ b_3 & &
\end{array}
$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:

$$\begin{array}{ccccc}
\mathbf{A}^\perp & \mathbf{B} & \mathbf{B}^\perp & & \mathbf{C} \\
 & \bullet b_1 & \circ b_1 & \xleftarrow{\quad\tau\quad} & \bullet c_1 \\
 & \downarrow \sigma & & & \\
 & \circ b_2 & \bullet b_2 & & \\
 & & \downarrow \tau & & \\
 & \bullet b_3 & \circ b_3 & &
\end{array}$$

Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

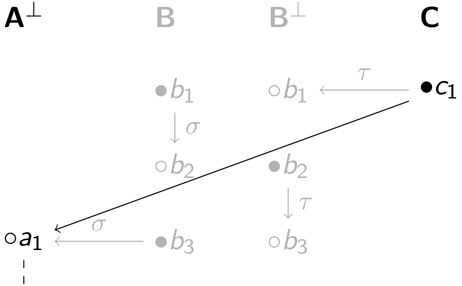
Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by ‘parallel composition plus hiding’. e.g.:

$$
\begin{tikzcd}
\mathbf{A}^\perp & \mathbf{B} & \mathbf{B}^\perp & \mathbf{C} \\
 & \bullet b_1 \arrow[d, "\sigma"] & \circ b_1 & \bullet c_1 \arrow[l, "\tau"'] \\
 & \circ b_2 & \bullet b_2 \arrow[d, "\tau"] & \\
\circ a_1 \arrow[d, dashed, no head] & \bullet b_3 \arrow[l, "\sigma"'] & \circ b_3 & \\
 & & &
\end{tikzcd}
$$

Games and history-free strategies form a symmetric monoidal closed category $\textbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

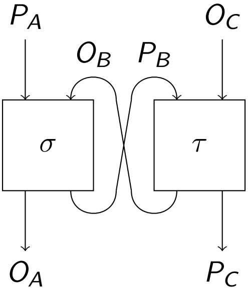
Given $\sigma : A \multimap B$ and $\tau : B \multimap C$, the composite $\tau \cdot \sigma : A \multimap C$ is given by 'parallel composition plus hiding'. e.g.:



Games and history-free strategies form a symmetric monoidal closed category $\mathbf{Games}_{HF}$.
[Abramsky, Jagadeesan, Malacaria 1994].

Simple games - composition of strategies

$\sigma : A \multimap B$ is a map $P_A + O_B \rightharpoonup O_A + P_B$,
$\tau : B \multimap C$ is a map $P_B + O_C \rightharpoonup O_B + P_C$.



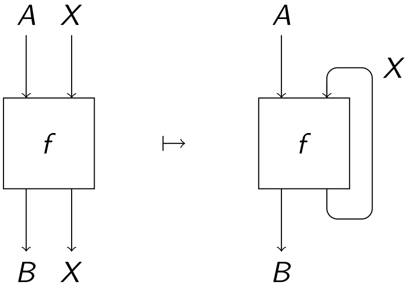
The composite $\tau \cdot \sigma : A \multimap C$ is a map $P_A + O_C \rightharpoonup O_A + P_C$.

Traces

A trace on a monoidal category $(\mathcal{C}, \otimes)$ is a natural family of functions

$$Tr^X_{A,B} : \mathcal{C}(A \otimes X, B \otimes X) \to \mathcal{C}(A, B)$$

satisfying some coherence axioms.

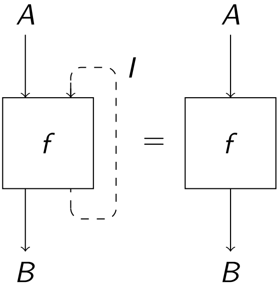


[Joyal, Street, Verity 1996]

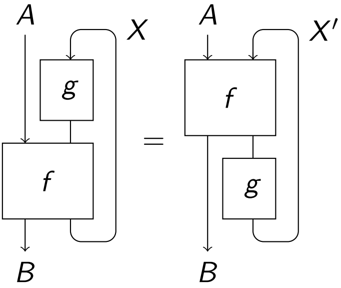
Traces

Coherence axiom examples:

$$Tr^{I}_{A,B}(f) = f$$



$$Tr^{X}_{A,B}((1 \otimes g)f) = Tr^{X'}_{A,B}(f(1 \otimes g))$$



Traces

Examples:

- In the category of real vector spaces, if f is a linear map $U \otimes W \to V \otimes W$, then the partial trace $U \to V$ is given by

$$(Tr^W_{U,V}(f))_{i,j} = \Sigma_k f_{i\otimes k, j\otimes k}$$

- In the category **Pfn** of sets and partial functions with $+$ as tensor, the trace of $f : A + X \rightharpoonup B + X$ is

$$Tr(f)(a) = \begin{cases} f^n(a) & \text{if } f^i(a) \in X, i < n \text{ and } f^n(a) \in B \\ & \text{for some } n \\ \perp \text{ (undefined)} & \text{otherwise.} \end{cases}$$

Traces - the Int construction

For any traced monoidal category $(\mathcal{C}, \otimes)$, there is an associated category $Int(\mathcal{C})$ with composition given by the trace in $\mathcal{C}$.

$Int(\mathcal{C})$ is the free compact closed category on $\mathcal{C}$:
A monoidal category is compact closed if every object A has a dual A^* with unit $\eta : I \to A^* \otimes A$ and counit $\epsilon : A \otimes A^* \to I$, satisfying some axioms.

Examples:

- Finite dimensional vector spaces
- Sets and relations with $\times$ as tensor, where $A^* = A$
- **Pfn** is not compact closed

Traces - the Int construction

$Int(\mathcal{C})$ is the free compact closed category on $\mathcal{C}$.

A compact closed category is always monoidal closed and has a canonical trace:

$$Tr^X_{A,B}(f) = A \xrightarrow{1\otimes\eta} A \otimes X \otimes X^* \xrightarrow{f\otimes 1} B \otimes X \otimes X^* \xrightarrow{1\otimes\epsilon} B$$

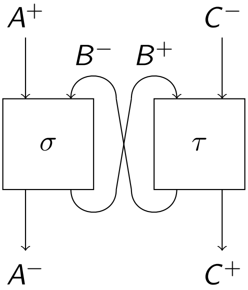
Compact closed categories form an abstract setting for modelling possibly non-terminating computation.
e.g. Geometry of Interaction for linear logic, quantum operators

[Girard 1989], [Abramsky, Haghverdi, Scott 2002]

Traces - the Int construction

In the category $Int(\mathcal{C})$:

- objects are pairs (A^+, A^-) of objects in $\mathcal{C}$
- morphisms $A \to B$ are morphisms $A^+ \otimes B^- \to A^- \otimes B^+$ in $\mathcal{C}$
- composition $A \xrightarrow{\sigma} B \xrightarrow{\tau} C$ is given by tracing out $B^- \otimes B^+$.



From games to traces

Composition of strategies is given by a trace.

Category of games:		Abstract category for computation:
$\mathbf{Games}_{HF}$	$\longrightarrow$	$Int(\mathbf{Pfn})$
A	$\longmapsto$	(P_A, O_A)
$A \xrightarrow{\sigma} B$	$\longmapsto$	$P_A + O_B \xrightarrow{\sigma} O_A + P_B$

This functor is faithful and preserves the monoidal closed structure.

History-sensitive strategies

A history-free strategy determines Player's moves from the previous move.

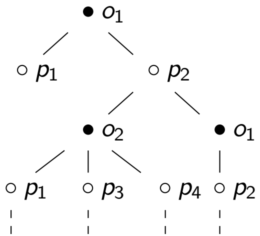
A history-sensitive strategy σ determines Player's moves from all the moves so far.

