

chocola seminar

13 March 2025

# KLEENE-CARTESIAN BICATEGORIES FOR PROGRAM LOGICS

Elena Di Lavoro  
University of Oxford

j.w.w. Filippo Bonchi and Alessandro Di Giorgio  
University of Pisa      Tallinn University of Technology

## OVERVIEW

- Introduce Kleene bicategories.
- Mix with cartesian bicategories of relations.
- Express Jloare triples and derive rules of Jloare logic.

# MORPHISMS AS PROGRAMS

$$f: A \rightarrow B$$

$A - \boxed{f} - B$

$$f': A' \rightarrow B'$$

$A' - \boxed{f'} - B'$

$$g: B \rightarrow C$$

$B - \boxed{g} - C$

- Sequential composition.  $f; g: A \rightarrow C$

$$A - \boxed{f} - \boxed{g} - C$$

- Identity.  $\text{id}_A: A \rightarrow A$

$$A - A$$

- Parallel composition.  $f \otimes f': A \otimes A' \rightarrow B \otimes B'$

$$A - \boxed{f} - B = A - \boxed{f} - B = A - \boxed{f} - B$$

$A' - \boxed{f'} - B' \qquad A' - \boxed{f'} - B' \qquad A' - \boxed{f'} - B'$

- Symmetries.  $\sigma_{A,B}: A \otimes B \rightarrow B \otimes A$

$$\begin{array}{c} A \\ B \end{array} \times \begin{array}{c} B \\ A \end{array}$$

$$A - \boxed{f} - B' = A - \boxed{f'} - B'$$

$A' - \boxed{f} - B \qquad A' - \boxed{f'} - B$

# MONOIDS IN STRING DIAGRAMS

$(\mathcal{C}, \otimes, I)$  symmetric monoidal category

A commutative monoid in  $\mathcal{C}$  is

$A \in \text{obj}$   
(carrier)

$\circlearrowright : A \otimes A \rightarrow A$   
(multiplication)

$\circ : I \rightarrow A$   
(unit)

such that

$$\circlearrowright \circlearrowleft = \overline{\circlearrowright} \circlearrowright$$

(associativity)

$$\circlearrowleft = \underline{\circlearrowleft} = \overline{\circlearrowleft}$$

(unitality)

$$\circlearrowright = \circlearrowleft \circlearrowright$$

(commutativity)

ex monoids in  $(\text{Set}, \times, \{*\})$  are monoids

# COMONOIDS IN STRING DIAGRAMS

$(\mathcal{C}, \otimes, I)$  symmetric monoidal category

A commutative comonoid in  $\mathcal{C}$  is

$A \in \text{obj}_{\mathcal{C}}$   
(carrier)

 :  $A \rightarrow A \otimes A$   
(comultiplication)

 :  $A \rightarrow I$   
(counit)

such that

$$\overbrace{\text{---}}^{\text{---}} = \overbrace{\text{---}}^{\text{---}}$$

(associativity)

$$\overbrace{\text{---}}^{\text{---}} = \text{---} = \overbrace{\text{---}}^{\text{---}}$$

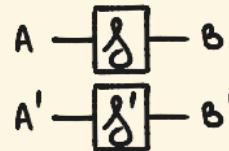
(unitality)

$$\overbrace{\text{---}}^{\text{---}} = \overbrace{\text{---}}^{\text{---}}$$

(commutativity)

ex  $\begin{cases} \text{---}(x) := (x, x) \\ \text{---}(x) := * \end{cases}$  is a commutative comonoid in  $(\text{Set}, \times, \{*\})$

## TWO KINDS OF PARALLEL COMPOSITION



execute  $f$  and  $f'$   $\rightsquigarrow$  data flow  
execute  $f$  or  $f'$   $\rightsquigarrow$  control flow

ex  $(\text{Set}, \times, \{\ast\}) \rightsquigarrow \text{'and'}$   
 $(\text{Set}, +, \emptyset) \rightsquigarrow \text{'or'}$

We need both kinds of parallel compositions

$\Rightarrow$  rig categories  $(\mathcal{C}, \otimes, I, \oplus, 0)$ :

$(\mathcal{C}, \otimes, I)$  and  $(\mathcal{C}, \oplus, 0)$  symmetric monoidal categories

$$\delta_{A,B,C}^L : A \otimes (B \oplus C) \rightarrow (A \otimes B) \oplus (A \otimes C)$$

$$\gamma_A^L : 0 \otimes A \rightarrow 0$$

$$\delta_{A,B,C}^R : (A \oplus B) \otimes C \rightarrow (A \otimes C) \oplus (B \otimes C)$$

$$\gamma_A^R : A \otimes 0 \rightarrow 0$$

# OUTLINE

- [• The category of relations ]
- Kleene bicategories
- Tape diagrams for Kleene-cartesian bicategories
- Program logics

# RELATIONS FORM A RIG CATEGORY

$f: A \rightarrow B$  is  $f \subseteq A \times B$

- Monoidal product  $\otimes$ .

$$f \otimes g := f \times g \quad I := \{*\}$$

- Monoidal product  $\oplus$ .

$$f \oplus g := f \cup g \quad 0 := \emptyset$$

- Distributivity.

$$(A \cup B) \times C \simeq (A \times C) \cup (B \times C) \quad A \times \emptyset \simeq \emptyset$$

⇒ For a monoidal monad  $T$  on a distributive category  $\mathcal{C}$ ,  
 $\text{Kleisli}(T)$  is rig.

