Concurrent Hyland-Ong Games

Pierre Clairambault
CNRS & ENS Lyon

Joint work with

Simon Castellan
ENS Lyon

Glynn Winskel
University of Cambridge

Shonan meeting
Higher-Order Model-Checking
14/03/2016

Game Semantics

What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \quad \to \quad \mathbb{B}) \quad \to \quad \mathbb{B}$$

Game Semantics

What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

$$\mathbf{q}^{(\mathcal{Q},-)}$$

Game Semantics

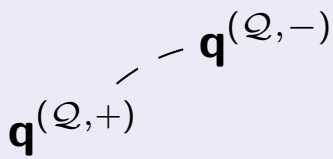
What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



Game Semantics

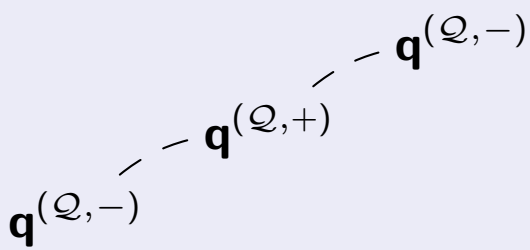
What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



Game Semantics

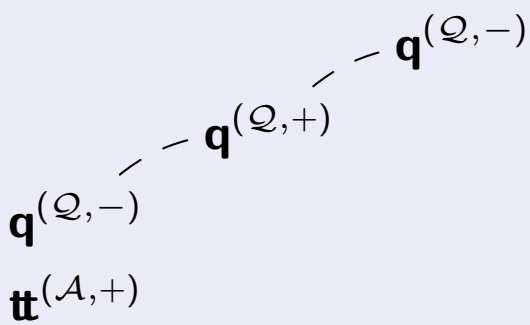
What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



Game Semantics

What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

$$\mathbf{q}^{(\mathcal{Q},-)}$$

$$\mathbf{q}^{(\mathcal{Q},+)}$$

$$\mathbf{q}^{(\mathcal{Q},-)}$$

$$\mathbf{tt}^{(\mathcal{A},+)}$$

Key points:

- *Syntactically: traversals*
- *Abstract machines*
- *Compositionality*

Game Semantics

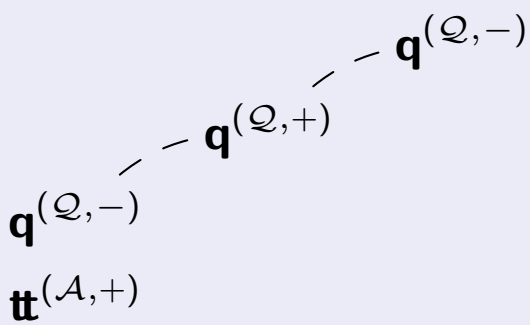
What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



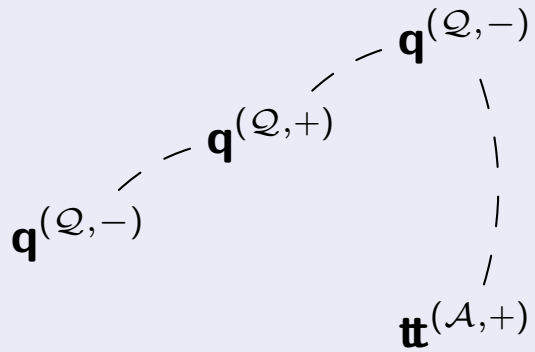
Key points:

- *Syntactically: traversals*
- *Abstract machines*
- *Compositionality*

Dynamic description of effects

Well-bracketing, innocence. . .

$$(\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



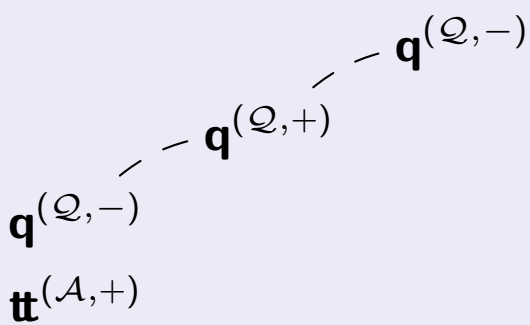
Game Semantics

What are Game Semantics?

Syntax-free operational semantics

Programs denoted by their executions:

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\ \mathbf{tt}]\!] =$$

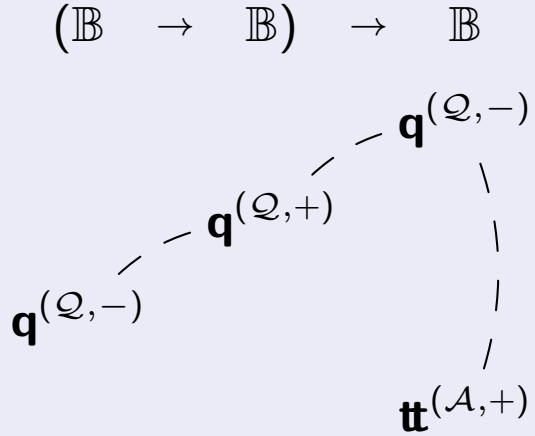


Key points:

- *Syntactically: traversals*
- *Abstract machines*
- *Compositionality*

Dynamic description of effects

Well-bracketing, innocence. . .



Key points:

- *Definability and full abstraction,*
- *Characterisation of observational equivalence*
- *Algorithmic game semantics*

IPA and its sub-languages

Types.

$$
\begin{array}{lcll}
A, B & ::= & \mathbb{B} \mid \mathbb{N} \mid A \to B & \textbf{PCF} \\
& & \mid \textbf{com} \mid \textbf{ref} & \textbf{Idealized Algol} \\
& & \mid \textbf{sem} & \textbf{Idealized Parallel Algol}
\end{array}
$$

Terms.

$$
\begin{array}{lcll}
M, N & ::= & x \mid M\,N \mid \lambda x.\, M \mid Y & \lambda Y\textbf{-calculus} \\
& & \mid \textbf{tt} \mid \textbf{ff} \mid \textbf{if}\ M\ N_1\ N_2 & \\
& & \mid n \mid \textbf{succ}\ M \mid \textbf{pred}\ M \mid \textbf{iszero}\ M & \textbf{PCF} \\
& & \mid \textbf{skip} \mid M; N & \\
& & \mid \textbf{newref}\ v := b\ \textbf{in}\ M \mid M := N \mid !M & \\
& & \mid \textbf{mkvar}\ M\ N & \textbf{IA} \\
& & \mid M \parallel N & \\
& & \mid \textbf{newsem}\ s\ \textbf{in}\ M \mid \textbf{grab}\ M & \\
& & \mid \textbf{release}\ M \mid \textbf{mksem}\ M\ N & \textbf{IPA}
\end{array}
$$

Standard typing rules and small-step call-by-name operational semantics.

This talk

Contents.

- New framework for Game Semantics based on event structures
- Causal / truly concurrent
- Illustrations on IPA and PCF

Outline

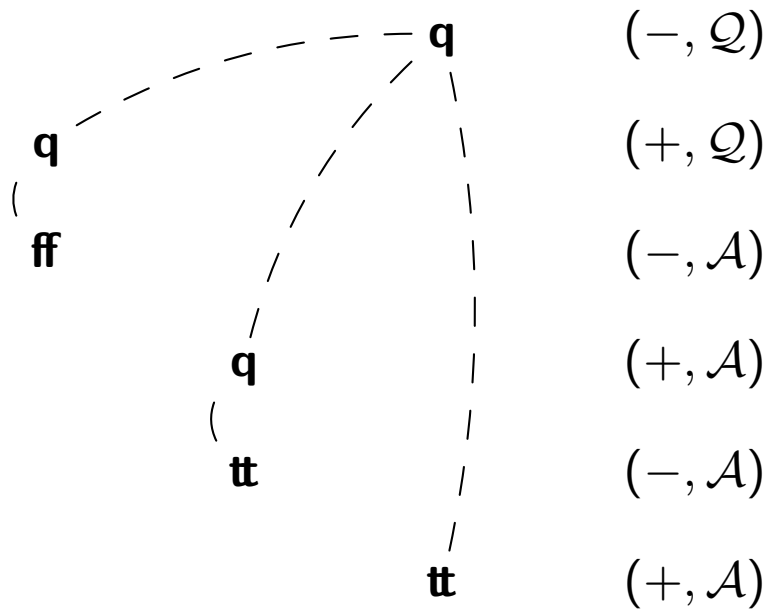
1. From plays to event structures
2. IPA
3. PCF (with a twist)

1. From plays to event structures

Hyland-Ong games and innocent strategies

Hyland-Ong games represent programs by enumerating their **dialogues** with the environment:

$$\mathrm{lor} \quad : \quad \mathbb{B} \to \mathbb{B} \to \mathbb{B}$$

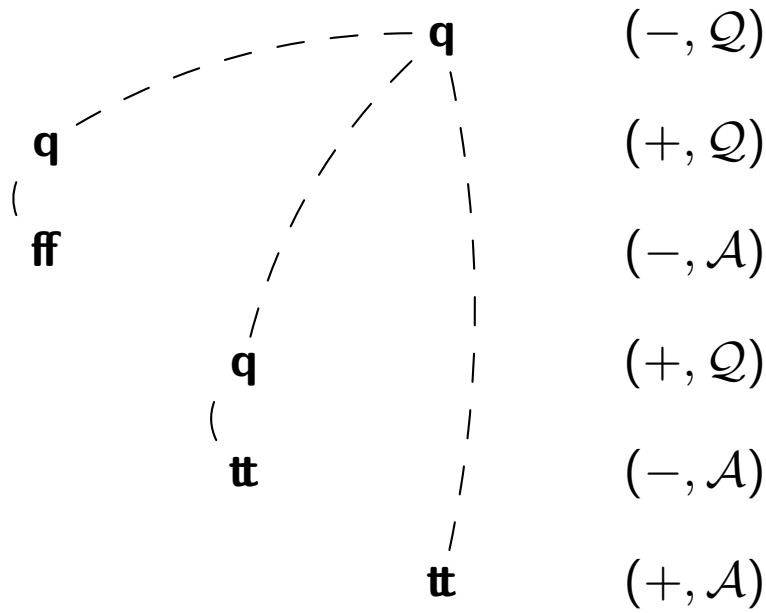


- The dashed lines represent the **justification** relation, indicating the function/argument relationship.
- The play above is a **P-view**, meaning that Opponent always point to the previous move.
- **Innocent strategies** are certain **sets of P-views**.

Definability of innocent strategies: P-views are branches of terms

For instance, this P-view:

$$\mathrm{lor} \quad : \quad \mathbb{B} \to \mathbb{B} \to \mathbb{B}$$



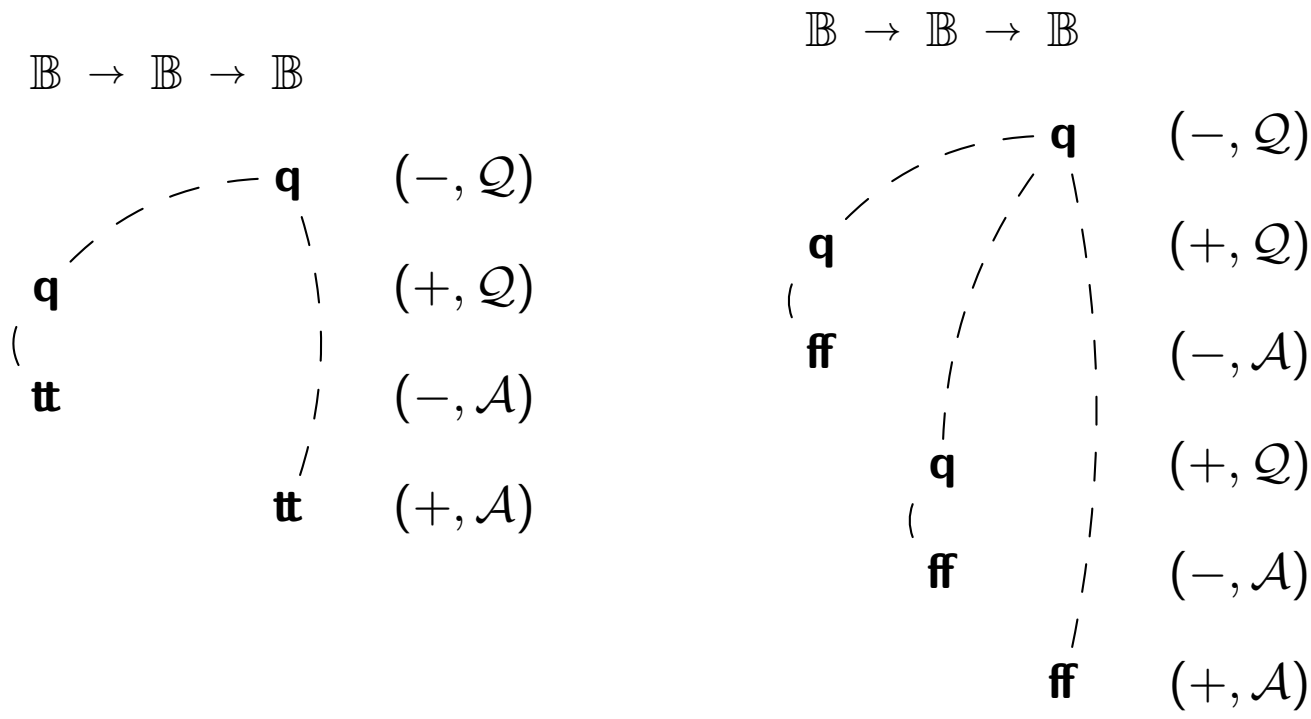
corresponds to the partial term:

$$\lambda b_1^{\mathbb{B}}.\, \lambda b_2^{\mathbb{B}}.\, \mathtt{if}\ b_1\ \mathtt{then}\ [\ ]\ \mathtt{else}\ (\mathtt{if}\ b_2\ \mathtt{then}\ \mathbf{tt}\ \mathtt{else}\ [\ ])$$

The holes [ ] appear where we have no information.

P-views and branches of terms

To fill in the holes, adjoin the two additional P-views:



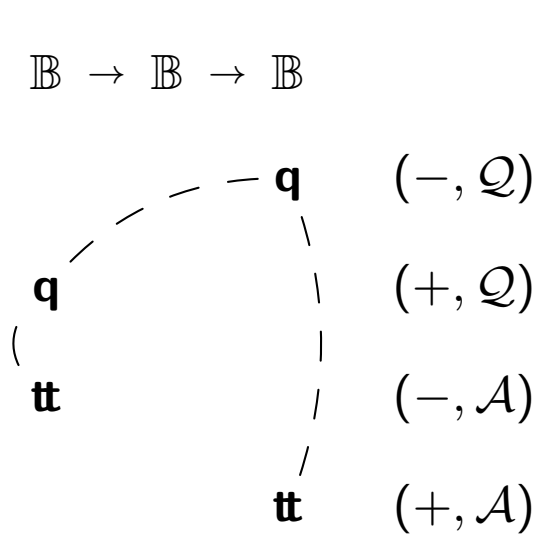
together, they form an **innocent strategy**, in total specifying the term:

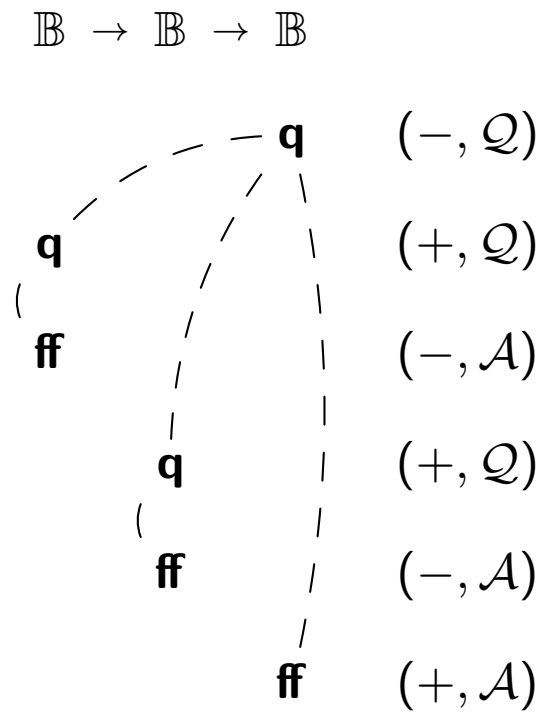
$$\lambda b_1^{\mathbb{B}}.\, \lambda b_2^{\mathbb{B}}.\, \texttt{if}\ b_1\ \texttt{then}\ [\ ]\ \texttt{else}\ (\texttt{if}\ b_2\ \texttt{then}\ \mathbf{tt}\ \texttt{else}\ [\ ])$$

which is often called the **PCF Böhm tree** of `lor`.

P-views and branches of terms

To fill in the holes, adjoin the two additional P-views:





together, they form an **innocent strategy**, in total specifying the term:

$$\lambda b_1^{\mathbb{B}}.\, \lambda b_2^{\mathbb{B}}.\, \texttt{if}\ b_1\ \texttt{then}\ [\mathbf{tt}]\ \texttt{else}\ (\texttt{if}\ b_2\ \texttt{then}\ \mathbf{tt}\ \texttt{else}\ [\mathbf{ff}])$$

which is often called the **PCF Böhm tree** of `lor`.

Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\,\mathbf{if}\,(f\,\mathbf{tt})\,\mathrm{then}\,\mathbf{tt}\,\mathrm{else}\,\bot]\!]$$

$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\,\mathbf{if}\,(f\,\mathbf{tt})\,\mathrm{then}\,\mathbf{tt}\,\mathrm{else}\,\bot]\!]$$

$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

$$\mathbf{q}^{(\mathcal{Q},-)}$$

Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

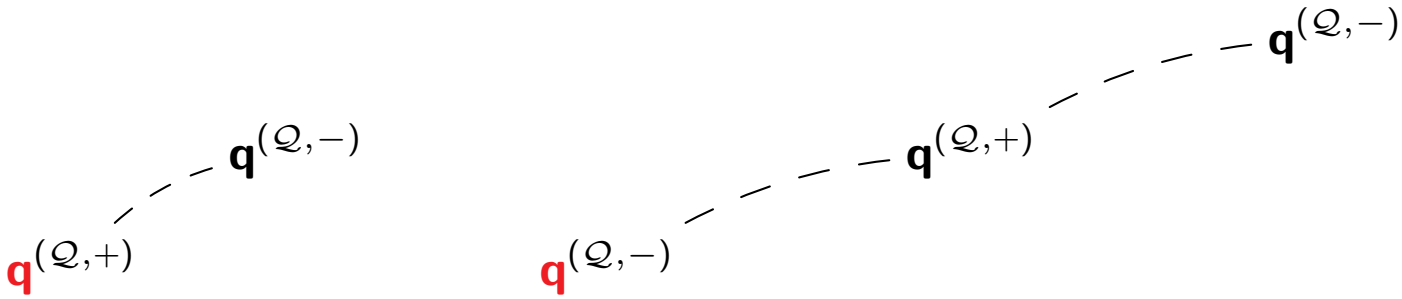
$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

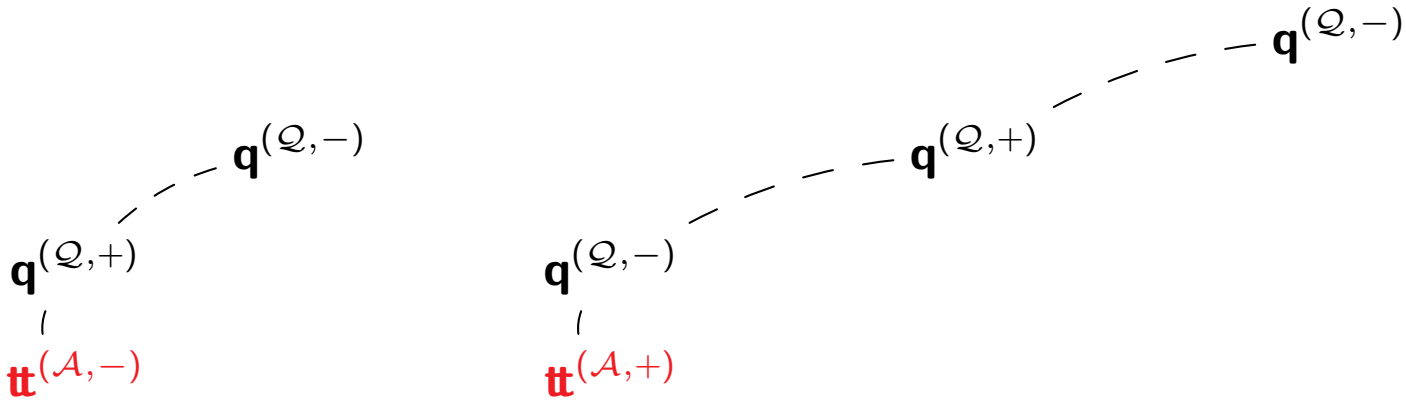


Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

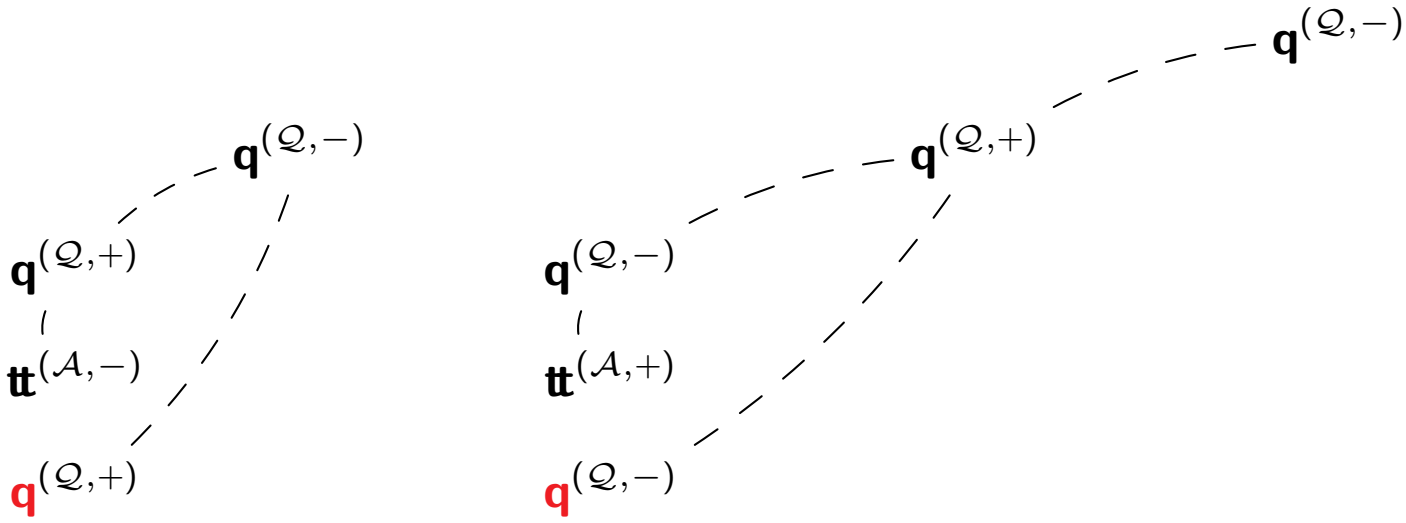


Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

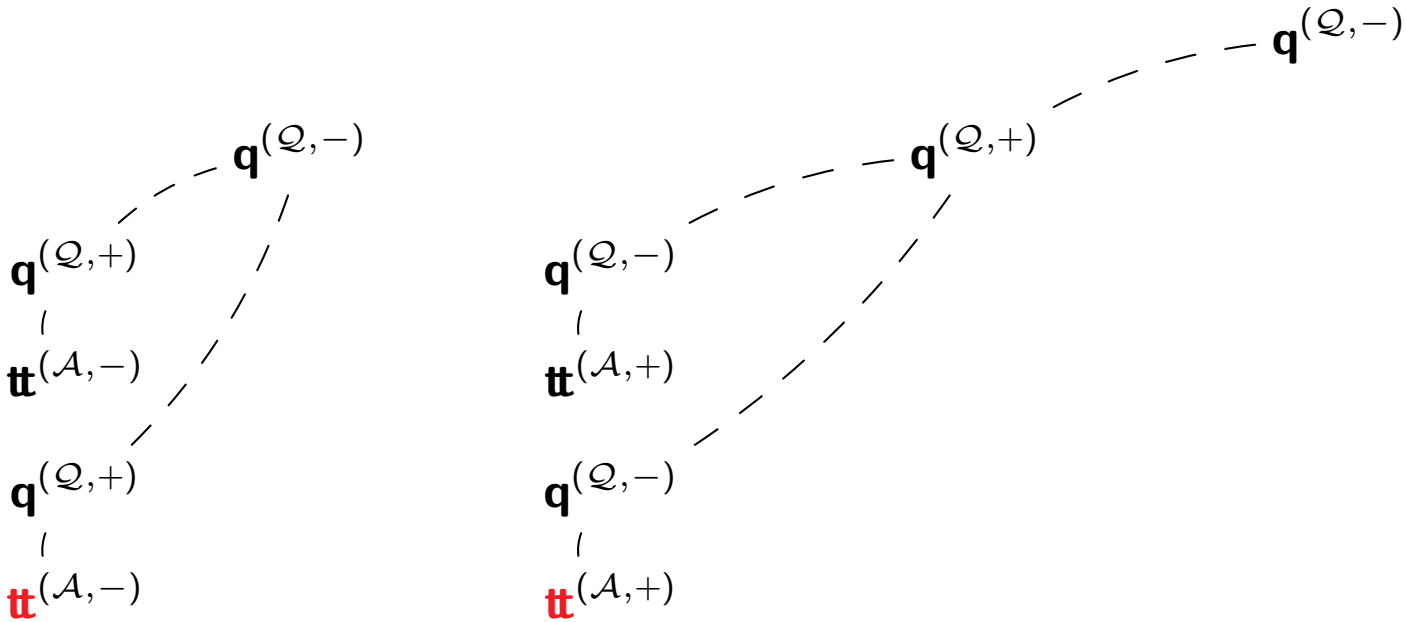


Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$

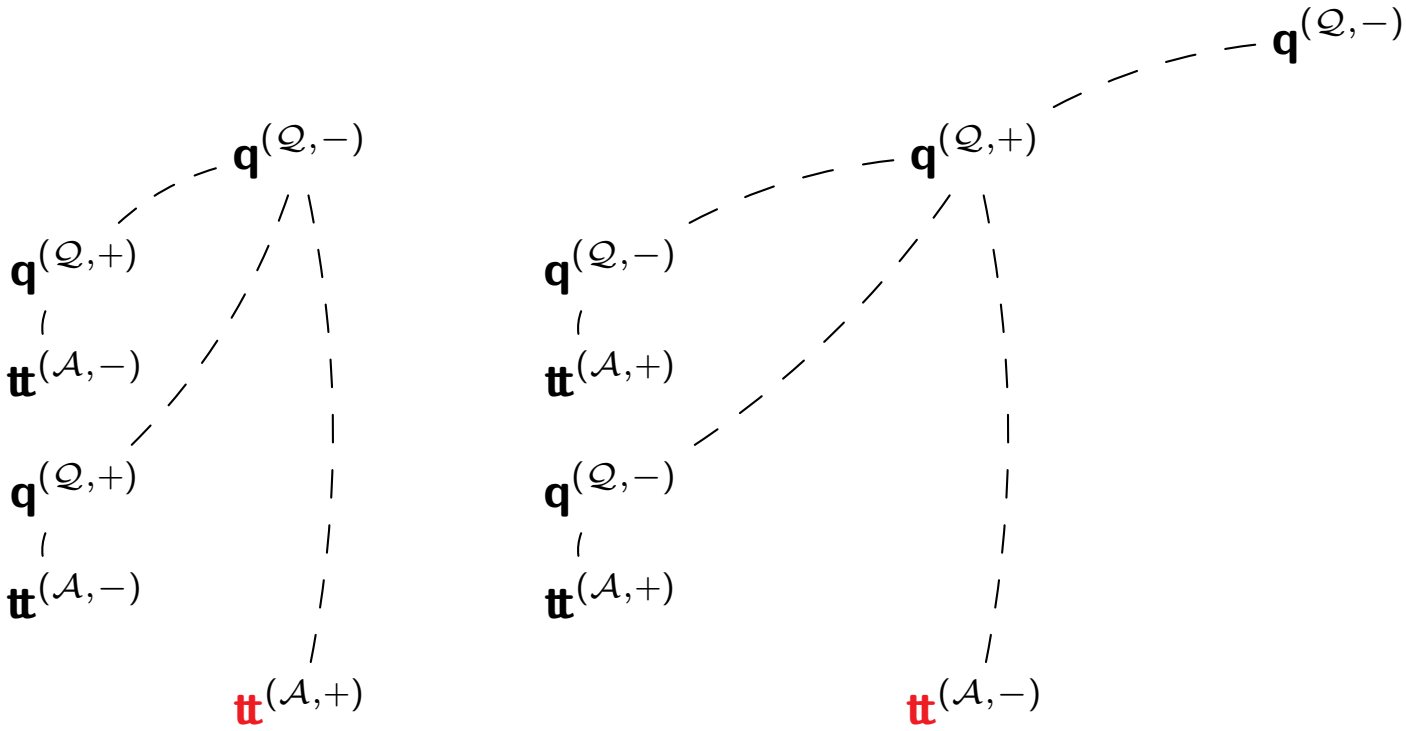


Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \textbf{if}\ (f\ \textbf{tt})\ \text{then}\ \textbf{tt}\ \text{else}\ \bot]\!]$$

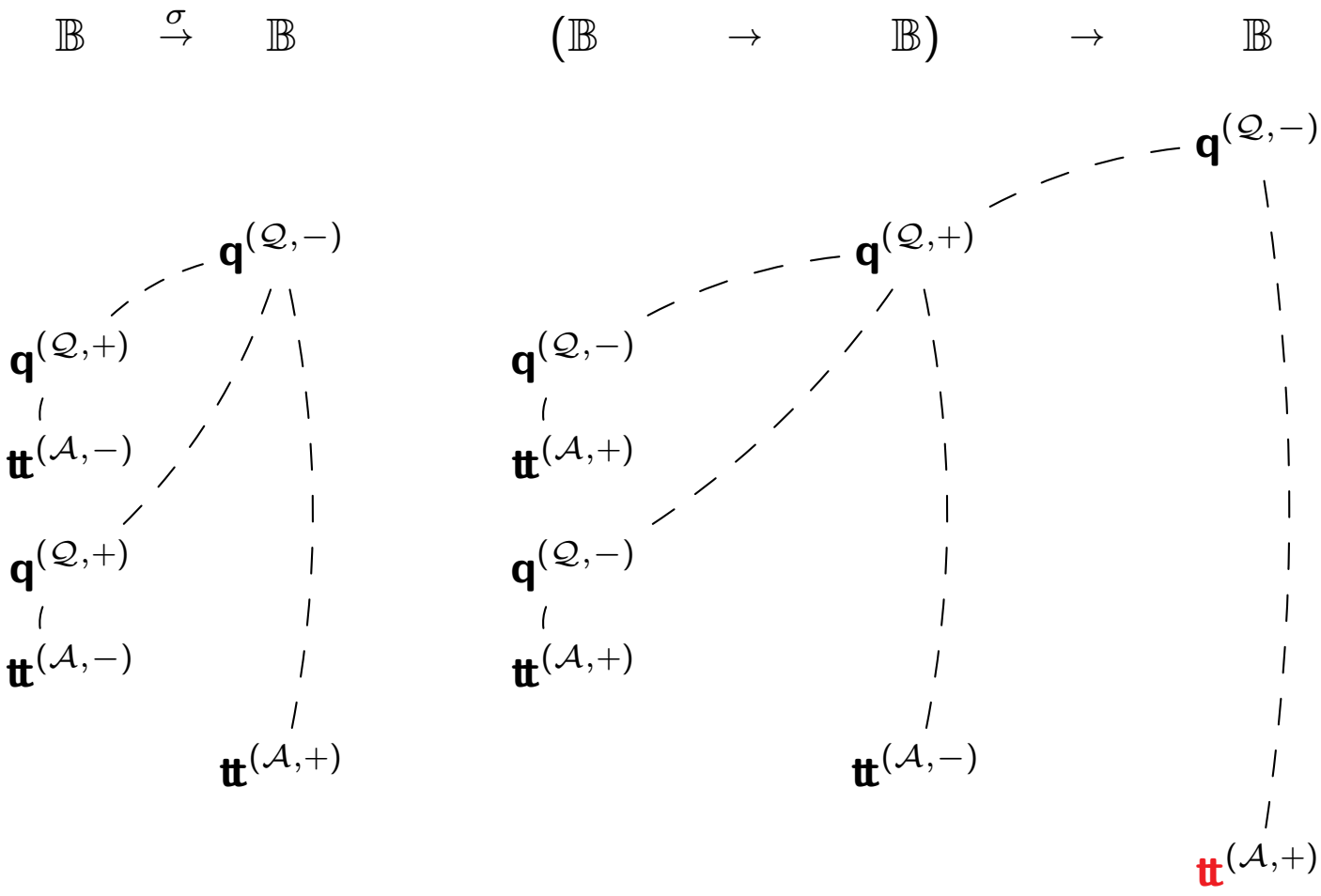
$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

