Asynchronous Template Games and the Gray Tensor Product of 2-Categories

Paul-André Melliès

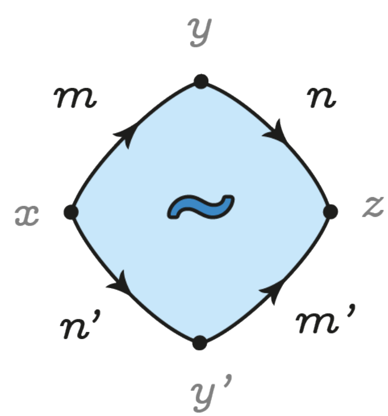
Institut de Recherche en Informatique Fondamentale (IRIF)
CNRS & Université de Paris

ACM/IEEE LICS 2021 ⚓ 29 June → 02 July 2021

Main purpose

We want to combine two styles of game semantics:

- ▷ **template games** played on **categories** of positions and trajectories,
- ▷ **asynchronous games** played on **graphs** with **permutation tiles**



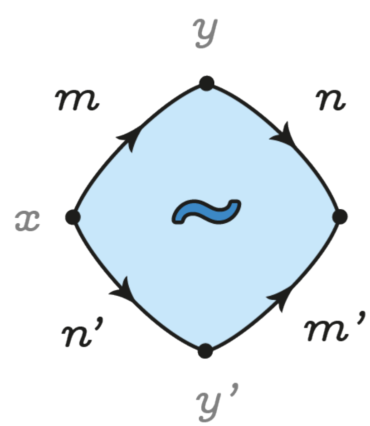
indicating when two moves m and n are **independent** in the game.

Asynchronous graphs

An **asynchronous graph** is defined as a graph

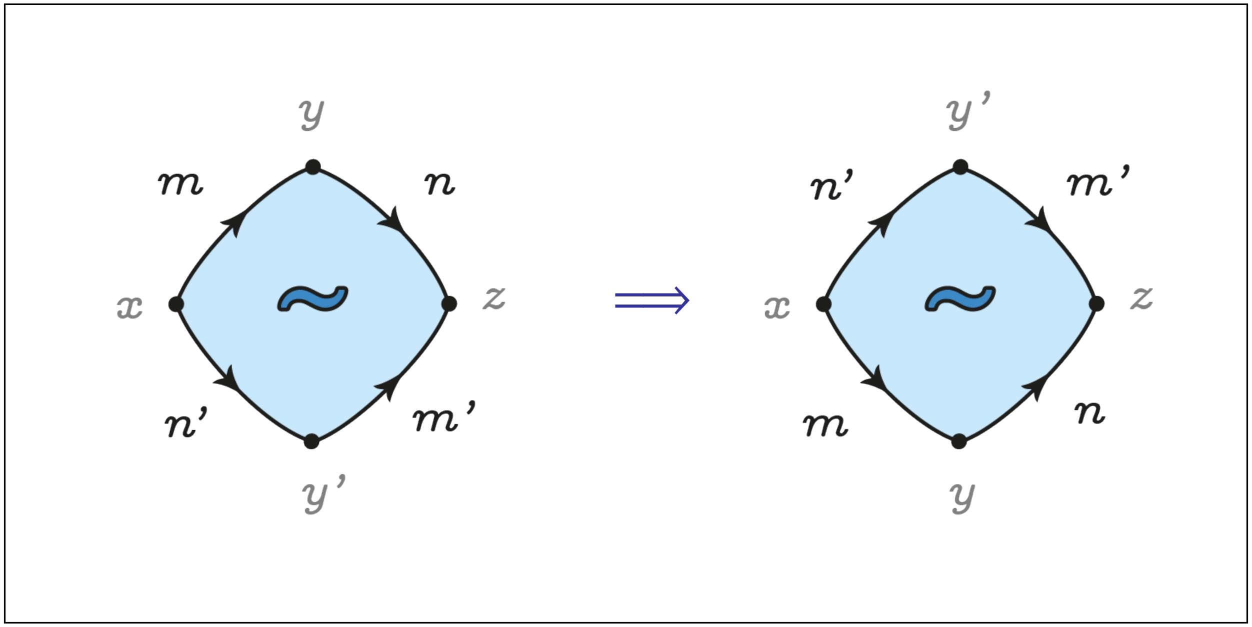
$$G = (V, E)$$

equipped with a set of **permutation tiles** of the form

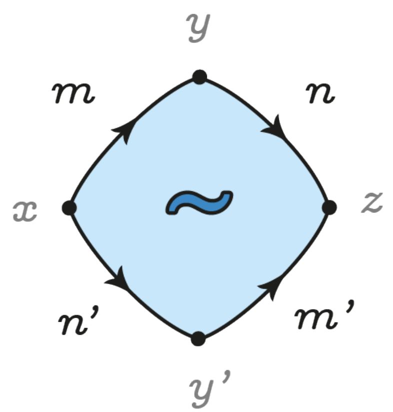


between **coinitial** and **cofinal** paths of length 2.

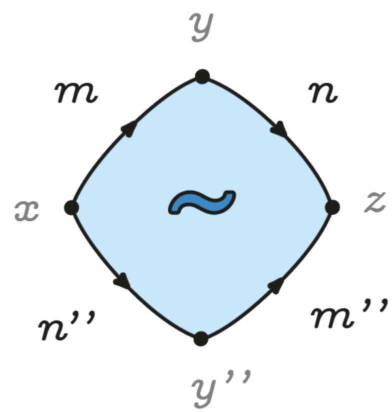
— Axiom 1 —
All permutations are symmetric



— Axiom 2 —
All permutations are deterministic

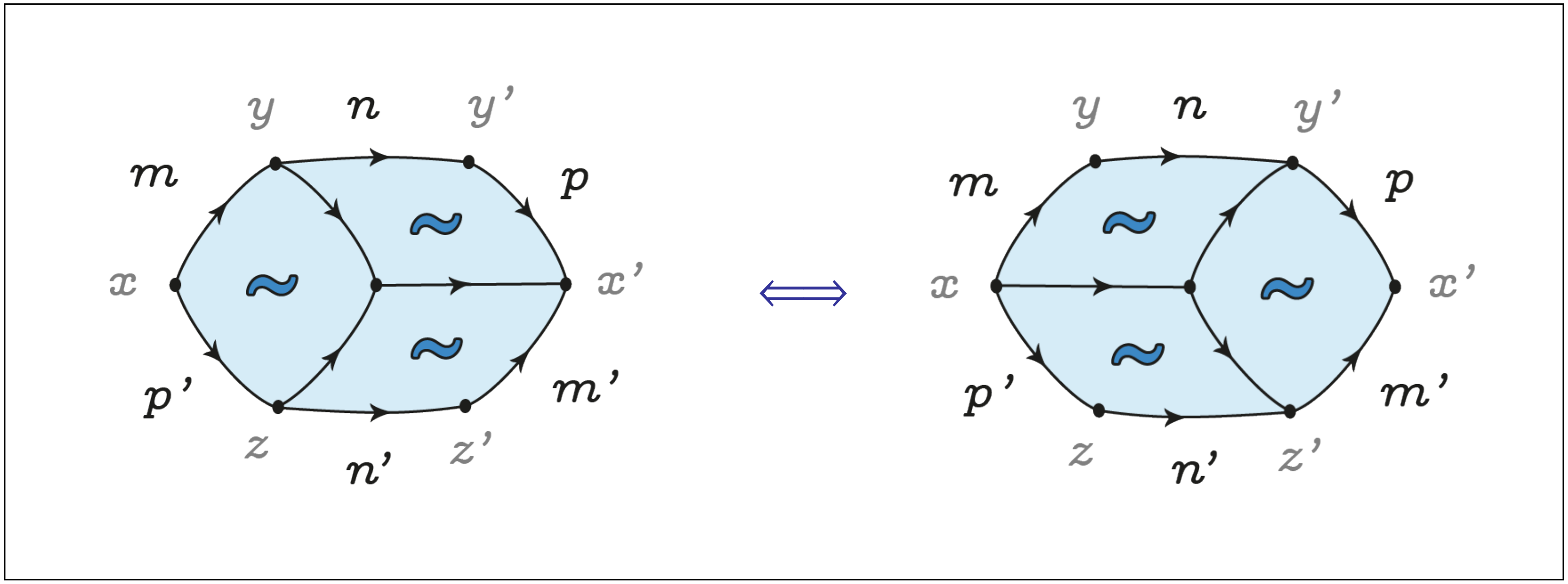


and



$\Longrightarrow \quad n' \cdot m' = n'' \cdot m''$

— Axiom 3 —
The cube axiom



The shuffle tensor product

The **shuffle tensor product**

$$G \sqcup\!\sqcup H \quad = \quad (G \sqcup\!\sqcup H, \diamond_{G \sqcup\!\sqcup H})$$