# CALCULUS OF RELATIONS

$$\begin{aligned} E ::= & R \mid id \mid E; E \\ & \mid E^\circ \mid T \mid E \circ E \\ & \mid E^* \mid \perp \mid E \cup E \end{aligned}$$

allegorical fragment  
~ cartesian bicategories

Kleene fragment  
~ Kleene bicategories

# THE STRUCTURE OF $(\text{Rel}, \otimes, I)$

- Commutative (co)monoid on every object. For a set  $A$ ,

$$\textcircled{\text{C}} : A \rightarrow A \otimes A$$

$$\textcircled{\text{C}} := \{(x, (x, x)) \mid x \in A\}$$

$$\textcircled{\text{D}} : A \otimes A \rightarrow A$$

$$\textcircled{\text{D}} := \{((x, x), x) \mid x \in A\}$$

$$\textcircled{\text{I}} : A \rightarrow I$$

$$\textcircled{\text{I}} := \{(x, *) \mid x \in A\}$$

$$\textcircled{\text{M}} : I \rightarrow A$$

$$\textcircled{\text{M}} := \{(*, x) \mid x \in A\}$$

- Opposite. For  $f : A \rightarrow B$ ,

$$f^{\text{op}} : B \rightarrow A$$

- Order. For  $f, g : A \rightarrow B$ ,

$$f \leq g \text{ iff } f \sqsubseteq g \text{ iff } f \circ g = f \text{ iff }$$

# CARTESIAN BICATEGORIES OF RELATIONS

a monoidal posetal bicategory with

( $\sqsubset$ ,  $\multimap$ ) coherent commutative comonoid

( $\sqsupset$ ,  $\multimap$ ) coherent commutative monoid

such that

$$\begin{array}{c} \text{---} \square \text{---} \sqsubset \leq \text{---} \bullet \text{---} \begin{array}{c} \text{---} \square \text{---} \\ \text{---} \square \text{---} \end{array} \text{---} \bullet \text{---} \leq \text{---} \bullet \text{---} \\ \text{(lax comonoid homomorphisms)} \end{array}$$

$$\begin{array}{c} \text{---} \sqsupset \text{---} \leq \text{---} \quad \bullet \bullet \leq \square \\ \text{---} \leq \text{---} \bullet \text{---} \quad \text{---} \leq \text{---} \bullet \bullet \end{array}$$

(comonoid  $\dashv$  monoid)

$$\begin{array}{c} \text{---} \sqsupset \text{---} = \text{---} \bullet \text{---} \end{array}$$

(Grobenius)

[Carboni, Walters (1987)]

# PREDICATES

Predicates are  $p : A \rightarrow 1$ .

$$p \wedge q = A - \text{C}_A(p, q)$$

$$\text{assert } p = A - \text{C}_A(p) = A - p - A$$

$$\Rightarrow \text{assert}(p \wedge q) = A - \text{C}_A(p, q) = A - \text{C}_A(p) - \text{C}_A(q) = A - p - q - A$$

$$= A - \text{C}_A(p) - \text{C}_A(q) = A - p - q - A$$

$$= \text{assert } p ; \text{ assert } q$$

# THE STRUCTURE OF $(\text{Rel}, \oplus, 0)$

- Biproducts.

$$\begin{array}{c} \diagup \\ \diagdown \end{array} : A \oplus A \rightarrow A$$

$$\begin{array}{c} \diagup \\ \diagdown \end{array} := \{((x, 0), x) | x \in A\} \cup \{((x, 1), x) | x \in A\}$$

$$\begin{array}{c} \diagup \\ \diagdown \end{array} : A \rightarrow A \oplus A$$

$$\begin{array}{c} \diagup \\ \diagdown \end{array} := \{(x, (x, 0)) | x \in A\} \cup \{(x, (x, 1)) | x \in A\}$$

$$\circ : 0 \rightarrow A$$

$$\circ := \emptyset$$

$$\multimap : A \rightarrow 0$$

$$\multimap := \emptyset$$

$(\begin{array}{c} \diagup \\ \diagdown \end{array}, \circ), (\begin{array}{c} \diagup \\ \diagdown \end{array}, \multimap)$  natural coherent commutative (co)monoid

$$\begin{array}{c} \diagup \\ \diagdown \end{array} = \begin{array}{c} \diagup \\ \diagdown \end{array} \quad \text{comonoid homomorphisms}$$

$$\begin{array}{c} \diagup \\ \diagdown \end{array} \circ = \circ$$

| ↗ comonoid homomorphisms

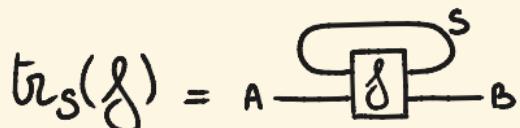
monoid homomorphisms ↗

$$\begin{array}{c} \diagup \\ \diagdown \end{array} \begin{array}{c} \diagup \\ \diagdown \end{array} = \begin{array}{c} \diagup \\ \diagdown \end{array} \quad \circ \begin{array}{c} \diagup \\ \diagdown \end{array} = \circ$$