σ is a partial function

$$L(O_A) \rightharpoonup P_A$$

compatible with the game structure T_A, where $L(O_A)$ is the set of lists of Opponent moves.

e.g. $\sigma([o_1]) = p_2, \sigma([o_1, o_2]) = p_3, \ldots$



History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$\begin{array}{cccc} \mathbf{A}^\perp & \mathbf{B} & \mathbf{B}^\perp & \mathbf{C} \\ & & & \bullet c_1 \end{array}$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$\begin{array}{cccc} \mathbf{A}^\perp & \mathbf{B} & \mathbf{B}^\perp & \mathbf{C} \\ & & \circ b_1 & \xleftarrow{\quad\tau\quad} \bullet c_1 \end{array}$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$\begin{array}{cccc} \mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\ & \bullet b_1 & \circ b_1 \xleftarrow{\tau} & \bullet c_1 \end{array}$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$\begin{array}{cccc} \mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & \mathbf{C} \\ & & & \\ & \bullet b_1 & \circ b_1 & \xleftarrow{\quad\tau\quad} \bullet c_1 \\ & \downarrow \sigma & & \\ & \circ b_2 & & \end{array}$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$
\begin{tikzcd}
\mathbf{A}^\perp & \mathbf{B} & \mathbf{B}^\perp & \mathbf{C} \\
 & \bullet b_1 \arrow[d, "\sigma"] & \circ b_1 & \bullet c_1 \arrow[l, "\tau"'] \\
 & \circ b_2 & \bullet b_2 &
\end{tikzcd}
$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$
\begin{array}{ccccc}
\mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & & \mathbf{C} \\
 & & & & \\
 & \bullet b_1 & \circ b_1 & \xleftarrow{\quad\tau\quad} & \bullet c_1 \\
 & \downarrow \sigma & & & \\
 & \circ b_2 & \bullet b_2 & & \\
 & & \downarrow \tau & & \\
 & & \circ b_3 & &
\end{array}
$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$
\begin{array}{ccccc}
\mathbf{A}^{\perp} & \mathbf{B} & \mathbf{B}^{\perp} & & \mathbf{C} \\
 & \bullet b_1 & \circ b_1 & \xleftarrow{\quad\tau\quad} & \bullet c_1 \\
 & \downarrow \sigma & & & \\
 & \circ b_2 & \bullet b_2 & & \\
 & & \downarrow \tau & & \\
 & \bullet b_3 & \circ b_3 & &
\end{array}
$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

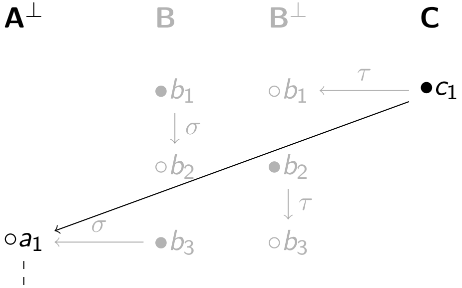
Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:

$$
\begin{array}{cccccc}
\mathbf{A}^{\perp} & & \mathbf{B} & \mathbf{B}^{\perp} & & \mathbf{C} \\
 & & \bullet b_1 & \circ b_1 & \xleftarrow{\quad\tau\quad} & \bullet c_1 \\
 & & \downarrow \sigma & & & \\
 & & \circ b_2 & \bullet b_2 & & \\
 & & & \downarrow \tau & & \\
\circ a_1 & \xleftarrow{\quad\sigma\quad} & \bullet b_3 & \circ b_3 & & \\
\vdots & & & & &
\end{array}
$$

where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

History-sensitive strategies

Composition of history-sensitive strategies $\sigma : A \multimap B$ and $\tau : B \multimap C$ is still parallel composition plus hiding. e.g.:



where $\tau([c_1]) = b_1$, $\sigma([b_1]) = b_2$, $\tau([c_1, b_1, b_2]) = b_3$, etc

From games to traces

Games and history-free strategies form a symmetric monoidal category $\mathbf{Games}_{HS}$.

$$
\begin{array}{ccc}
\text{Category of games:} & & \text{Category for computation:} \\
\mathbf{Games}_{HF} & \longrightarrow & \mathit{Int}(\mathbf{Pfn}) \\
\mathbf{Games}_{HS} & \longrightarrow & \mathit{Int}(?)
\end{array}
$$