of two asynchronous graphs

$$G = (G, \diamond_G) \qquad\qquad H = (H, \diamond_H)$$

is the asynchronous graph

- whose vertices (x, y) are the pairs of vertices $x \in G$ and $y \in H$,

The shuffle tensor product

- whose edges are of two kinds: the pairs

$$(x, y) \xrightarrow{(u,y)} (x', y)$$

consisting of an edge in the graph G

$$x \xrightarrow{\quad u \quad} x'$$

and of a vertex $y \in H$; and pairs

$$(x, y) \xrightarrow{(x,v)} (x, y')$$

consisting of an edge in the graph H

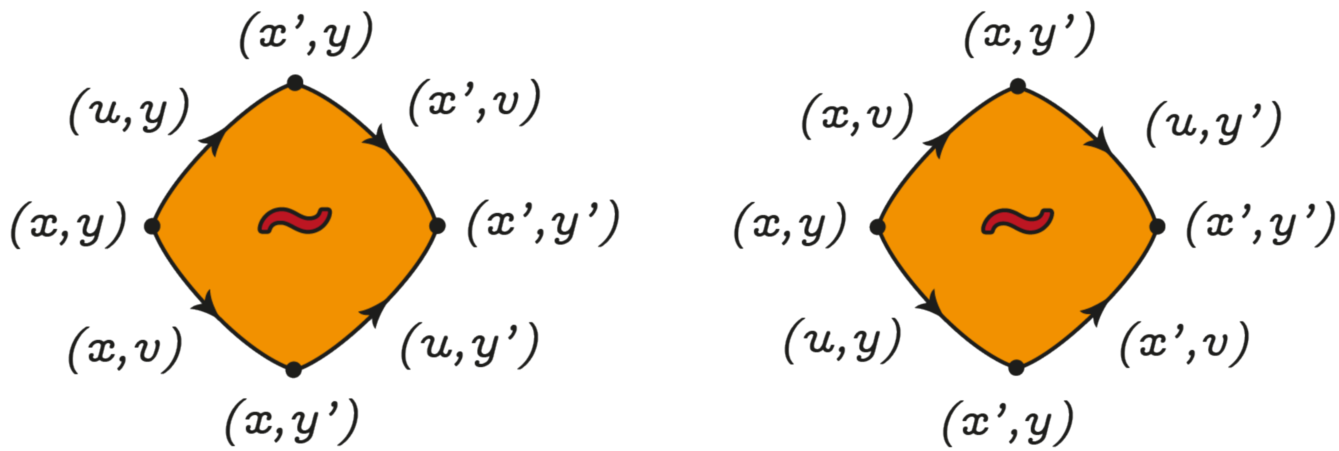
$$y \xrightarrow{\quad v \quad} y'$$

and of a vertex $x \in G$.

The shuffle tensor product

- whose permutation tiles are of three kinds:

1. two permutation tiles



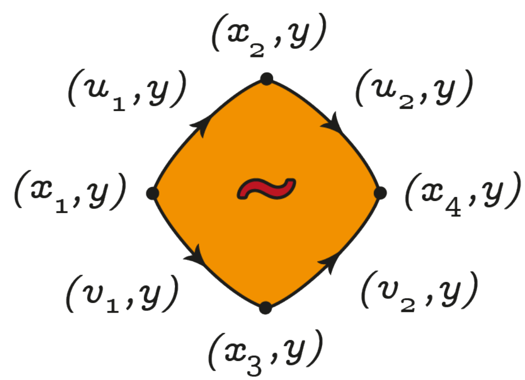
for every pair of edges

$$x \xrightarrow{u} x' \qquad\qquad y \xrightarrow{v} y'$$

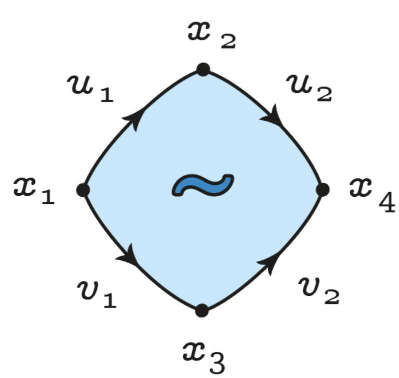
in the graphs G and H respectively ;

The shuffle tensor product

2. a permutation tile



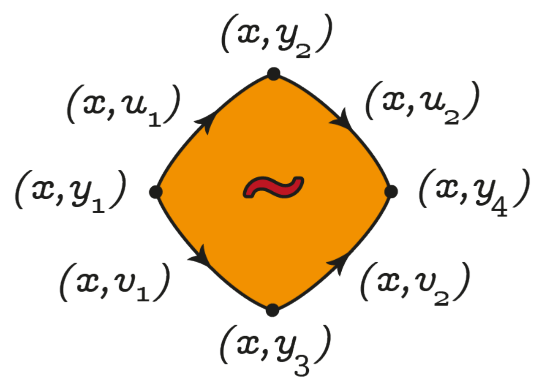
for every permutation tile



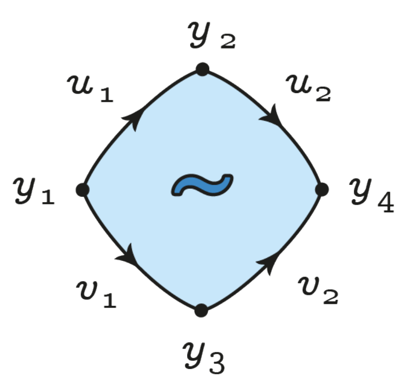
in the asynchronous graph G and every vertex $y \in H$;

The shuffle tensor product

3. a permutation tile



for every permutation tile



in the asynchronous graph H and every vertex $x \in G$.

The category of asynchronous graphs

The category **Asynch** of asynchronous graphs has its morphisms

$$f \quad : \quad (G, \diamond_G) \longrightarrow (H, \diamond_H)$$

graph homomorphisms

$$f \quad : \quad G \longrightarrow H$$

transporting every permutation tile of G to a permutation tile of H.

Theorem. The shuffle tensor product

$$G, H \quad \mapsto \quad G \sqcup\!\sqcup H \quad : \quad \textbf{Asynch} \times \textbf{Asynch} \longrightarrow \textbf{Asynch}$$

turns the category **Asynch** into a **symmetric monoidal category.**

Basic illustration

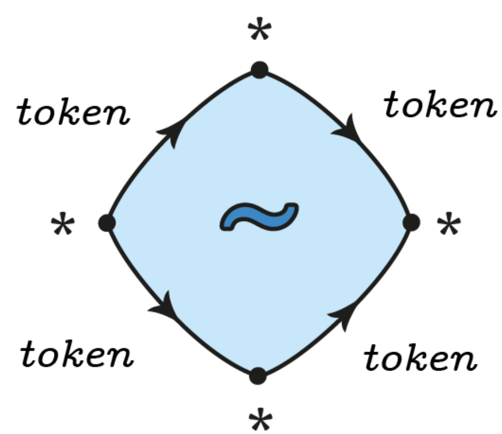
For every label *token*, we define the asynchronous graph

$$\underline{+}[token]$$

with a unique vertex $*$ and a unique edge

$$token \quad : \quad * \longrightarrow *$$

together with a unique permutation tile



The template of games

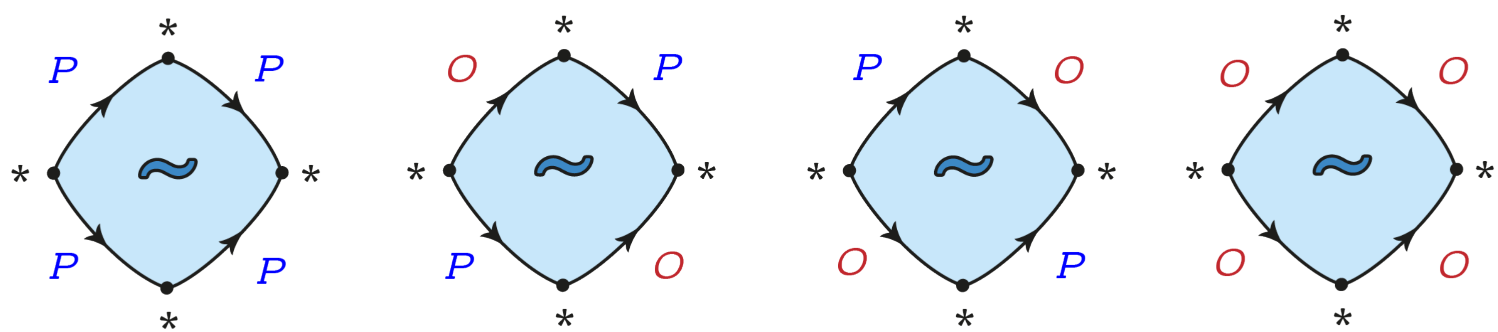
The template of games $\pm_{\mathrm{game}}$ is the asynchronous graph

$$\pm[O,P] \quad = \quad \pm[O] \sqcup \pm[P]$$

with one unique vertex $*$ and two edges

$$O, P \quad : \quad * \longrightarrow *$$

together with four permutation tiles



expressing that all edges in $\pm[O,P]$ are pairwise independent.