# THE STRUCTURE OF $(\text{Rel}, \oplus, 0)$

- Trace. For  $f: S \oplus A \rightarrow S \oplus B$ ,

$$\text{tr}_S(f): A \rightarrow B$$



such that

$$A - \boxed{\begin{array}{c} f \\ g \end{array}} - B = A - \boxed{f} - B$$

The diagram shows two parallel horizontal lines labeled A and B. Between them is a box labeled f. Below the lines are two more boxes, one labeled u and one labeled v. A dashed box encloses the f and u boxes. Above the top line A is a curved arrow labeled s, and above the bottom line B is another curved arrow labeled s.

$$A - \boxed{f} - B \stackrel{SOT}{=} A - \boxed{g} - B$$

The diagram shows two parallel horizontal lines labeled A and B. Between them is a box labeled f. Below the lines are two more boxes, one labeled u and one labeled v. A dashed box encloses the f and v boxes. Above the top line A is a curved arrow labeled s, and above the bottom line B is another curved arrow labeled s.

$$A' - \boxed{\begin{array}{c} u \\ f \\ v \end{array}} - B' = A' - \boxed{u} - \boxed{v} - B'$$

The diagram shows two parallel horizontal lines labeled A' and B'. Between them are three boxes: u, f, and v. The f box is between u and v. Above the top line A' is a curved arrow labeled s, and above the bottom line B' is another curved arrow labeled s.

$$A - \boxed{f} - B = A - \boxed{f} - B$$

The diagram shows two parallel horizontal lines labeled A and B. Between them is a box labeled f. Above the top line A is a curved arrow labeled s, and above the bottom line B is another curved arrow labeled s.

$$A - \boxed{\begin{array}{c} f \\ u \end{array}} - B = A - \boxed{u} - \boxed{f} - B$$

The diagram shows two parallel horizontal lines labeled A and B. Between them are two boxes: f and u. The f box is to the left of the u box. Above the top line A is a curved arrow labeled s, and above the bottom line B is another curved arrow labeled s.

$$A - \overbrace{\quad}^A - A = A - A$$

The diagram shows a single horizontal line labeled A. Above it is a curved arrow labeled s.

# OUTLINE

- The category of relations
- Kleene bicategories
- Tape diagrams for Kleene-cartesian bicategories
- Program logics

# KLEENE BICATEGORIES

How do biproducts and trace interact?

A Kleene bicategory is a category with

- finite biproducts s.t.  $a \vee a = a$
- posetally uniform trace s.t.  $\text{id}_n^* \leq \text{id}_n$

# KLEENE BICATEGORIES

A Kleene bicategory is a category with

- finite biproducts s.t.  $a \vee a = a$

$(\bullet, \circ), (\dashv, \dashv)$  natural coherent commutative (co)monoid

s.t.  $\text{---} \circ \text{---} = \text{---}$   $\rightsquigarrow$  this gives a posetal enrichment.

$$f \leq g \text{ iff } \begin{array}{c} f \\ \square \\ g \end{array} = \begin{array}{c} g \end{array} \text{ iff } f \vee g = g$$

- posetally uniform trace s.t.  $\text{id}_A^* \leq \text{id}_A$

$$\begin{array}{c} f \\ \square \\ u \end{array} \leq \begin{array}{c} u \\ \square \\ g \end{array} \Rightarrow \begin{array}{c} f \\ \square \\ u \end{array} \leq \begin{array}{c} g \\ \square \\ u \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \end{array} \leq \text{---}$$

# TRACES AND FIXPOINTS

For  $f, f': A \rightarrow B$ ,  $g: B \rightarrow C$  and  $a: A \rightarrow A$ ,

$$f \vee f' := - \circ \begin{array}{c} f \\ \sqcap \\ f' \end{array} \circ -$$

$$0_{A,B} := - \circ -$$

$$a^* := \begin{array}{c} \text{---} \\ \text{---} \circ \text{---} \circ \text{---} \\ \text{---} \end{array}$$

$$f \cdot g := - \begin{array}{|c|c|} \hline f & g \\ \hline \end{array} -$$

$$1_A := -$$

$$\rightsquigarrow a^* \text{ is the least fixpoint of } \phi_a : f \mapsto - \circ \begin{array}{|c|c|} \hline a & f \\ \hline \end{array} \circ -$$

## PROPOSITION

Kleene bicategories are typed Kleene algebras.