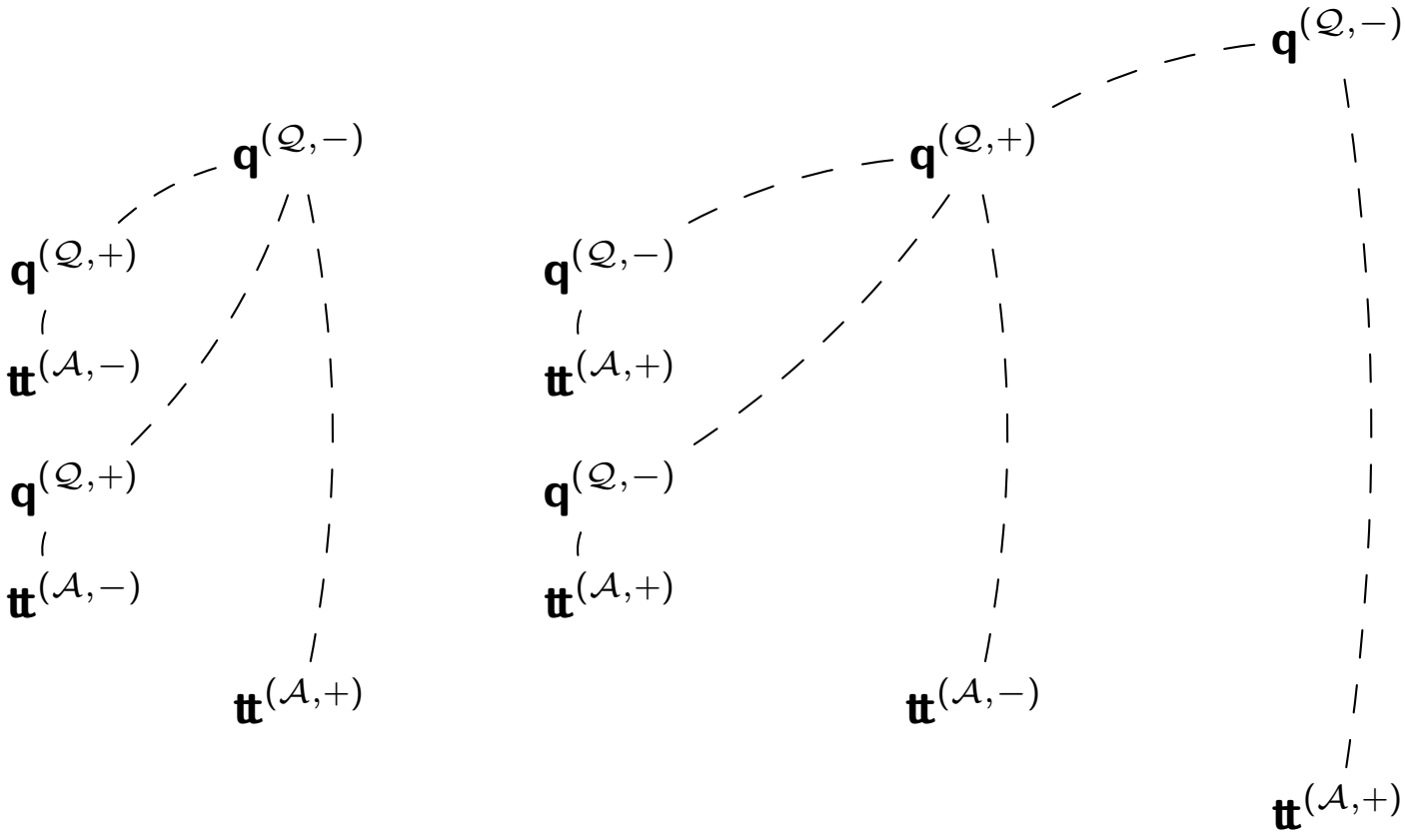


Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\ \mathbf{if}\ (f\ \mathbf{tt})\ \mathrm{then}\ \mathbf{tt}\ \mathrm{else}\ \bot]\!]$$

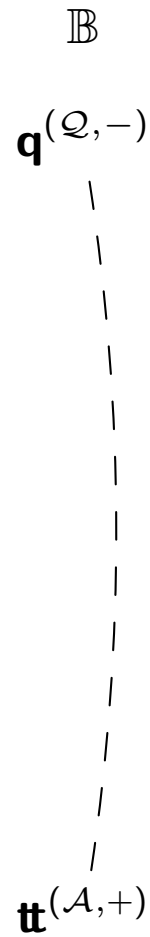
$$\mathbb{B} \xrightarrow{\sigma} \mathbb{B} \qquad (\mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\,\mathbf{if}\,(f\,\mathbf{tt})\,\mathrm{then}\,\mathbf{tt}\,\mathrm{else}\,\bot]\!]$$

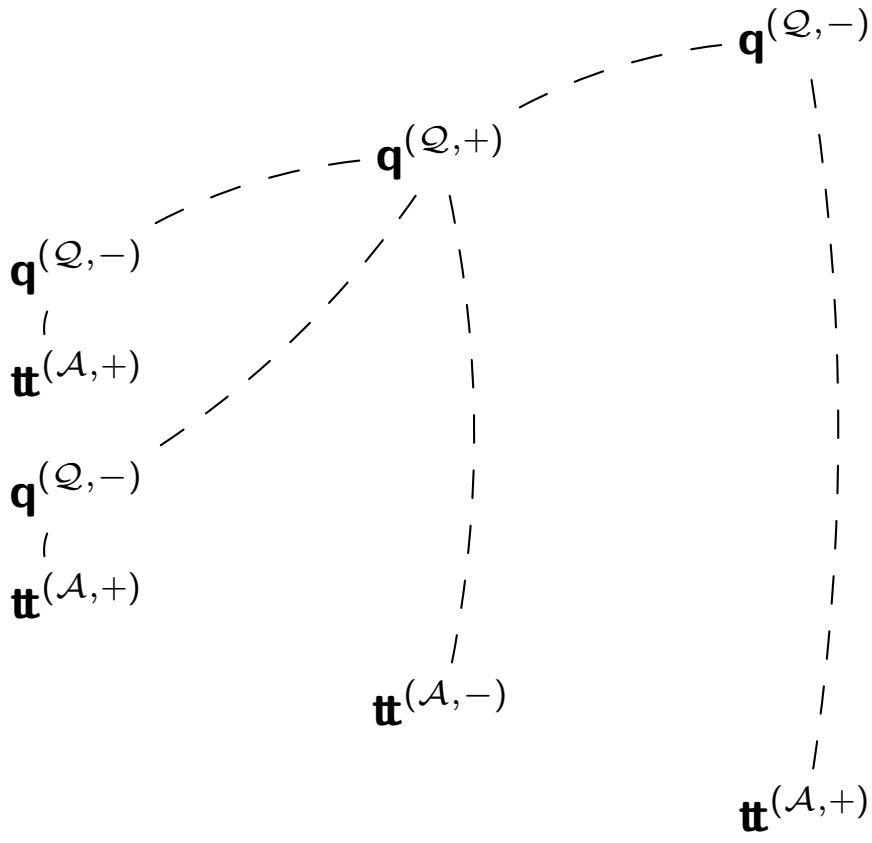


Composition and replication

Application is computed via **parallel interaction plus hiding**.

$$[\![\lambda f^{\mathbb{B}\to\mathbb{B}}.\,\mathbf{if}\,(f\,\mathbf{tt})\,\mathrm{then}\,\mathbf{tt}\,\mathrm{else}\,\bot]\!]$$

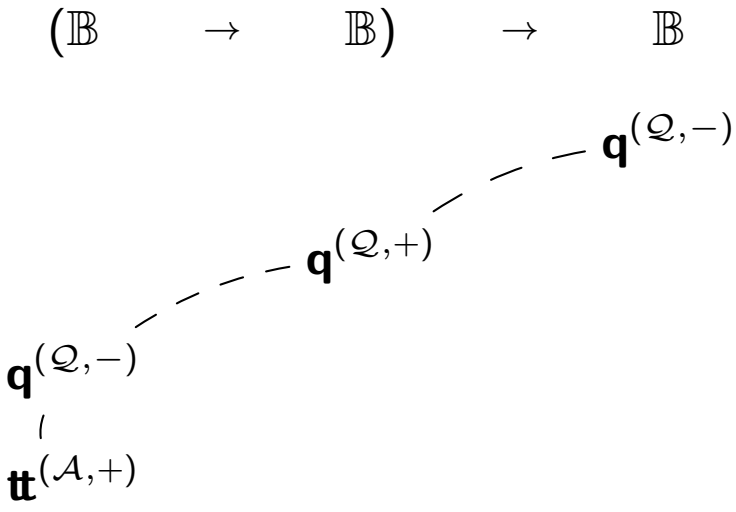
$$(\mathbb{B} \quad \to \quad \mathbb{B}) \quad \to \quad \mathbb{B}$$



↪ Freely complete strategies to plays where **Opponent is unrestricted**.

Causal *vs.* expanded innocent strategies

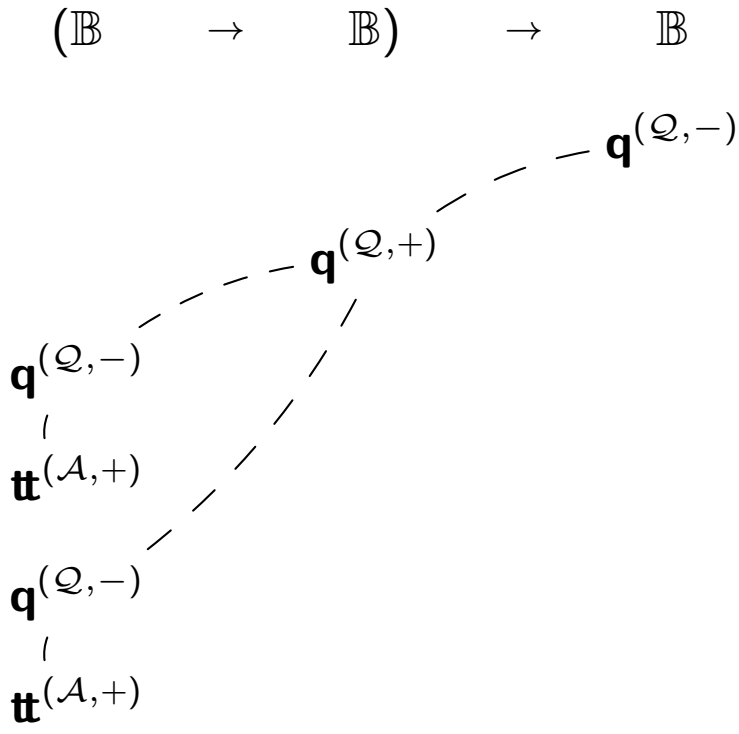
Causal form



Set of **P-views**

Syntax (Böhm tree)

Expanded form



Set of **plays**

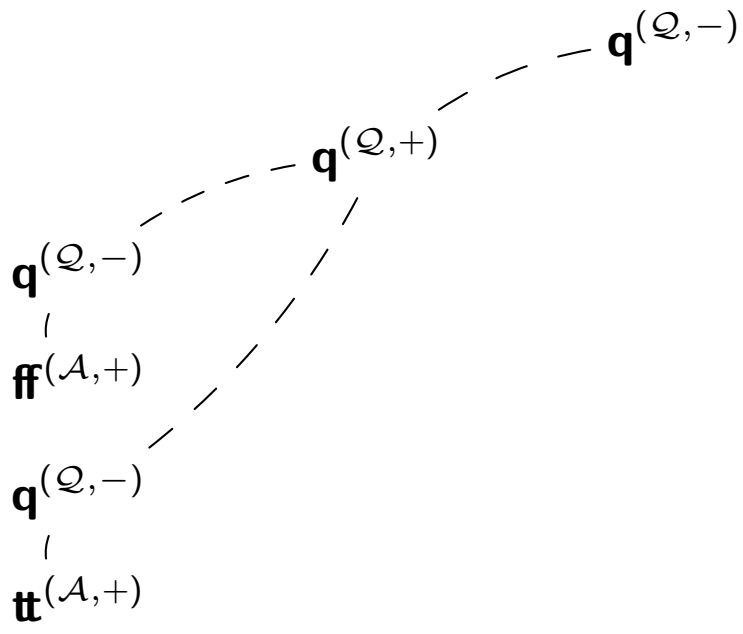
Behaviour under execution

Composition of causal forms goes to expanded forms and back.

State through expanded plays

What if strategies can **change their mind?**

$$(\mathbb{B} \quad \to \quad \mathbb{B}) \quad \to \quad \mathbb{B}$$



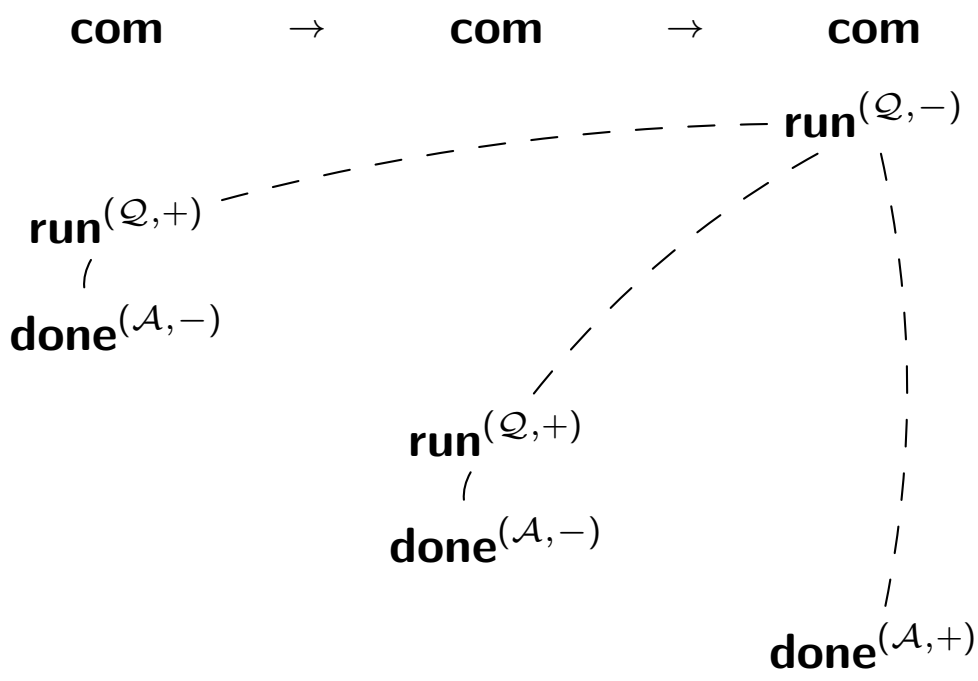
$$\in [\![\mathbf{newref}\, x := \mathbf{ff}\, \mathbf{in}\, \lambda f^{\mathbb{B}\to\mathbb{B}}.\, f\, (\mathbf{if}\, !x\, \mathbf{tt}\, (x := \mathbf{tt}; \mathbf{ff}))]\!]$$

We get **full abstraction** for **higher-order stateful languages** (Abramsky& McCusker, . . .).

Price: lost the causal representation.

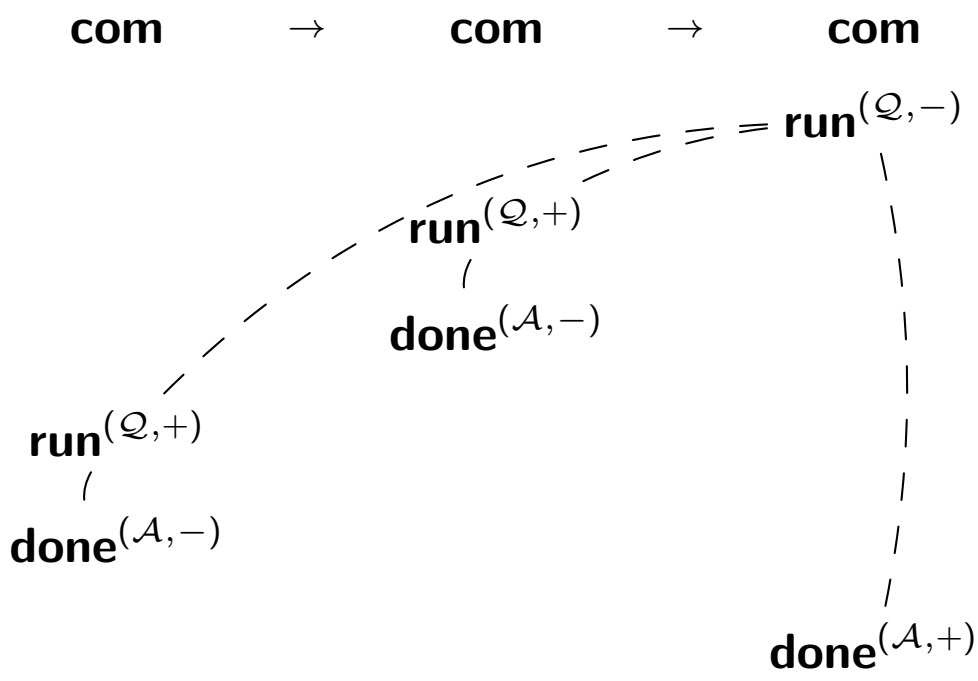
Into concurrency

The strategy for parallel composition [Laird, Ghica&Murawski] has:



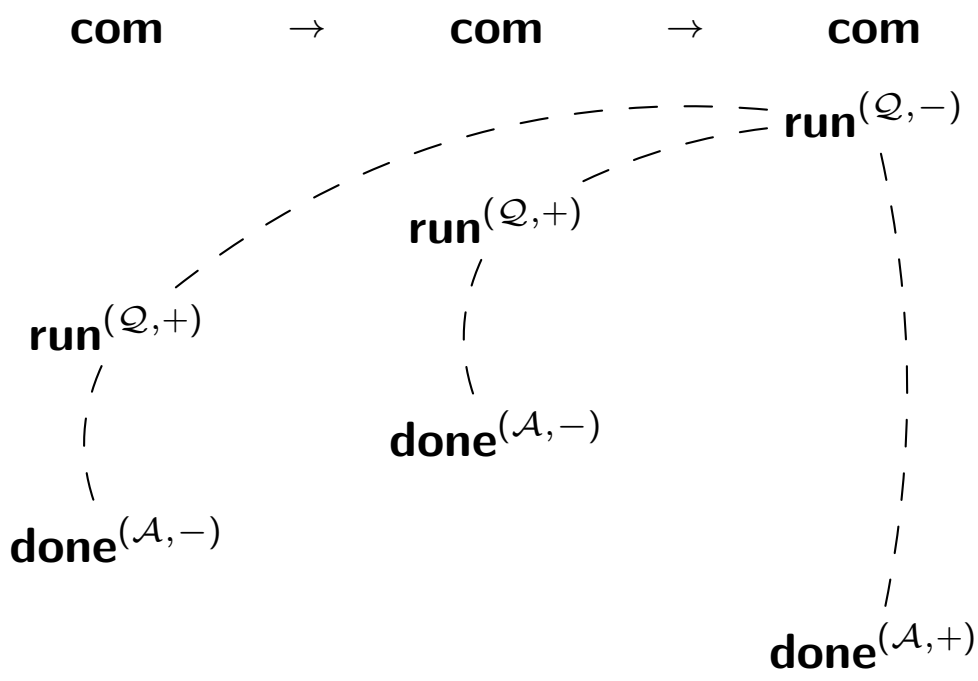
Into concurrency

The strategy for parallel composition [Laird, Ghica&Murawski] has:



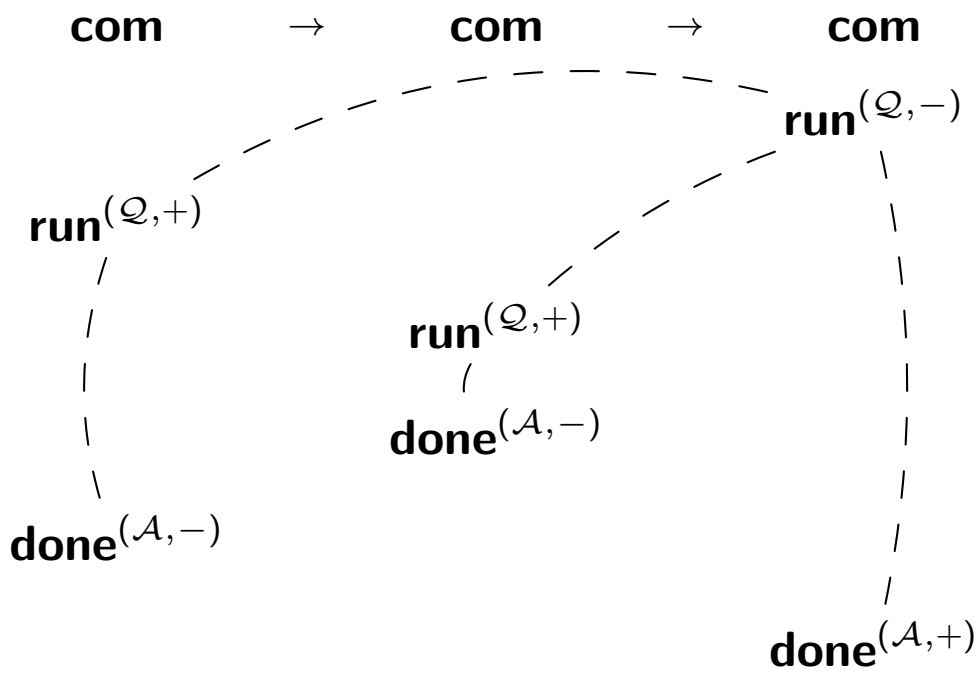
Into concurrency

The strategy for parallel composition [Laird, Ghica&Murawski] has:



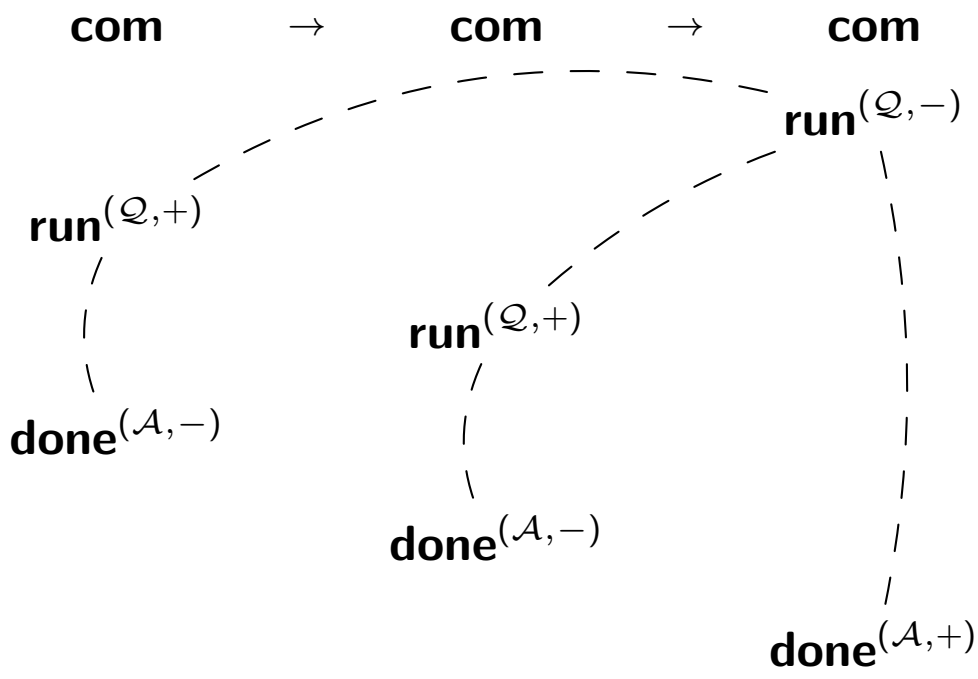
Into concurrency

The strategy for parallel composition [Laird, Ghica&Murawski] has:



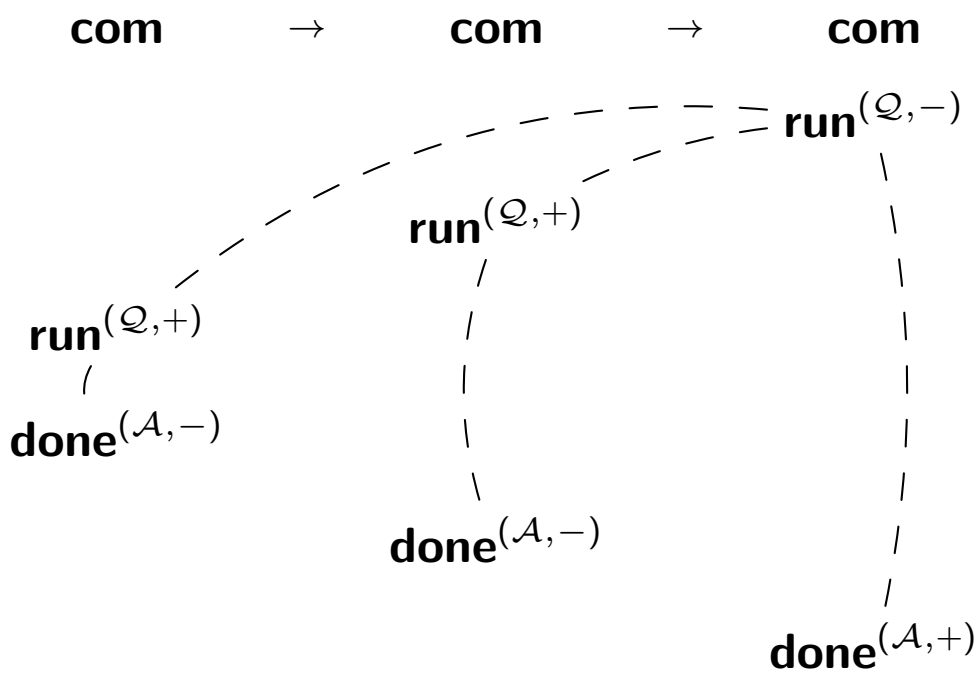
Into concurrency

The strategy for parallel composition [Laird, Ghica&Murawski] has:



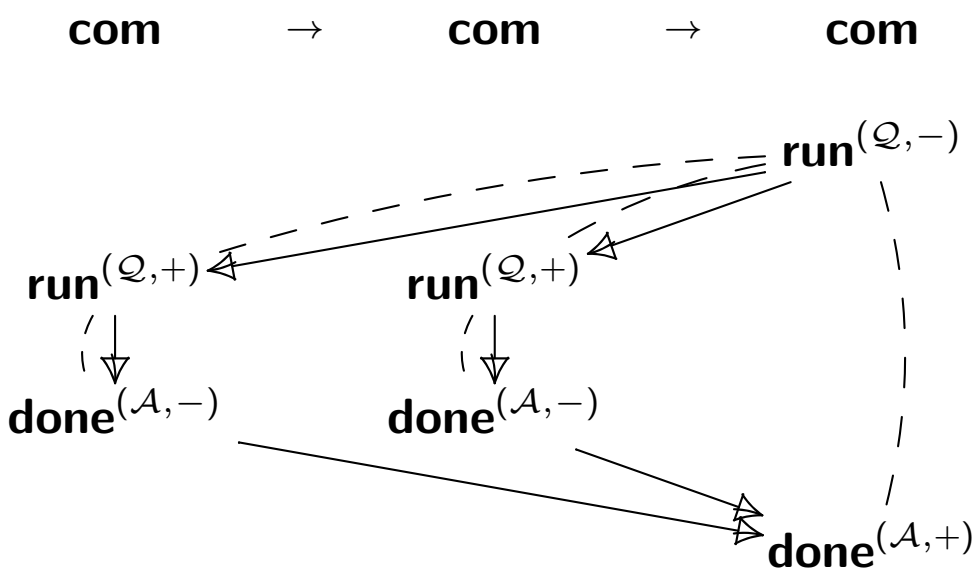
Into concurrency

The strategy for parallel composition [Laird, Ghica&Murawski] has:



Alternative: strategies as partial orders

A **causal** representation [Abramsky&Melliès, Melliès&Mimram, Faggian&Piccolo].



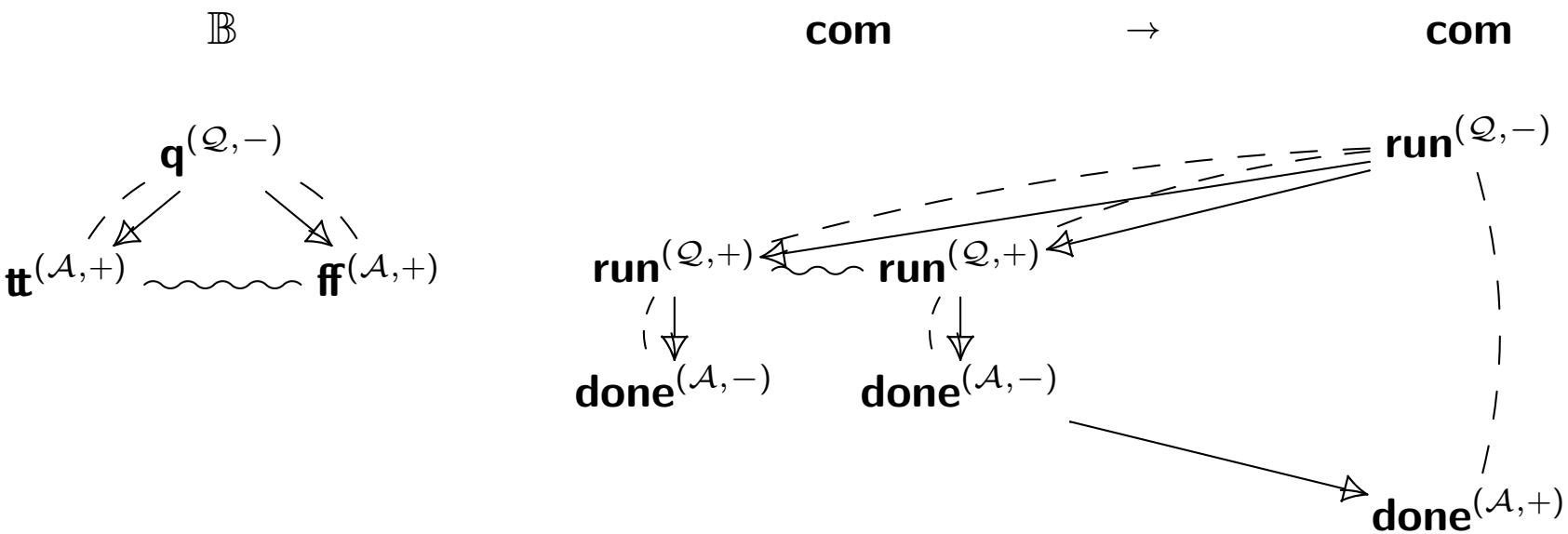
Deterministic programs are **partial orders**: more **compact**, and **readable**.

Questions:

- How to **systematically compute** such representations for programs? (deterministic case: inferred from plays [Melliès&Mimram])
- How to exploit it, and for which programs?

Causality + non-determinism = event structures

Adding **non-determinism** to the representation.



Event structures

Definition

An **event structure** (with binary conflict) is a tuple $(E, \leq_E, \sharp_E)$ where $(E, \leq_E)$ is a **partial order** and $\sharp_E$ is an irreflexive symmetric **conflict** relation. This data satisfies:

- For each $e \in E$, $\{e' \in E \mid e' \leq_E e\}$ is finite
- For each $e_1 \sharp_E e_2$, if $e_2 \leq_E e_2'$, then $e_1 \sharp_E e_2'$.