Asynchronous games

By definition, an **asynchronous game**

$$(A, \lambda_A)$$

is an asynchronous graph equipped with a **polarity map**

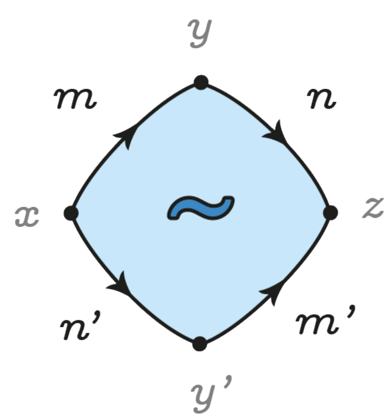
$$\lambda_A \quad : \quad (A, \diamond_A) \longrightarrow \pm_{\text{game}}$$

What this means...

The polarity map assigns a polarity O or P to each edge of the graph:

- an **Opponent move** $m : x \to y$ is an edge mapped to the polarity O ,
- a **Player move** $m : x \to y$ is an edge mapped to the polarity P .

Polarities of moves are moreover preserved by permutation tiles:



in the sense that $\lambda_A(m) = \lambda_A(m')$ and that $\lambda_A(n) = \lambda_A(n')$.

Asynchronous games as 2-categories

At this stage, we revisit one main principle of template games:

think of games as categories with polarities

which we lift one dimension up to the following mantra:

think of asynchronous games as 2-categories with polarities

Asynchronous games as 2-categories

To that purpose, we use the basic observation that

every asynchronous graph $(G, \diamond_G)$ **generates a 2-category** $\langle G, \diamond_G \rangle$

The 2-category $\langle G, \diamond_G \rangle$ is defined in the following way:

- its objects = the vertices of the graph,
- its morphisms = the paths of the graph,
- its 2-cells = the **reshufflings** induced by the permutation tiles.

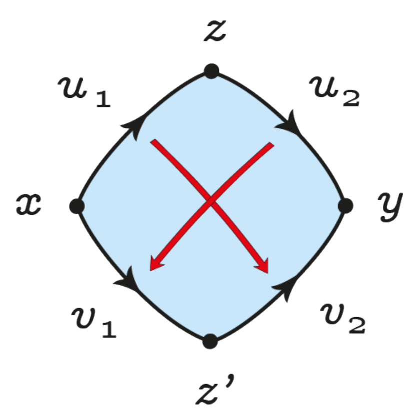
Reshufflings between paths

Definition: a **reshuffling** is a **bijective function**

$$\varphi \quad : \quad \{1, ..., n\} \quad \longrightarrow \quad \{1, ..., n\}$$

which "keeps track" of a sequence of tiles on a path of length n.

Typically, the reshuffling $\begin{pmatrix} 1 \mapsto 2 \\ 2 \mapsto 1 \end{pmatrix}$ is associated to any permutation tile:

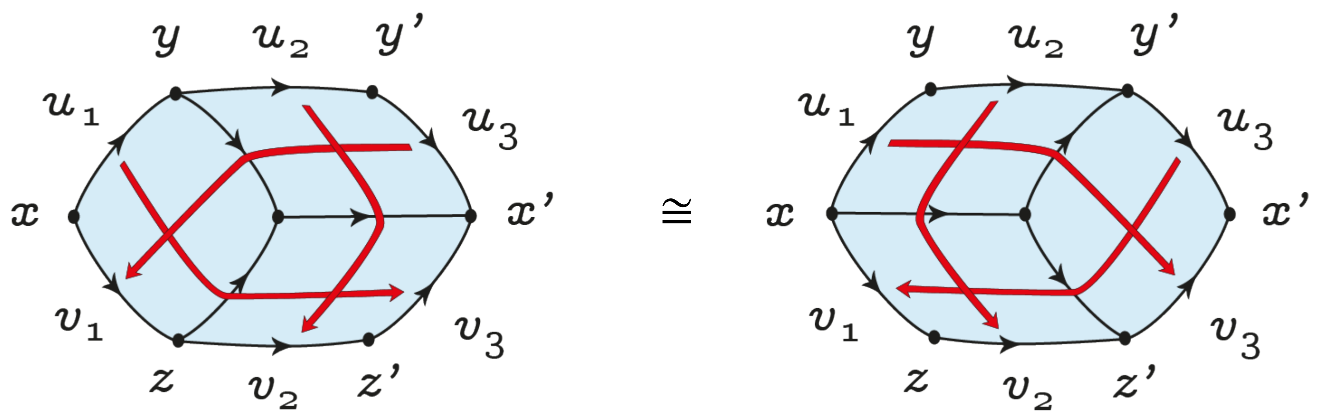


Reshufflings between paths

Similarly, the reshuffling on three indices

$$\begin{pmatrix} 1 \mapsto 3 \\ 2 \mapsto 2 \\ 3 \mapsto 1 \end{pmatrix} \quad : \quad \{1,2,3\} \quad \longrightarrow \quad \{1,2,3\}$$

keeps track and identifies the two sequences of tiles:



Related to the braid equation and the Yang-Baxter equation

From asynchronous graphs to 2-categories, functorially...

The translation induces a functor

$$\langle - \rangle \quad : \quad \textbf{Asynch} \longrightarrow \textbf{TwoCat}$$

where **TwoCat** is the category of 2-categories and 2-functors.

Key observation:

The functor $\langle - \rangle$ comes equipped with a family of isomorphisms

$$\langle G \sqcup H \rangle \;\cong\; \langle G \rangle \boxtimes \langle H \rangle \qquad\qquad \langle \mathbf{I} \rangle \;\cong\; \mathbf{1}$$

and thus defines a symmetric monoidal functor

$$\langle - \rangle \quad : \quad (\textbf{Asynch}, \sqcup, \mathbf{I}) \longrightarrow (\textbf{TwoCat}, \boxtimes, \mathbf{1})$$

where we write $\boxtimes$ for the **Gray tensor product** of 2-categories.

Polarities and Gray comonoids

Here, something like an algebraic miracle occurs!

Big surprise and main observation of the paper:

Fact. A **polarity structure** on an asynchronous graph $(A, \diamond_A)$

$$\lambda_A \; : \; E_A \longrightarrow \{O, P\}$$

is the same thing as a **Gray comonoid structure**

$$\text{comult} \; : \; \langle A, \diamond_A \rangle \longrightarrow \langle A, \diamond_A \rangle \boxtimes \langle A, \diamond_A \rangle$$

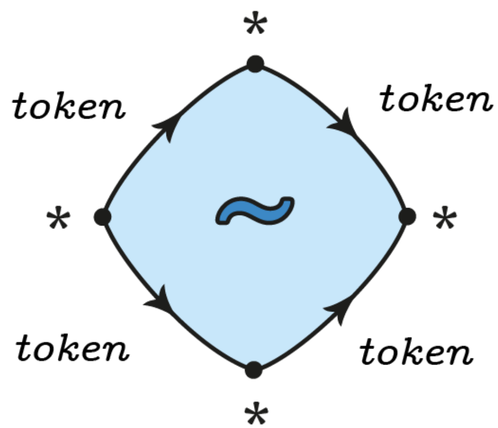
on the associated 2-category $\langle A, \diamond_A \rangle$.