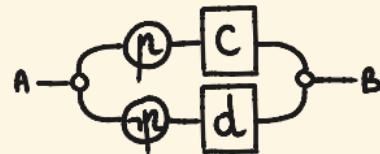
[Kozen (1994), (1998)]

# COPRODUCTS FOR IF-ELSE STATEMENTS, TRACES FOR WHILE LOOPS

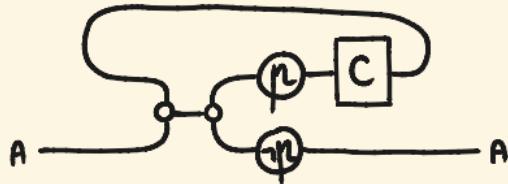
$$c, d : A \rightarrow B \quad a : A \rightarrow A$$

$$p : A \rightarrow 1 \quad \Rightarrow \text{ assert } p : A \rightarrow A$$

if  $p$  then  $c$  else  $d$   $\mapsto$



while  $p$  do  $c$   $\mapsto$



# KLEENE AND CARTESIAN BICATEGORIES

$(\text{Rel}, \sqcup, \emptyset)$

$(\rightarrowtail, \multimap), (\multimap, \rightarrowtail)$

natural

monoids  $\dashv$  comonoids

$$\begin{array}{c} a \\ \square \\ a \end{array} = \boxed{a}$$

$$f \leq g \text{ iff } \begin{array}{c} f \\ \square \\ g \end{array} = \boxed{g}$$

$$A - \boxed{a^*} - A := \begin{array}{c} A \\ \swarrow \\ \square \\ \searrow \\ A \end{array} \quad \begin{array}{c} a \\ \square \\ a \end{array} \quad A$$

Kleene fragment  
control flow

$(\text{Rel}, \times, \{\ast\})$

$(\multimap, \rightarrowtail), (\rightarrowtail, \multimap)$

lax natural

comonoids  $\dashv$  monoids

$$\begin{array}{c} a \\ \square \\ a \end{array} = \boxed{a}$$

$$f \leq g \text{ iff } \begin{array}{c} f \\ \square \\ g \end{array} = \boxed{g}$$

$$B - \boxed{f^{\text{op}}} - A := \begin{array}{c} B \\ \swarrow \\ \square \\ \searrow \\ A \end{array}$$

allegorical fragment  
data flow

# KLEENE-CARTESIAN BICATEGORIES

A Kleene-cartesian bicategory is a rig category  $(\mathcal{C}, \otimes, I, \oplus, 0)$  such that

- $(\mathcal{C}, \otimes, I)$  is a cartesian bicategory
- $(\mathcal{C}, \oplus, 0)$  is a Kleene bicategory
- the black and white (co)monoids interact as

$$\dashv_{A \otimes B} = (\dashv_A \oplus \dashv_B); (\dashv_{A \otimes A} \oplus \dashv_{A \otimes B} \oplus \dashv_{B \otimes A} \oplus \dashv_{B \otimes B}); (\delta_{A,A,B}^L \oplus \delta_{B,A,B}^L); \delta_{A,B,A \otimes B}^R$$

$$\triangleright_{A \otimes B} = \delta_{A,B,A \otimes B}^R; (\delta_{A,A,B}^L \oplus \delta_{B,A,B}^L); (\dashv_{A \otimes A} \oplus \dashv_{A \otimes B} \oplus \dashv_{B \otimes A} \oplus \dashv_{B \otimes B}); (\triangleright_A \oplus \triangleright_B)$$

$$\dashv_{A \oplus B} = (\dashv_A \oplus \dashv_B); \triangleright_I$$

$$\dashv_{A \oplus B} = -\dashv_I; (\dashv_A \oplus \dashv_B)$$

# OUTLINE

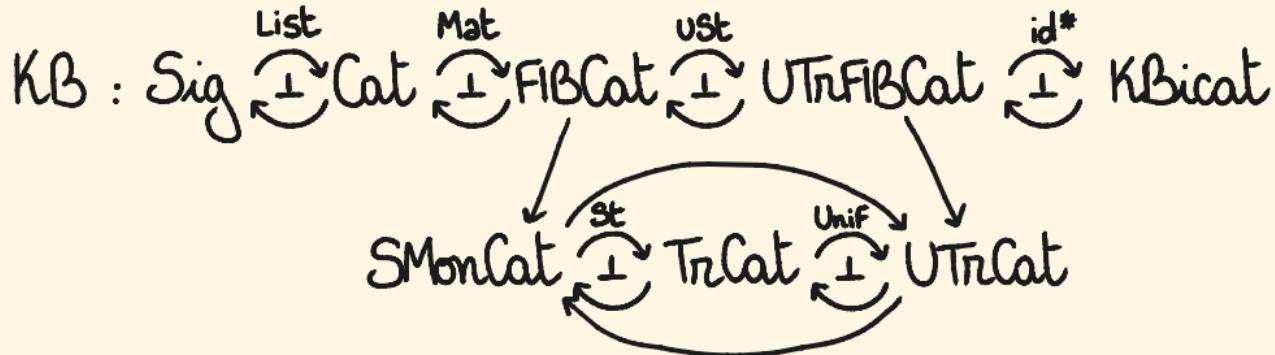
- The category of relations

- Kleene bicategories

[ • Tape diagrams for Kleene-cartesian bicategories ]

