Effectful Mealy Machines

Filippo Bonchi[1], Elena Di Lavore[2], Mario Román[2]

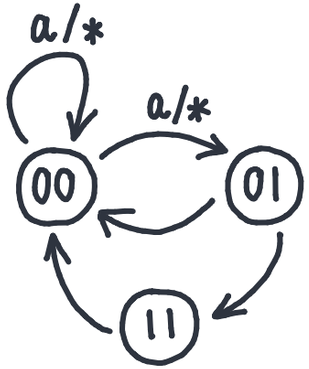
[1] Università di Pisa, [2] University of Oxford.

23rd June, LiCS '25 Singapore

Filippo Bonchi,
Università di Pisa,

Elena Di Lavore,
University of Oxford.

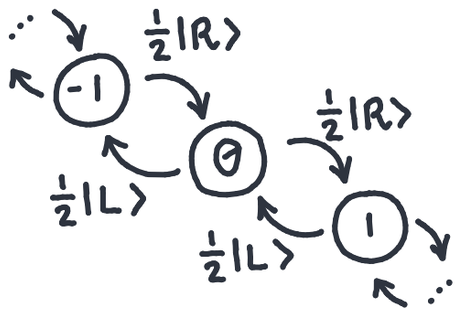
Part 0: Motivation



Mealy machine.

$s : I \to S$

$t : S \times A \to \mathcal{P}_\omega(S \times B)$

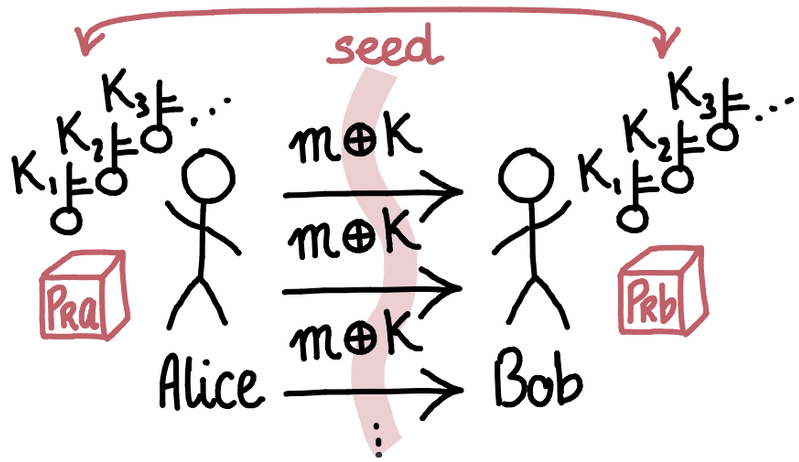


Markov decision process.

$t : S \times A \to \mathcal{D}(S)$

$r : S \times A \to \mathcal{D}(B)$

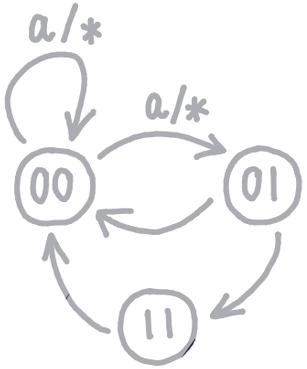
$s : 1 \to \mathcal{D}(S)$



Stateful sequential process.

$i : I \to M$

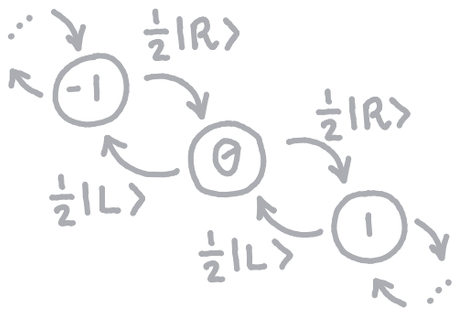
$t : S \otimes M \otimes A \to \mathcal{D}(S \otimes M \otimes B)$



Mealy machine.

$s : I \to S$

$t : S \times A \to \mathcal{P}_\omega(S \times B)$

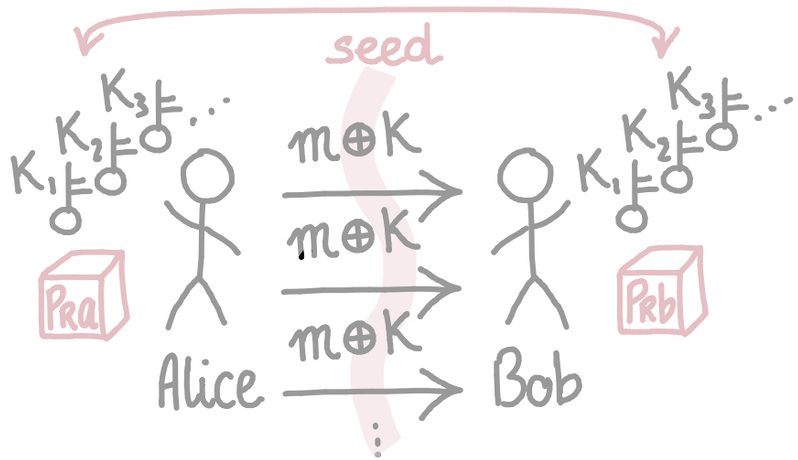


Markov decision process.