Distributive laws - the list comonad

Let $L(A)$ be the set of non-empty lists with elements in A.

$$L(A) = \mu X.(A + A \times X) \quad \text{(least fixed point)}$$

L is a comonad on **Set**.
The counit ε gives the head of the list:

$$\begin{aligned} L(A) &\xrightarrow{\varepsilon_A} A \\ [a_1, \ldots, a_n] &\mapsto a_n \end{aligned}$$

The comultiplication δ gives the list of prefixes:

$$\begin{aligned} L(A) &\xrightarrow{\delta_A} LL(A) \\ [a_1, \ldots, a_n] &\mapsto [[a_1], [a_1, a_2], \ldots, [a_1, \ldots, a_n]] \end{aligned}$$

Distributive laws

History-sensitive strategies are partial functions $L(A) \rightharpoonup B$.

We have:

$$\begin{tikzcd} \mathbf{Set} \arrow[loop above, "\text{Monad } (-+1)"] \arrow[loop below, "\text{Comonad } L"'] \end{tikzcd}$$

- The coKleisli category for the comonad L has as objects sets and as morphisms total functions $L(A) \to B$.

- The Kleisli category **Pfn** for the lifting monad $X \mapsto X + 1$ has as objects sets and as morphisms partial functions $A \rightharpoonup B$.

Distributive laws

Let R be a comonad and T a monad on a category $\mathcal{C}$.

R extends to a comonad on $Kl(T)$ iff there is a distributive law of R over T,
i.e. if there exists $\lambda : RT \Rightarrow TR$ compatible with the monad and comonad structure.

$$\begin{tikzcd} RTA \arrow[r, "\lambda_A"] \arrow[d, "\varepsilon_T"'] & TRA \arrow[dl, "T\varepsilon"] \\ TA & \end{tikzcd} \qquad \begin{tikzcd} & RA \arrow[dl, "R\eta"'] \arrow[d, "\eta_R"] \\ RTA \arrow[r, "\lambda_A"'] & TRA \end{tikzcd}$$ etc

Distributive laws

There is no distributive law of the list comonad L over the monad $T = (- + 1)$.

There is a natural transformation $\lambda : LT \Rightarrow TL$:

$$L(A+1) \xrightarrow{\lambda_A} L(A) + 1$$

$$[a_1, \ldots, a_n] \mapsto \begin{cases} [a_1, \ldots, a_n] & \text{if all } a_i \in A \\ \perp & \text{otherwise.} \end{cases}$$

But λ is not compatible with the counit ε:

$$\begin{tikzcd} L(A+1) \arrow[r, "\lambda_A"] \arrow[d, "\varepsilon"'] \arrow[dr, phantom, "\neq"] & L(A)+1 \arrow[dl, "\varepsilon+1"] \\ A+1 & \end{tikzcd}$$

e.g. $\lambda_A([a_1, \perp]) = \perp \neq a_1$.

Distributive laws - near-comonads

A near-comonad is a endofunctor R on $\mathcal{C}$ with

- a natural transformation $\delta : R \to RR$
- a (not necessarily natural) family of morphisms $\{\varepsilon_A : RA \to A\}_{A \in \mathcal{C}}$

satisfying the usual axioms of a comonad.

A near-comonad R has a near-coKleisli category $coKl(R)$ with

- objects the same as objects of $\mathcal{C}$
- morphisms from A to B those morphisms $R(A) \xrightarrow{f} B$ in $\mathcal{C}$ satisfying $f = \varepsilon_B \circ Rf \circ \delta_A$.

[Hyland, Nagayama, Power, Rosolini 2006]

Distributive laws - near-comonads

A near-distributive law is a natural transformation satisfying all the axioms of a distributive law except for compatibility with the counit.

The near-distributive law

$$L(A+1) \xrightarrow{\lambda_A} L(A)+1$$

gives an extension of L to a near-comonad $\tilde{L}$ on $Kl(-+1)$.

The near-comonad $\tilde{L}$ has a near-coKleisli category $coKl(\tilde{L})$ with

- objects sets
- morphisms from A to B partial functions $f : L(A) \rightharpoonup B$ such that if f is defined on a list then it is defined on all prefixes.