We write $\rightarrowtriangle$ for **immediate causality**, and $\sim$ for **immediate conflict**.

Definition

A **strategy** on game A is an event structure S labeled by A:

$$\sigma : S \to A$$

subject to some further conditions.

Here A is a **game**, *i.e.* an event structure representing the type whose events are annotated with $\{\mathcal{Q}, \mathcal{A}\} \times \{-, +\}$.

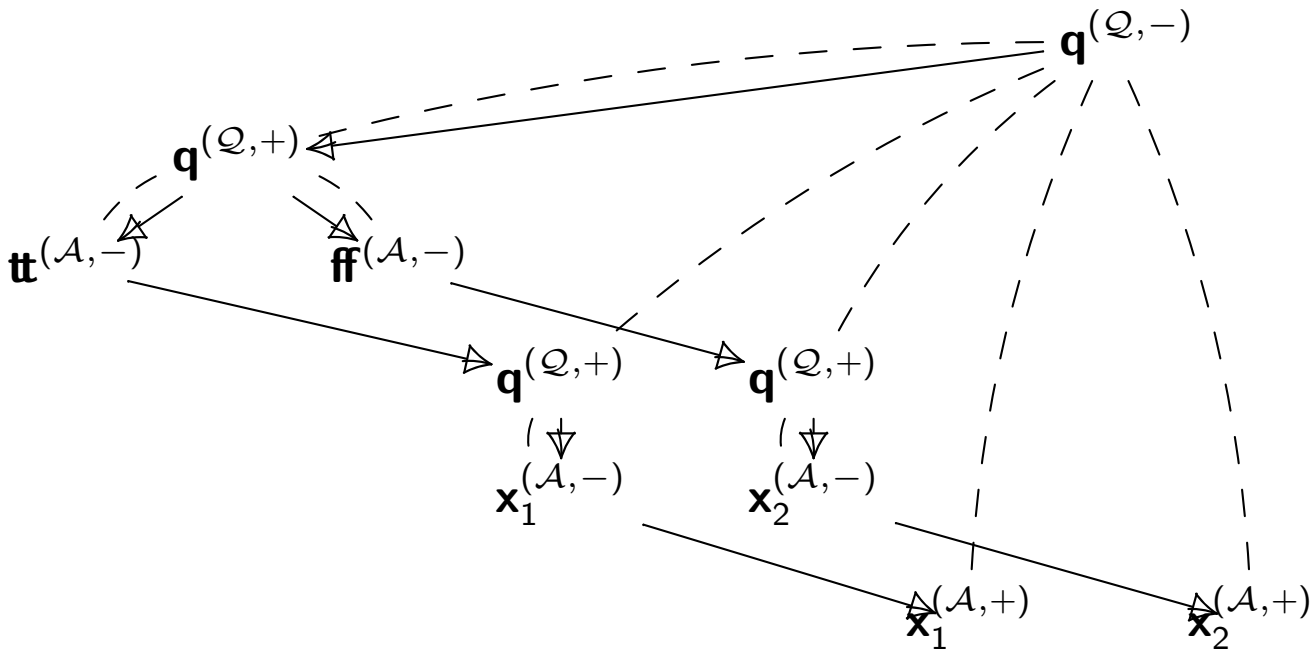
3. Interpretation of IPA

Interpretation: boolean constants/combinators

Combinators on booleans are interpreted as:

$$[\![\mathbf{tt}]\!] : \mathbb{B} \qquad [\![\mathbf{ff}]\!] : \mathbb{B}$$

$$[\![\mathbf{if} - - -]\!] : \mathbb{B} \to \mathbb{X} \to \mathbb{X} \to \mathbb{X}$$



Similar for $\mathbb{N}$, **com** combinators.

Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

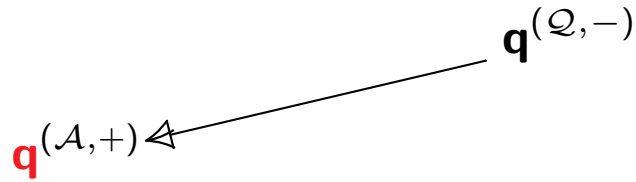
$$\mathbf{q}^{(\mathcal{Q},-)}$$

Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

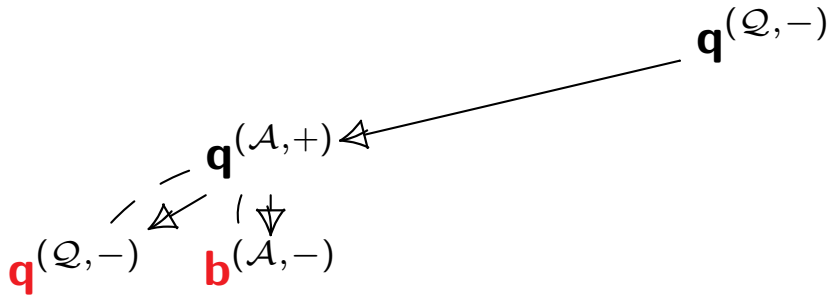


Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

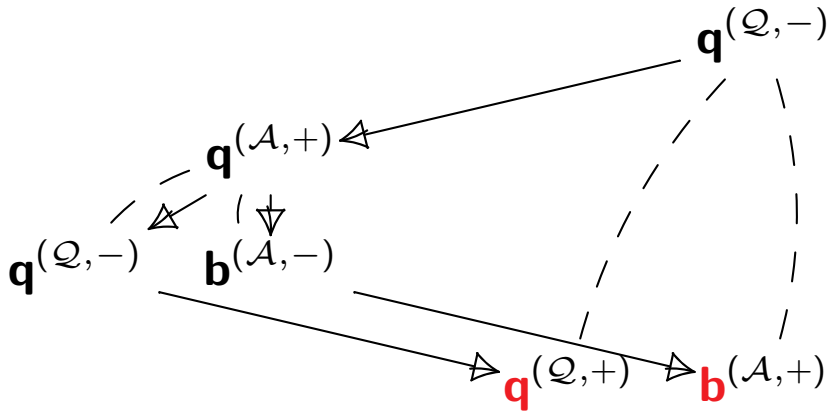


Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

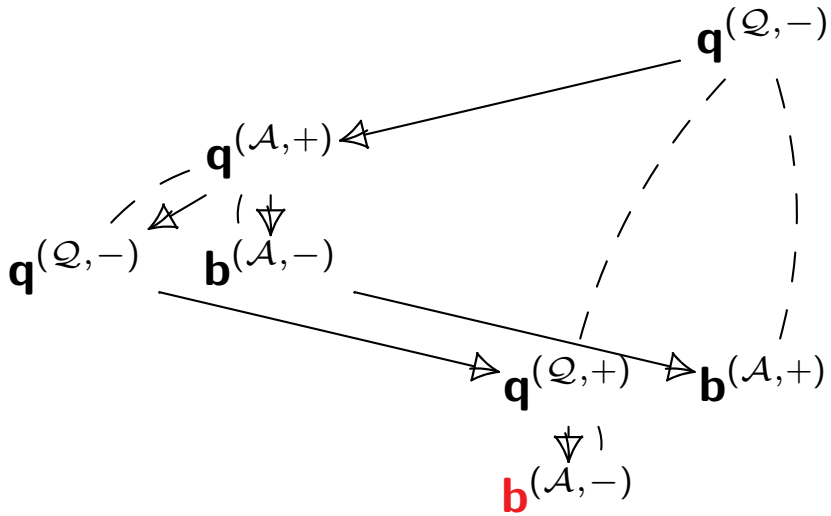


Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

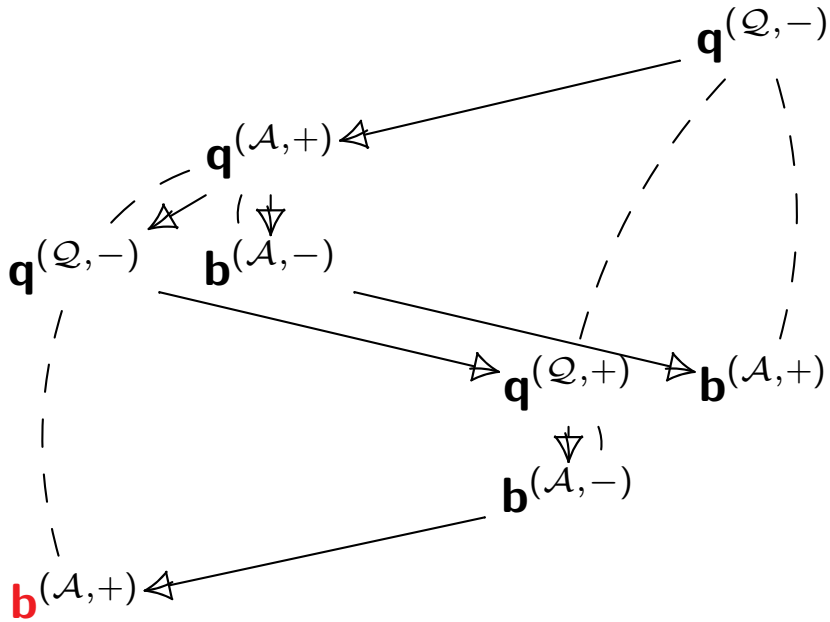


Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$

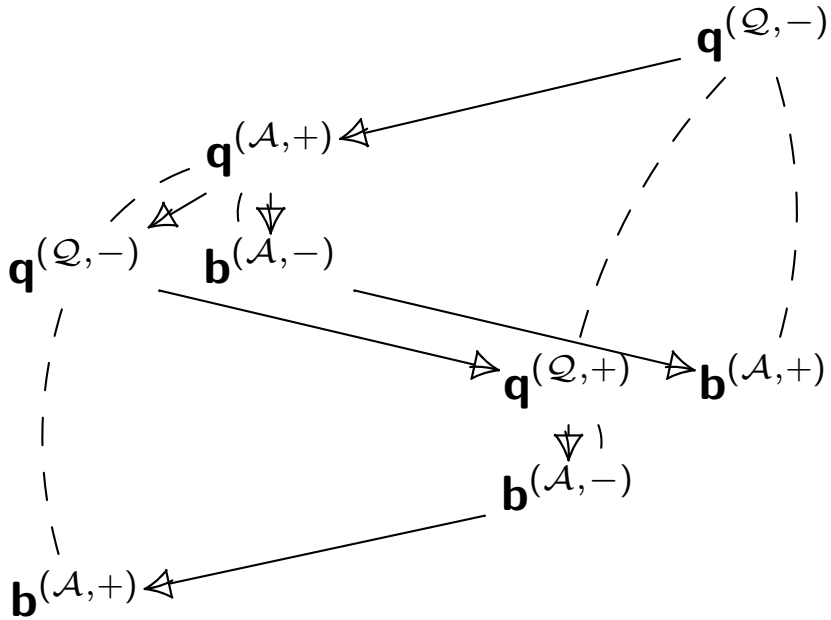


Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$



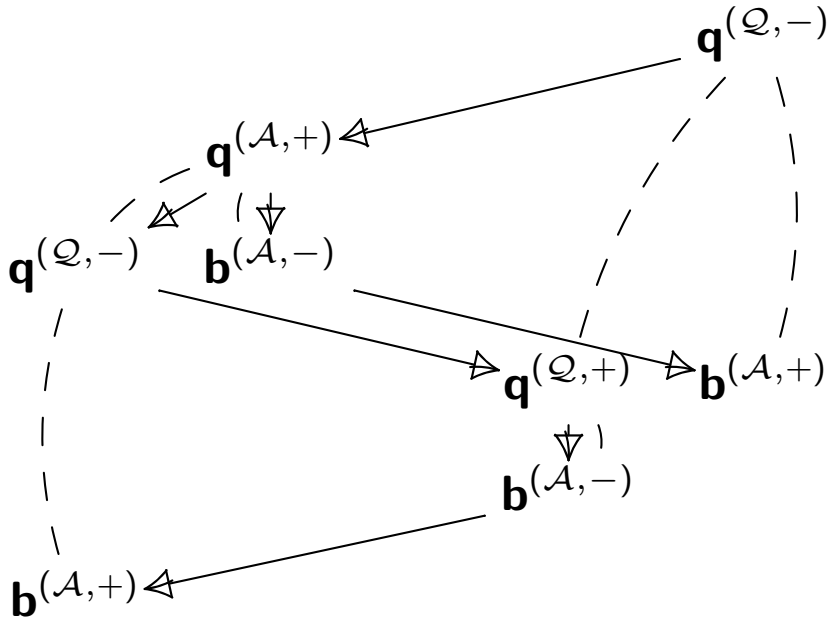
Abstraction is silent.

Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$



Abstraction is silent.

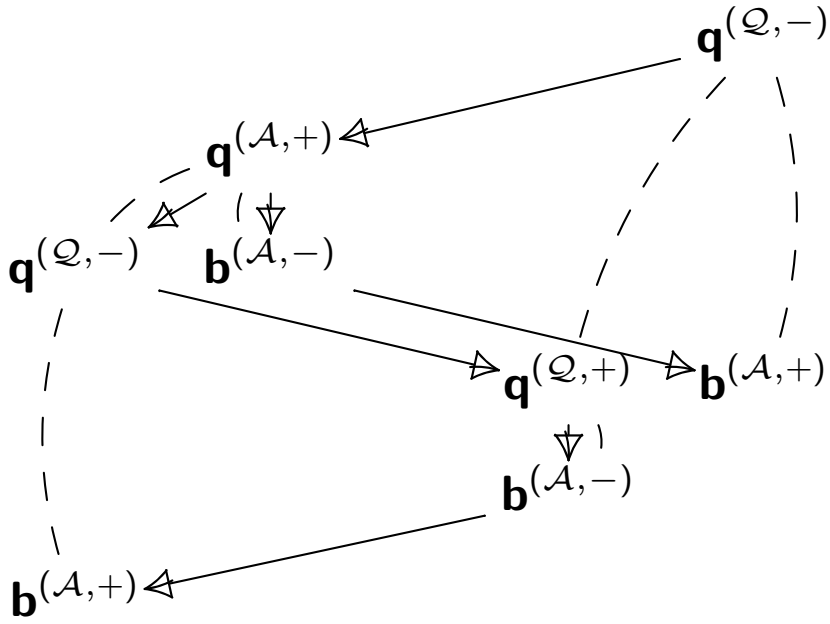
Recursion is the upper bound of finite iterations.

Interpretation: λY-calculus

Identifiers are interpreted as **copycat strategies**.

$$[\![\Gamma, x : \mathbb{B} \to \mathbb{B}, \Delta \vdash x : \mathbb{B} \to \mathbb{B}]\!] =$$

$$\Gamma, \quad \mathbb{B} \quad \to \quad \mathbb{B}, \quad \Delta \quad \vdash \quad \mathbb{B} \quad \to \quad \mathbb{B}$$



Abstraction is silent.

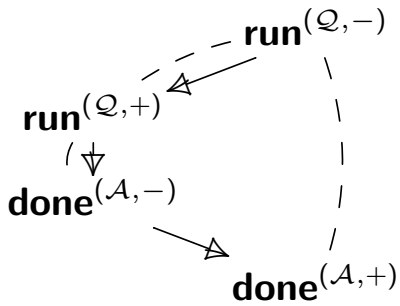
Recursion is the upper bound of finite iterations.

Application is interaction plus hiding. . .

An example of composition

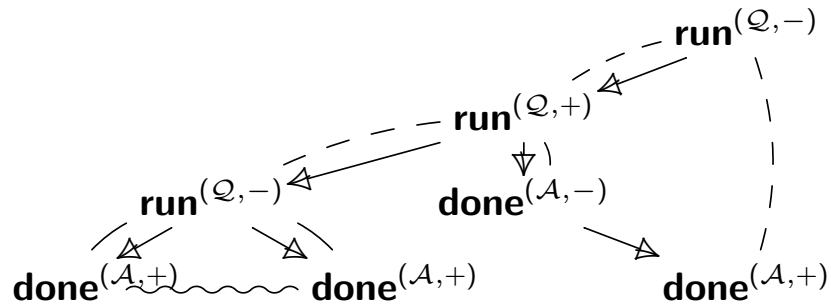
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ $\quad \mathbf{com} \quad \to \quad \mathbf{com}$



$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ $\quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbf{com}$



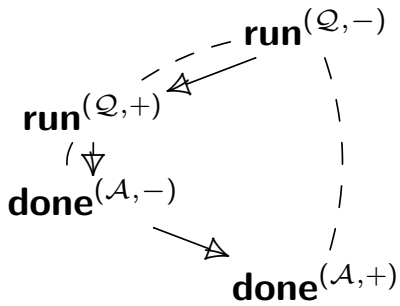
$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

$\tau \circledast \sigma :$ $\quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbf{com}$

An example of composition

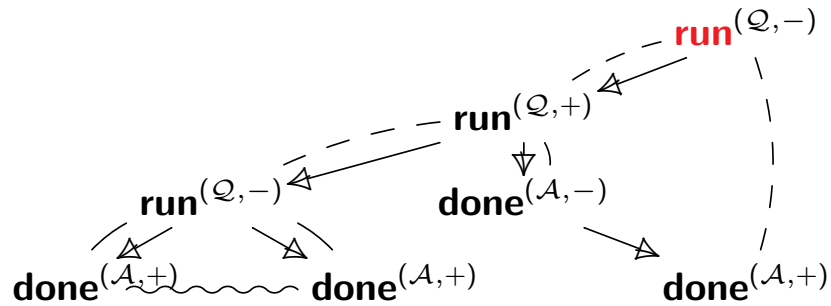
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

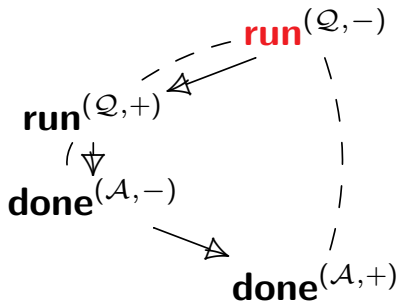
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**

run$^{(\mathcal{Q},-)}$

An example of composition

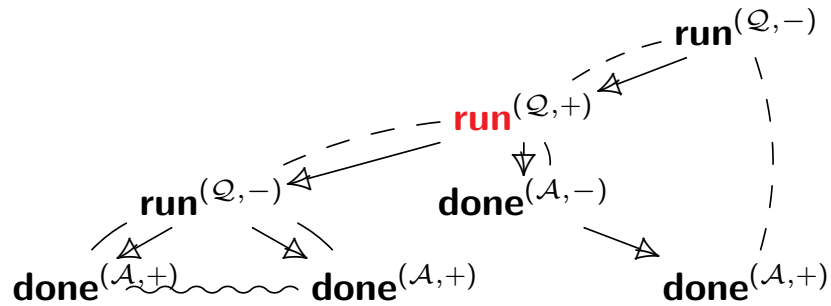
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



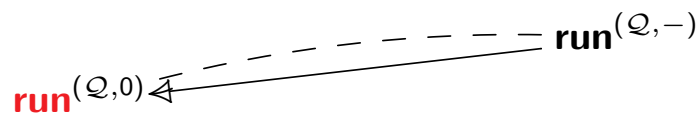
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

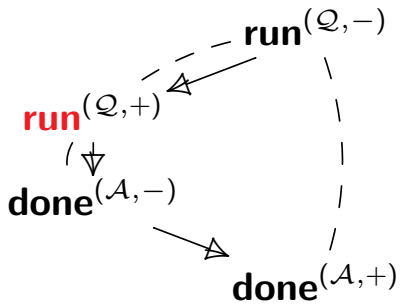
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**



An example of composition

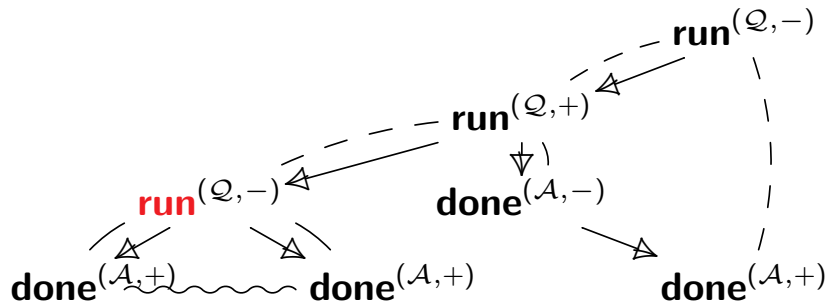
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



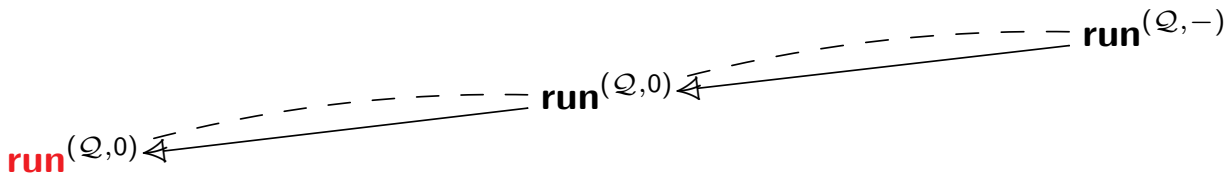
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

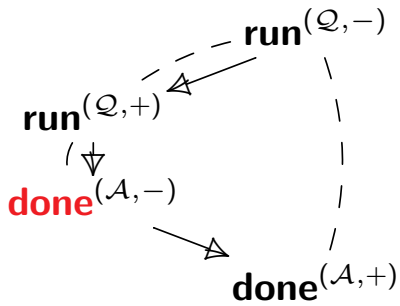
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**



An example of composition

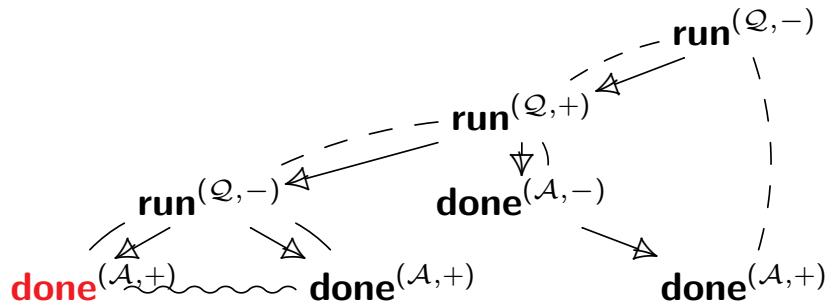
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ $\mathbf{com}$ $\to$ $\mathbf{com}$



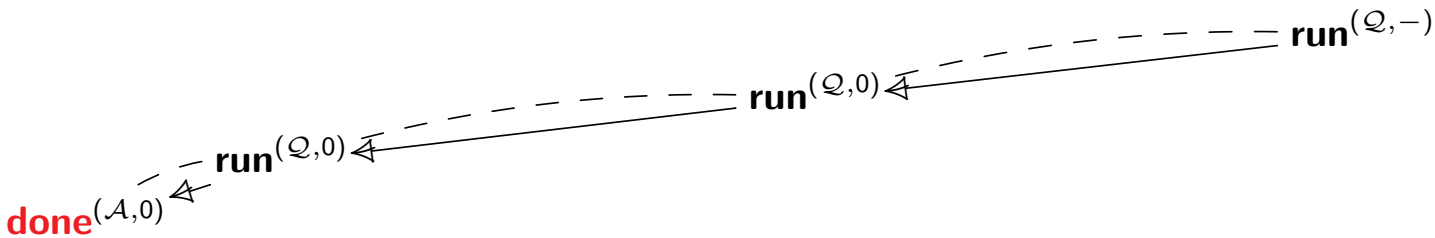
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ $(\mathbf{com}$ $\to$ $\mathbf{com})$ $\to$ $\mathbf{com}$



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

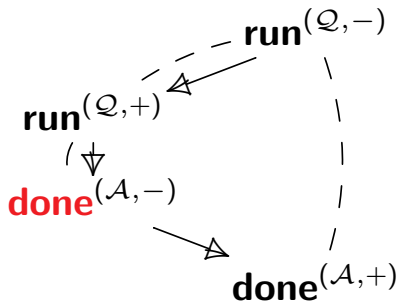
$\tau \circledast \sigma :$ $(\mathbf{com}$ $\to$ $\mathbf{com})$ $\to$ $\mathbf{com}$



An example of composition

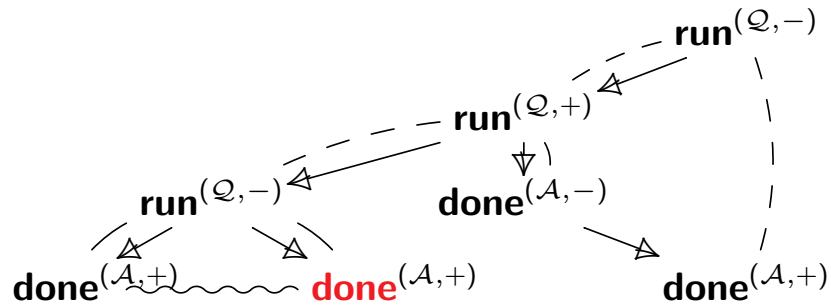
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



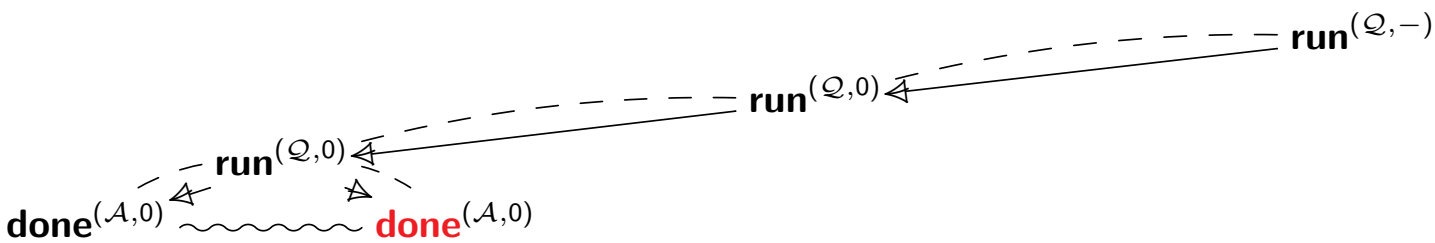
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

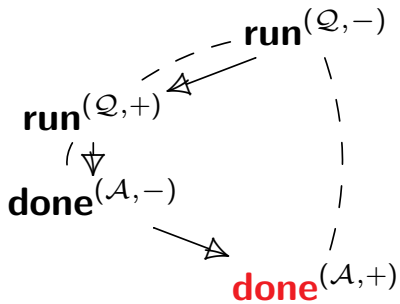
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**



An example of composition

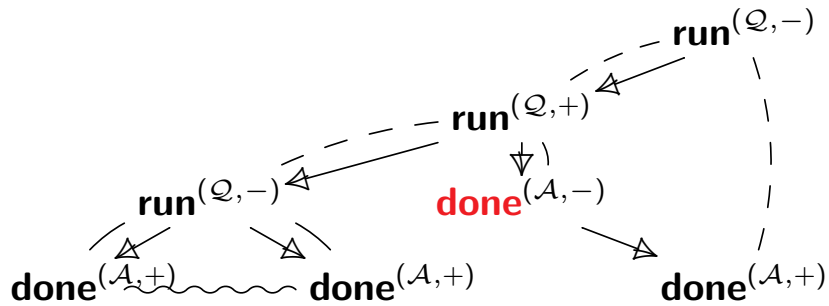
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



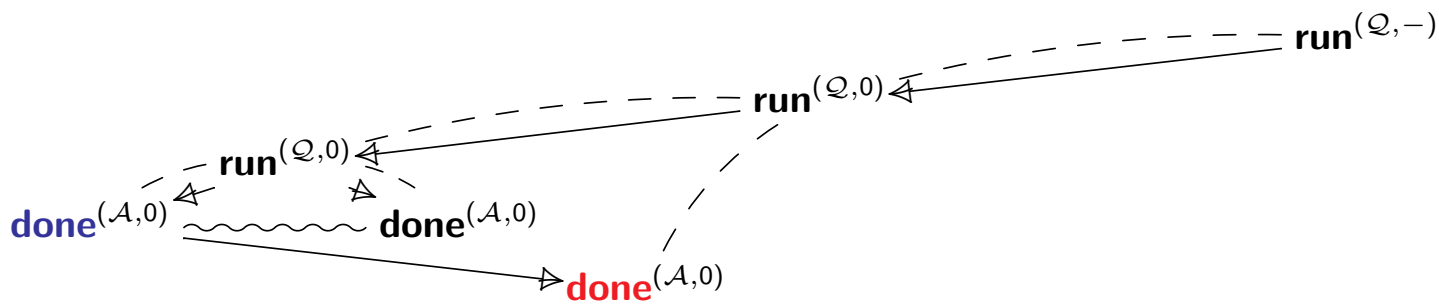
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

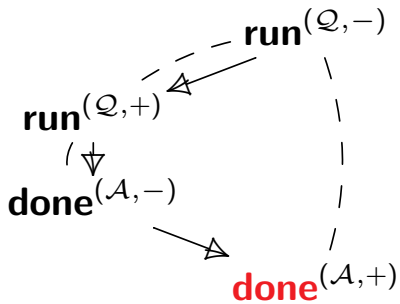
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**



An example of composition

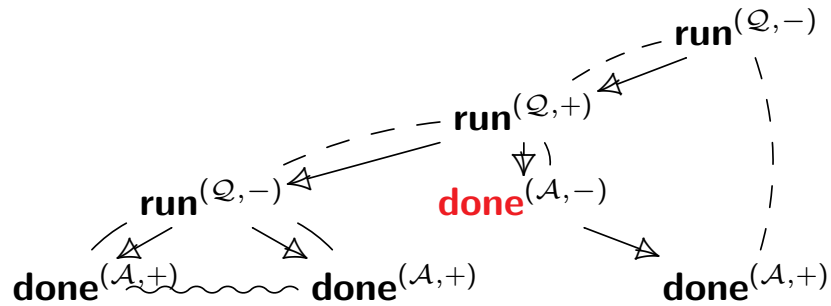
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



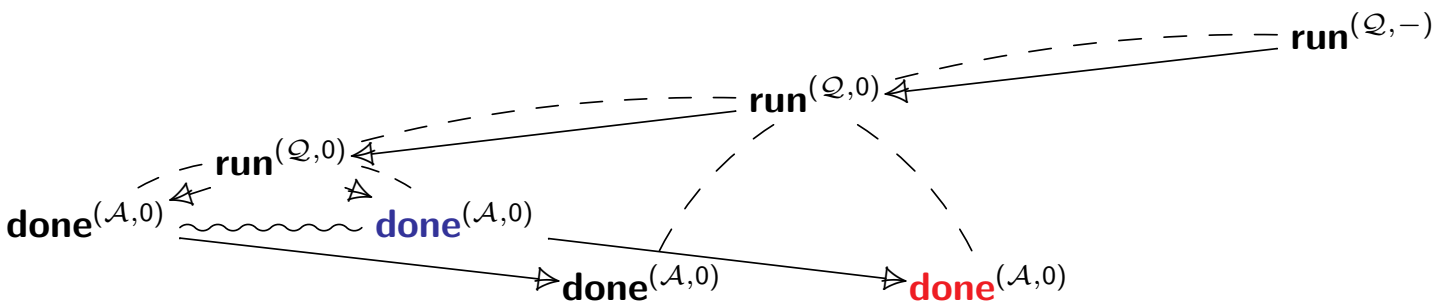
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

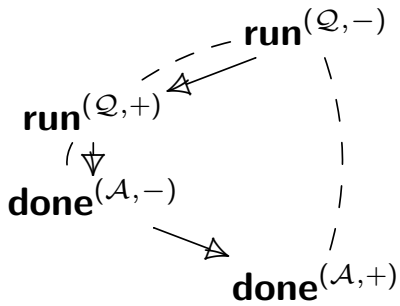
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**



An example of composition

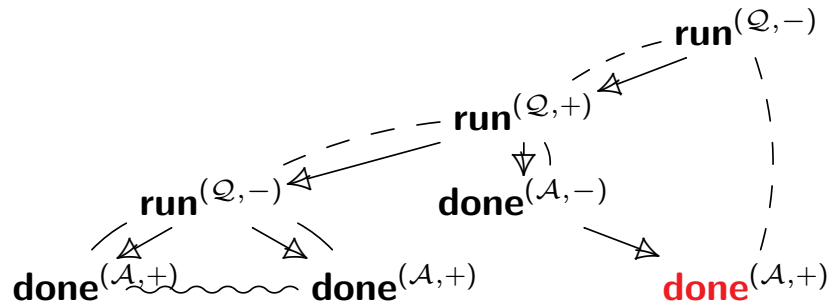
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ **com** $\to$ **com**