$t : S \times A \to \mathcal{D}(S)$

$r : S \times A \to \mathcal{D}(B)$

$s : 1 \to \mathcal{D}(S)$

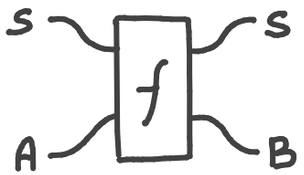


Stateful sequential process.

$i : I \to M$

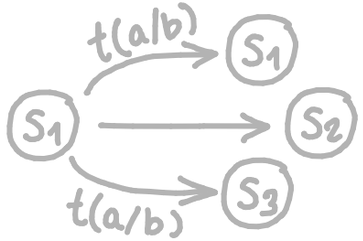
$t : S \otimes M \otimes A \to \mathcal{D}(S \otimes M \otimes B)$

Monoidal Composition, Coalgebraic Behaviour



$$S \otimes A \longrightarrow S \otimes B$$

- Composable: sequential and parallel.
- Trace-like iteration.
- Any symmetric monoidal category.

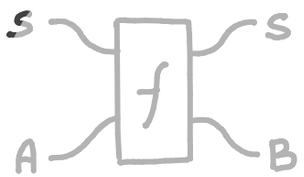


$$S \longrightarrow (S \otimes B)^A$$

- Coalgebraic description.
- Bisimilarity / Trace equivalence.
- Closed structure.

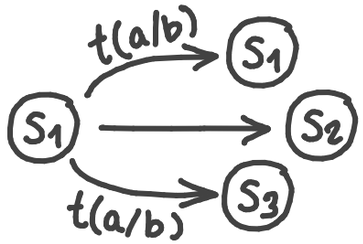
$S \times A \to S \times B$	Set	Deterministic	$S \to (S \times B)^A$
$S \times A \to S \times B + 1$	Par	Partial	$S \to (S \times B + 1)^A$
$S \times A \to \mathcal{P}(S \times B)$	Rel	Nondeterministic	$S \to \mathcal{P}(S \times B)^A$
$S \times A \to D(S \times B)$	KL(D)	Probabilistic	$S \to D(S \times B)^A$
$S \times A \to Q(S \times B)$	KL(Q)	Quantum	$S \to Q(S \times B)^A$

Monoidal Composition, Coalgebraic Behaviour



$$S \otimes A \longrightarrow S \otimes B$$

- Composable: sequential and parallel.
- Trace-like iteration.
- Any symmetric monoidal category.



$$S \longrightarrow (S \otimes B)^A$$

- Coalgebraic description.
- Bisimilarity / Trace equivalence.
- Closed structure.

$S \times A \to S \times B$	Set	$S \to (S \times B)^A$
$S \times A \to S \times B + 1$	Par	$S \to (S \times B + 1)^A$
$S \times A \to \mathcal{P}(S \times B)$	Rel	$S \to \mathcal{P}(S \times B)^A$
$S \times A \to D(S \times B)$	KL(D)	$S \to D(S \times B)^A$
$S \times A \to Q(S \times B)$	KL(Q)	$S \to Q(S \times B)^A$

Deterministic
Partial
Nondeterministic
Probabilistic
Quantum

Summary

1. Effectful triples: theories of processes.
2. Machines over an effectful triple: unifying bisimulation.
3. Traces over an effectful triple: effectful streams; causal processes.