Illustration

Typically, the 2-category

$$\pm\{\!\!|\, token \,|\!\!\} \quad = \quad \langle\, \pm[token]\,\rangle$$

generated by the asynchronous graph



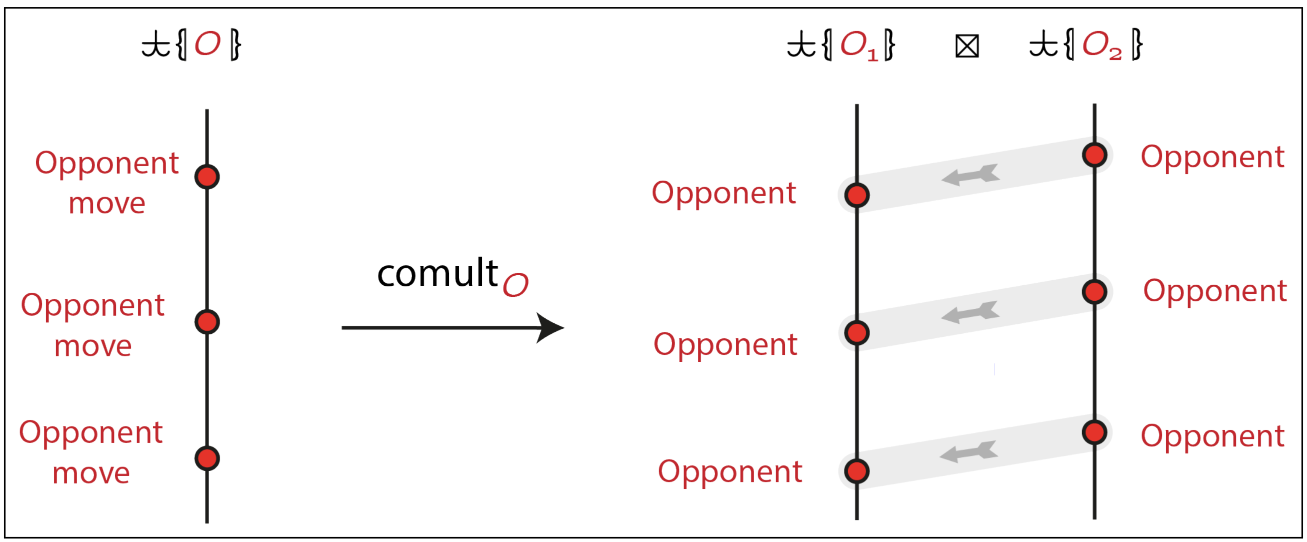
has exactly two Gray comonoid structures, noted $\pm\{\!\!|\, O \,|\!\!\}$ and $\pm\{\!\!|\, P \,|\!\!\}$.

The negative Gray comonoid

The comultiplication of the Gray comonoid

$$\mathsf{comult}_{O} \quad : \quad \pitchfork\{\!\!|\, O \,|\!\!\} \longrightarrow \pitchfork\{\!\!|\, O_1 \,|\!\!\} \boxtimes \pitchfork\{\!\!|\, O_2 \,|\!\!\}$$

transports the unique edge O to the path $O_2 \cdot O_1$ of length 2:



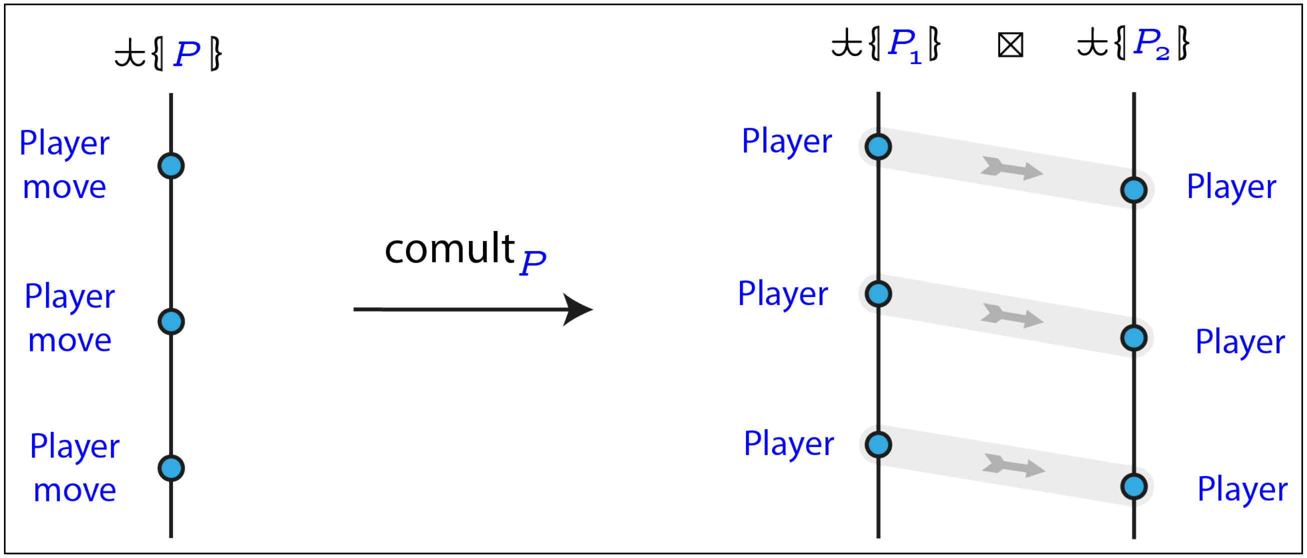
This is the scheduling of an **Opponent move** in a copycat strategy.

The positive Gray comonoid

The comultiplication of the Gray comonoid

$$\text{comult}_P \quad : \quad ⅄\{\!\!| P |\!\!\} \longrightarrow ⅄\{\!\!| P_1 |\!\!\} \boxtimes ⅄\{\!\!| P_2 |\!\!\}$$

transports the unique edge P to the path $P_1 \cdot P_2$ of length 2.



This is the scheduling of a **Player move** in a copycat strategy.

The mixed Gray comonoid

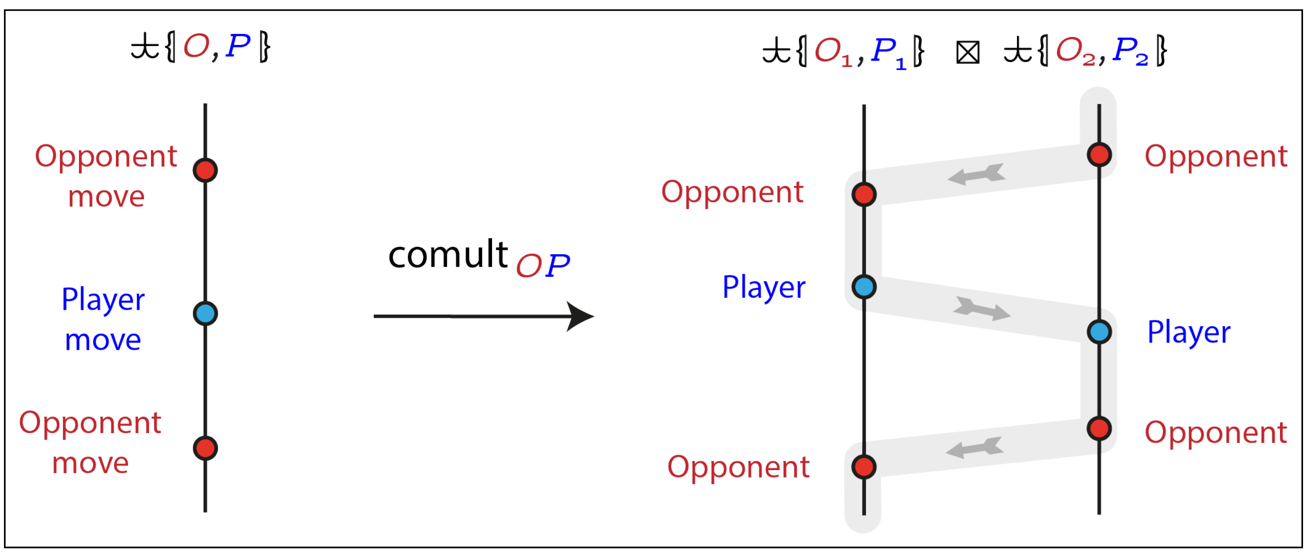
By tensoring the two Gray comonoids

$$\pitchfork\{\!|\, O, P \,|\!\} \quad = \quad \pitchfork\{\!|\, O \,|\!\} \boxtimes \pitchfork\{\!|\, P \,|\!\}$$

one obtains the Gray comonoid whose comultiplication

$$\text{comult}_{OP} \quad : \quad \pitchfork\{\!|\, O, P \,|\!\} \longrightarrow \pitchfork\{\!|\, O_1, P_1 \,|\!\} \boxtimes \pitchfork\{\!|\, O_2, P_2 \,|\!\}$$

reflects the "mixed" scheduling of the usual copycat strategy:



Asynchronous template games

This leads us to the following definition:

Definition. An **asynchronous template game** is a Gray comonoid

$$\text{comult} : A \longrightarrow A \boxtimes A \qquad \text{counit} : A \longrightarrow \mathbf{1}$$

equipped with a **polarity map** of Gray comonoids

$$\lambda_A : A \longrightarrow \pm_{\text{game}}$$

where the **template of games** is defined as

$$\pm_{\text{game}} = \pm\{\!\!\{O, P\}\!\!\}$$

Asynchronous strategies

Definition. An **asynchronous strategy**

$$\sigma = (S, \mathsf{coact}_\sigma, \lambda_\sigma) \;:\; (A, \lambda_A) \longrightarrow\!\!\!\!\!\!\!\!+\;\;\; (B, \lambda_B)$$

is a triple consisting of

- ► a 2-category S called the **support** of the strategy,
- ► a bicomodule structure $\mathsf{coact}_\sigma \;:\; S \longrightarrow A \boxtimes S \boxtimes B$
- ► a scheduling 2-functor $\lambda_\sigma \;:\; S \longrightarrow ⅄_{\mathrm{strat}}$

where the **template of strategies** is defined as

$$⅄_{\mathrm{strat}} \quad = \quad ⅄_{\mathrm{game}} \boxtimes ⅄_{\mathrm{game}} \quad = \quad ⅄\{\!|\, O_s\, ,\, P_s\, ,\, O_t\, ,\, P_t \,|\!\}$$

The template of strategies

Here, each of the four labels

$$O_s \qquad P_s \qquad O_t \qquad P_t$$

describes a specific kind of Opponent and Player move

O_s	:	Opponent move	played at	the source game
P_s	:	Player move	played at	the source game
O_t	:	Opponent move	played at	the target game
P_t	:	Player move	played at	the target game

which may appear on the interactive trajectory played by a strategy

$$\sigma \quad : \quad A \longrightarrow\!\!\!\!\!\!\!\!+\quad B.$$

Asynchronous strategies

One requires moreover that the diagram commutes:

$$
\begin{array}{ccc}
S & \xrightarrow{\quad \mathsf{coact}_{\sigma} \quad} & A \boxtimes S \boxtimes B \\
{\scriptstyle \lambda_{\sigma}} \big\downarrow & & \big\downarrow {\scriptstyle \lambda_{A} \boxtimes \lambda_{\sigma} \boxtimes \lambda_{B}} \\
\pitchfork_{\mathrm{strat}} & \xrightarrow{\quad \mathsf{coact}_{\pitchfork} \quad} & \pitchfork_{\mathrm{game}} \boxtimes \pitchfork_{\mathrm{strat}} \boxtimes \pitchfork_{\mathrm{game}}
\end{array}
$$

The diagram says that the **scheduling 2-functor**

$$\lambda_{\sigma} \; : \; S \longrightarrow \pitchfork_{\mathrm{strat}}$$

is a **map of Gray comodules** to the Gray comodule

$$\mathsf{coact}_{\pitchfork} \; : \; \pitchfork_{\mathrm{strat}} \longrightarrow \pitchfork_{\mathrm{strat}} \boxtimes \pitchfork_{\mathrm{game}}$$

Asynchronous strategies

Equivalently, an **asynchronous strategy**

$$\sigma \;=\; (S, \mathsf{coact}_{\sigma}, \lambda_{\sigma}) \;:\; (A, \lambda_A) \longrightarrow\!\!\!\!\!\!\!\!\!\!\!\!\;\;\;+\;\;\;\;\;\; (B, \lambda_B)$$

is a **double cell**

$$\begin{array}{ccccc}
A & \overset{S}{\longrightarrow\!\!\!\!\!\!\!+\;\;\;\;} & & & B \\
\Big\downarrow \lambda_A & & \Big\Downarrow \lambda_{\sigma} & & \Big\downarrow \lambda_B \\
\text{よ}_{\text{game}} & & \underset{\text{よ}_{\text{strat}}}{\longrightarrow\!\!\!\!\!\!\!+\;\;\;\;} & & \text{よ}_{\text{game}}
\end{array}$$

in the double category **BiComod** of **Gray comonoids** and **bicomodules**.

The template of interactions

Key observation: the **template of interactions**

$$\pitchfork_{\mathrm{int}} \quad = \quad \left(\pitchfork_{\mathrm{game}} \xrightarrow{\ \pitchfork_{\mathrm{strat}}\ } \pitchfork_{\mathrm{game}} \xrightarrow{\ \pitchfork_{\mathrm{strat}}\ } \pitchfork_{\mathrm{game}} \right)$$

coincides with the 2-category

$$\pitchfork_{\mathrm{int}} \quad = \quad \pitchfork\{\!\!|\ O_s\ ,\ P_s\ ,\ OP\ ,\ PO\ ,\ O_t\ ,\ P_t\ |\!\!\}$$

generated by the **six polarities** of moves on three games:

O_s	:	Opponent move	played at	the source game	A
P_s	:	Player move	played at	the source game	A
OP	:	internal move	played at	the intermediate game	B
PO	:	internal move	played at	the intermediate game	B
O_t	:	Opponent move	played at	the target game	C
P_t	:	Player move	played at	the target game	C

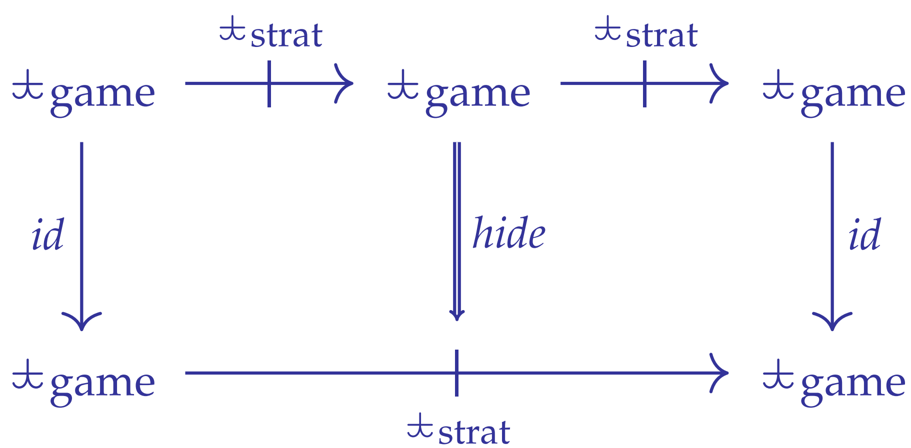
The template of interactions

The template of interactions comes equipped with a 2-functor

$$hide \quad : \quad \mathfrak{t}_{\text{int}} \longrightarrow \mathfrak{t}_{\text{strat}}$$

which hides the internal polarities PO and OP.

This defines a double cell:



in the double category **BiComod**.

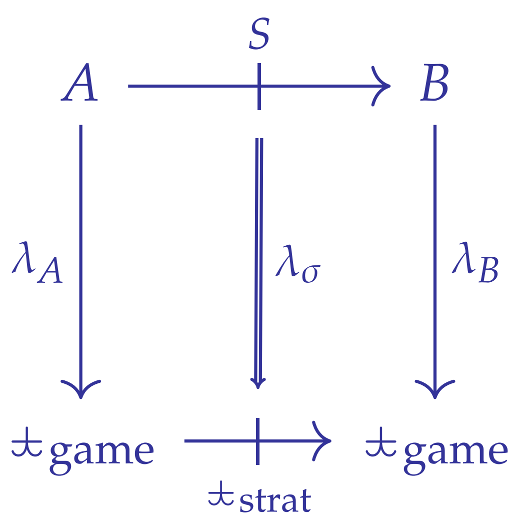
Composition of strategies

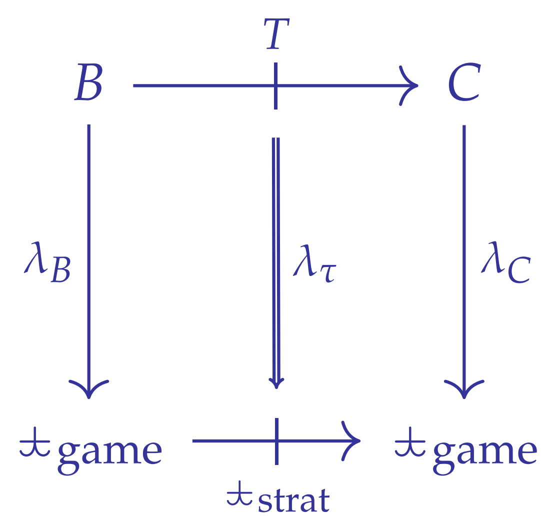
Now, suppose given a pair of **asynchronous strategies**

$$\sigma = (S, \mathsf{coact}_{\sigma}, \lambda_{\sigma}) \;:\; (A, \lambda_A) \longrightarrow\!\!\!\!\!\!\!\!+\;\;\;\; (B, \lambda_B)$$

$$\tau = (T, \mathsf{coact}_{\tau}, \lambda_{\tau}) \;:\; (B, \lambda_B) \longrightarrow\!\!\!\!\!\!\!\!+\;\;\;\; (C, \lambda_C)$$

represented as a pair of double cells:



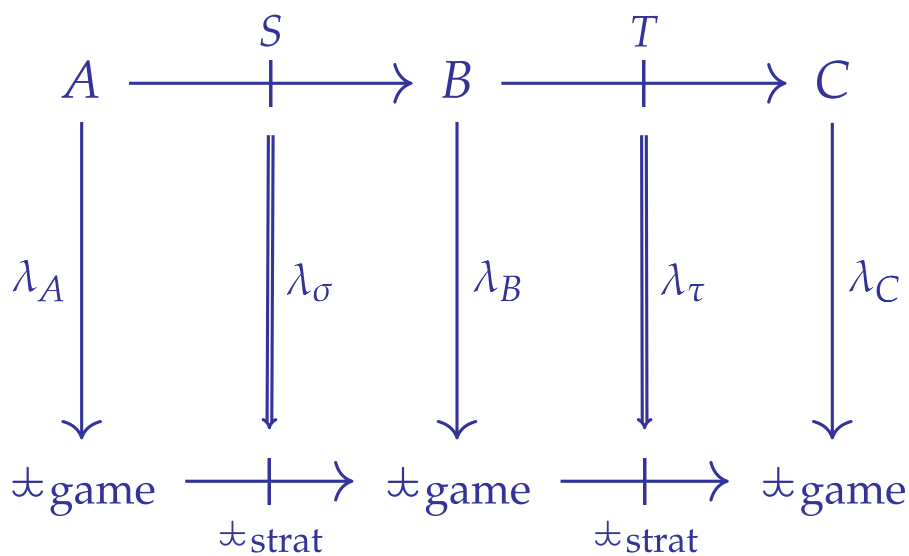


Composition of strategies

The composite

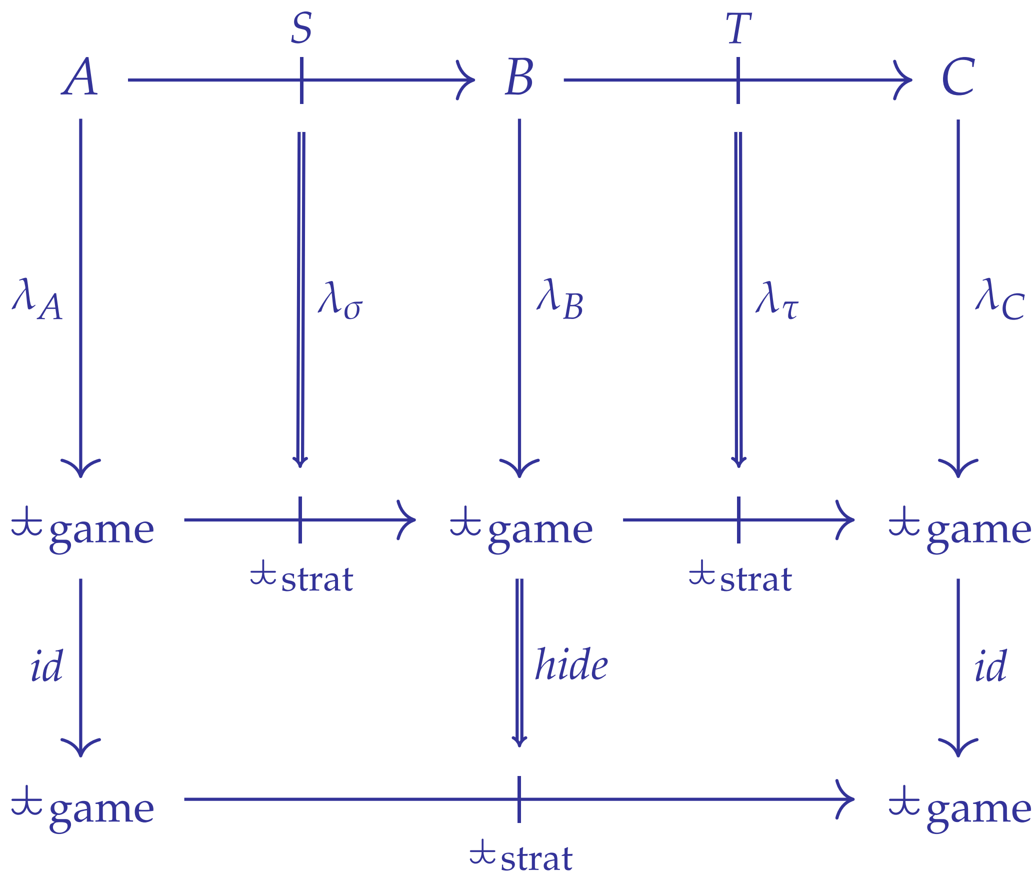
$$\tau \circ \sigma \ = \ (S \boxtimes_B T, \mathsf{coact}_{\tau\circ\sigma}, \lambda_{\tau\circ\sigma}) \ : \ (A, \lambda_A) \longrightarrow\!\!\!\!\!\!\!+\quad (C, \lambda_C)$$

is obtained by composing the two cells horizontally:



and then composing vertically with the double cell *hide*.

Composition of strategies



Composition of strategies

This definition of composition implements the slogan that

composition	=	synchronization	+	hiding

Illustration

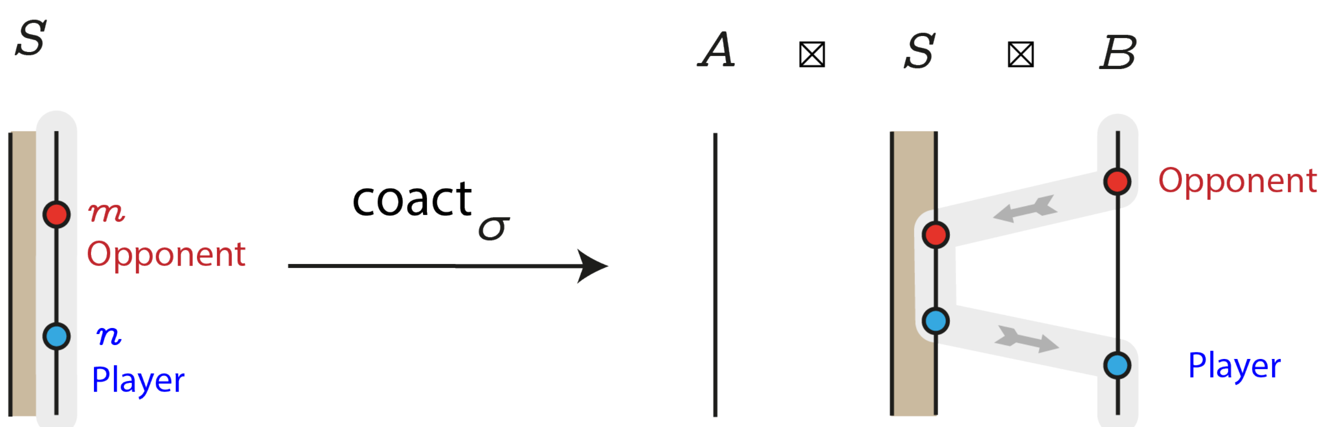
A nice diagrammatic way to represent a strategy

$$\sigma = (S, \mathsf{coact}_\sigma, \lambda_\sigma) \ : \ (A, \lambda_A) \longrightarrow\!\!\!\!\!\!\!\!\!+\;\;\;\; (B, \lambda_B)$$

and its comodule structure

$$S \xrightarrow{\mathsf{coact}_\sigma} A \boxtimes S \boxtimes B$$

is to draw them as follows:

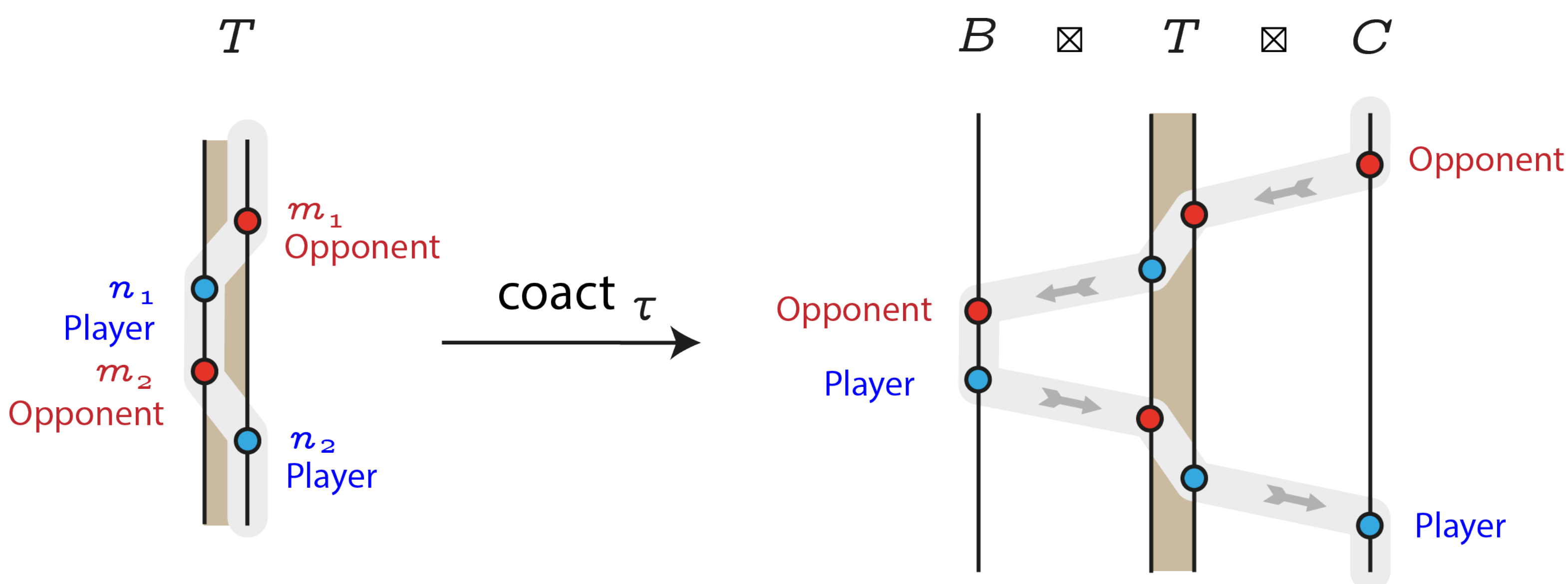


Illustration

We may also consider an asynchronous strategy

$$\tau \ = \ (T, \mathsf{coact}_\tau, \lambda_\tau) \ : \ (B, \lambda_B) \longrightarrow\!\!\!\!\!\!\!+\ \ (C, \lambda_C)$$

and its comodule structure represented as

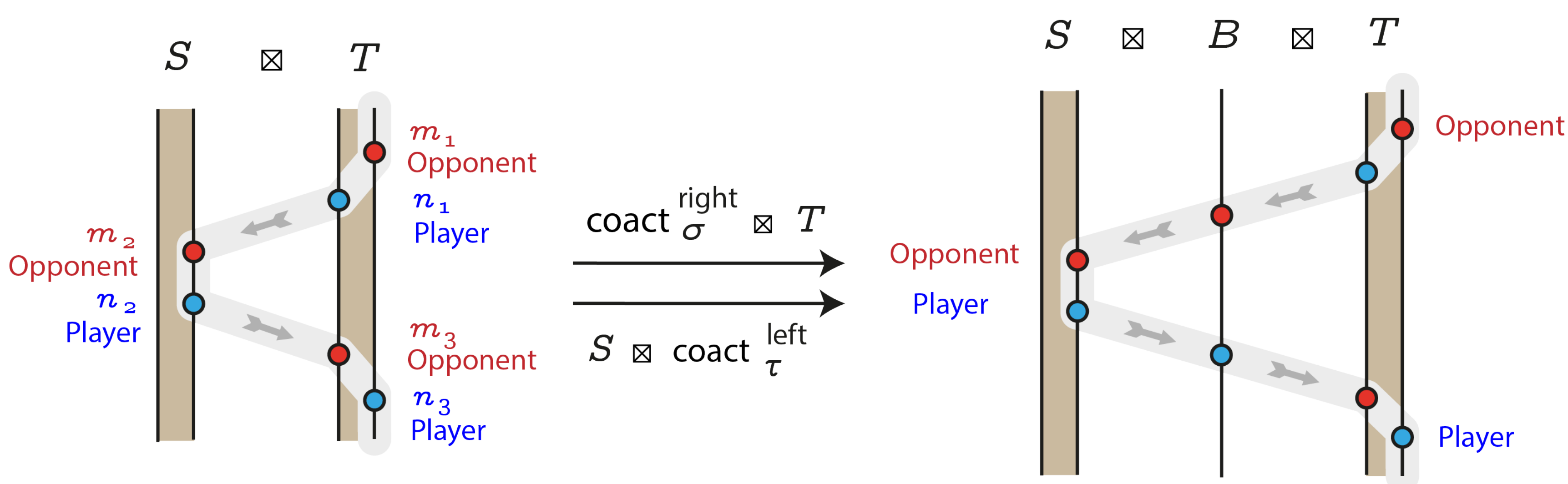


Illustration

The tensor of **Gray comodules**

$$S \boxtimes_B T \overset{equ}{\dashrightarrow} S \boxtimes T \underset{S \boxtimes \mathrm{coact}^{\mathrm{left}}_{\tau}}{\overset{\mathrm{coact}^{\mathrm{right}}_{\sigma} \boxtimes T}{\rightrightarrows}} S \boxtimes B \boxtimes T$$

may be understood as a **synchronization scheme** between strategies:

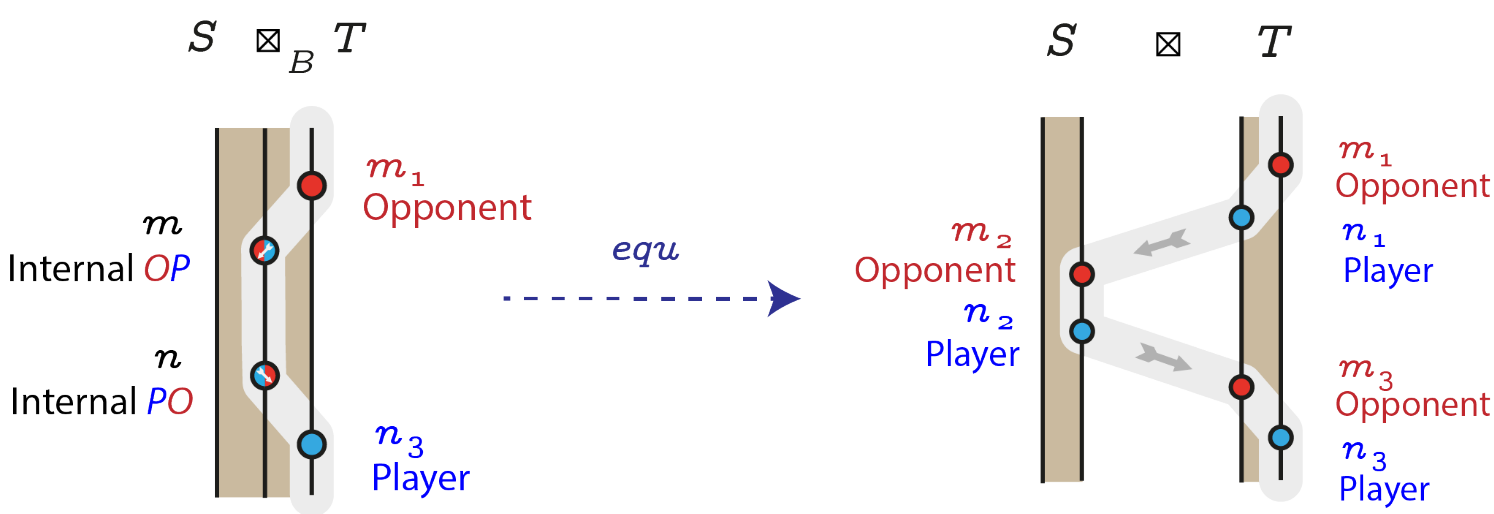


Illustration

In this diagrammatic representation, the equalizer

$$S \boxtimes_B T \xrightarrow{equ} S \boxtimes T$$

is depicted as follows:



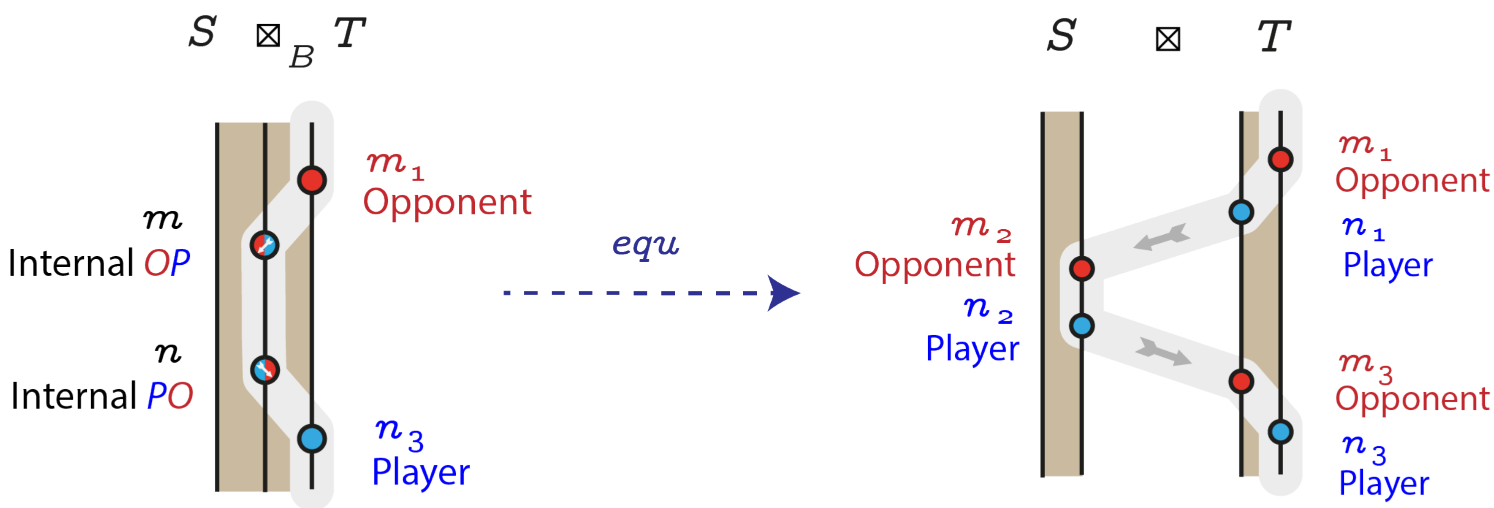
where the synchronized OP move m is mapped to the path $n_1 \cdot m_2$