Distributive laws - monoidal structure

The comonad L is compatible with the tensor $+$ on **Set**:
There is a natural transformation

$$L(A+B) \to L(A) + L(B)$$
$$[x_1, \ldots, x_n] \mapsto \begin{cases} [x_i, \ldots, x_n] \in L(A) \text{ if } x_n \in A \\ [x_i, \ldots, x_n] \in L(B) \text{ if } x_n \in B \end{cases}$$

which commutes with the counit and comultiplication.

This gives $coKl(\tilde{L})$ the structure of a symmetric monoidal category with $+$ as tensor.

The trace on **Pfn** induces a trace on $coKl(\tilde{L})$.

From games to traces

Category of games: Abstract category for computation:

$$\begin{array}{ccc}
\mathbf{Games}_{HS} & \longrightarrow & Int(\mathrm{coKl}(\tilde{L})) \\
A & \longmapsto & (P_A, O_A) \\
A \xrightarrow{\sigma} B & \longmapsto & L(P_A + O_B) \xrightarrow{\sigma} O_A + P_B
\end{array}$$

This functor is faithful and preserves the monoidal closed structure.

From games to traces - generalising

The above construction generalises.
Given:

- a monad T corresponding to a class of partial maps,
- a comonad R defined by a least fixed point which is compatible with T and the monoidal structure,

then R extends to a near-comonad $\tilde{R}$ on $Kl(T)$.

The near-coKleisli category $coKl(\tilde{R})$ is a symmetric monoidal category, and the trace on $Kl(T)$ induces a trace on $coKl(\tilde{R})$.

Finally, we get a compact closed category $Int(\text{coKl}(\tilde{R}))$.

From games to traces - generalising

Example:

$$\text{Instead of} \quad L(A) = \mu X.A \times (1 + X),$$
$$\text{use} \quad R(A) = \mu X.A \times \mathcal{P}_f(X).$$

Category of games: Category for computation:

$$\mathbf{Games}_{HF} \longrightarrow \mathit{Int}(\mathbf{Pfn})$$

$$\mathbf{Games}_{HS} \longrightarrow \mathit{Int}(\mathrm{coKl}(\tilde{L}))$$

$$? \longrightarrow \mathit{Int}(\mathrm{coKl}(\tilde{R}))$$

From games to traces - generalising

$$R(A) = \mu X.A \times \mathcal{P}_f(X)$$

An element of $R(A)$ is a finite rooted tree of elements of A.

If a strategy is represented by some partial map $R(A) \rightharpoonup B$ then the next move will depend on a partially ordered set of previous plays, not a list. Moves might be played concurrently rather than sequentially.

This has similarities to the category of concurrent games.

From games to traces - generalising

A concurrent game E consists of

- Two sets P_E and O_E of Player moves and Opponent moves
- A partial order $\leq$ on moves $P_E + O_E$ specifying the prerequisites for a move to be played
- A consistency predicate on finite sets of moves in $P_E + O_E$ specifying which moves may occur together

satisfying some axioms.

A strategy in a concurrent game E is given by another game S and a map $S \rightarrow E$ which preserves downward-closed consistent sets and is locally injective.

[Castellan, Clairambault, Rideau, Winskel 2016]

From games to traces - generalising

Composition of strategies is given by a trace.

Category of games:		Category for computation:
$\mathbf{Games}_{HF}$	$\longrightarrow$	$Int(\mathbf{Pfn})$
$\mathbf{Games}_{HS}$	$\longrightarrow$	$Int(\mathrm{coKl}(\tilde{L}))$
Simple concurrent games	$\longrightarrow$	$Int(\mathrm{coKl}(\tilde{R}))$
$\vdots$		$\vdots$

Questions and future work

- The functor from games to a compact closed category appears to lose some of the game structure. How much of it can be recovered?
- Near-monads and near-comonads arise in other situations, but their general theory is not well-understood.
- What other categories of games can be described this way? What is the right abstract notion of a category of games?