Part 1. Effectful Triples

Effectful Triples

cartesian category

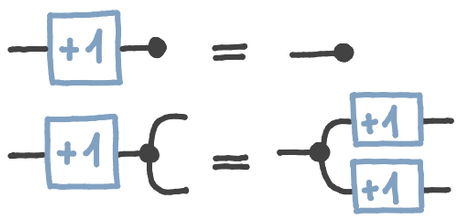
symmetric monoidal

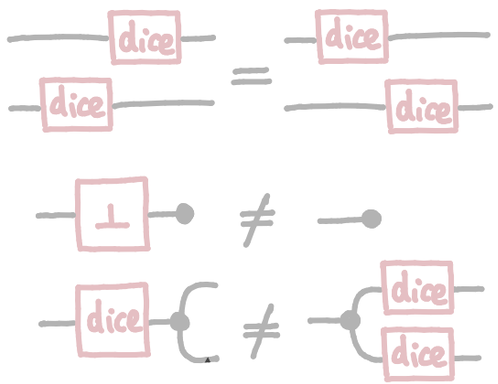
symmetric premonoidal

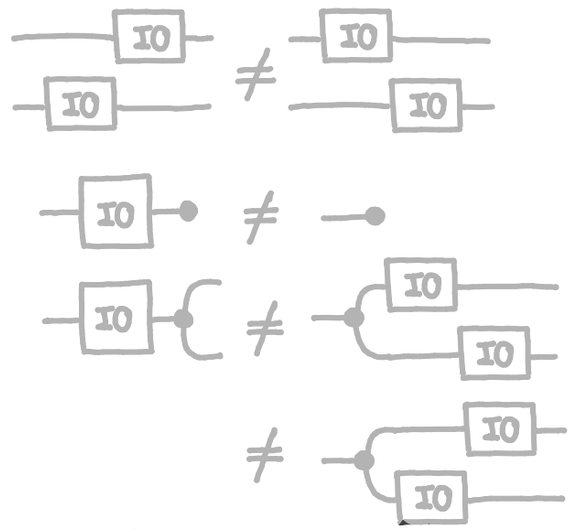
Values that are both deterministic and total

Local effects that interchange, but are not always deterministic nor total.

GLOBAL EFFECTS that do not interchange.







Jeffrey, 1998. Power and Robinson, 1997. Román, Sobociński, 2025.

Effectful Triples

cartesian category

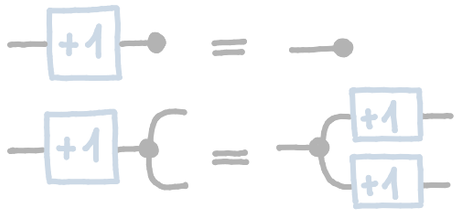
symmetric monoidal

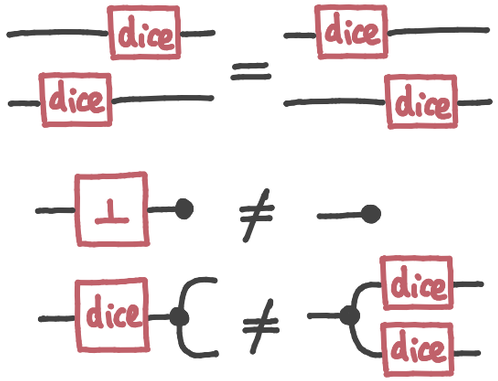
symmetric premonoidal

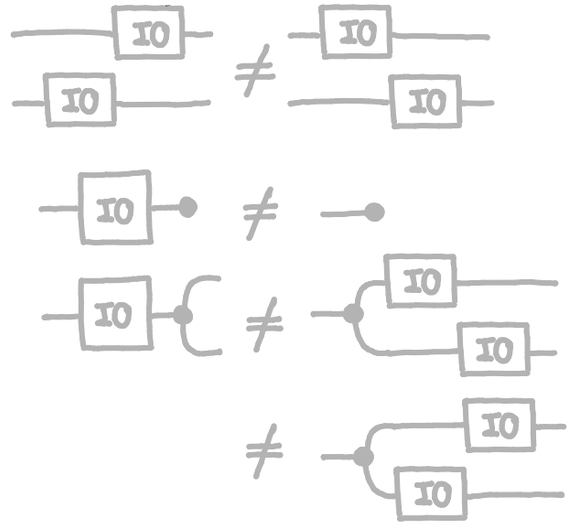
Values that are both deterministic and total

Local effects that interchange, but are not always deterministic nor total.

GLOBAL EFFECTS that do not interchange.







Jeffrey, 1998. Power and Robinson, 1997. Román, Sobociński, 2025.

Effectful Triples

cartesian category

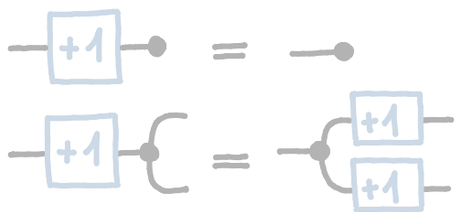
symmetric monoidal

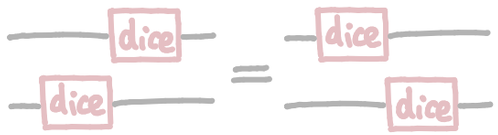
symmetric premonoidal

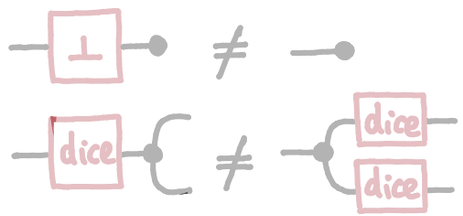
Values that are both deterministic and total