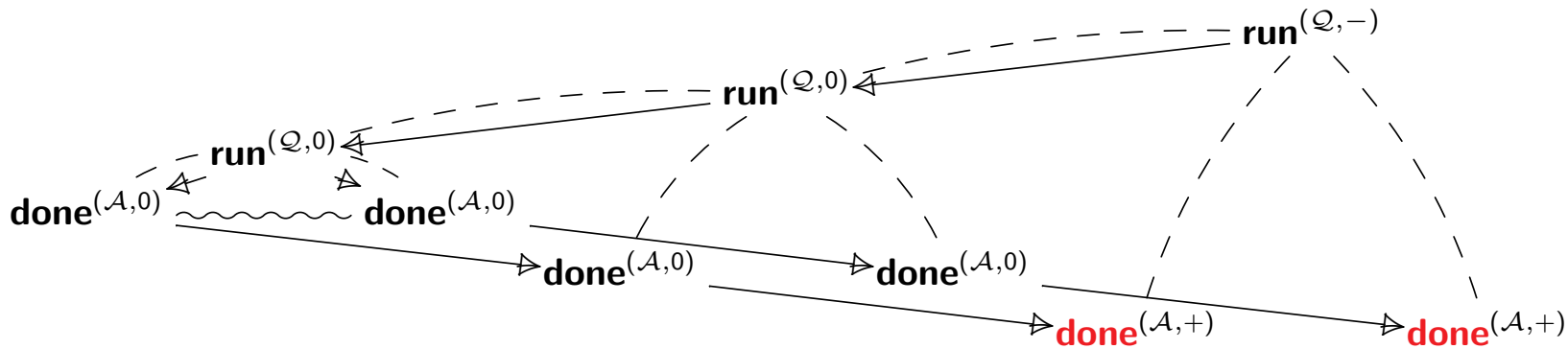
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ (**com** $\to$ **com**) $\to$ **com**



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\ (\mathbf{skip} + \mathbf{skip}))\ (\lambda x^{\mathbf{com}}.\, x)]\!] =$

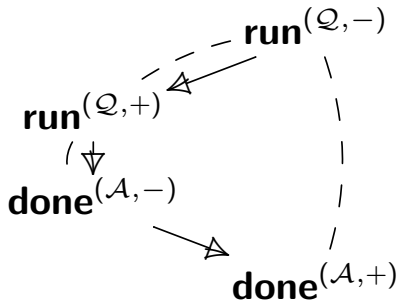
$\tau \circledast \sigma :$ (**com** $\to$ **com**) $\to$ **com**



An example of composition

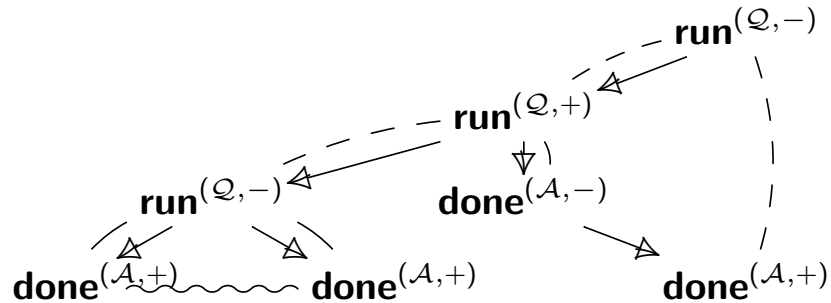
$[\![\lambda x^{\mathbf{com}}.\, x]\!] =$

$\sigma :$ $\quad \mathbf{com} \quad \to \quad \mathbf{com}$



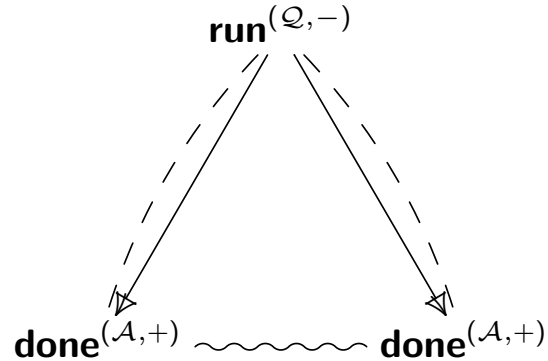
$[\![\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(\mathbf{skip} + \mathbf{skip})]\!] =$

$\tau :$ $\quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbf{com}$



$[\![(\lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(\mathbf{skip} + \mathbf{skip}))\,(\lambda x^{\mathbf{com}}.\, x)]\!] =$

$\tau \odot \sigma :$



The Cartesian Closed Category CHO

With a lot more work and details, we get:

Theorem

There is a Cartesian Closed Category, having the **usual Hyland-Ong arenas** *as objects, and as morphisms,* **single-threaded concurrent strategies**.

Proof.

Quite involved:

- Relies on a category of concurrent games by Rideau and Winskel for the basic composition mechanism.
- Add explicit copy indices for replications,
- Deal with uniformity problems.
- Introduce negativity and single-threadedness.

□

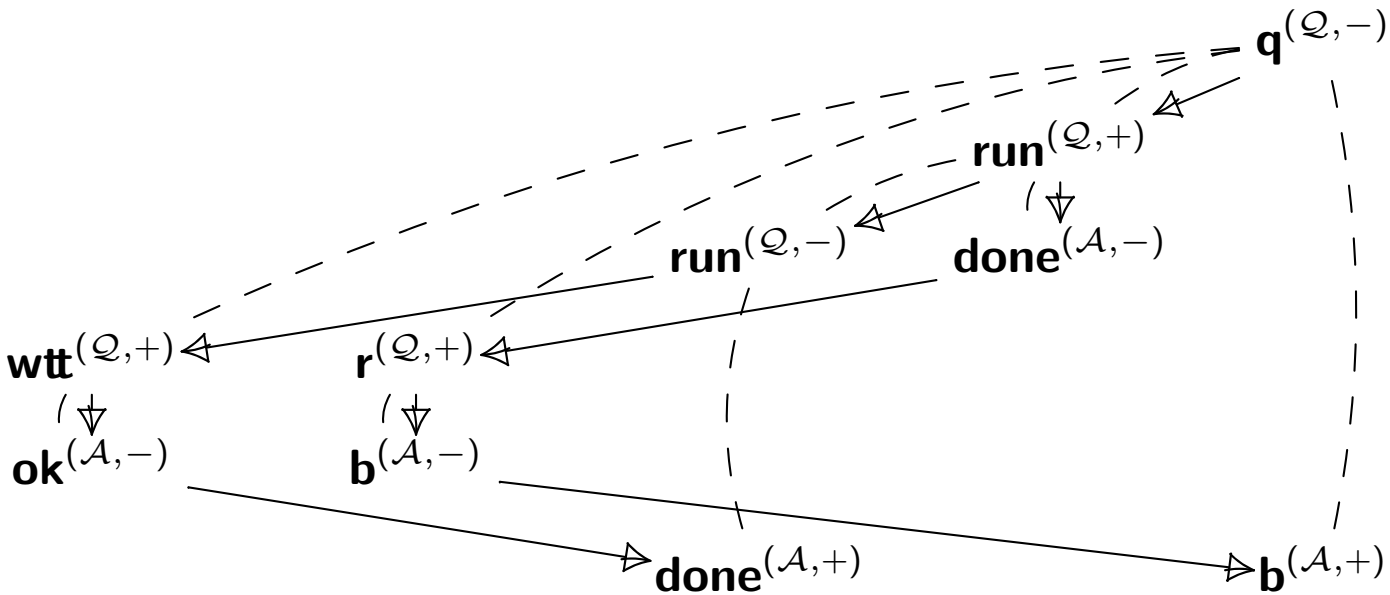
Terms of PCF are interpreted by the O-branching forest of their P-views, *i.e.* the **causal** representation.

Interpreting state (1/3)

As a **first step**, we leave the variables un-interpreted.

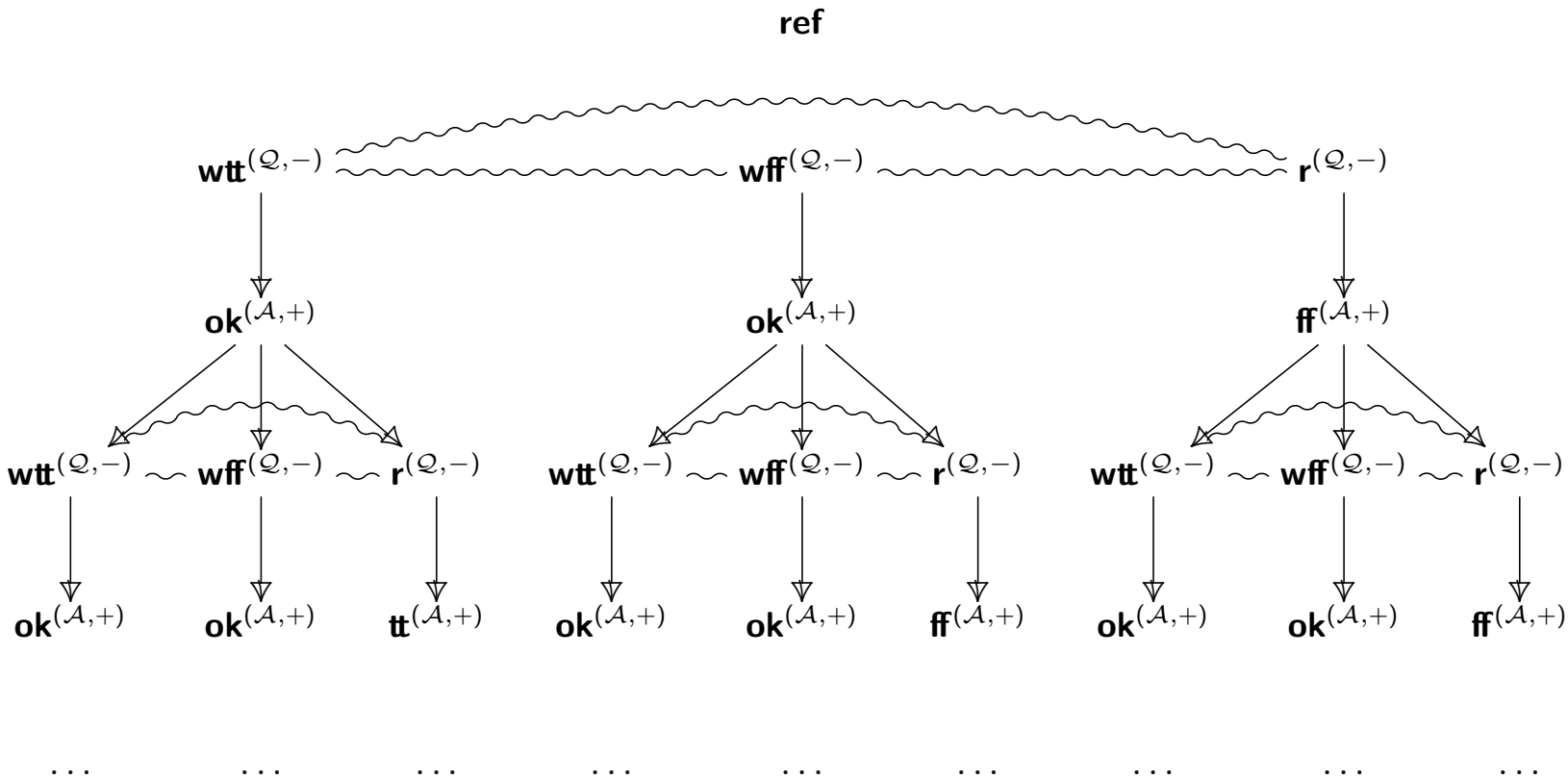
$$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$$

$$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$$



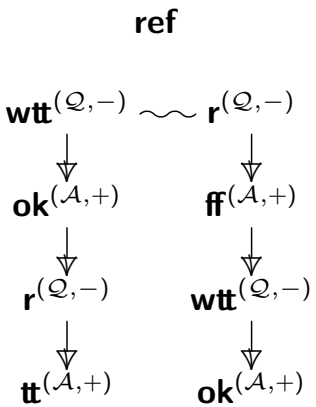
Interpreting state (2/3)

As a **second step**, we compose with the **sequential memory cell**.



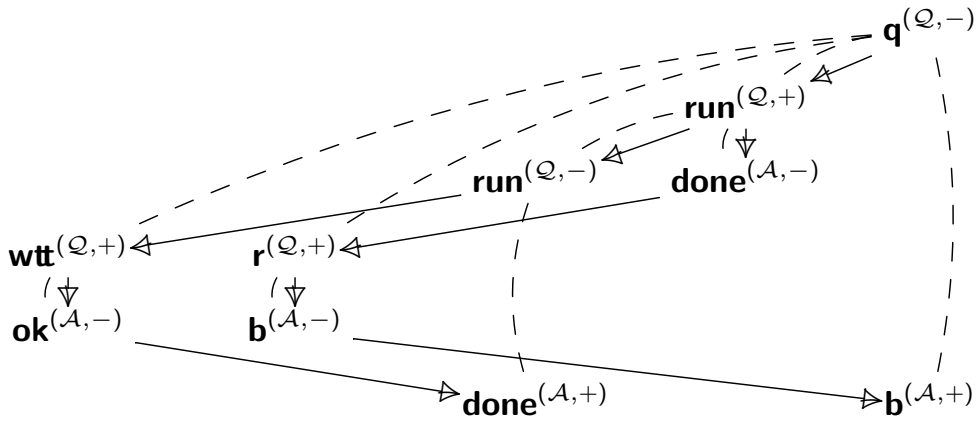
Interpreting state (3/3)

$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$

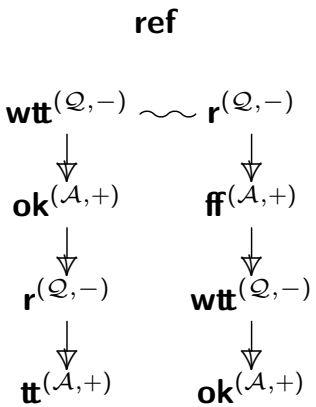


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$

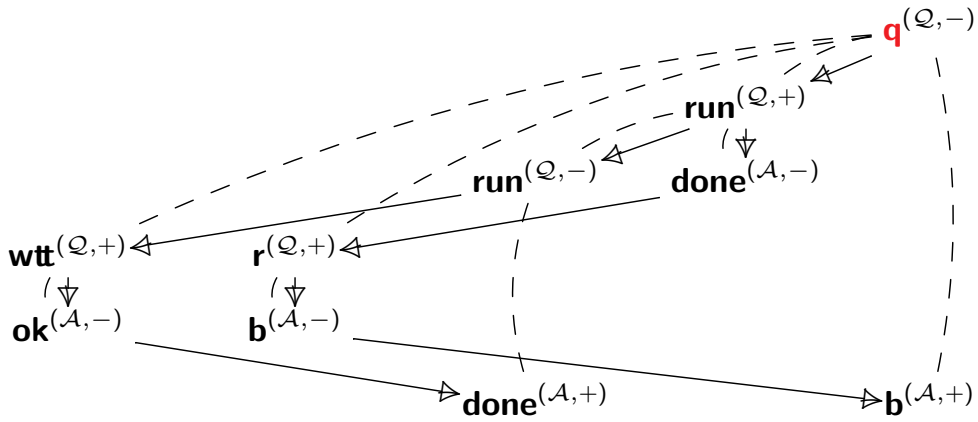
Interpreting state (3/3)

$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$



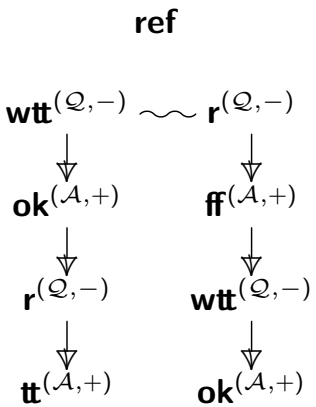
$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$

$\mathbf{q}^{(\mathcal{Q},-)}$

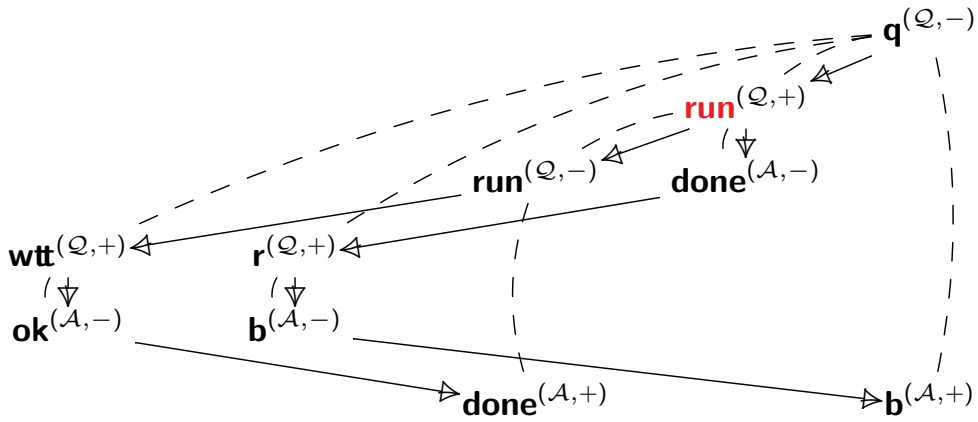
Interpreting state (3/3)

$\mathbf{cell} =$



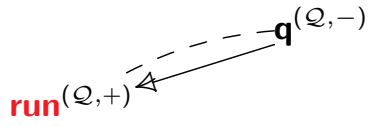
$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \to (\mathbf{com} \to \mathbf{com}) \to \mathbb{B}$



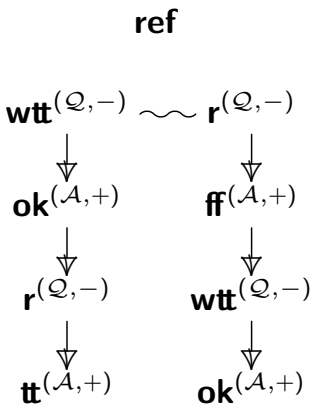
$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \to (\mathbf{com} \to \mathbf{com}) \to \mathbb{B}$



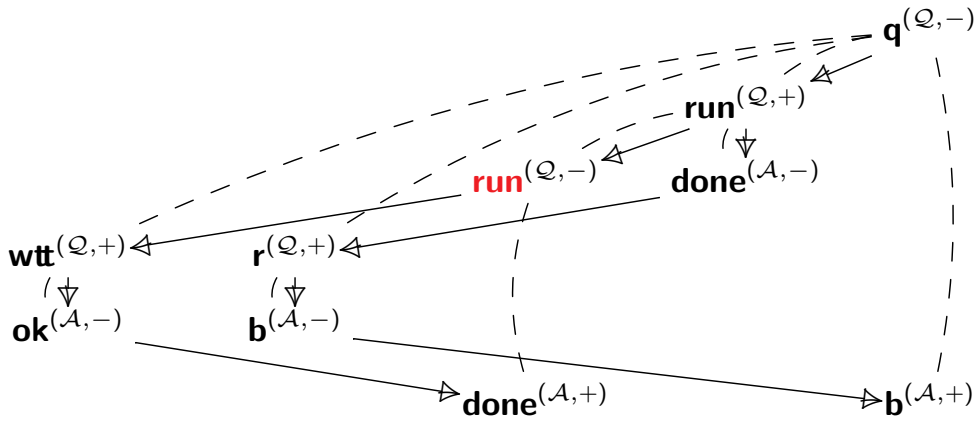
Interpreting state (3/3)

$\mathbf{cell} =$



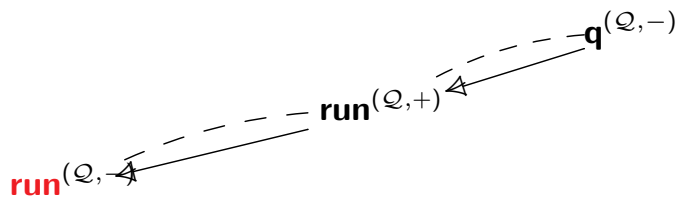
$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \to (\mathbf{com} \to \mathbf{com}) \to \mathbb{B}$



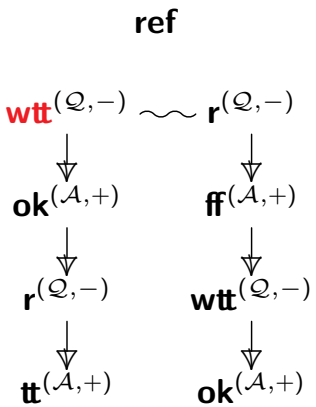
$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \to (\mathbf{com} \to \mathbf{com}) \to \mathbb{B}$

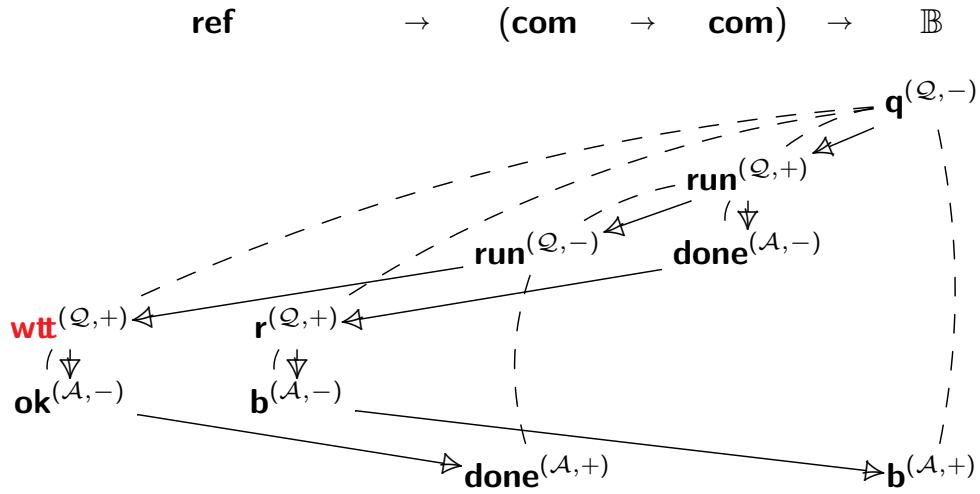


Interpreting state (3/3)

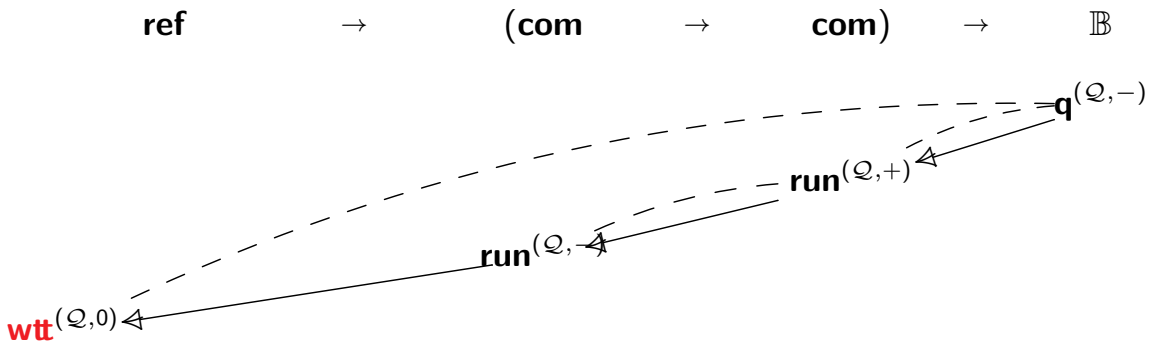
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

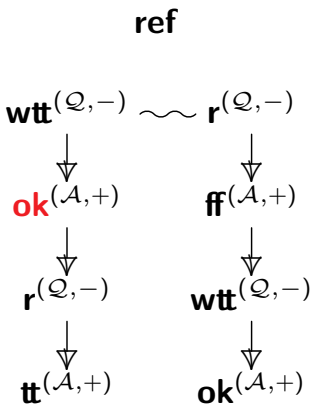


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

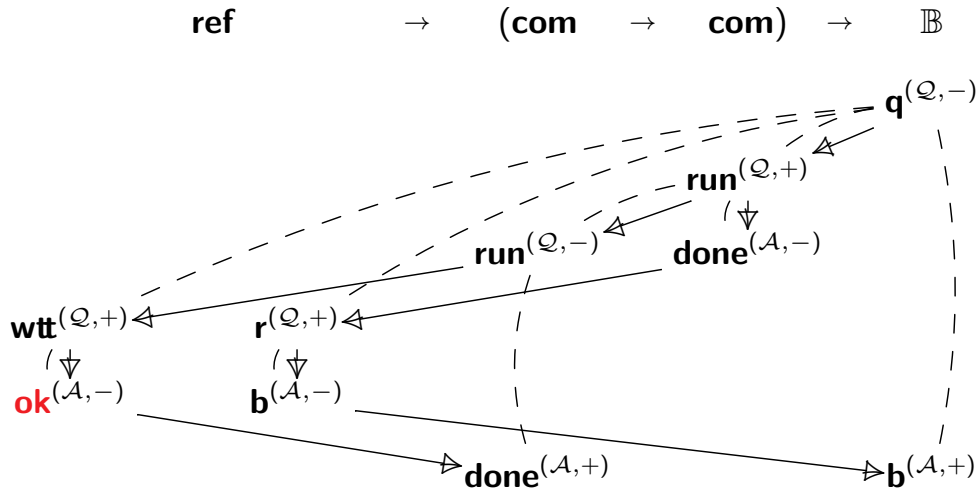


Interpreting state (3/3)

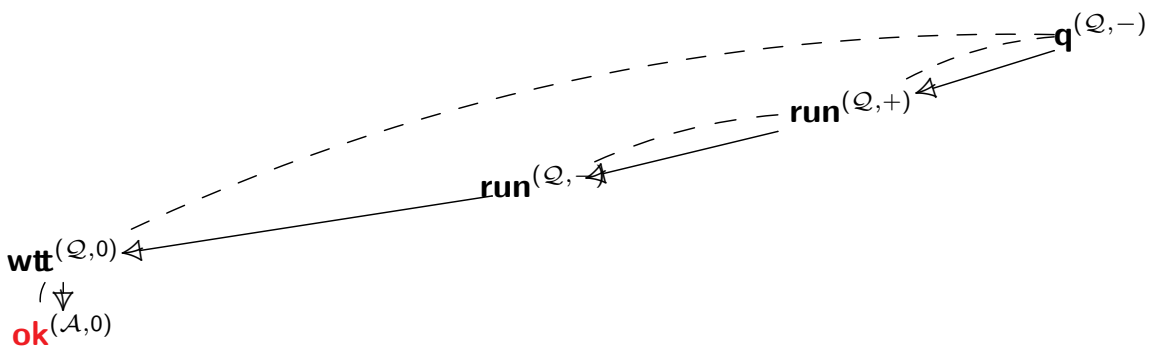
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

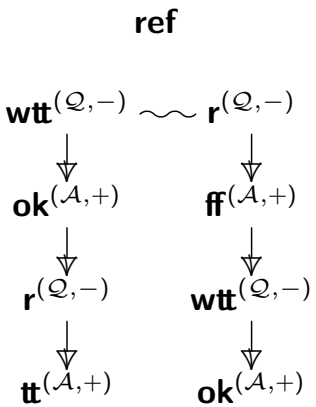


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

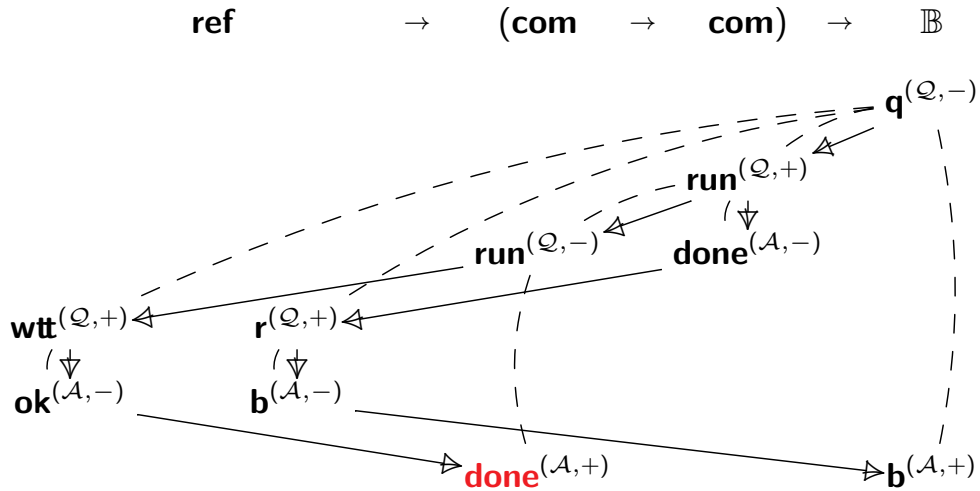


Interpreting state (3/3)

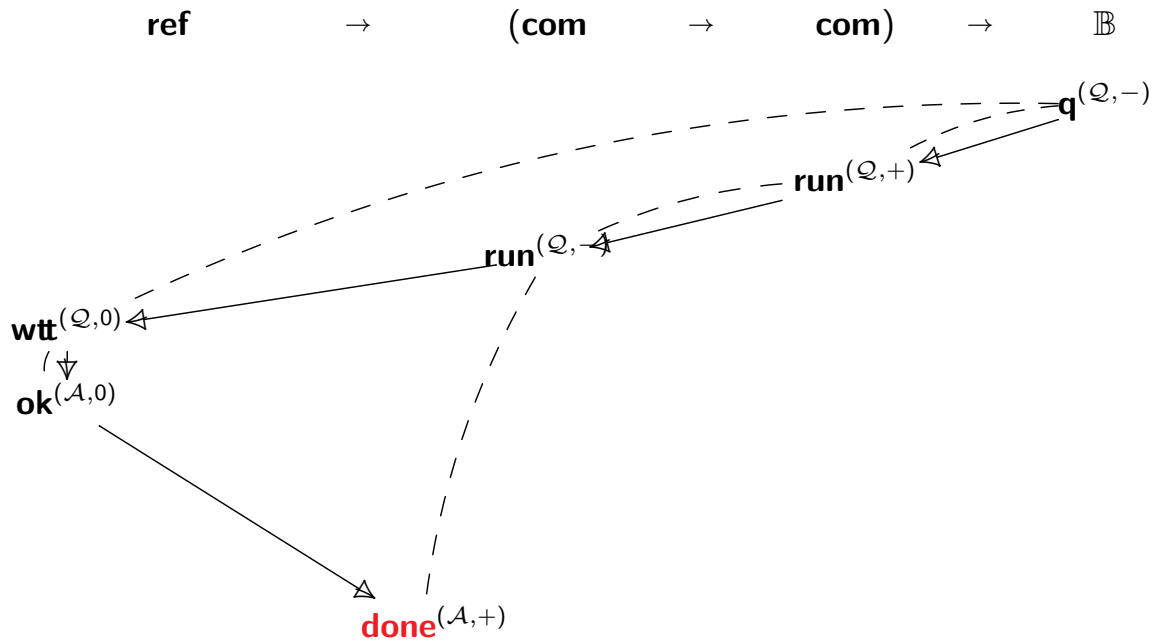
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