- Program logics

# FREE KLEENE BICATEGORIES



Objects of  $\text{KB}_\Sigma$  are lists of types

Morphisms  $A_1 \dots A_m \rightarrow B_1 \dots B_m$  are matrices with entries typed Kleene algebra expressions over  $\Sigma$ :

$$f_{ij} = b_{ij} a_{ij}^* c = -b_{ij} \overbrace{a_{ij}}^c c_{ij}$$

# FREE CARTESIAN BICATEGORIES

$$\text{CB} : \text{MonSig} \xrightarrow{\perp} \text{CBicat}$$

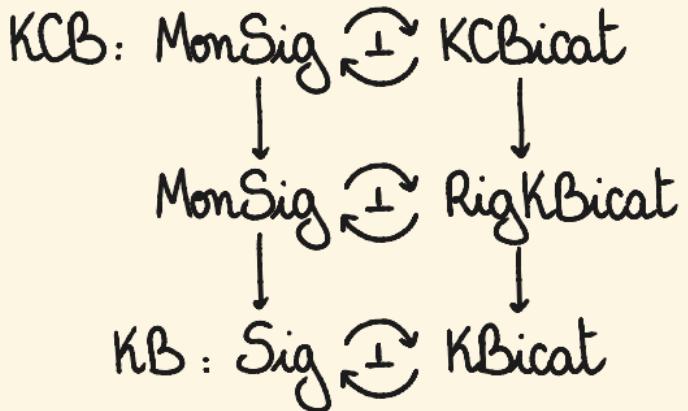
Objects of  $\text{CB}_\Sigma$  are lists of types

Morphisms  $A_1 \dots A_m \rightarrow B_1 \dots B_m$  are discrete cospan  
of directed hypergraphs over  $\Sigma$  :

$$n \rightarrow H \leftarrow m$$

[Rosebrugh, Sabadini, Walters (2005), Bonchi, Seeger, Sobociński (2018)]

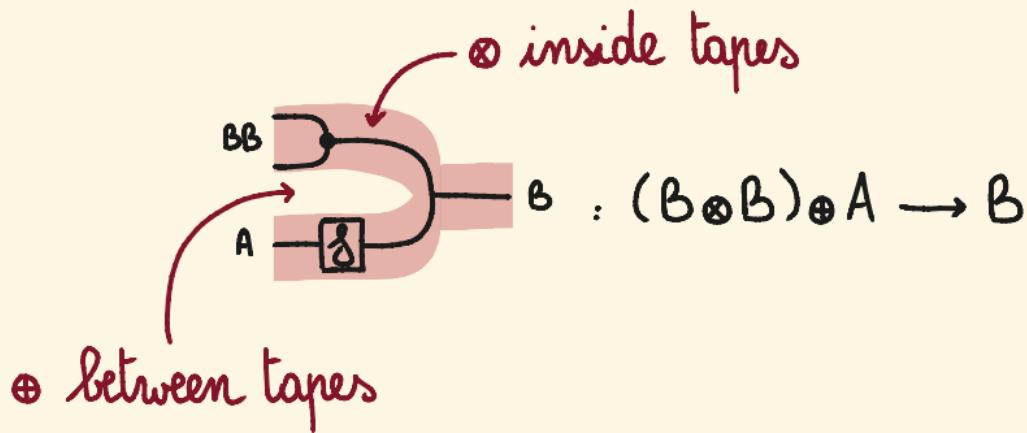
# FREE KLEENE-CARTESIAN BICATEGORIES



Objects of  $\text{KCB}_\Sigma$  are polynomials of types  
~~ what are the morphisms ?

# TAPE DIAGRAMS

String diagrams inside string diagrams.



[Bonchi, Di Giorgio, Santamaria (2023)]  
[clomfort, Delpuech, Jledges (2022)]

# TAPE DIAGRAMS FOR KLEENE-CARTESIAN BICATEGORIES

$(\text{---} \cup, \text{---} \cdot)$   $(\text{---} \cap, \text{---} \bullet)$  such that

$$\begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \bullet \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \bullet \end{array} = \begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \bullet \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \bullet \end{array} = \begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \boxed{\text{---}} \\ \text{---} \cup \\ \text{---} \end{array} \leq \begin{array}{c} \text{---} \\ \boxed{\text{---}} \\ \text{---} \cap \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \boxed{\text{---}} \\ \text{---} \bullet \\ \text{---} \end{array} \leq \begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \end{array} \leq \begin{array}{c} \text{---} \\ \text{---} \end{array}$$

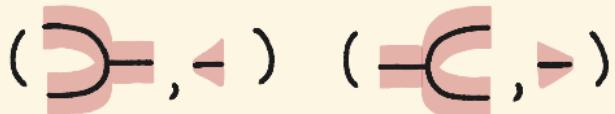
$$\begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array} \leq \begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \end{array} \leq \begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \bullet \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array} \leq \begin{array}{c} \text{---} \\ \text{---} \bullet \\ \text{---} \bullet \end{array}$$

$$\begin{array}{c} \text{---} \\ \text{---} \cap \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \cup \\ \text{---} \end{array}$$