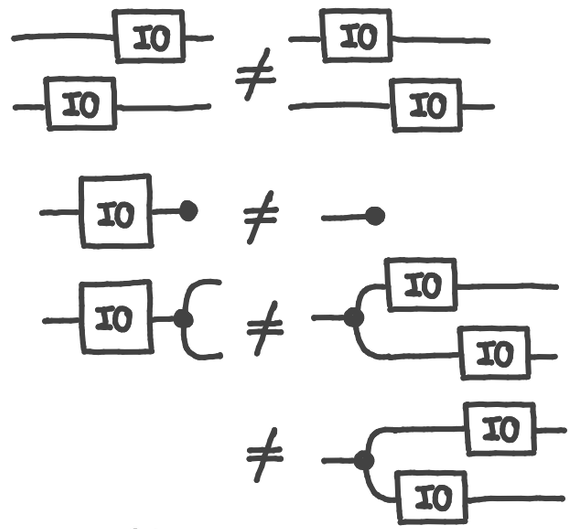
Local effects that interchange, but are not always deterministic nor total.

GLOBAL EFFECTS that do not interchange.









Jeffrey, 1998.

Power and Robinson, 1997.

Román, Sobociński, 2025.

Effectful Triples

cartesian category

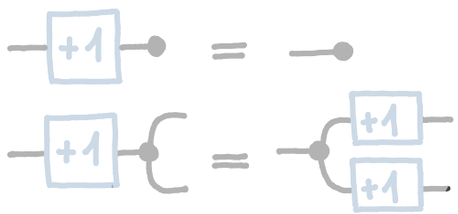
symmetric monoidal

Alan Jeffrey's string diagrams with extra wire

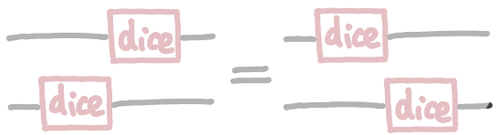
↓

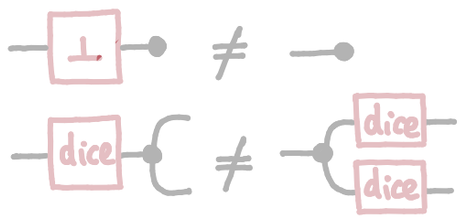
symmetric premonoidal

Values that are both deterministic and total

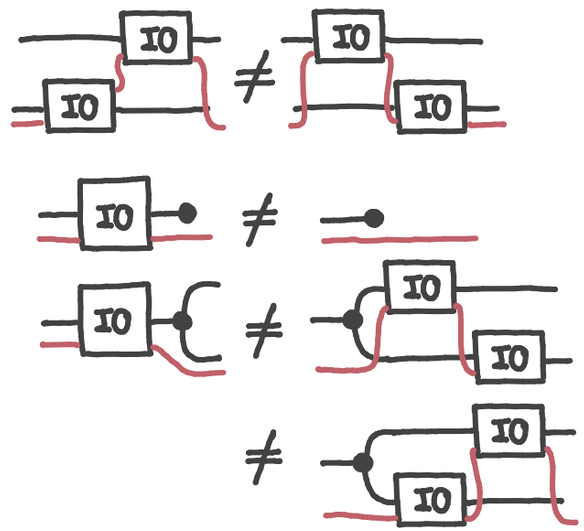


Local effects that interchange, but are not always deterministic nor total.





Global Effects that do not interchange.



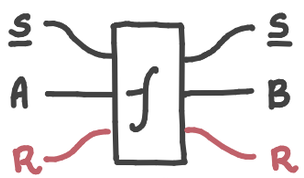
Jeffrey, 1998. Power and Robinson, 1997. Román, Sobociński, 2025.

Part 2. Effectful Machines

Effectful Machines

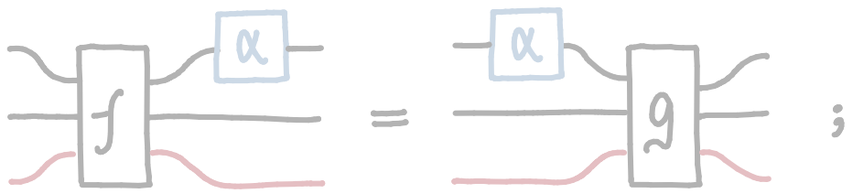
DEFINITION. An effectful machine over $\mathbb{V}, \mathbb{P}, \mathbb{C}$ is
- a state object, $S \in \mathrm{obj}$;
- an initial state, $i \in \mathbb{P}(I; S)$;
- a transition morphism, $t : \mathbb{C}(S \otimes A; S \otimes B)$.

initial state



transition

DEFINITION (Homomorphism of machines).
A homomorphism $\alpha : (S, i, f) \to (T, j, g)$ is $\alpha \in \mathbb{V}(A; B)$ such that



REMARK. A shadow of the uniformity axiom of traced monoidal categories.