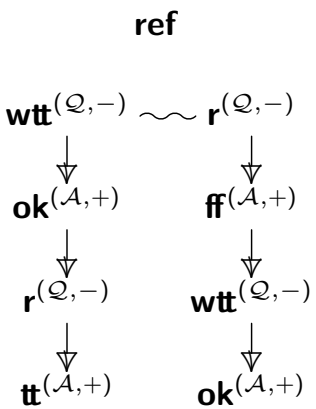


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

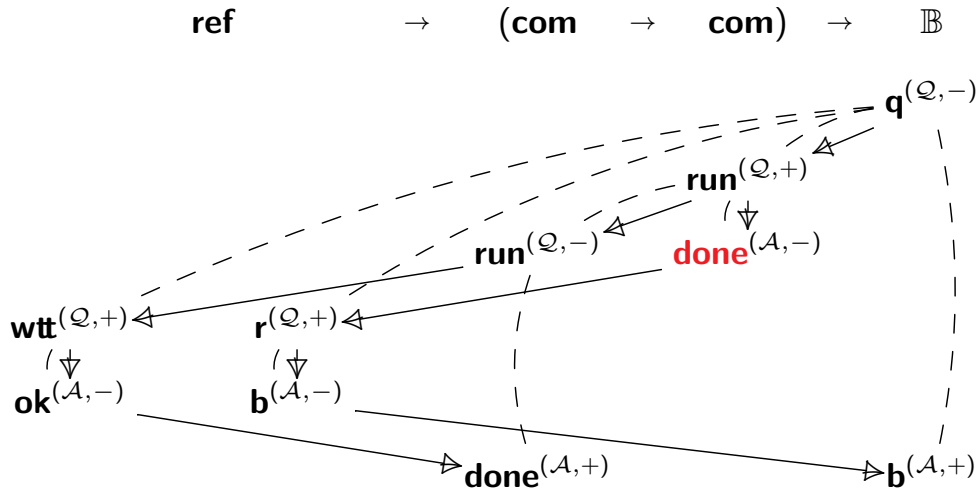


Interpreting state (3/3)

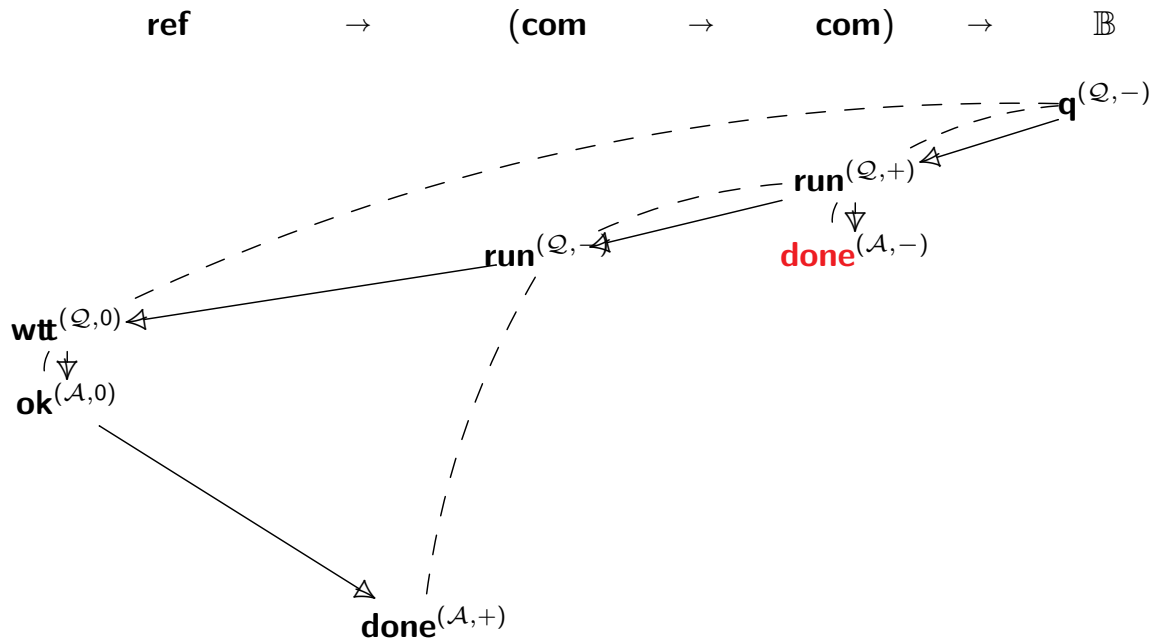
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

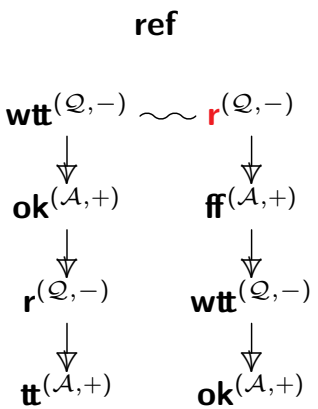


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$



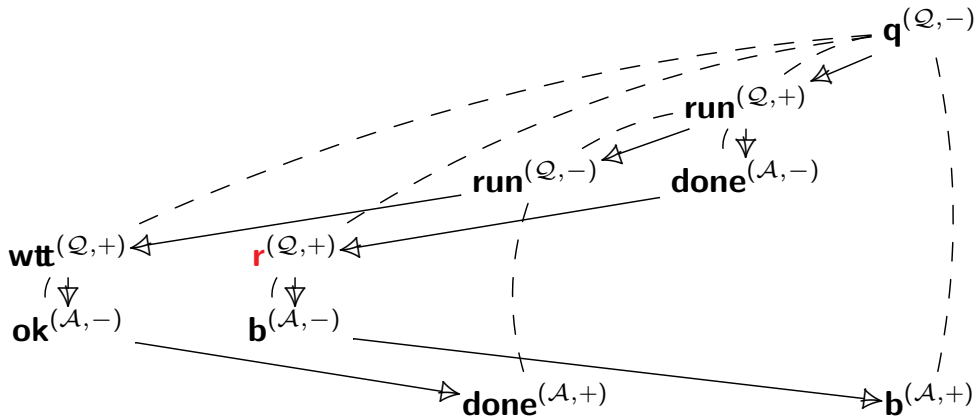
Interpreting state (3/3)

$\mathbf{cell} =$



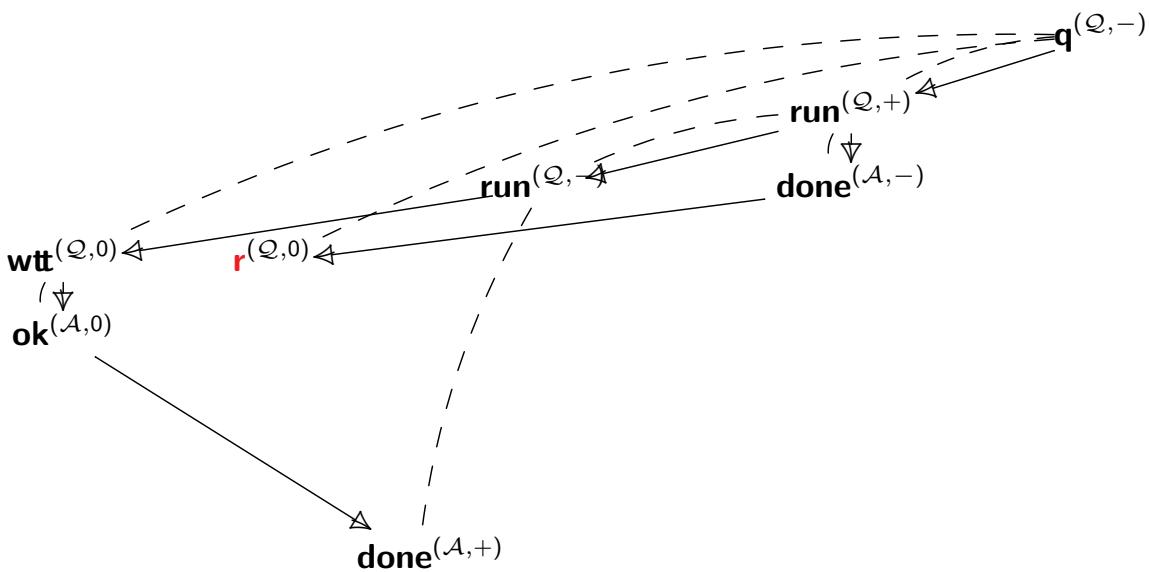
$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$



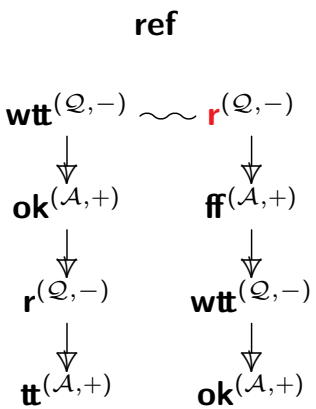
$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

$\mathbf{ref} \quad \to \quad (\mathbf{com} \quad \to \quad \mathbf{com}) \quad \to \quad \mathbb{B}$

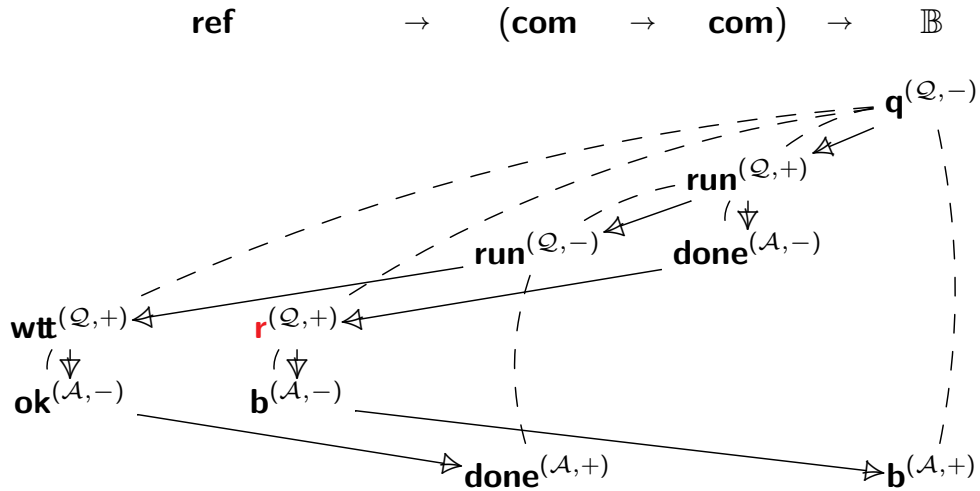


Interpreting state (3/3)

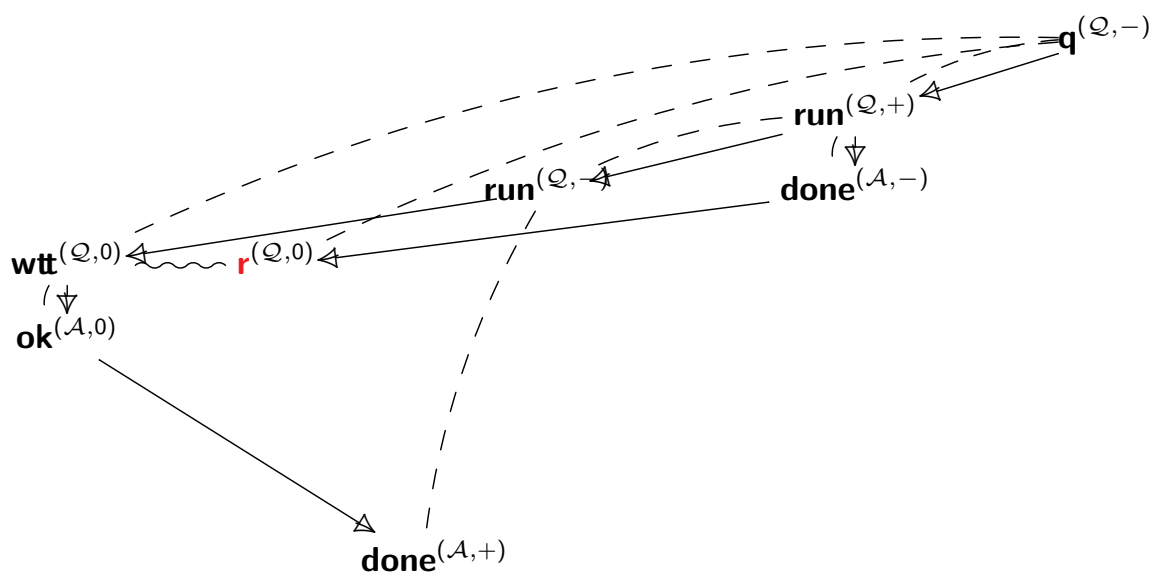
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

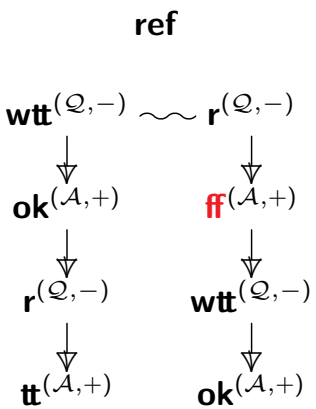


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

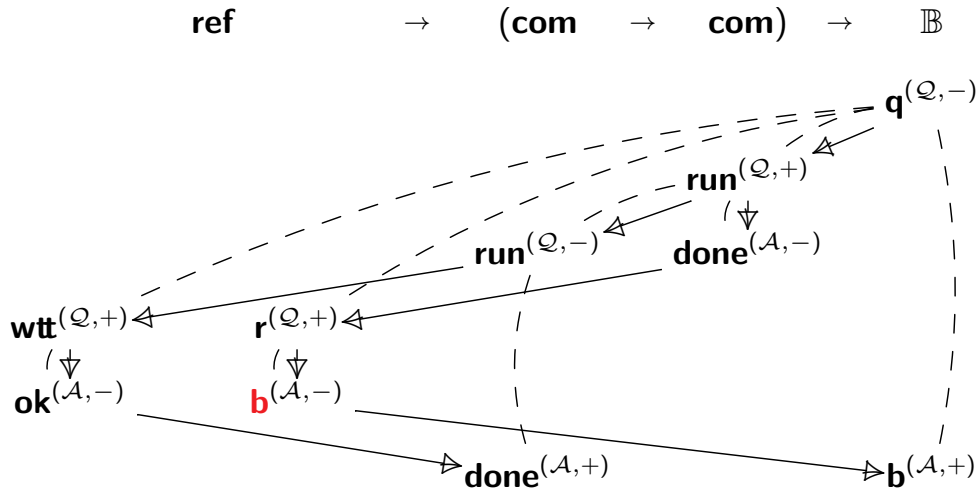


Interpreting state (3/3)

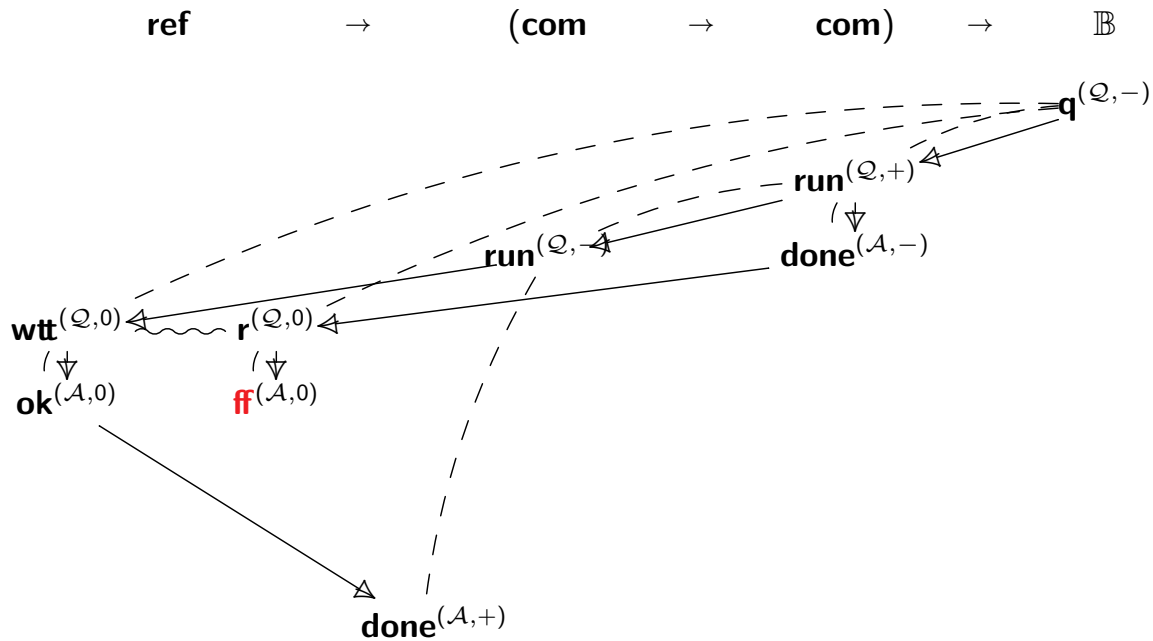
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

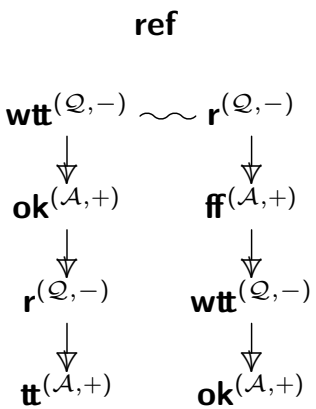


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

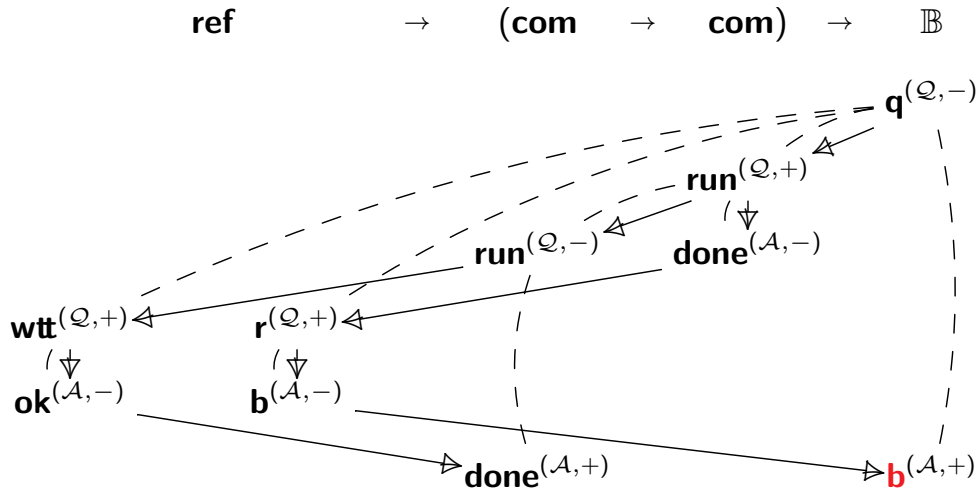


Interpreting state (3/3)

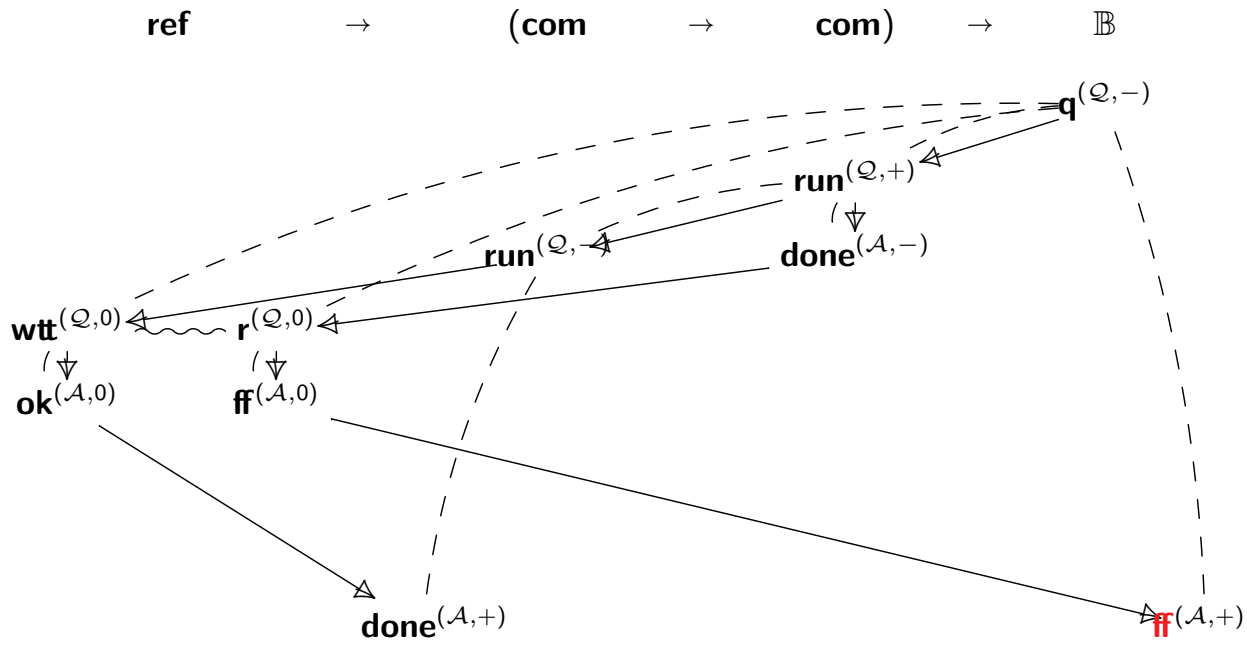
$\mathbf{cell} =$



$[\![\lambda r^{\mathbf{ref}}\, f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

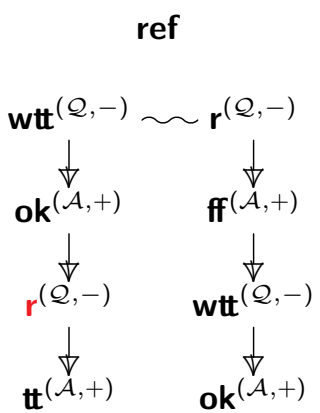


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathbf{com}\to\mathbf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

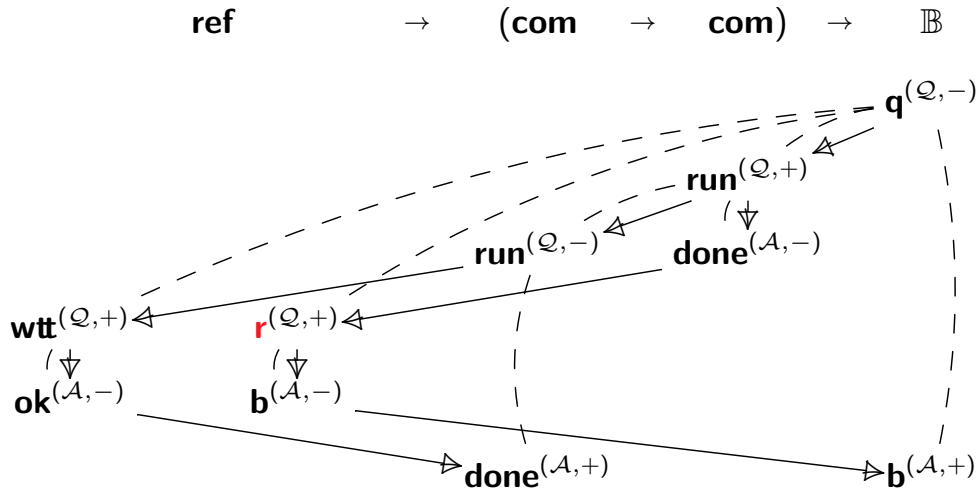


Interpreting state (3/3)

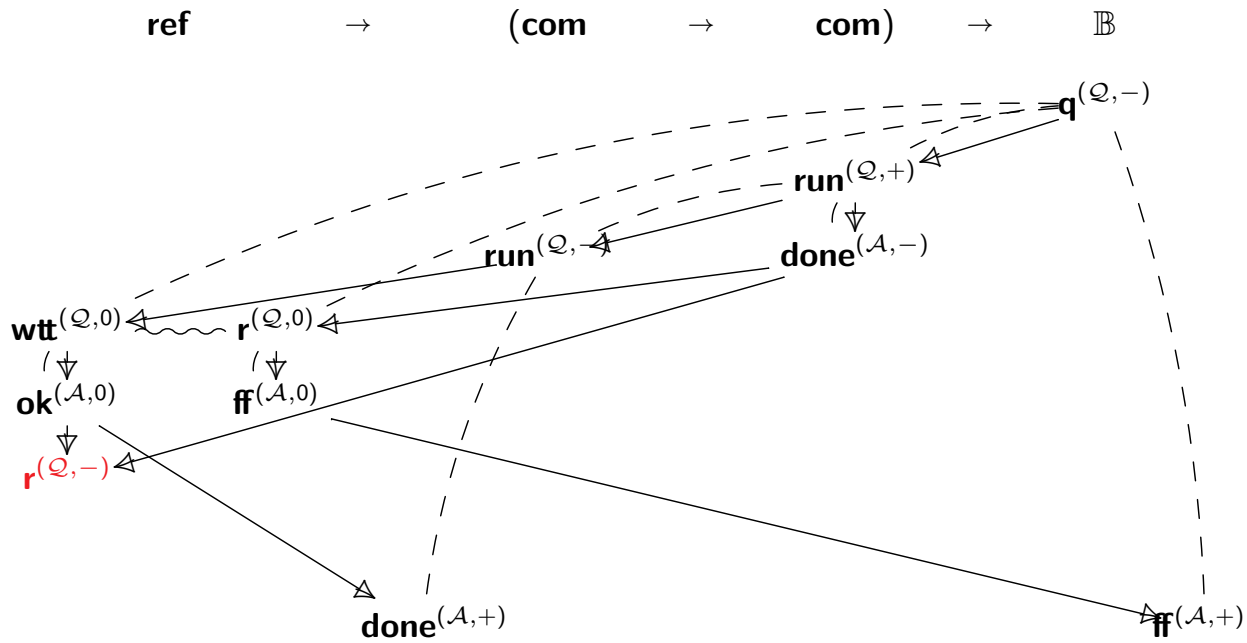
cell $=$



$[\![\lambda r^{\mathsf{ref}}\, f^{\mathsf{com}\to\mathsf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

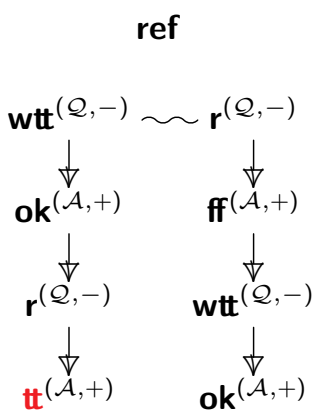


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathsf{com}\to\mathsf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$

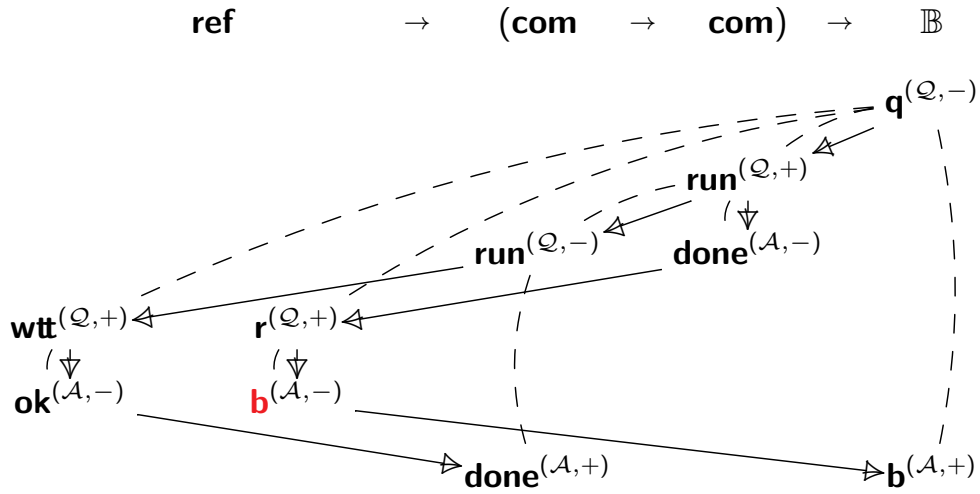


Interpreting state (3/3)

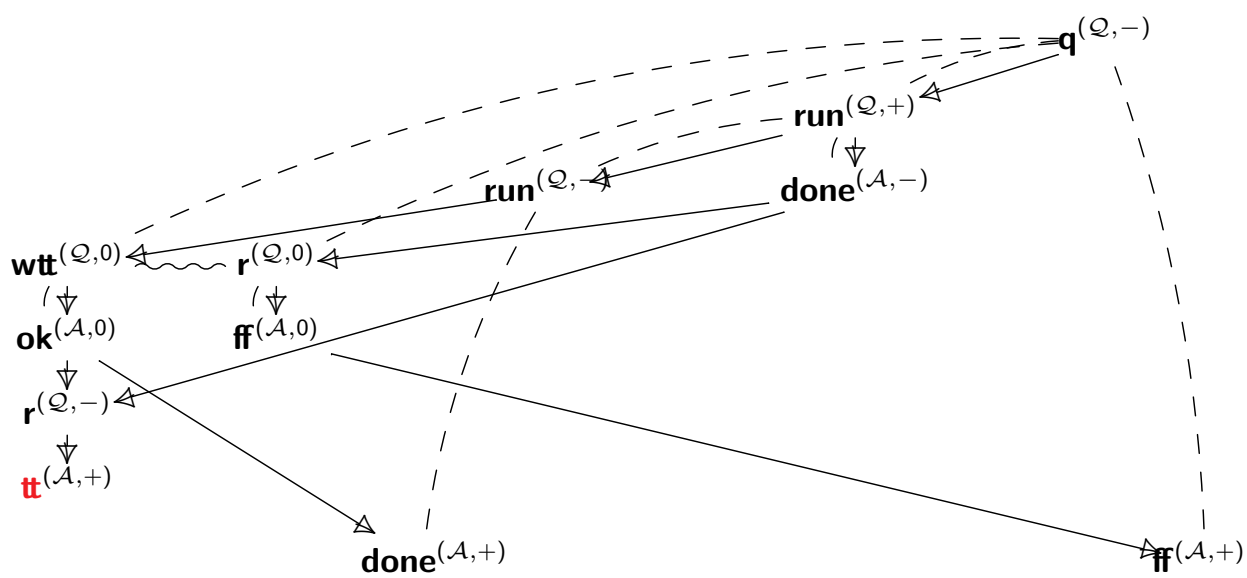
cell =



$[\![\lambda r^{\mathsf{ref}}\, f^{\mathsf{com}\to\mathsf{com}}.\, f\,(r := \mathsf{tt});\, !r]\!]$ =

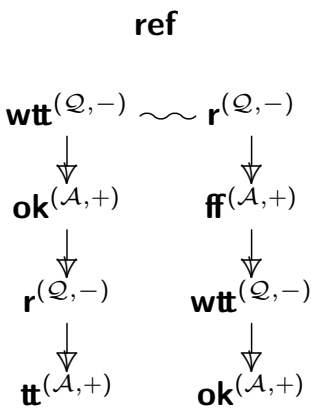


$[\![\mathsf{newref}\ r := \mathsf{ff}\ \mathsf{in}\ \lambda f^{\mathsf{com}\to\mathsf{com}}.\, f\,(r := \mathsf{tt});\, !r]\!]$ =

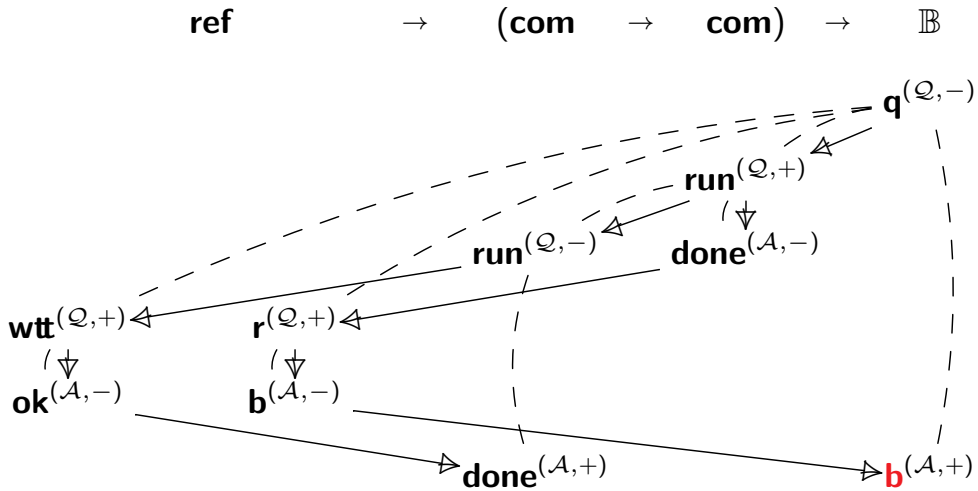


Interpreting state (3/3)