# TAPE DIAGRAMS FOR KLEENE-CARTESIAN BICATEGORIES



$$\text{---} = \text{---} \quad \text{---} = \text{---} \quad \text{---} = \text{---}$$

$$\text{---} = \text{---} \quad \text{---} = \text{---} \quad \text{---} = \text{---}$$

$$\text{---} = \text{---} \quad \text{---} = \text{---}$$

# TAPE DIAGRAMS FOR KLEENE-CARTESIAN BICATEGORIES

$$\text{lefthand bracket } A \oplus B = \begin{array}{c} \text{A} \\ \diagdown \quad \diagup \\ \text{---} \\ \diagup \quad \diagdown \\ \text{B} \end{array}$$
$$\text{righthand bracket } A \oplus B = \begin{array}{c} \text{A} \\ \rightarrow \\ \text{B} \end{array}$$

$$\text{lefthand bracket } A \otimes B = \begin{array}{c} \text{A} \\ \diagup \quad \diagdown \\ \text{---} \\ \diagup \quad \diagdown \\ \text{B} \end{array}$$
$$\text{righthand bracket } A \otimes B = \begin{array}{c} \text{A} \\ \rightarrow \\ \text{B} \end{array}$$

# OUTLINE

- The category of relations
- Kleene bicategories
- Tape diagrams for Kleene-cartesian bicategories
- [• Program logics ]

# INTERPRETING A PROGRAMMING LANGUAGE

Var,  $\Sigma_E$ ,  $\Sigma_P$ ,  $\Sigma_C$

$E ::= x \in \text{Var} \mid e(E, \dots, E), e \in \Sigma_E(m)$

P ::=  $p \in \Sigma_P \mid \neg p, p \in \Sigma_P \mid P \wedge P \mid P \vee P \mid E = E$

C ::= c  $\in \Sigma_C \mid \text{skip} \mid CC \mid x \leftarrow E \mid \text{assert } P \mid \text{if } P \text{ then } C \text{ else } C \mid \text{while } P \text{ do } C$

choose a type X for variables

N := # Var

$\Rightarrow$  expressions are  $X^N \rightarrow X$

predicates are  $X^N \rightarrow I$

programs are  $X^N \rightarrow X^N$

# INTERPRETING A PROGRAMMING LANGUAGE

$E ::= x_i \in \text{Var} \mid e(E, \dots, E), e \in \Sigma_E(m)$

expressions are  $X^N \rightarrow X$  deterministic

$$x_i \mapsto \pi_i = i \begin{array}{c} x \\ \hline x \\ \vdots \\ x \end{array} x$$

$$e(t_1, \dots, t_m) \mapsto -\begin{bmatrix} m \\ x_N \end{bmatrix}; ([t_1] \otimes \dots \otimes [t_m]); [e]^{m=2} = \begin{array}{c} x \\ \hline \boxed{\begin{array}{c} t_1 \\ \otimes \\ t_2 \end{array}} \\ \hline e \\ \hline x \end{array}$$

# INTERPRETING A PROGRAMMING LANGUAGE

$$P ::= p \in \Sigma_p \mid \neg p, p \in \Sigma_p \mid P_1 P_2 \mid P_1 P_2 \mid E = E$$

predicates are  $X^N \rightarrow I$

$$p \mapsto [p] = \begin{array}{c} x \\ \text{---} \\ x \end{array} \text{D} \quad \text{for } p \in \Sigma_p \cup \neg \Sigma_p$$

$$p \wedge q \mapsto \neg \sqsubset_{X^N}; ([p] \otimes [q]) = \begin{array}{c} x \\ \text{---} \\ x \end{array} \text{C} \quad \text{with } \begin{array}{c} p \\ \text{---} \\ q \end{array}$$

$$p \vee q \mapsto \neg \sqsubset_{X^N}; ([p] \oplus [q]); \exists_I = \begin{array}{c} x \\ \text{---} \\ x \end{array} \text{C} \quad \text{with } \begin{array}{c} p \\ \text{---} \\ q \end{array}$$

$$e_1 = e_2 \mapsto \neg \sqsubset_{X^N}; ([e_1] \otimes [e_2]); \exists_x = \begin{array}{c} x \\ \text{---} \\ x \end{array} \text{C} \quad \text{with } \begin{array}{c} e_1 \\ \text{---} \\ e_2 \end{array}$$

# INTERPRETING A PROGRAMMING LANGUAGE (1)

$C ::= c \in \Sigma_C \mid \text{skip} \mid CC \mid x \leftarrow E \mid \text{assert } P \mid \text{if } P \text{ then } C \text{ else } C \mid \text{while } P \text{ do } C$   
programs are  $X^N \rightarrow X^N$

$$c \mapsto [c] = \begin{array}{c} x \\ \vdots \\ x \end{array} \boxed{c} \begin{array}{c} x \\ \vdots \\ x \end{array} \quad \text{for } c \in \Sigma_C$$

$$\text{skip} \mapsto id_{X^N} = \begin{array}{c} x \\ \vdots \\ x \end{array}$$

$$fg \mapsto [f];[g] = \begin{array}{c} x \\ \vdots \\ x \end{array} \boxed{f} \boxed{g} \begin{array}{c} x \\ \vdots \\ x \end{array}$$

$$x_i \leftarrow e \mapsto -\epsilon_{X^N}; (\pi_{-i} \otimes [e]); \bar{x}_{x^{N-i}, x} =$$



$$\text{assert } p \mapsto -\epsilon_{X^N}; (id_{X^N} \otimes [p]) = \begin{array}{c} x \\ \vdots \\ x \end{array} \boxed{p} \begin{array}{c} x \\ \vdots \\ x \end{array}$$