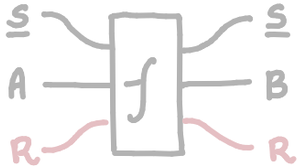
Katis, Sabadini, Walters, 1997.

Di Lavore, Gianola, Román, Sabadini, Sobociński, 2022.

Effectful Machines

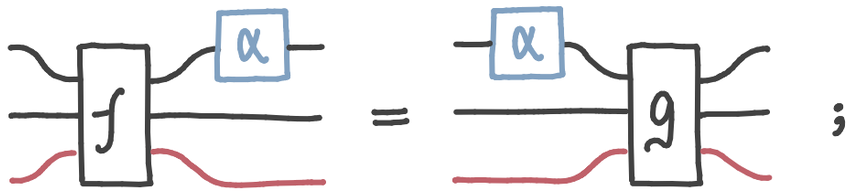
DEFINITION. An effectful machine over $\mathbb{V}, \mathbb{P}, \mathbb{C}$ is
- a state object, $S \in \mathrm{obj}$;
- an initial state, $i \in \mathbb{P}(I; S)$;
- a transition morphism, $t : \mathbb{C}(S \otimes A; S \otimes B)$.

initial state



transition

DEFINITION (Homomorphism of machines).
A homomorphism $\alpha : (S, i, f) \to (T, j, g)$ is $\alpha \in \mathbb{V}(A; B)$ such that



REMARK. A shadow of the uniformity axiom of traced monoidal categories.

Katis, Sabadini, Walters, 1997.

Di Lavore, Gianola, Román, Sabadini, Sobociński, 2022.

Bisimilarity

In Kleisli categories for a monad T, machines are $F_T(S) = T(S \times B)^A$ - coalgebras.

$$\frac{S \times A \to T(S \times B)}{S \to T(S \times B)^A}$$

Definition (Bisimilarity). Two machines are bisimilar when connected by coalgebra span.

$$\begin{array}{ccccc} M & \xleftarrow{\pi_1} & R & \xrightarrow{\pi_2} & N \\ {\scriptstyle f}\downarrow & & \downarrow & & \downarrow{\scriptstyle g} \\ F_T(M) & \leftarrow & F_T(R) & \rightarrow & F_T(N) \end{array}$$

Theorem (Rutten, 2000). When $F_T : \mathrm{Set} \to \mathrm{Set}$ preserves pullbacks, bisimilarity is an equivalence.

Definition (Bisimilarity). Two machines are bisimilar when connected by a sequence of spans of machine homomorphisms.

$$(S, i, f) \quad \leftarrow (R_1, r_1, t_1) \rightarrow \quad \cdots \quad \leftarrow (R_n, r_n, t_n) \rightarrow \quad (T, j, g)$$

Proposition. For commutative monads preserving weak pullbacks, effectful bisimulation is coalgebraic bisimulation.

Recovers: bisimulation (REL), deterministic machine bisimulation (SET), partial machine bisimulation (e.g. Baier, Katoen, PAR), and probabilistic (Larsen, Skou, KL(Dist)).

Bisimilarity

In Kleisli categories for a monad T, machines are $F_T(S) = T(S \times B)^A$ - coalgebras.

$$\frac{S \times A \to T(S \times B)}{S \to T(S \times B)^A}$$

Definition (Bisimilarity). Two machines are bisimilar when connected by coalgebra span.

$$\begin{array}{ccccc} M & \xleftarrow{\pi_1} & R & \xrightarrow{\pi_2} & N \\ {\scriptstyle f}\downarrow & & \downarrow & & \downarrow{\scriptstyle g} \\ F_T(M) & \leftarrow & F_T(R) & \rightarrow & F_T(N) \end{array}$$

Theorem (Rutten, 2000). When $F_T : Set \to Set$ preserves pullbacks, bisimilarity is an equivalence.