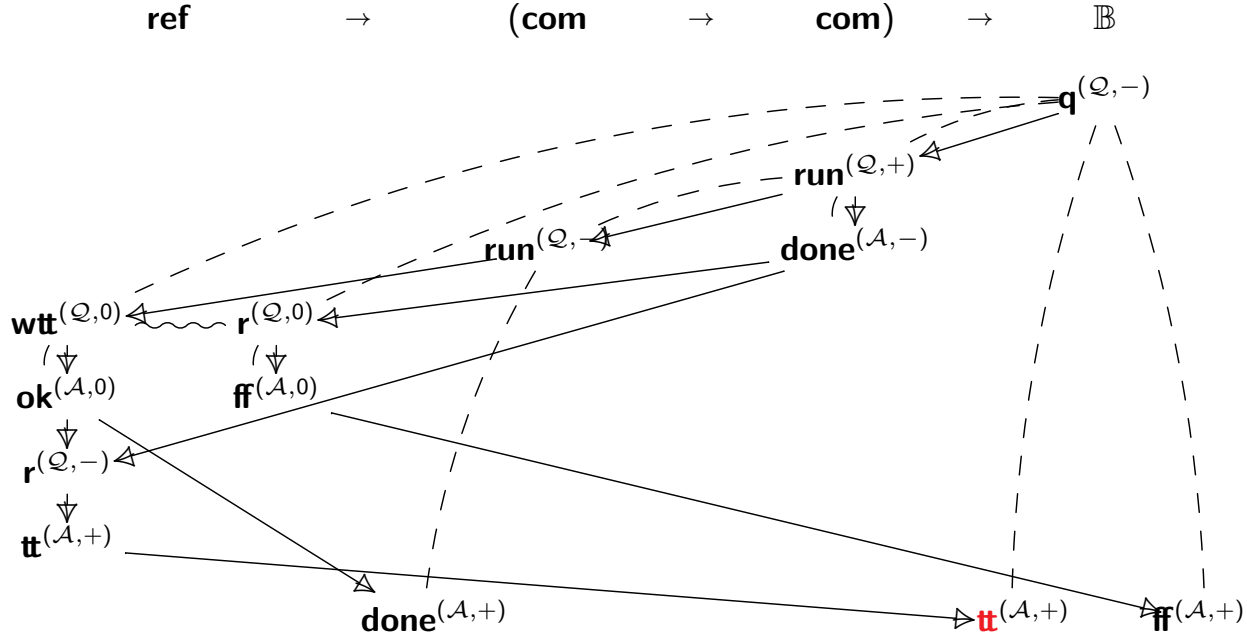
cell =



$[\![\lambda r^{\mathsf{ref}}\, f^{\mathsf{com}\to\mathsf{com}}.\, f\ (r := \mathsf{tt});\ !r]\!]$ =

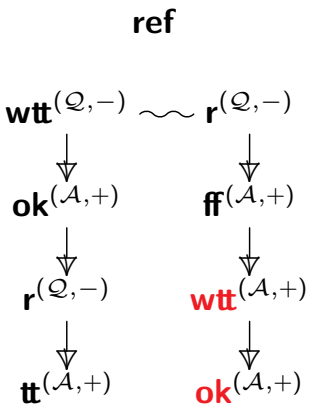


$[\![\mathsf{newref}\ r := \mathsf{ff}\ \mathsf{in}\ \lambda f^{\mathsf{com}\to\mathsf{com}}.\, f\ (r := \mathsf{tt});\ !r]\!]$ =

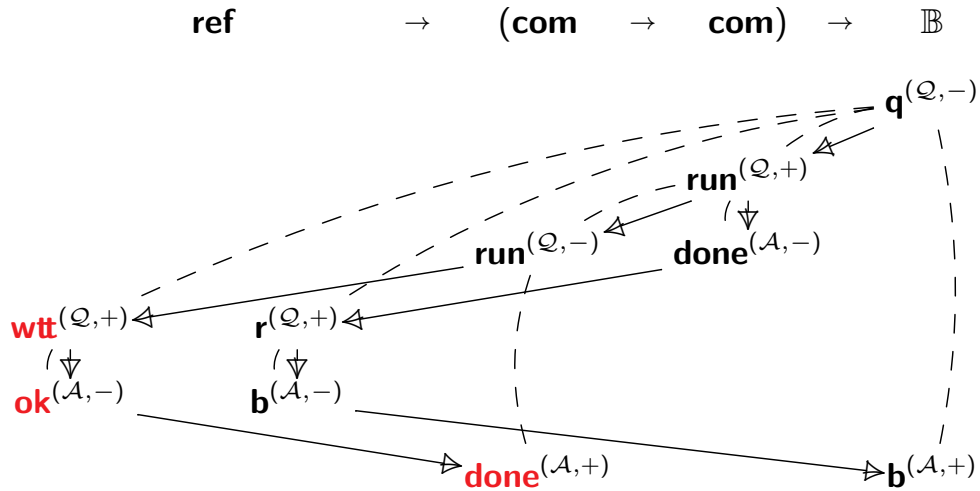


Interpreting state (3/3)

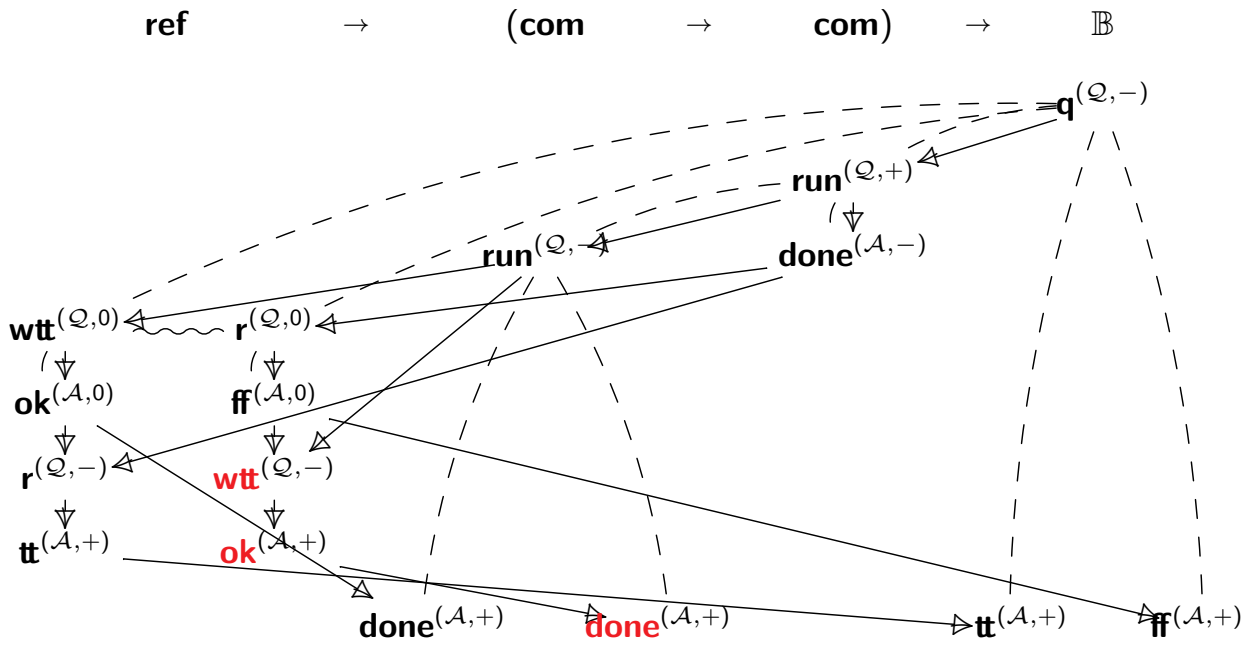
cell =



$[\![\lambda r^{\mathsf{ref}}\, f^{\mathsf{com}\to\mathsf{com}}.\, f\ (r := \mathbf{tt});\ !r]\!] =$

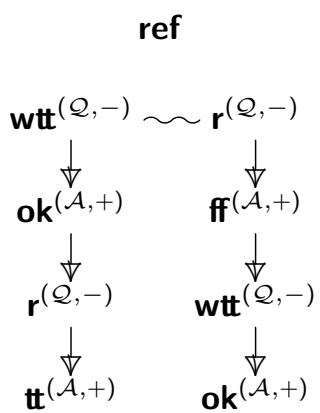


$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda f^{\mathsf{com}\to\mathsf{com}}.\, f\ (r := \mathbf{tt});\ !r]\!] =$

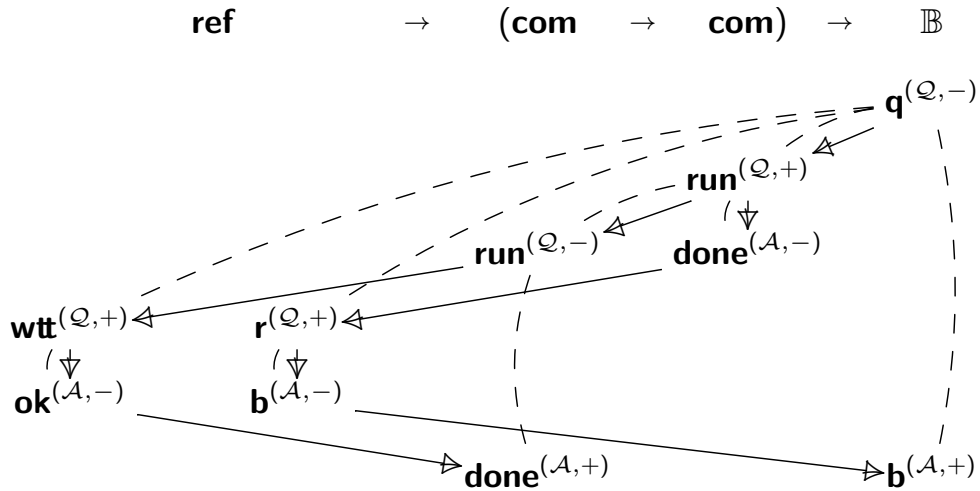


Interpreting state (3/3)

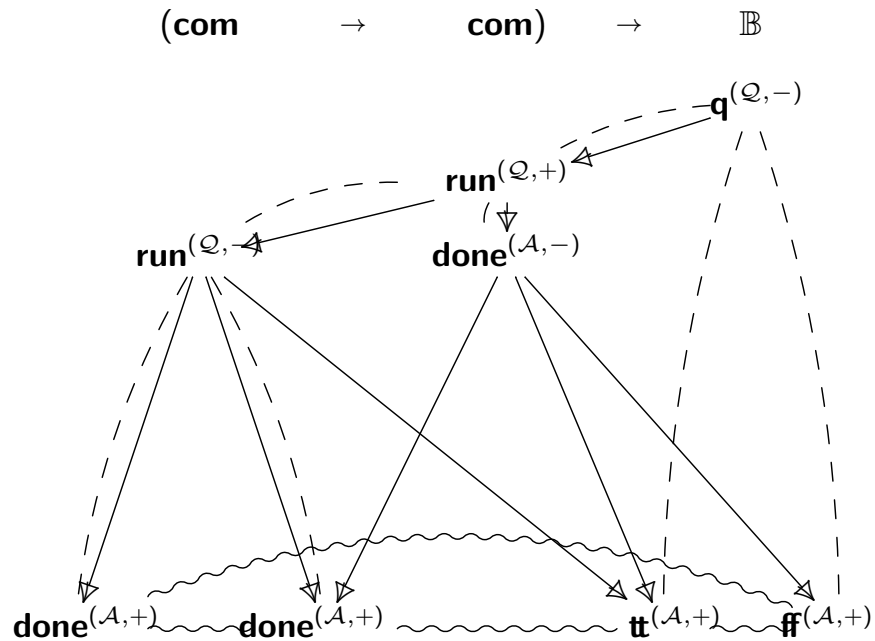
cell $=$



$[\![\lambda r^{\mathsf{ref}}\, f^{\mathsf{com}\to\mathsf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$



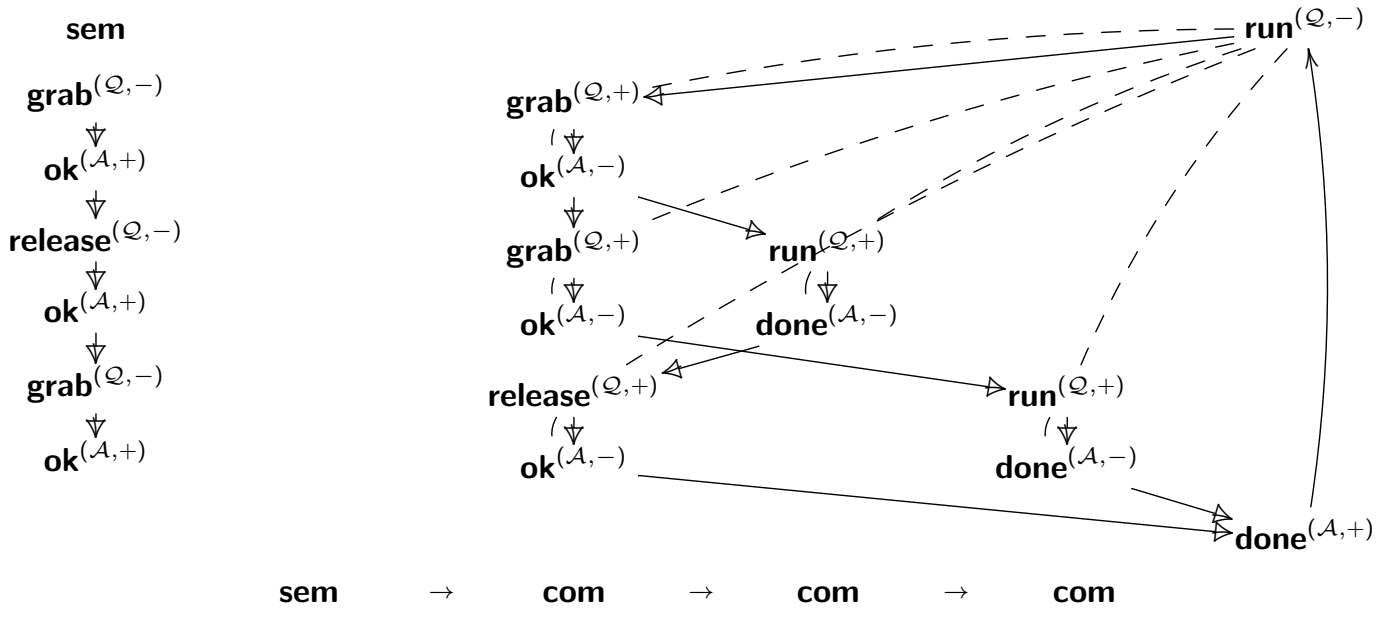
$[\![\mathbf{newref}\; r := \mathbf{ff}\; \mathbf{in}\; \lambda f^{\mathsf{com}\to\mathsf{com}}.\, f\,(r := \mathbf{tt});\, !r]\!] =$



Example: sequential composition through semaphores

$[\![\lambda s^{\mathsf{sem}}, x^{\mathsf{com}}, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$

sem → **com** → **com** → **com**

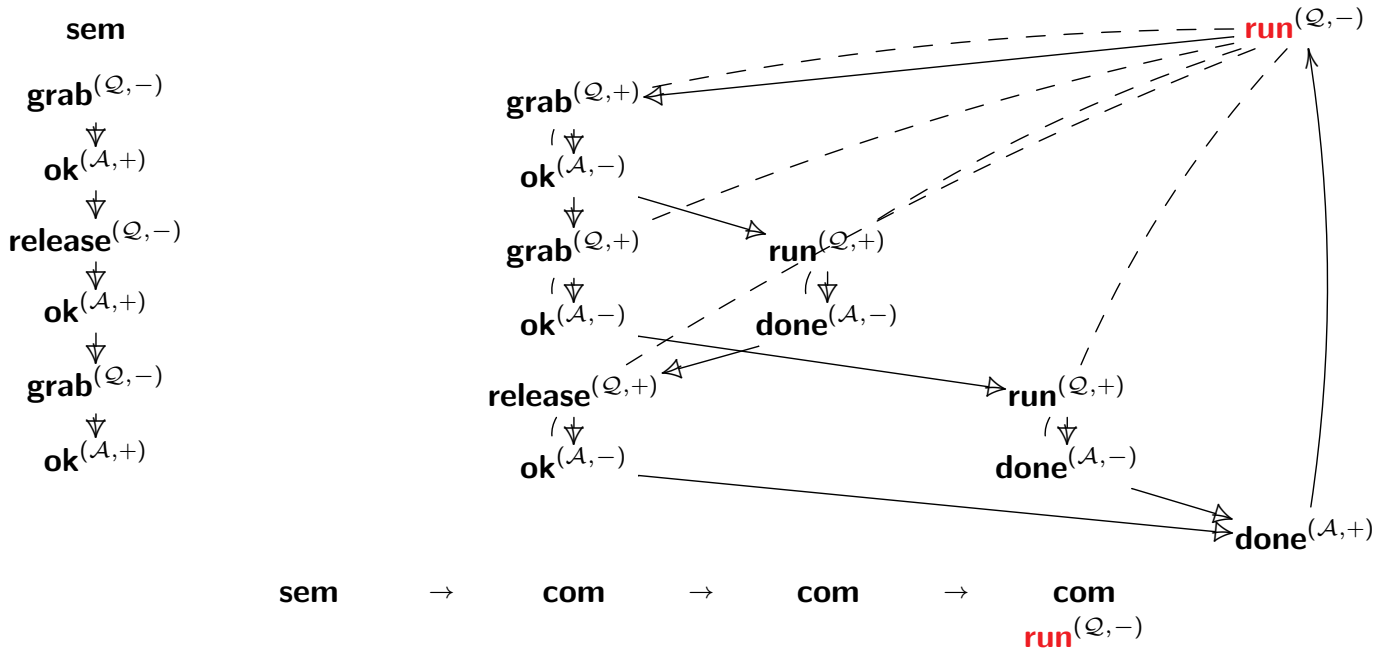


sem → **com** → **com** → **com**

Example: sequential composition through semaphores

$[\![\lambda s^{\mathsf{sem}}, x^{\mathsf{com}}, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$

sem → **com** → **com** → **com**

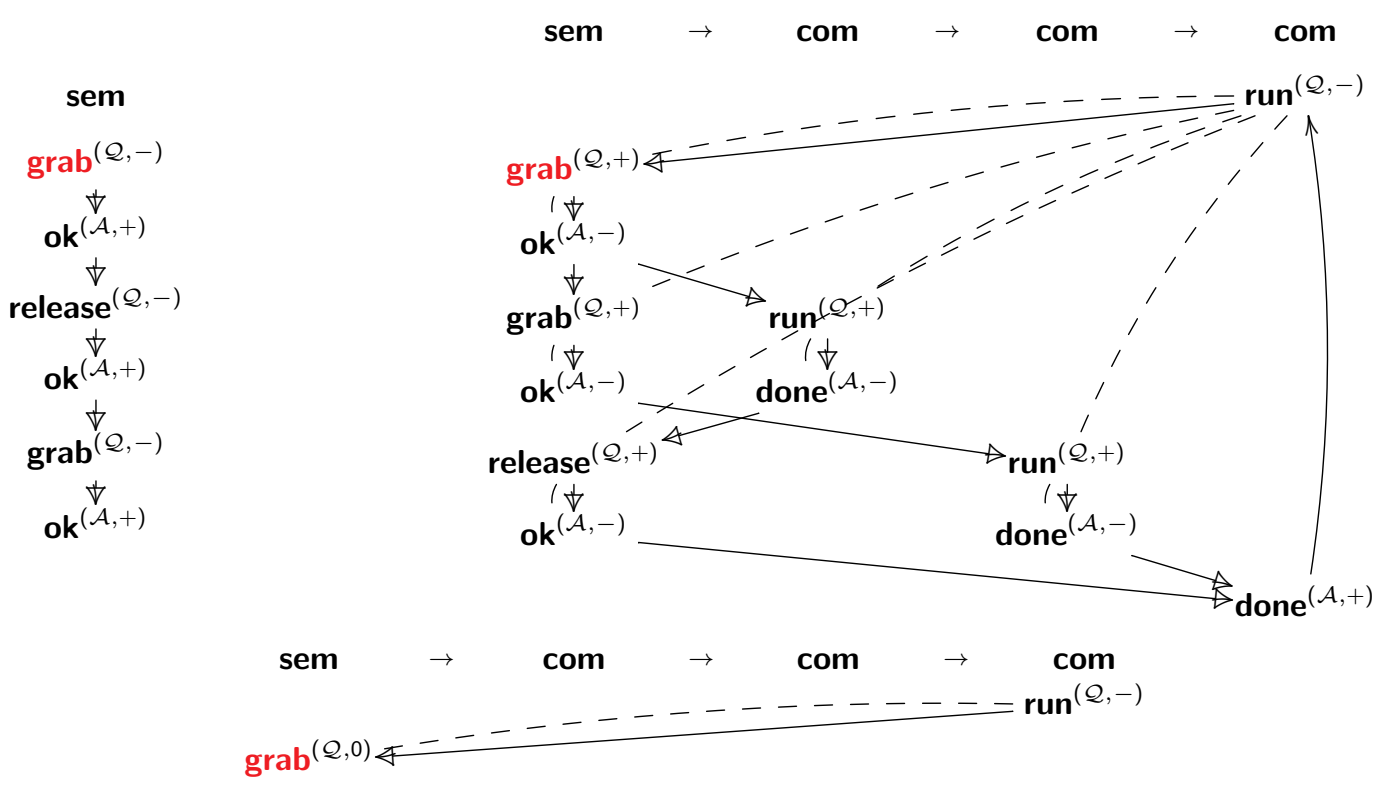


sem → **com** → **com** → **com**

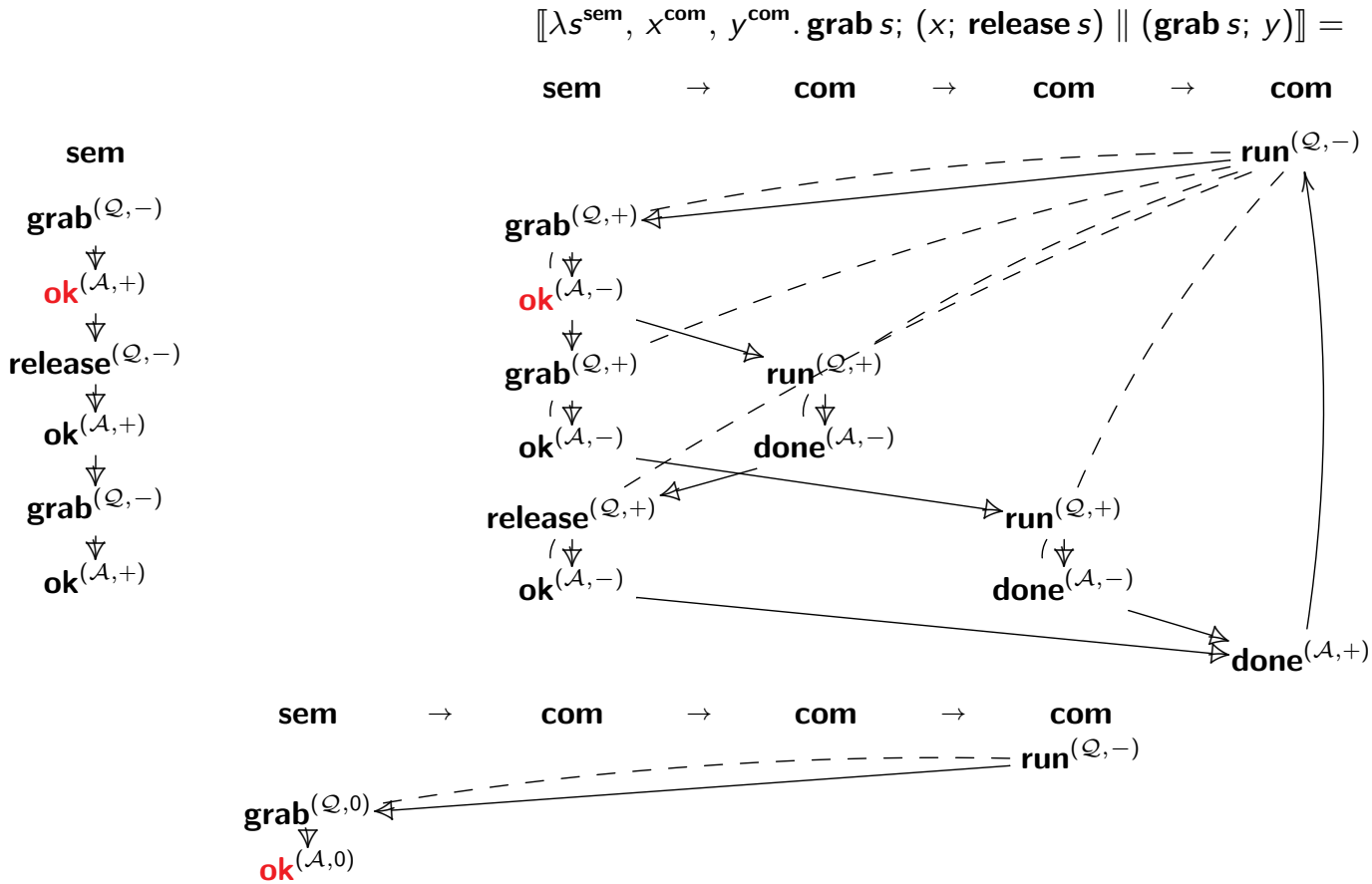
run$^{(\mathcal{Q},-)}$

Example: sequential composition through semaphores

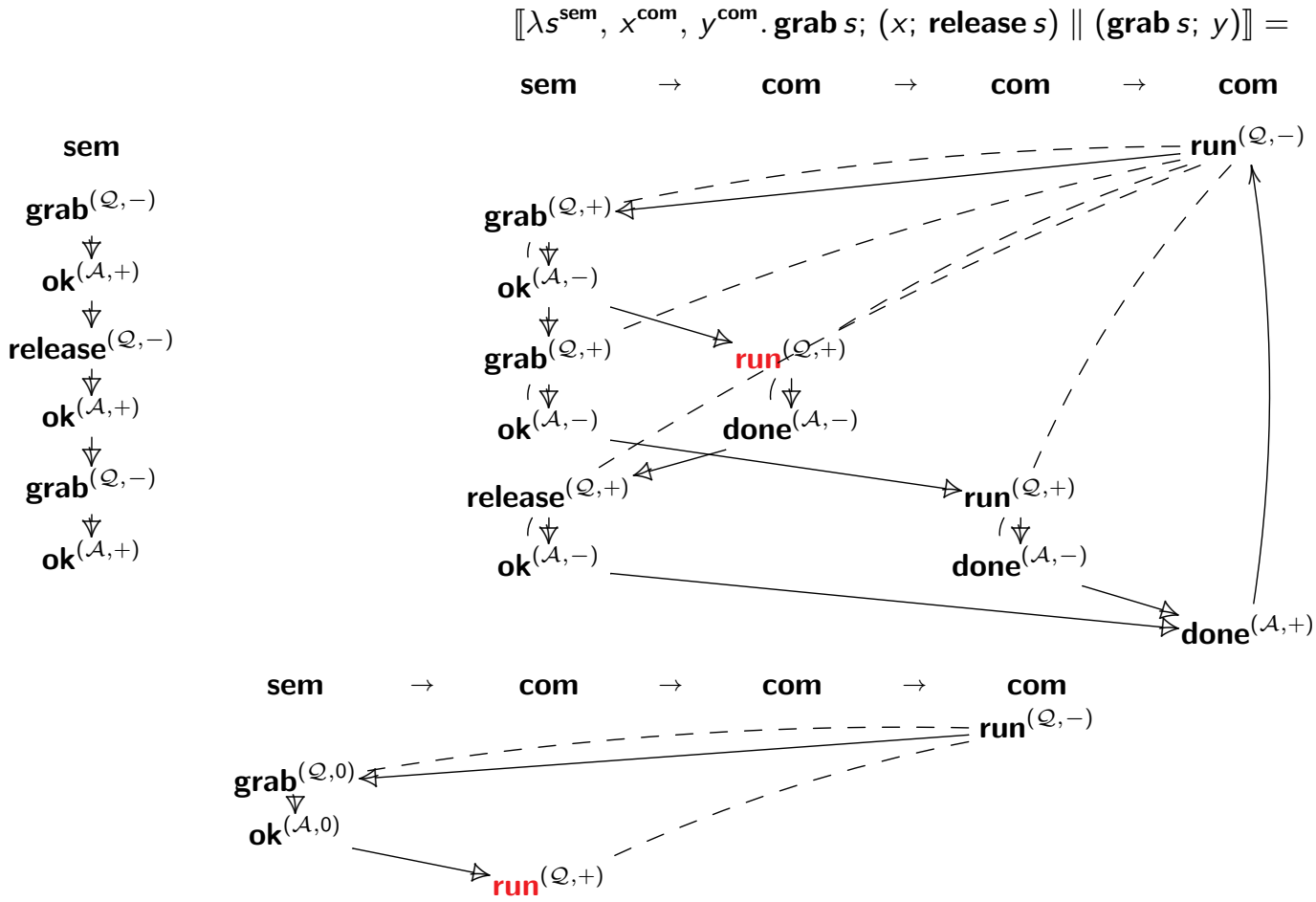
$[\![\lambda s^{\mathsf{sem}},\, x^{\mathsf{com}},\, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$



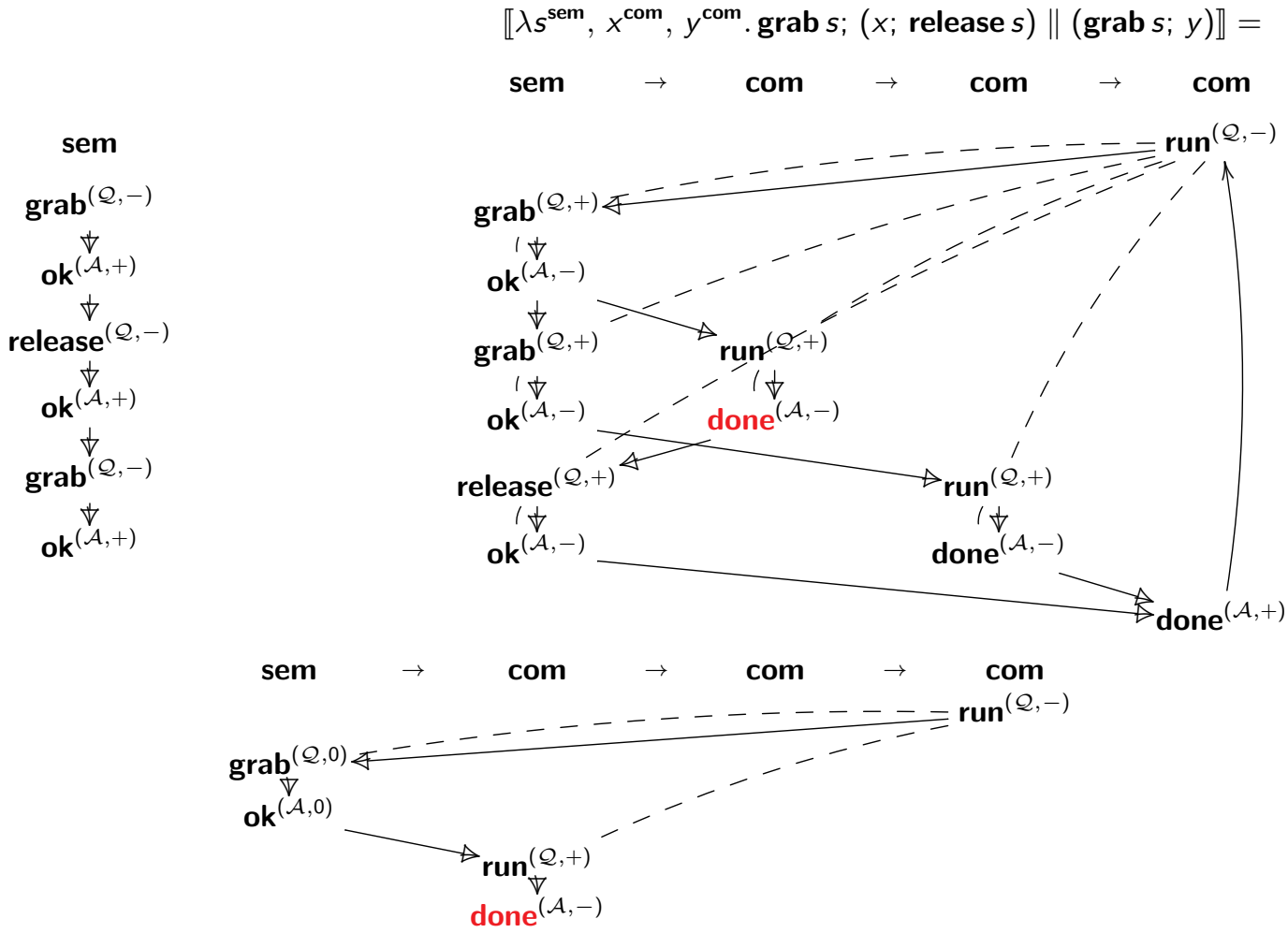
Example: sequential composition through semaphores