## INTERPRETING A PROGRAMMING LANGUAGE (2)

$C ::= c \in \Sigma_C \mid \text{skip} \mid CC \mid x \leftarrow E \mid \text{assert } P \mid \text{if } P \text{ then } C \text{ else } C \mid \text{while } P \text{ do } C$

$$\begin{aligned} \text{if } p \\ \text{then } g \\ \text{else } g' &\rightarrow \neg C; (([\text{assert } p]); [g]) \oplus ([\text{assert } \neg p]; [g']); \exists \\ &= \text{Diagram showing two parallel regions: one for } p \text{ (top) and one for } \neg p \text{ (bottom). Both regions contain an assertion node } g \text{ and a guard node } g'. \end{aligned}$$

$$\begin{aligned} \text{while } p \\ \text{do } g &\rightarrow \text{tr}(\exists; \neg C; (([\text{assert } p]); [g]) \oplus ([\text{assert } \neg p])) \\ &= \text{Diagram showing a large loop structure. It starts with a guard node } p \text{ (top), followed by an assertion node } g \text{ (middle), and ends with a guard node } p \text{ (bottom). The entire loop is enclosed in a large rounded rectangle.} \end{aligned}$$

# HOARE LOGIC

$\{p\} c \{q\}$  ↪ if the input satisfies  $p$  and  $c$  succeeds,  
then its output satisfies  $q$

$$\overline{\{p\} \text{skip} \{p\}}$$

$$\overline{\{p[e/x]\} x \leftarrow e \{p\}}$$

$$\overline{\{q\} \text{assert } p \{p \wedge q\}}$$

$$\frac{p \subseteq p' \quad \{p'\} c \{q'\} \quad q' \subseteq q}{\{p\} c \{q\}}$$

$$\frac{\{p\} c \{q\} \quad \{q\} d \{r\}}{\{p\} cd \{r\}}$$

$$\frac{\{p \wedge b\} c \{q\} \quad \{p \wedge \neg b\} d \{q\}}{\{p\} \text{if } b \text{ then } c \text{ else } d \{q\}}$$

$$\frac{\{p \wedge b\} c \{p\}}{\{p\} \text{while } b \text{ do } c \{p \wedge \neg b\}}$$

# INTERPRETING HOARE TRIPLES

## THEOREM

$\{c\}$  Kleene-cartesian bicategory  
 $[\cdot] : KCB_{\Sigma} \rightarrow \{c\}$

$\{p\} c \{q\}$  is derivable with the rules of Hoare logic  
 $\Rightarrow [p]^{\circ\circ}; [c] \leq [q]^{\circ\circ}$

# DERIVATION RULES (1)

$$\frac{}{\{p\} \text{skip}\{p\}}$$

$$[p]^\infty; [skip]$$

$$= \boxed{p}$$

$$= [p]^\infty$$

$$\frac{p \sqsubseteq p' \quad \{p'\} c \{q'\} \quad q' \sqsubseteq q}{\{p\} c \{q\}}$$

$$[p]^\infty; [c]$$

$$= \boxed{p \boxed{c}}$$

$$\leq \boxed{p \boxed{c}}$$

$$\leq \boxed{q'}$$

$$\leq \boxed{q}$$

$$= [q]^\infty$$

$$\frac{\{p\} c \{q\} \quad \{q\} d \{r\}}{\{p\} cd \{r\}}$$

$$[p]^\infty; [cd]$$

$$= \boxed{p \boxed{c} \boxed{d}}$$

$$\leq \boxed{q \boxed{d}}$$

$$\leq \boxed{r}$$

$$= [r]^\infty$$

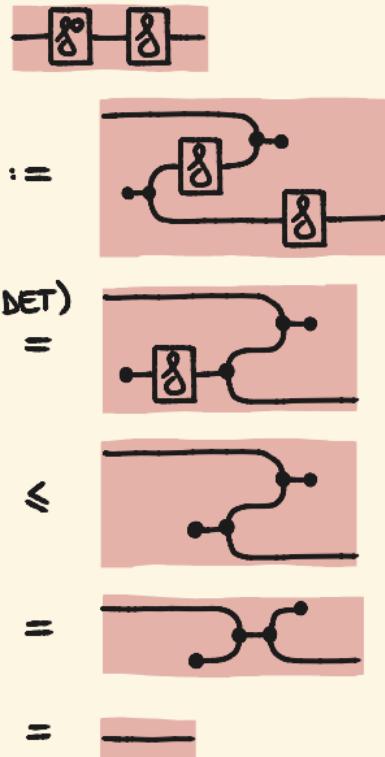
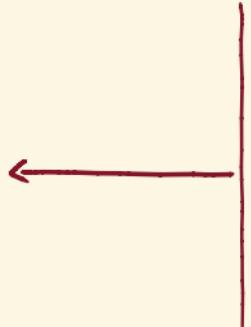
## DERIVATION RULES (2)