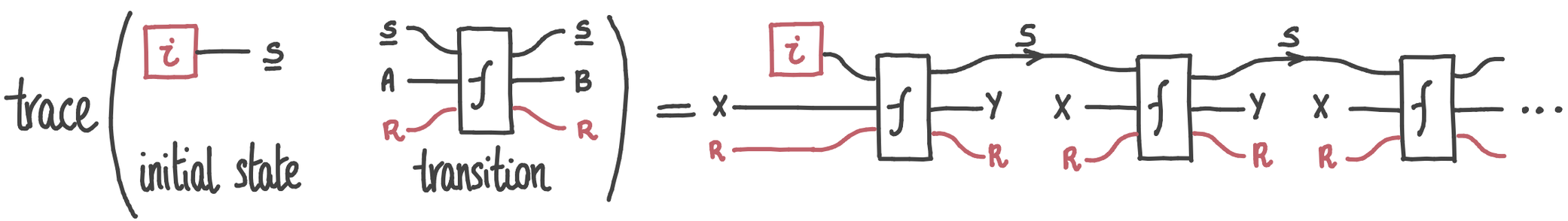
Definition (Bisimilarity). Two machines are bisimilar when connected by a sequence of spans of machine homomorphisms.

$$(S, i, f) \quad \leftarrow (R_1, r_1, t_1) \rightarrow \quad \cdots \quad \leftarrow (R_n, r_n, t_n) \rightarrow \quad (T, j, g)$$

Proposition. For commutative monads preserving weak pullbacks, effectful bisimulation is coalgebraic bisimulation.

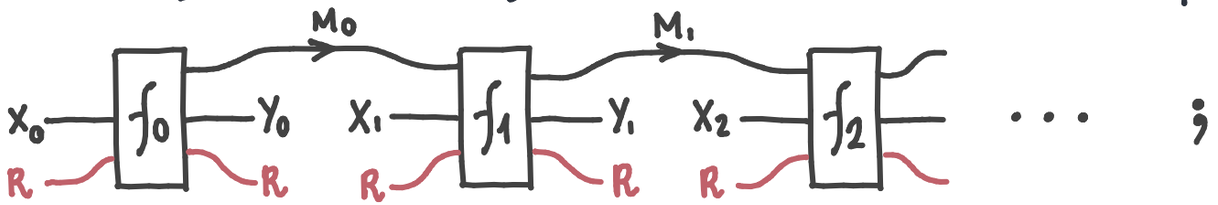
Recovers: bisimulation (Rel), deterministic machine bisimulation (Set), partial machine bisimulation (e.g. Baier, Katoen, Par), and probabilistic (Larsen, Skou, KL(Dist)).

Part 3. Stream Semantics

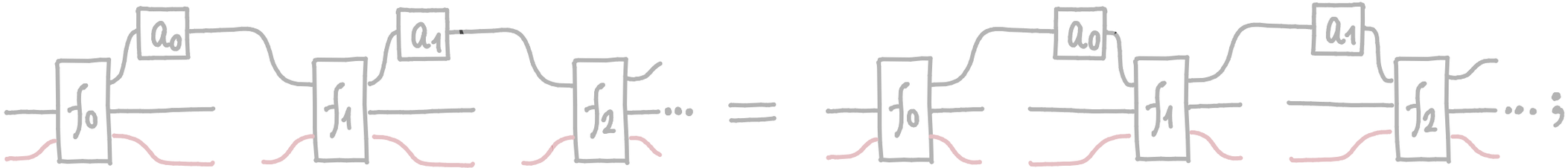


Effectful Streams

Definition (Effectful Stream). An effectful stream, $f : \mathrm{Stream}(A; B)$ consists of a morphism $f_0 : \mathbb{C}(A_0; M \otimes B_0)$, followed by a stream, $f : \mathrm{Stream}(M \cdot A^+; B^+)$. They are quotiented by sliding.



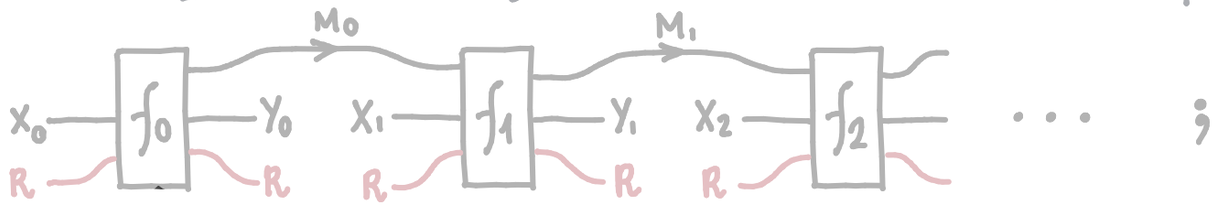
Sliding is defined in terms of dinaturality, $\phi(Q)(X,Y) = \int^M \mathbb{C}(X_0; M \otimes Y_0) \times Q(M \cdot X_+; Y_+)$.