Example: sequential composition through semaphores

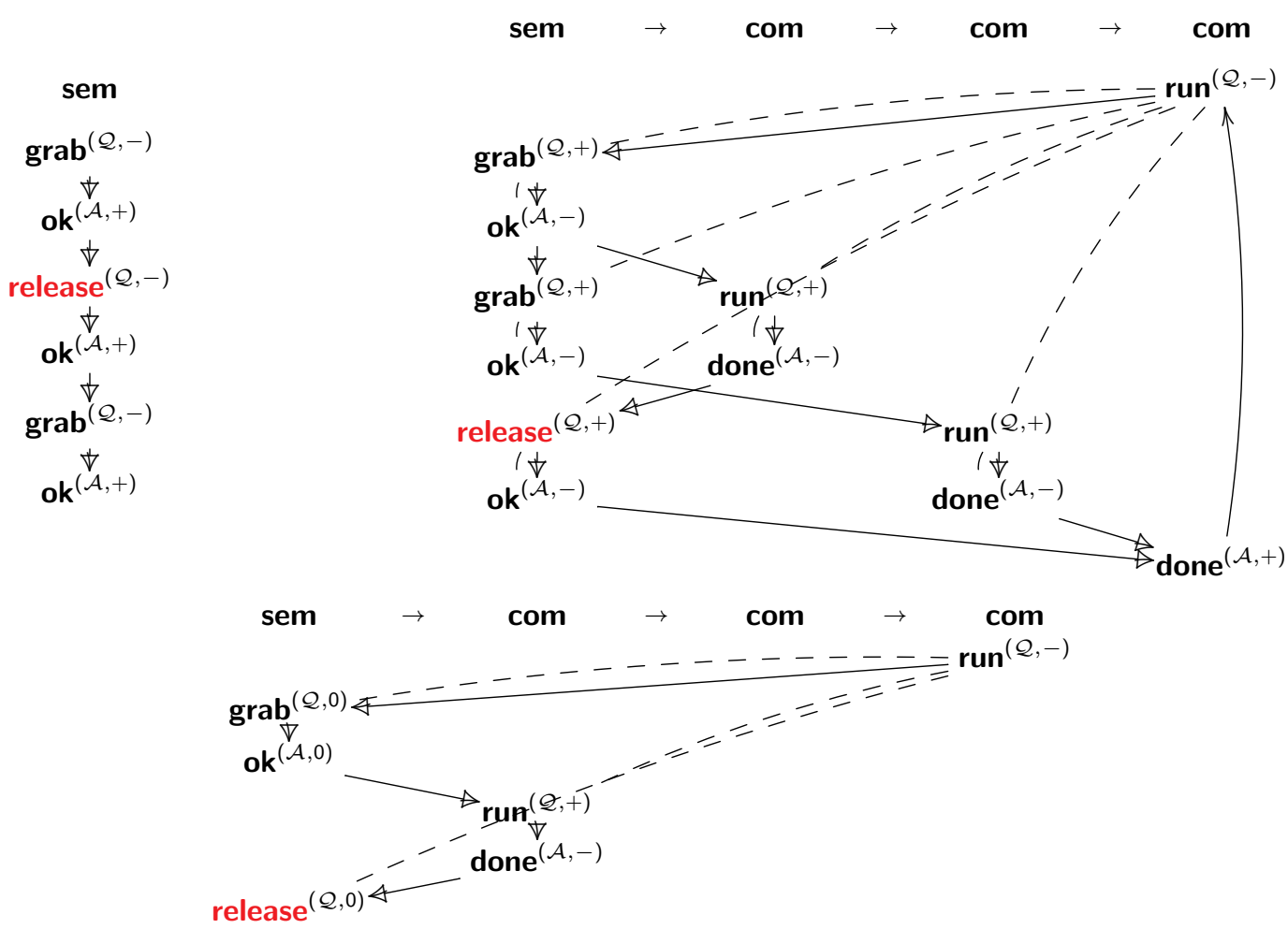


Example: sequential composition through semaphores



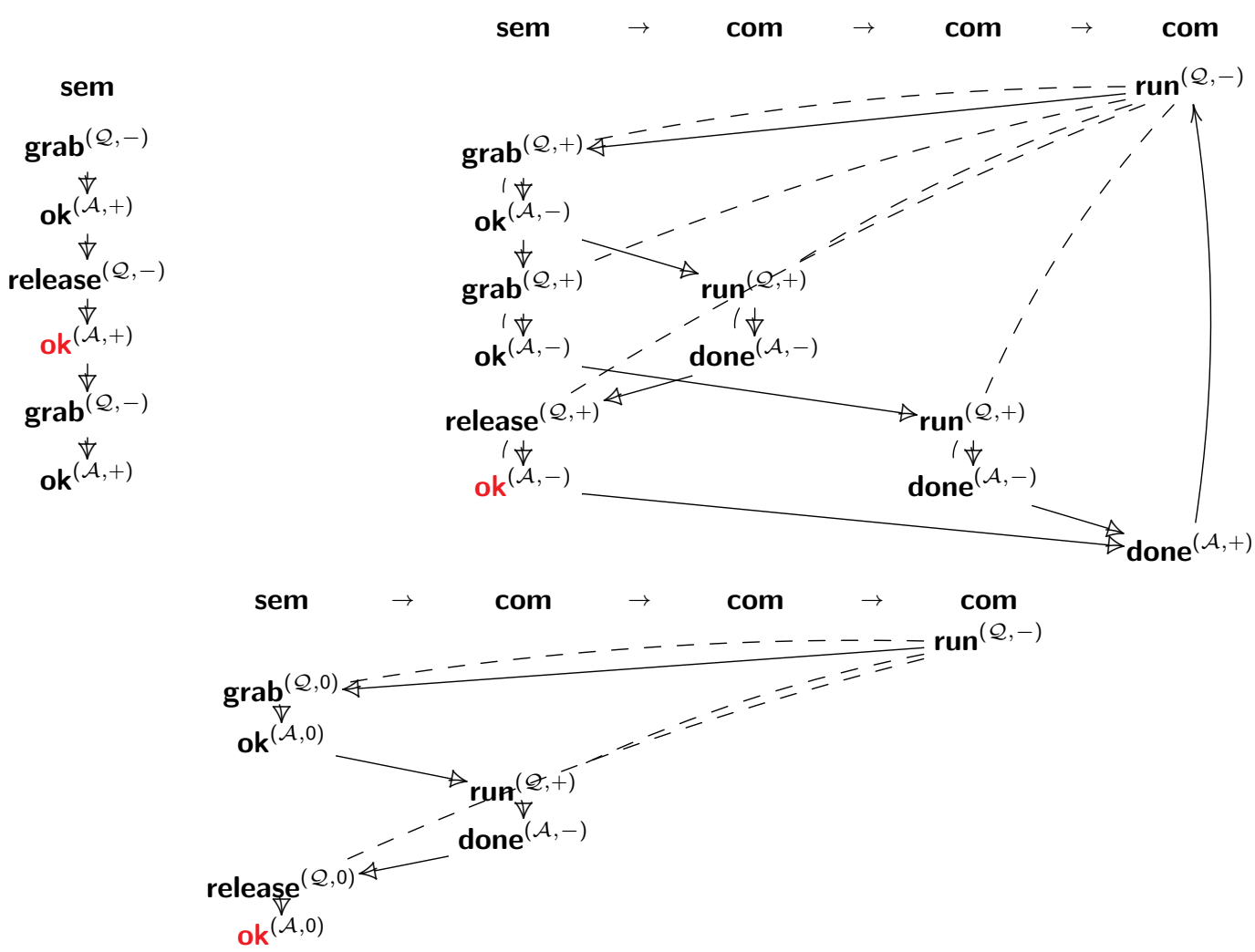
Example: sequential composition through semaphores

$[\![\lambda s^{\mathsf{sem}},\, x^{\mathsf{com}},\, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$



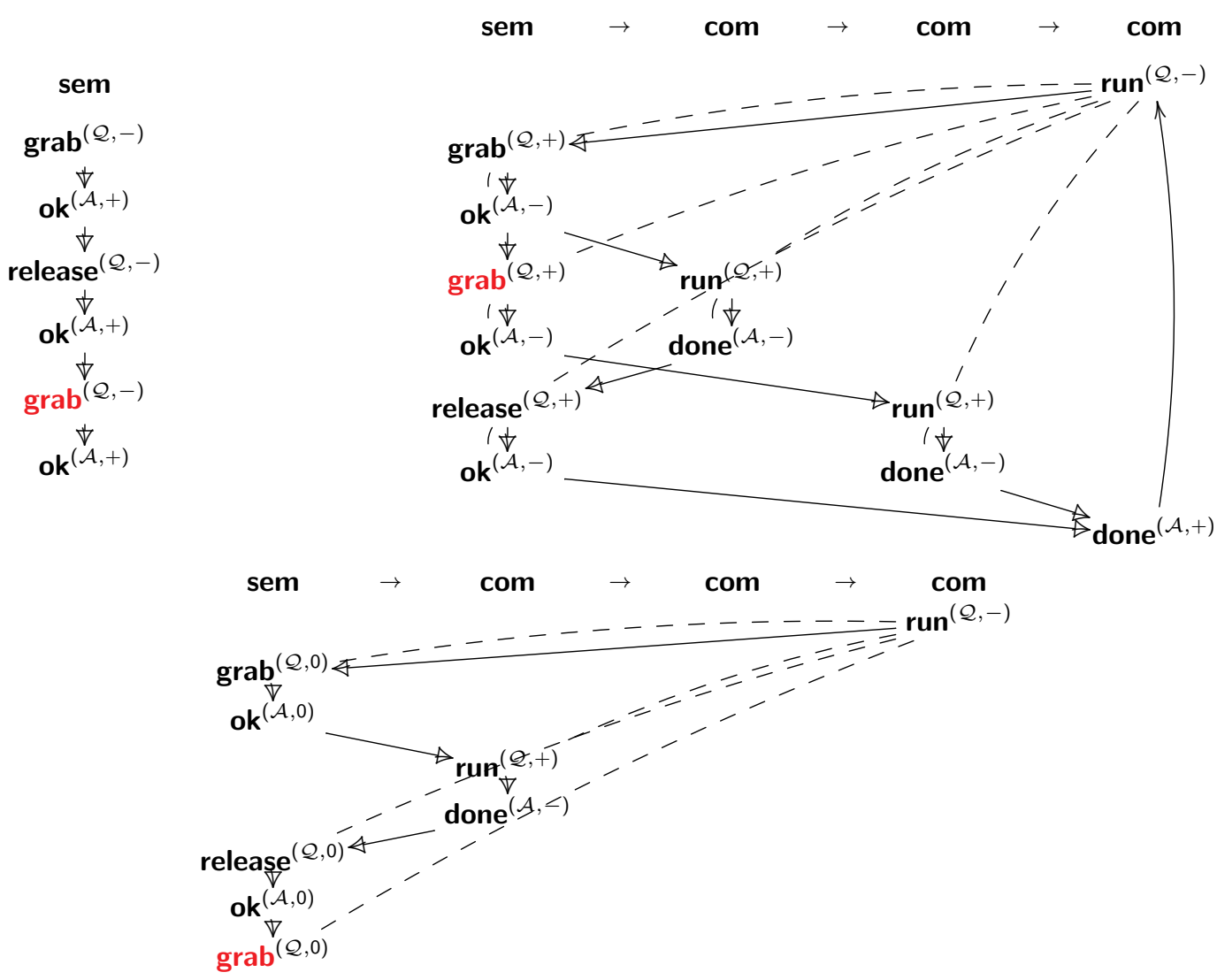
Example: sequential composition through semaphores

$[\![\lambda s^{\mathsf{sem}},\, x^{\mathsf{com}},\, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$



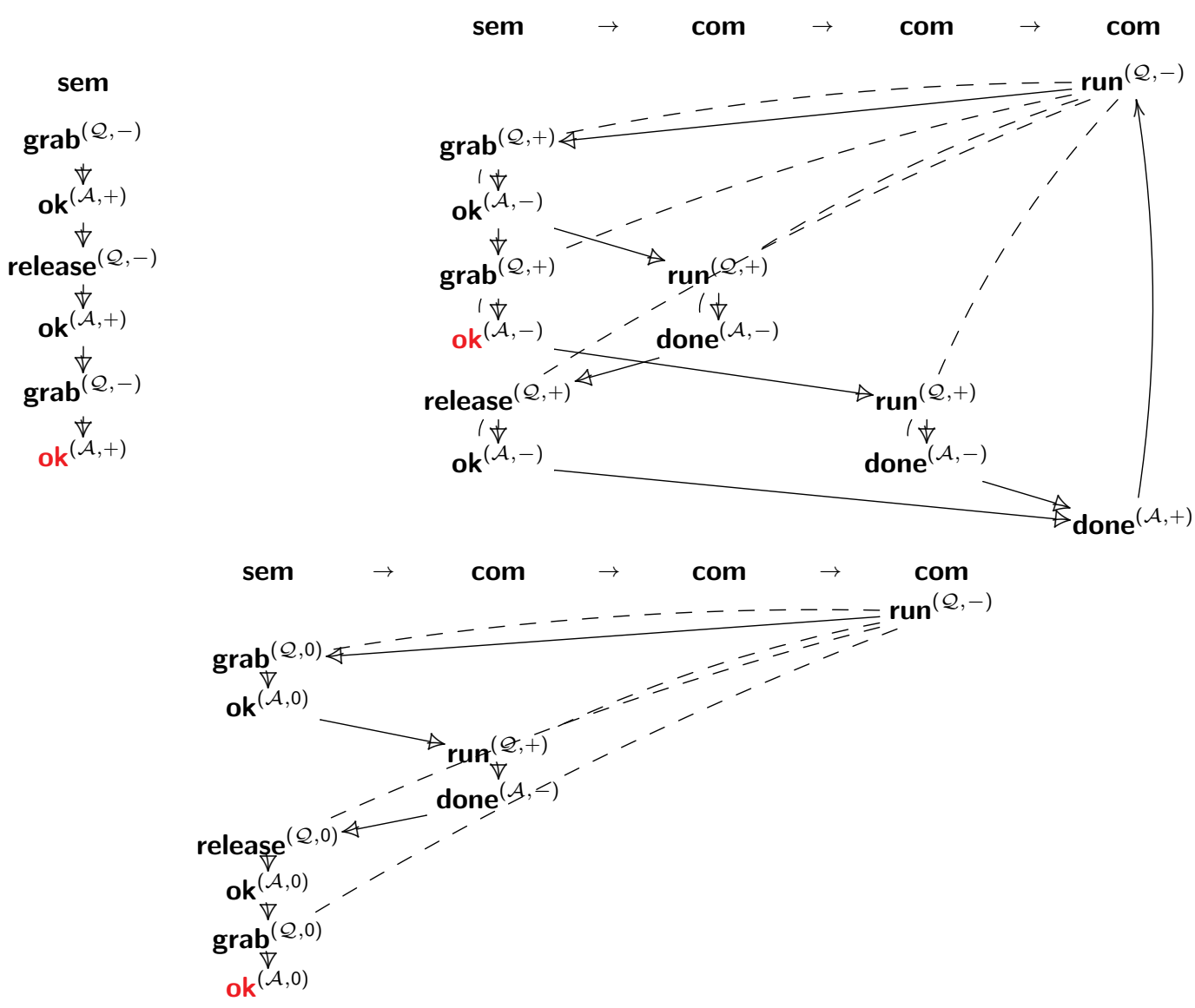
Example: sequential composition through semaphores

$[\![\lambda s^{\mathsf{sem}},\, x^{\mathsf{com}},\, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$



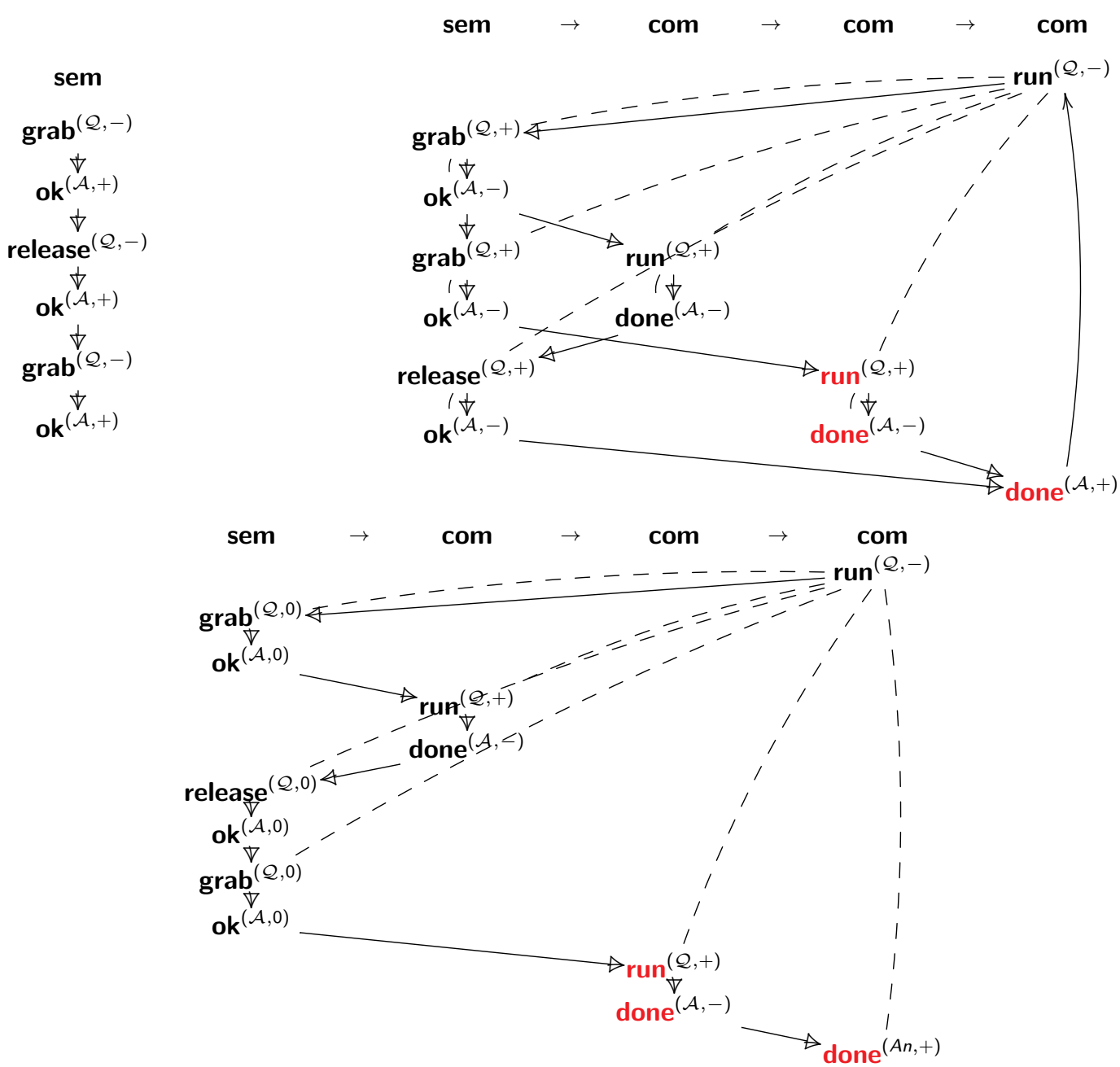
Example: sequential composition through semaphores

$[\![\lambda s^{\mathsf{sem}},\, x^{\mathsf{com}},\, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$

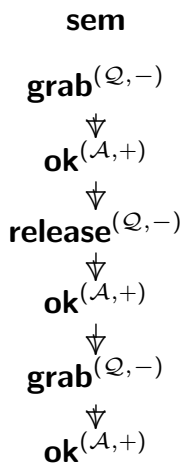


Example: sequential composition through semaphores

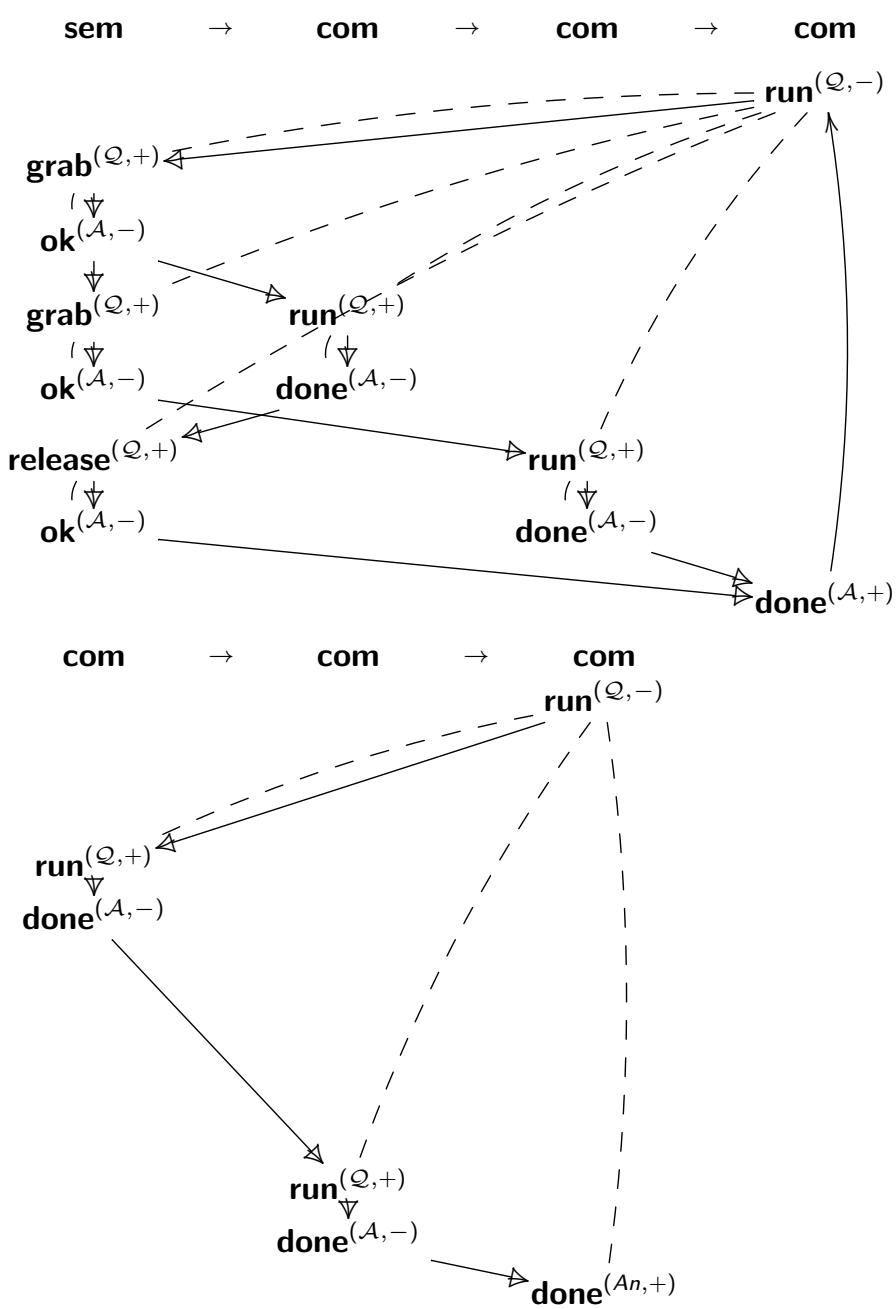
$[\![\lambda s^{\mathsf{sem}},\, x^{\mathsf{com}},\, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$



Example: sequential composition through semaphores

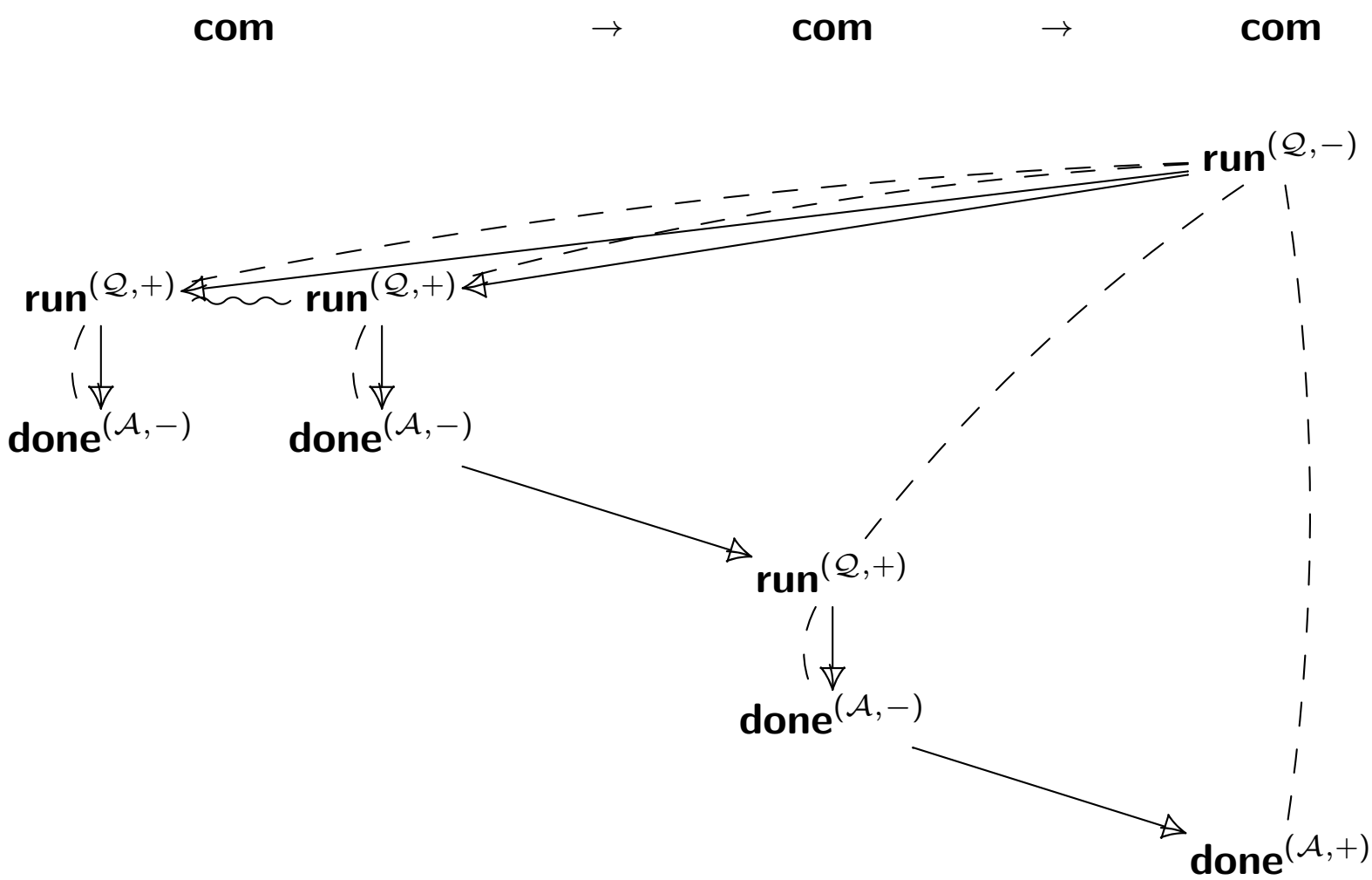


$[\![\lambda s^{\mathsf{sem}}, x^{\mathsf{com}}, y^{\mathsf{com}}.\, \mathbf{grab}\, s;\, (x;\, \mathbf{release}\, s) \parallel (\mathbf{grab}\, s;\, y)]\!] =$



Another example : sequential composition through state (may)

$$[\![\mathbf{newref}\ r := \mathbf{ff}\ \mathbf{in}\ \lambda x^{\mathbf{com}}, y^{\mathbf{com}}.\ (\mathbf{if}\ (!r)\ \bot\ (x; r := \mathbf{tt}))\ \|\ (\mathbf{if}\ (!r)\ y\ \bot)]\!] =$$



State and concurrency, summary

Proposition

There is a cartesian closed category **CHO** *of* **concurrent strategies on arenas**.

Proposition

If one restricts to **well-bracketed** *strategies, there is a functor:*

$$\mathrm{Plays} : \mathbf{CHO}_{\mathrm{wb}} \to \mathbf{GM}$$

with a finite definability property. As a consequence, the observational collapse of $\mathbf{CHO}_{\mathrm{wb}}$ *is fully abstract for IPA.*

4. Parallel full abstraction for PCF

Towards parallelization

Evaluating PCF in parallel: Consider a term:

$$\texttt{if } M_1 \texttt{ then } M_2 \texttt{ else } M_3 : \mathbb{B}$$

The standard games interpretation of `if` forces the evaluation order:

$$\begin{aligned} \texttt{if } M_1 \texttt{ then } M_2 \texttt{ else } M_3 &\rightsquigarrow \texttt{if tt then } M_2 \texttt{ else } M_3 \\ &\rightsquigarrow M_2 \\ &\rightsquigarrow \texttt{ff} \end{aligned}$$

But with a parallel execution, we could do instead:

$$\begin{aligned} \texttt{if } M_1 \texttt{ then } M_2 \texttt{ else } M_3 &\ (\rightsquigarrow \| \rightsquigarrow \| \rightsquigarrow) \ \texttt{if tt then ff else tt} \\ &\rightsquigarrow \texttt{ff} \end{aligned}$$

Question: Build a notion of strategy representing the independence of these sub-computations?