$\overline{\{p[e/x]\}} \ x \leftarrow e \ \{p\}$

$\{p[e/x]\}^\alpha; [x \leftarrow e]$



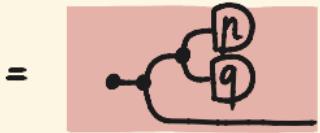
$= [p]^\alpha$



# DERIVATION RULES (3)

$\overline{[p] \text{ assert } q \{ p \wedge q \}}$

$[p]^{\text{op}}; [\text{assert } q]$

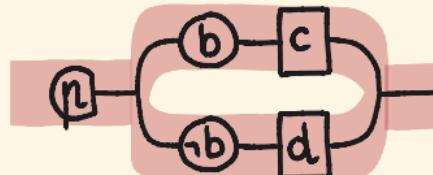


=  $[p \wedge q]^{\text{op}}$

# DERIVATION RULES (4)

$$\frac{\{p \wedge b\} c \{q\} \quad \{p \wedge \neg b\} d \{q\}}{\{p\} \text{ if } b \text{ then } c \text{ else } d \{q\}}$$

$$[p]^{\circ\circ}; [\text{if } b \text{ then } c \text{ else } d] =$$

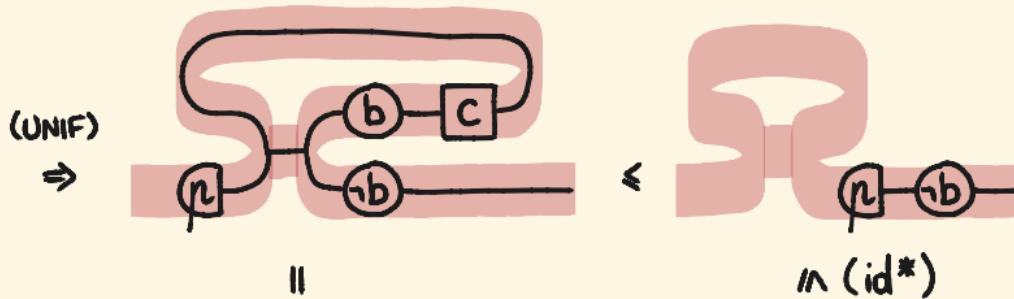
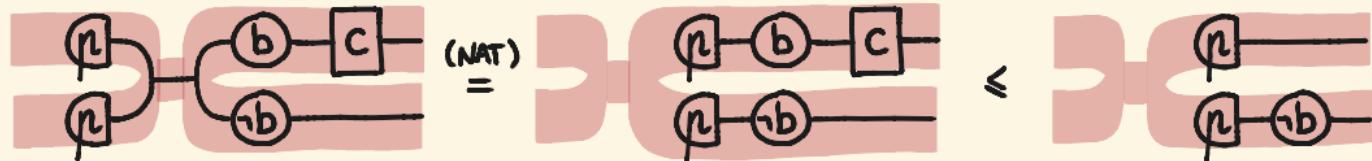


$$\begin{aligned} & (\text{NAT}) \\ &= \text{[red box with circuit]} \leq \text{[red box with q-q connection]} \end{aligned}$$

$$\begin{aligned} & (\text{NAT}) \\ &= \text{[red box with q-q connection]} \stackrel{(\text{DEM})}{=} \text{[red box with q-q connection]} = [q]^{\circ\circ} \end{aligned}$$

# DERIVATION RULES (5)

$$\frac{\{p \wedge b\} c \{p\}}{\{p\} \text{ while } b \text{ do } c \{p \wedge \neg b\}}$$



$[p]^{op}; [\text{while } b \text{ do } c]$

$\neg p \neg b = [p \wedge \neg b]^{op}$

# OTHER PROGRAM LOGICS

HOARE

$$\{p\} c \{q\} \quad [p]^\infty; [c] \leq [q]^\infty$$

if c terminates from p,  
its output is in q

INCORRECTNESS

$$[p] c [q] \quad [p]^\infty; [c] \geq [q]^\infty$$

all outputs in q can be  
reached by c starting in q

SUFFICIENT INCORRECTNESS

$$\ll p \gg c \ll q \gg \quad [p] \leq [c]; [q]$$

starting in p, c can  
terminate in q

NECESSARY

$$(p) c (q) \quad [p] \geq [c]; [q]$$

if c terminates in q,  
its input was in p

# SUMMARY

- Kleene bicategories  $\approx$  structure of  $(\text{Rel}, \sqcup, \emptyset)$ .  
 $(\rightarrow, \circ)$ ,  $(\leftarrow, \circ)$  natural coherent commutative (co)monoid

s.t.

$$\text{---} \circ \text{---} = \text{---}$$

$$\text{---} \circ \text{---} \leq \text{---}$$

$$\text{---} \xrightarrow{\delta} \text{---} \leq \text{---} \xrightarrow{u} \text{---} \Rightarrow \text{---} \xrightarrow{\delta} \text{---} \leq \text{---} \xrightarrow{g} \text{---}$$

- Kleene-cartesian bicategories express both control flow and data flow of classical programs.
- Kleene-cartesian bicategories are sound for floare logic.

## FUTURE WORK

- Effectful programs and their logics.
- Normal form for morphisms in  $\text{KB}_\ell$  and  $\text{KCB}_\Sigma$ .