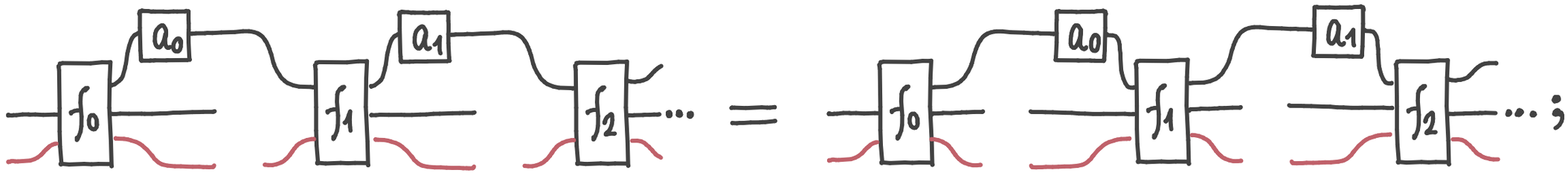
Di Lavore, de Felice, Román, 2022. Sprunger, Katsumata, 2019.

Effectful Streams

DEFINITION (Effectful Stream). An effectful stream, $f : \mathrm{Stream}(A; B)$ consists of a morphism $f_0 : \mathbb{C}(A_0; M \otimes B_0)$, followed by a stream, $f : \mathrm{Stream}(M \cdot A^+; B^+)$. They are quotiented by sliding.



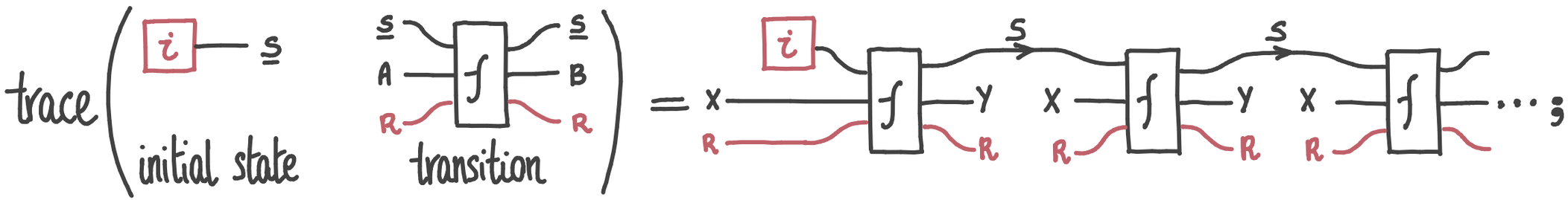
Sliding is defined in terms of dinaturality, $\phi(Q)(X,Y) = \int^M \mathbb{C}(X_0; M \otimes Y_0) \times Q(M \cdot X_+; Y_+)$.



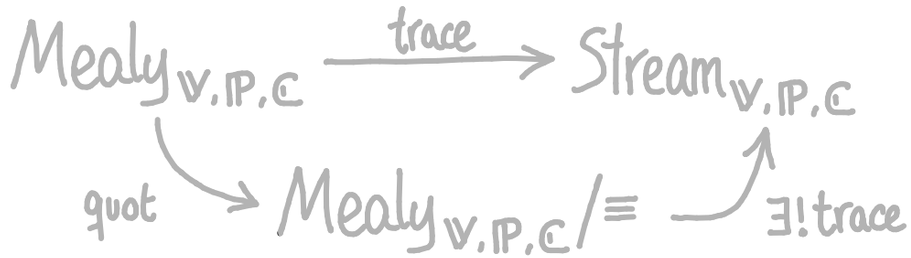
Di Lavore, de Felice, Román, 2022. Sprunger, Katsumata, 2019.

Effectful Streams

Definition (Trace of an effectful machine).



Theorem. Bisimilarity implies trace equivalence.

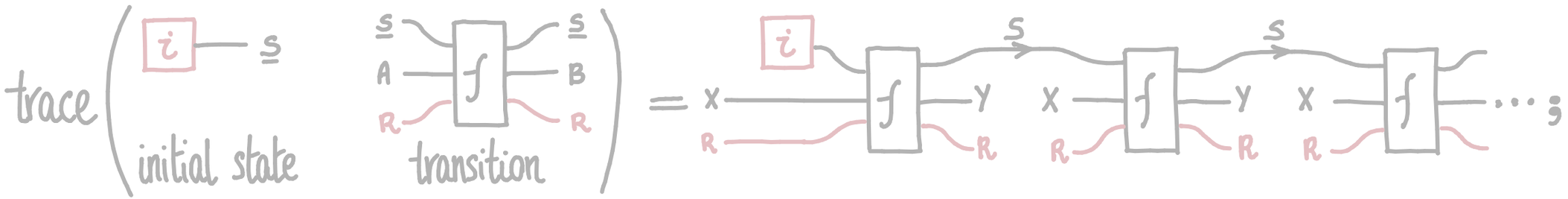


Di Lavore, de Felice, Román, 2022.