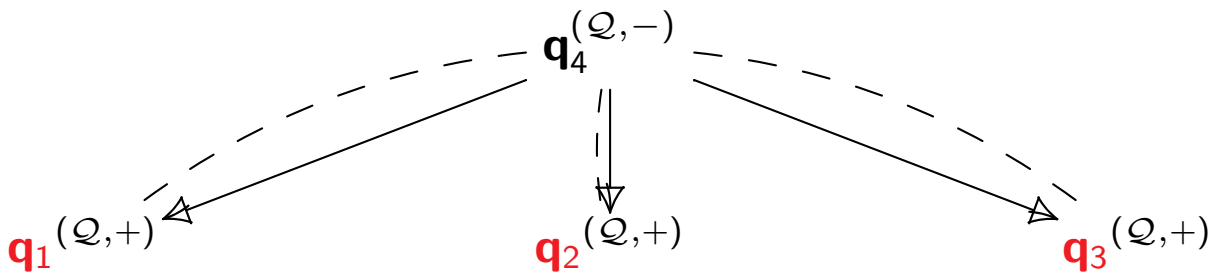
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.

$$\mathbf{q}_4{}^{(\mathcal{Q},-)}$$

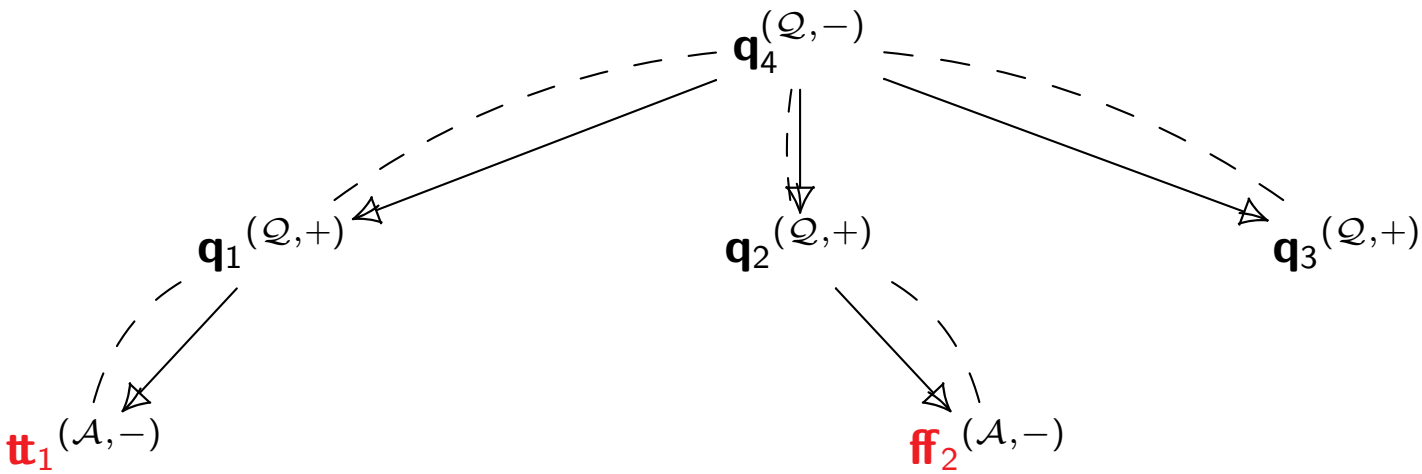
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



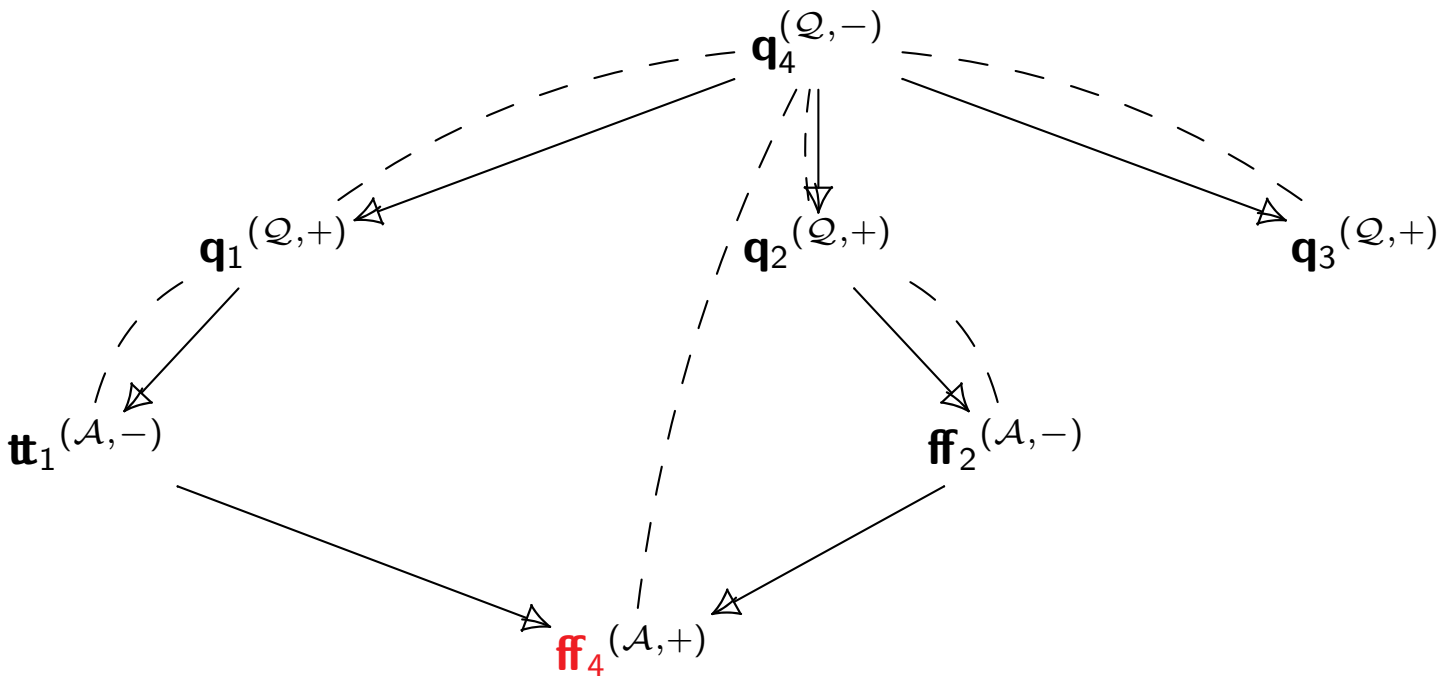
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



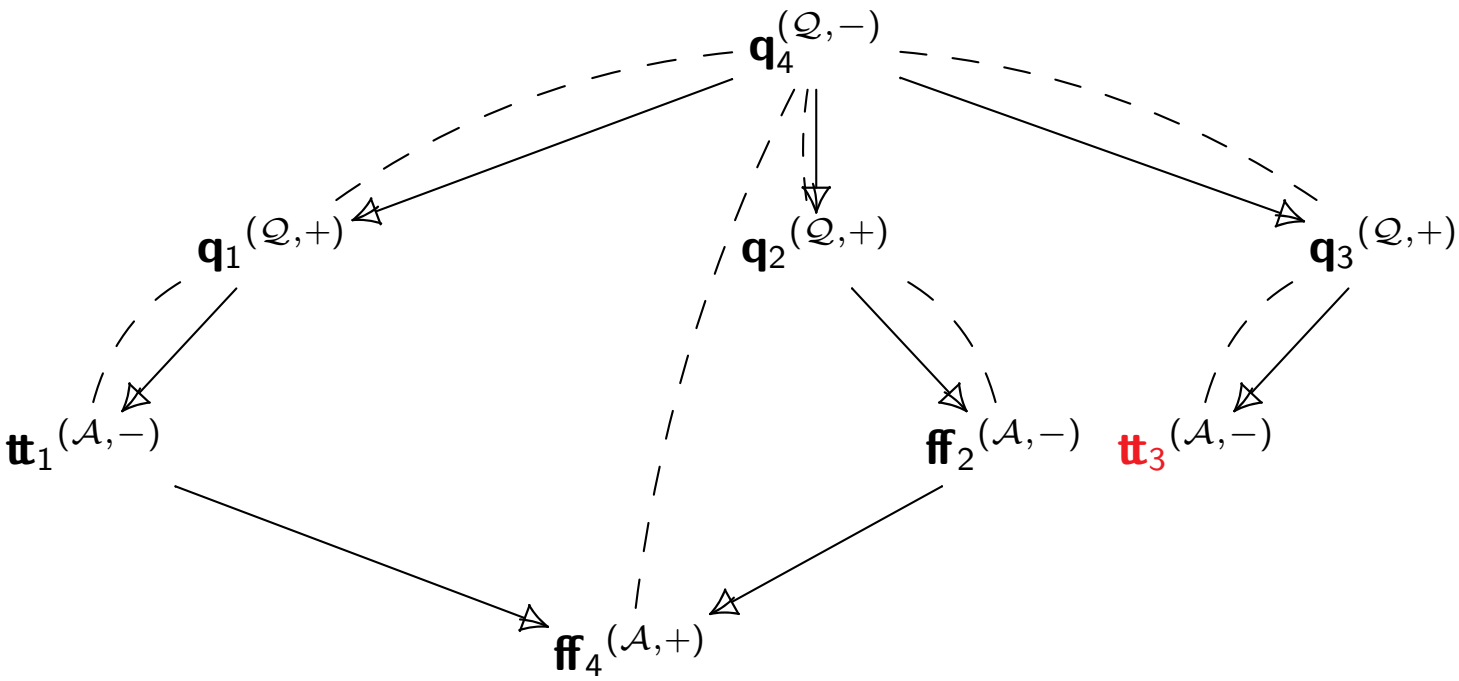
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



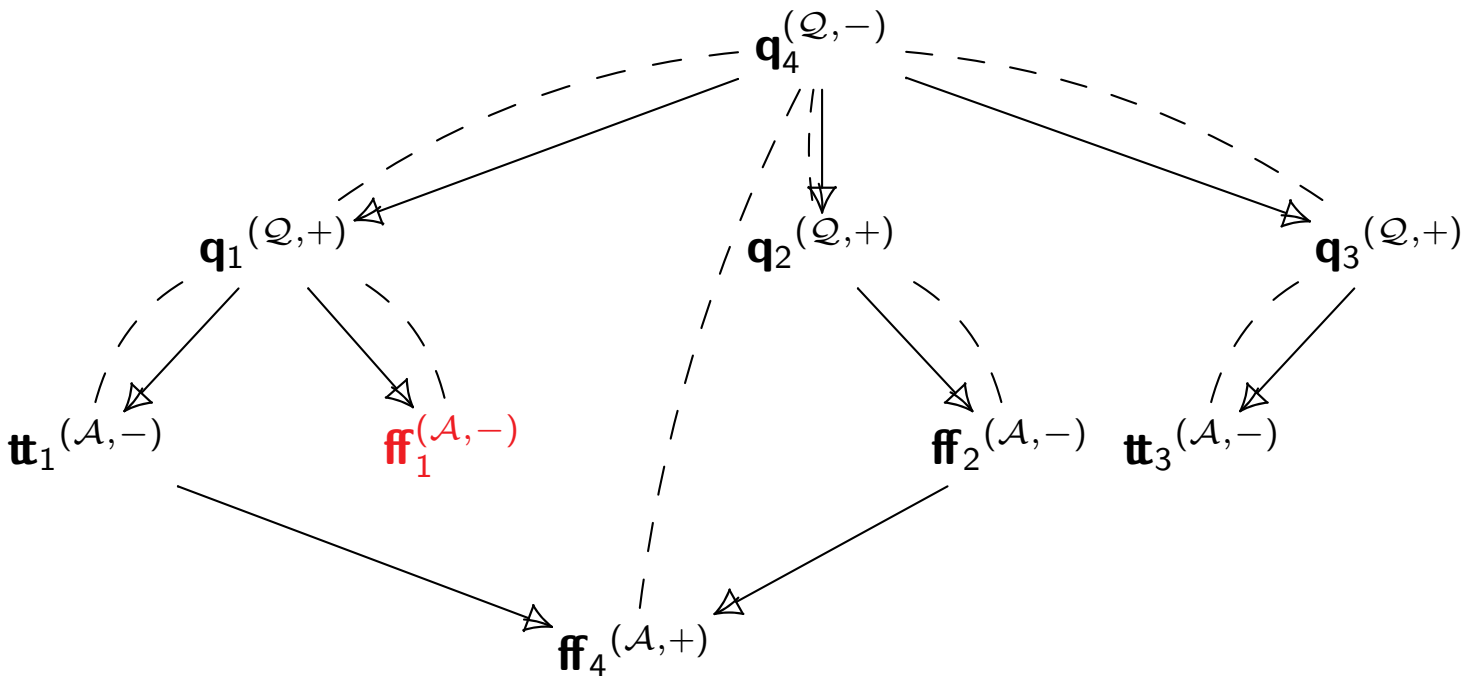
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



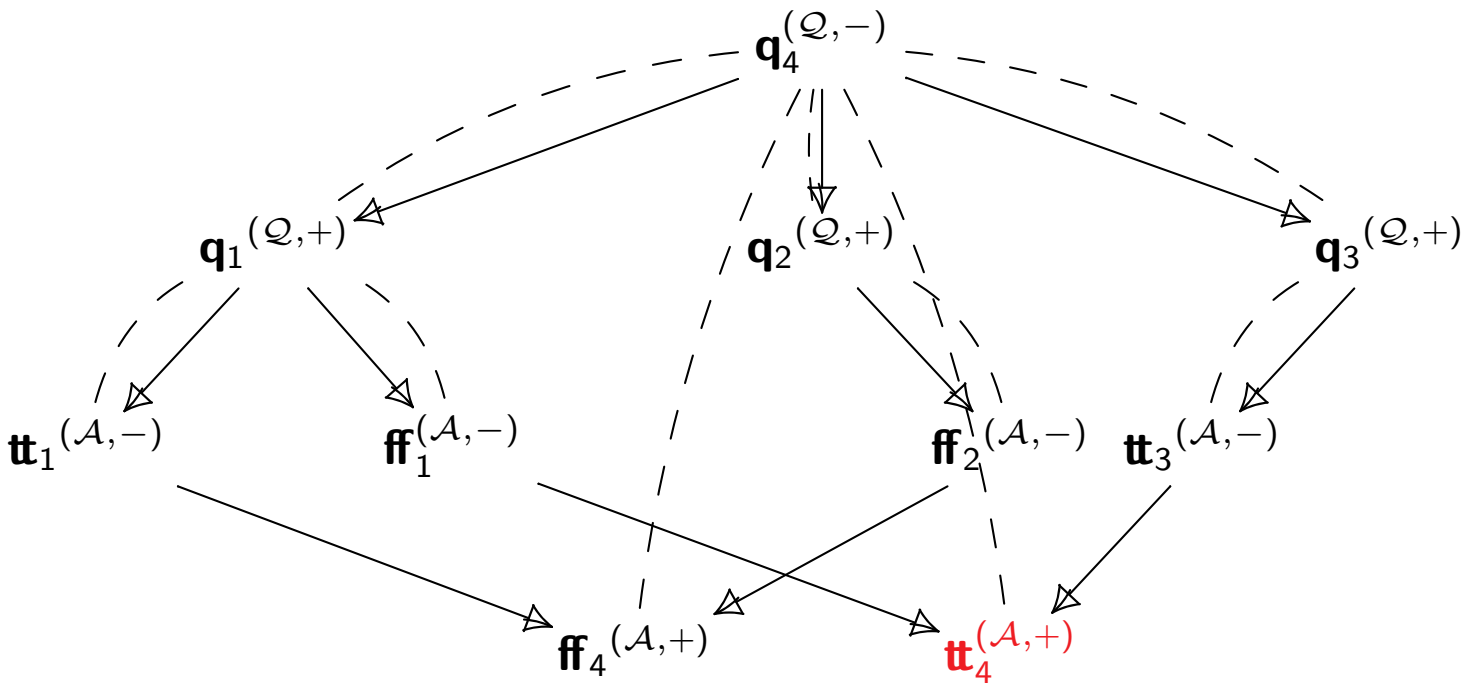
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



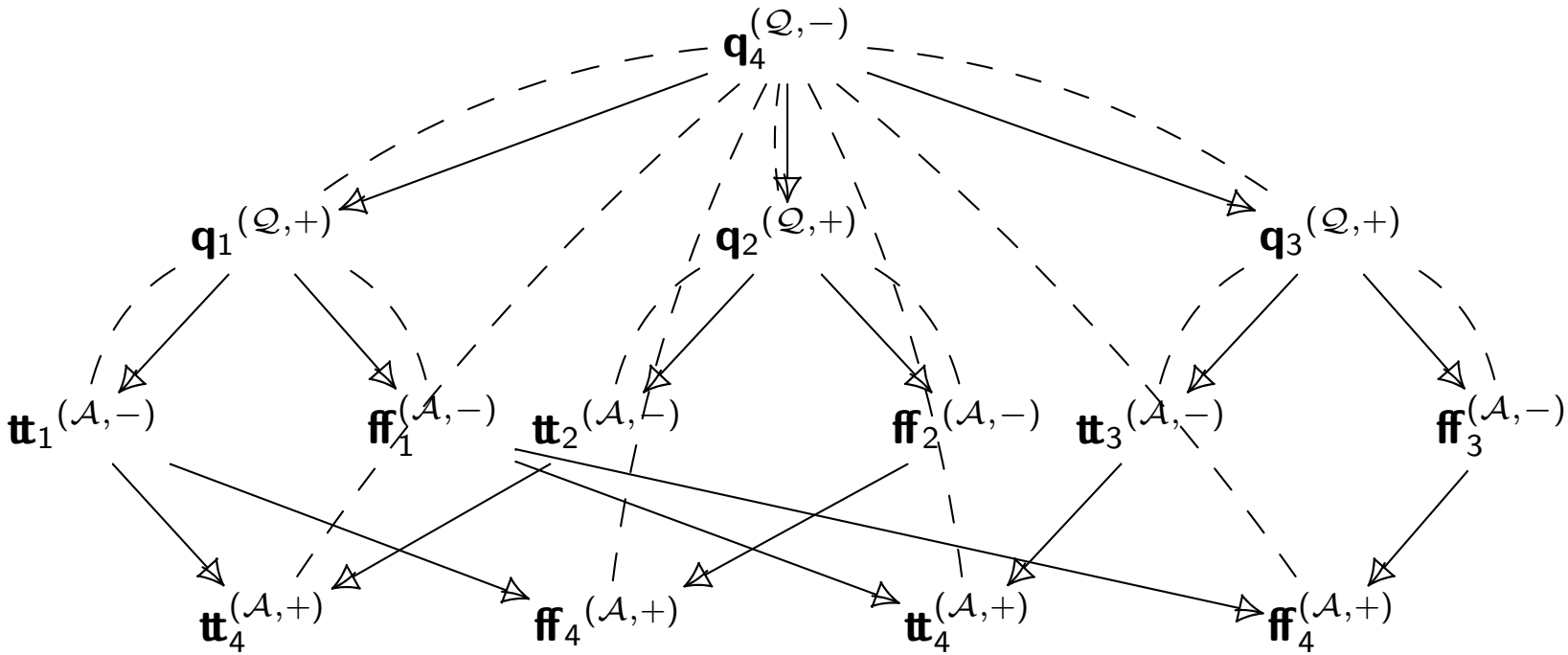
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



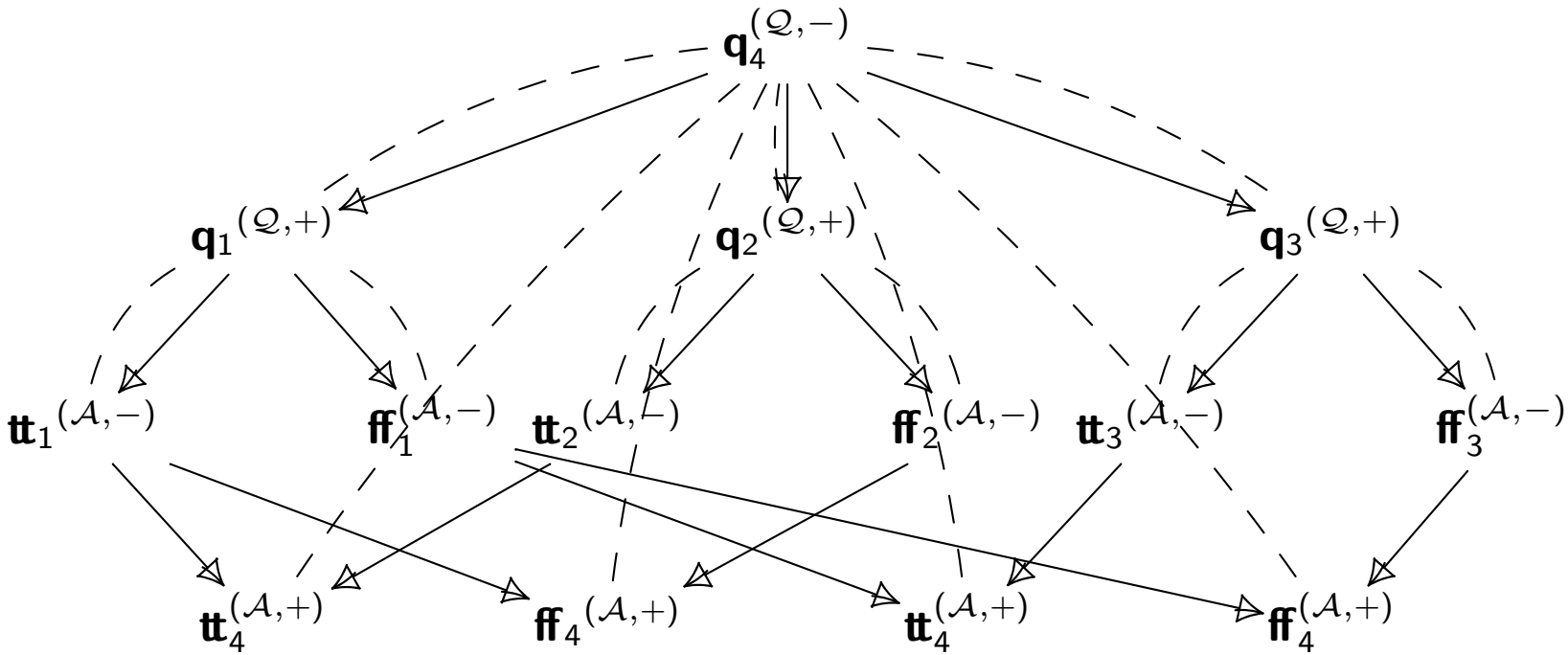
A parallel strategy for **if**

A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



A parallel strategy for **if**

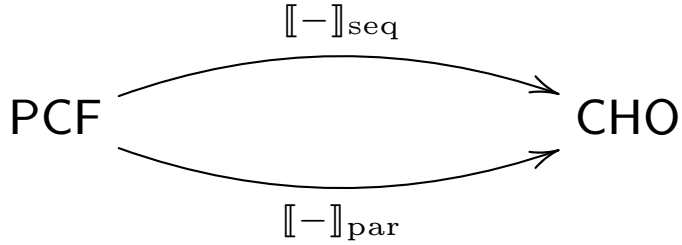
A concurrent strategy playing on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3 \to \mathbb{B}_4$.



It is parallel, but computes **the usual sequential function for if!**

A parallel interpretation of PCF

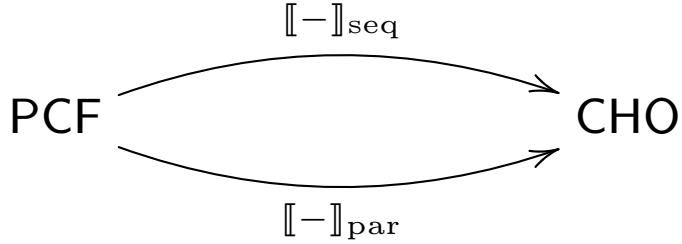
The interpretation of other combinators is unchanged. We have:



Sound and adequate: for all $\vdash M : \mathbb{B}$, $M \downarrow b \Leftrightarrow [\![M]\!] = [\![b]\!]$.

A parallel interpretation of PCF

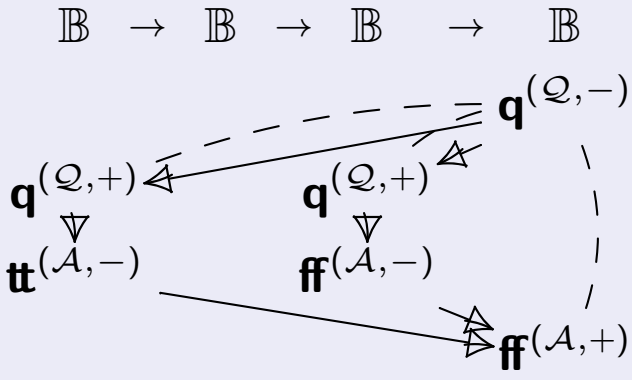
The interpretation of other combinators is unchanged. We have:



Sound and adequate: for all $\vdash M : \mathbb{B}$, $M \downarrow b \Leftrightarrow [\![M]\!] = [\![b]\!]$.

Syntax-free operational semantics

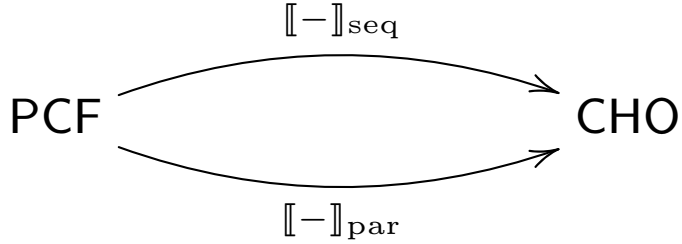
Parallel operational semantics *for PCF*



Traversals are **DAGs***!*

A parallel interpretation of PCF

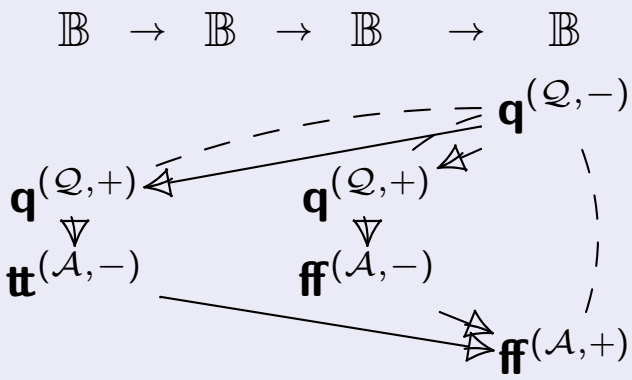
The interpretation of other combinators is unchanged. We have:



Sound and adequate: for all $\vdash M : \mathbb{B}$, $M \downarrow b \Leftrightarrow [\![M]\!] = [\![b]\!]$.

Syntax-free operational semantics

Parallel operational semantics *for PCF*



Traversals are **DAGs***!*

Dynamic description of effects

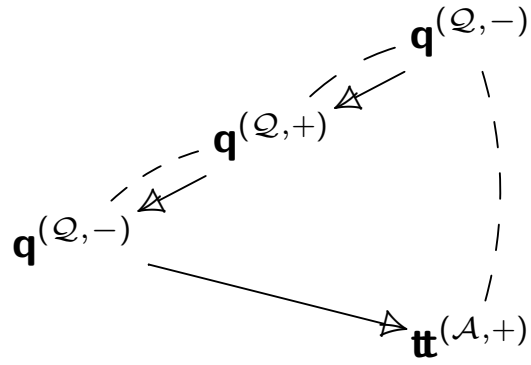
Cannot reuse usual notions of innocence, well-bracketing, . . .

- *What is a* **pure/effect-free** *parallel strategy?*
- *What is* **concurrent innocence***?*
- **Criterion***: should never distinguish* $[\![M]\!]_{\mathrm{seq}}$ *and* $[\![M]\!]_{\mathrm{par}}$*.*

Well-bracketing and innocence

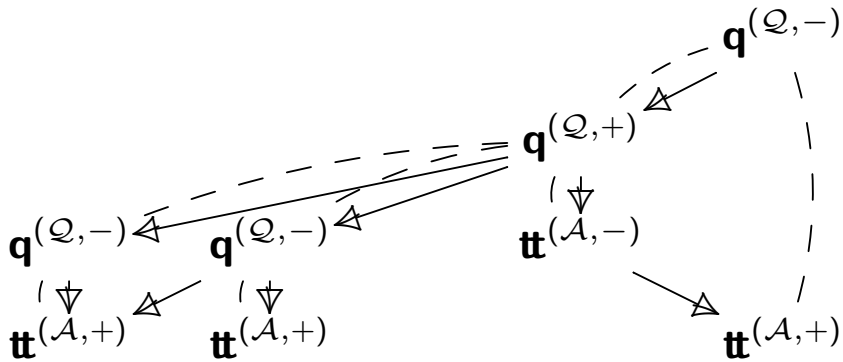
The following strategies distinguish $[\![\mathbf{if}]\!]_{\mathrm{seq}}$ and $[\![\mathbf{if}]\!]_{\mathrm{par}}$.

$$(\mathbb{B} \to \mathbb{B} \to \mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



$\hookrightarrow$ failure of **well-bracketing**.

$$(\mathbb{B} \to \mathbb{B} \to \mathbb{B} \to \mathbb{B}) \to \mathbb{B}$$



$\hookrightarrow$ failure of **innocence** (actually visibility).

The parallel intensionally fully abstract model for PCF

We have successfully captured a notion of **parallel effect-free computation**.

Proposition

CHO *has a subcategory* **PCFPAR** *of deterministic, visible, well-bracketed innocent strategies.*

Theorem

The interpretation of PCF *in* **PCFPAR** *is intensionally fully abstract.*

(just like the standard HO games model for PCF)

5. Conclusions

Conclusions and perspectives

Contributions.

- A **causal**, **concurrent** and **non-deterministic** version of Hyland-Ong games.
- An intensionally fully abstract **interpretation of IPA** in this setting,
- An intensionally fully abstract **parallel interpretation of PCF** in its subcategory of deterministic, well-bracketed, visible and innocent strategies.

Other developments.

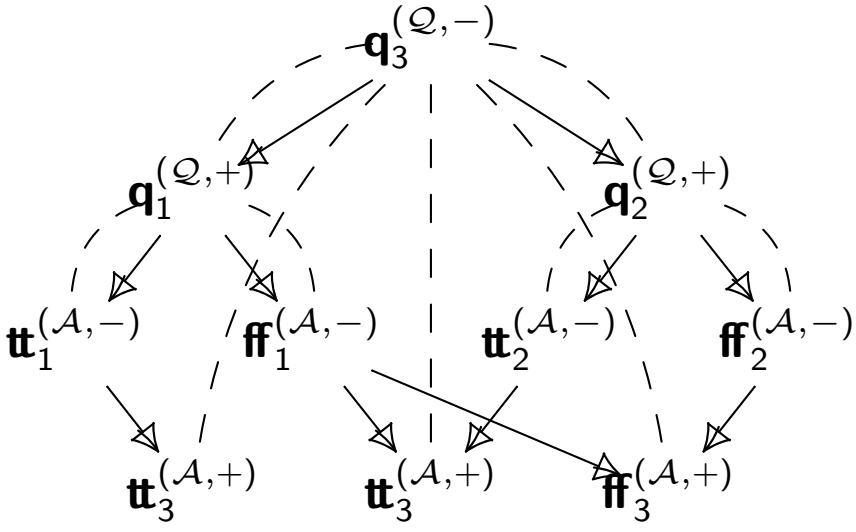
- **Essential events.** Keep in composition events carrying must-information.
- **Disjunctive causes.** Design a notion of determinism accommodating disjunctive strategies such as parallel or.
- **Weak memory models (Simon Castellan, JFLA 2016).** Design variants of the interpretation of state dealing with weak memory architectures.

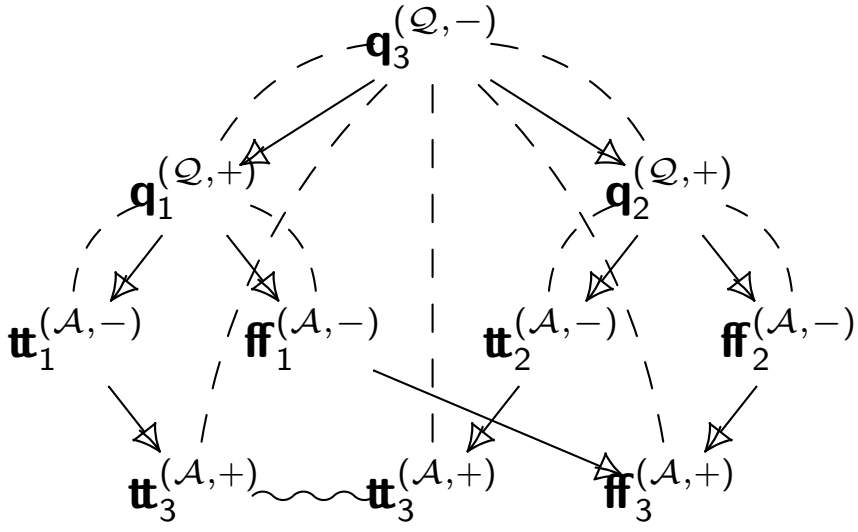
Perspectives.

- **Control of interference.** A deterministic interpretation of SCI.
- **Probabilities.** Mixing with probabilistic event structures.
- **Algorithmic concurrent games.**
- ...

Parallel **lor** *vs* parallel **or**

Compare the two following strategies on $\mathbb{B}_1 \to \mathbb{B}_2 \to \mathbb{B}_3$:





$$\begin{array}{lcccccc} \mathrm{lor}_{\|} & : & \mathbb{B}_1 & \Rightarrow & \mathbb{B}_2 & \Rightarrow & \mathbb{B}_3 \\ & & \bot & , & b & \mapsto & \bot \\ & & \mathbf{tt} & , & b & \mapsto & \mathbf{tt} \\ & & \mathbf{ff} & , & b & \mapsto & b \end{array} \qquad \begin{array}{lcccccc} \mathrm{por} & : & \mathbb{B}_1 & \Rightarrow & \mathbb{B}_2 & \Rightarrow & \mathbb{B}_3 \\ & & \bot & , & \mathbf{tt} & \mapsto & \mathbf{tt} \\ & & \mathbf{tt} & , & \bot & \mapsto & \mathbf{tt} \\ & & \mathbf{ff} & , & \mathbf{ff} & \mapsto & \mathbf{ff} \end{array}$$

So **parallel strategies** can compute **sequential functions**!