Illustration

In this diagrammatic representation, the equalizer

$$S \boxtimes_B T \overset{equ}{\dashrightarrow} S \boxtimes T$$

is depicted as follows:



where the synchronized PO move n is mapped to the path $n_2 \cdot m_3$

What about the identities?

The templates $\pitchfork_{\mathrm{game}}$ and $\pitchfork_{\mathrm{strat}}$ are related by the 2-functor

$$\mathit{copycat} \; : \; \pitchfork_{\mathrm{game}} \longrightarrow \pitchfork_{\mathrm{game}} \boxtimes \pitchfork_{\mathrm{game}}$$

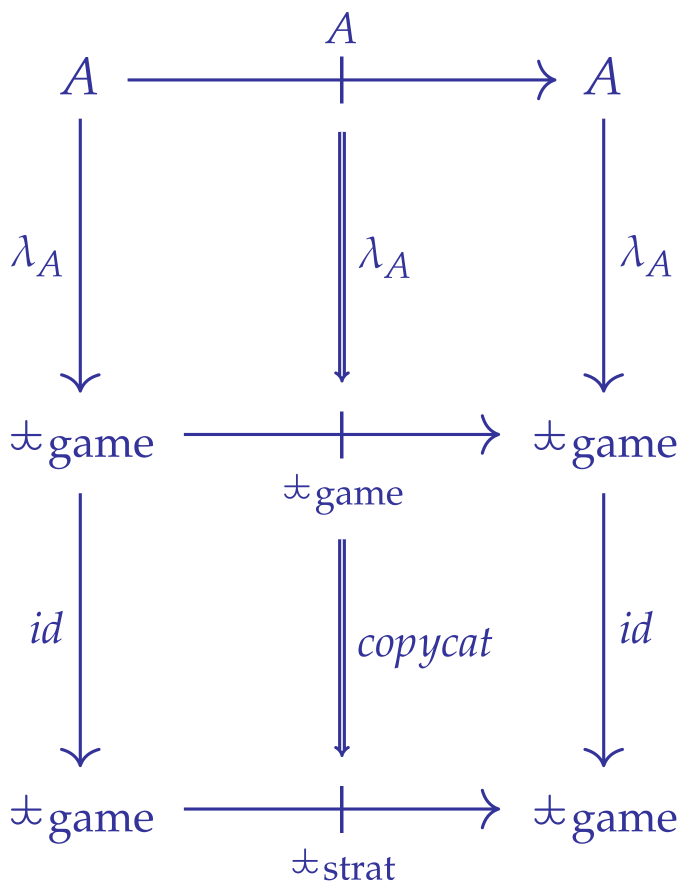
defined by the comultiplication of the Gray comonoid $\pitchfork_{\mathrm{game}}$.

This defines a double cell

$$\begin{array}{ccc} \pitchfork_{\mathrm{game}} & \xrightarrow{\;\pitchfork_{\mathrm{game}}\;} & \pitchfork_{\mathrm{game}} \\ {\scriptstyle id}\downarrow & \Downarrow{\scriptstyle \mathit{copycat}} & \downarrow{\scriptstyle id} \\ \pitchfork_{\mathrm{game}} & \xrightarrow[\;\pitchfork_{\mathrm{strat}}\;]{} & \pitchfork_{\mathrm{game}} \end{array}$$

in the double category **BiComod**.

The identity strategy



What makes everything work...

The family of 2-categories

$$\perp[0] = \perp_{\text{game}} \qquad \perp[1] = \perp_{\text{strat}} \qquad \perp[2] = \perp_{\text{int}}$$

defines an **internal category** $\perp_{\text{asynch}}$ in the monoidal category

$$(\,\mathbf{TwoCat}\,,\,\boxtimes\,,\,\mathbf{1}\,)$$

of 2-categories equipped with the Gray tensor product.

An important point: we use here the definition by Marcelo Aguiar of

an internal category in a monoidal category

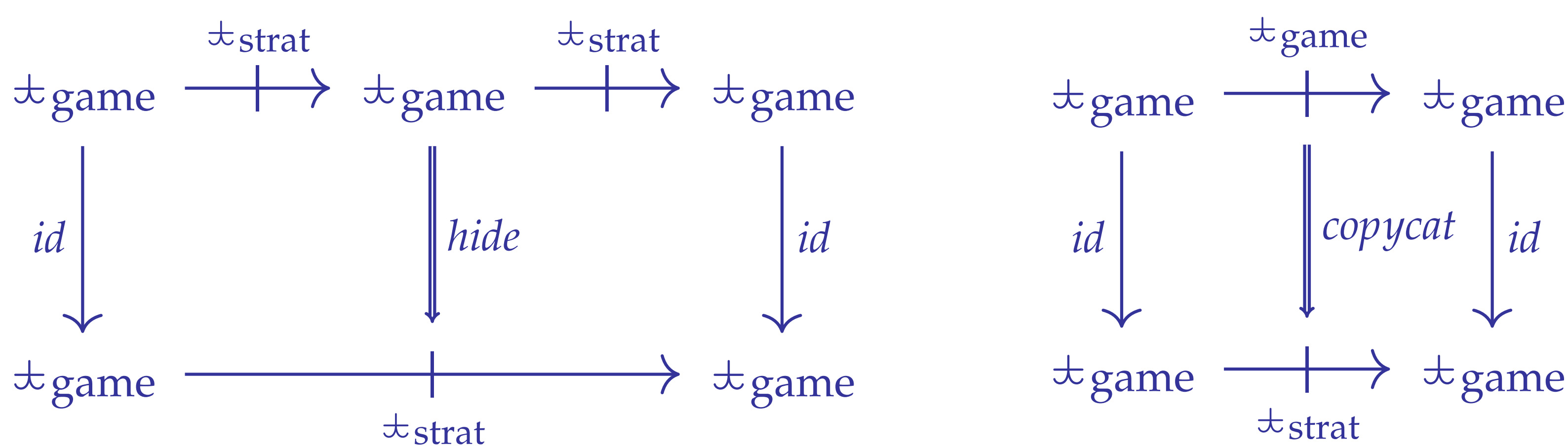
because the Gray tensor product is not the cartesian product of **TwoCat**.

What makes everything work...

In other words, the horizontal map

$$\pitchfork_{\text{strat}} \quad : \quad \pitchfork_{\text{game}} \longrightarrow\!\!\!\!\!\!\!\!+\quad \pitchfork_{\text{game}}$$

defines a **formal monad** $\pitchfork_{\text{asynch}}$ with multiplication and unit



in the double category **BiComod** of **Gray comonoids** and **bicomodules**.

As an immediate consequence...

Theorem A. The construction just given defines a **bicategory**

$$\textbf{Games}(\pm_{\text{asynch}})$$

of asynchronous games, strategies and simulations.

Main technical result of the paper

Theorem B. The bicategory of asynchronous games

$$\textbf{Games}(\pm_{\mathrm{asynch}})$$

is **symmetric monoidal closed.**

Main technical result of the paper

Theorem C. The bicategory of asynchronous games

$$\textbf{Games}(\pm_{\mathrm{asynch}})$$

is **star-autonomous**.

Conclusion and perspectives

- asynchronous games played on **2-categories** with
  - positions of the game as objects
  - trajectories as morphisms
  - reshufflings as 2-cells
- the **Gray tensor of 2-categories** as a shuffle tensor product
- an unexpected connection with **bicomodules on Gray comonoids**
- a model of **differential linear logic** based on **homotopy**

Thank you and stay safe !