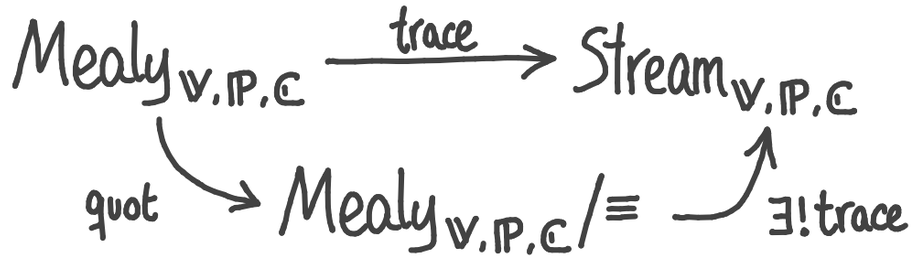
Sprunger, Katsumata, 2019.

Effectful Streams

DEFINITION (Trace of an effectful machine).



THEOREM. Bisimilarity implies trace equivalence.



Di Lavore, de Felice, Román, 2022.

Sprunger, Katsumata, 2019.

Part 3. Causal Processes.

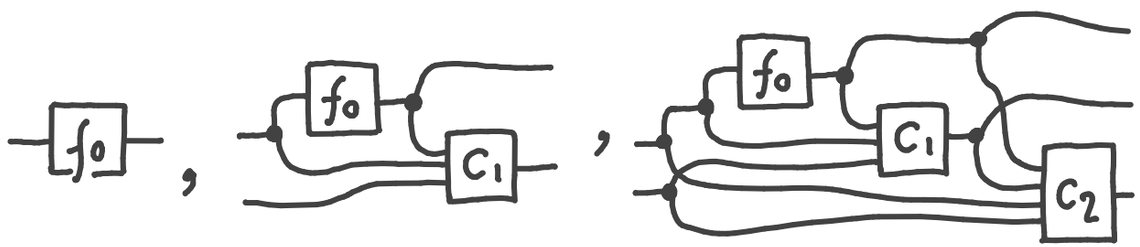
Causal Processes

Usually, controlled traces are simpler than streams: sliding equivalence may be difficult.

THEOREM (Raney, 1958). Causal functions are the executions of deterministic Mealy machines,

$$f_0 : X_0 \to Y_0,$$
$$f_1 : X_0 \times X_1 \to Y_1,$$
$$f_2 : X_0 \times X_1 \times X_2 \to Y_2,$$
$$\cdots$$

Generalizing requires some care: e.g. joint distributions cannot be split into marginals.



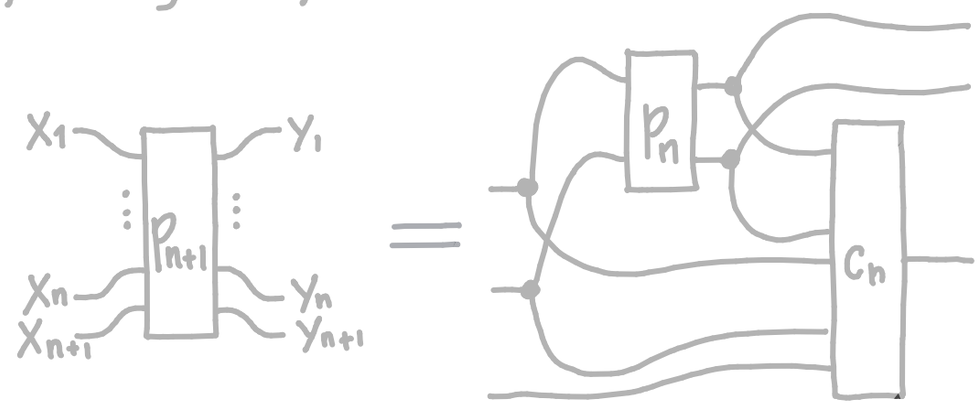
DEFINITION. In a copy-discard category, a causal process,

$$p : X_1, X_2, X_3, \ldots \to Y_1, Y_2, Y_3, \ldots$$

is a family of morphisms,

$$p_n : X_1 \otimes \cdots \otimes X_n \to Y_1 \otimes \cdots \otimes Y_n$$

factoring as follows



for some morphisms,

$$C_{n+1} : Y_0 \otimes \cdots \otimes Y_n \otimes X_0 \otimes \cdots \otimes X_n \otimes X_{n+1} \to Y_{n+1}.$$

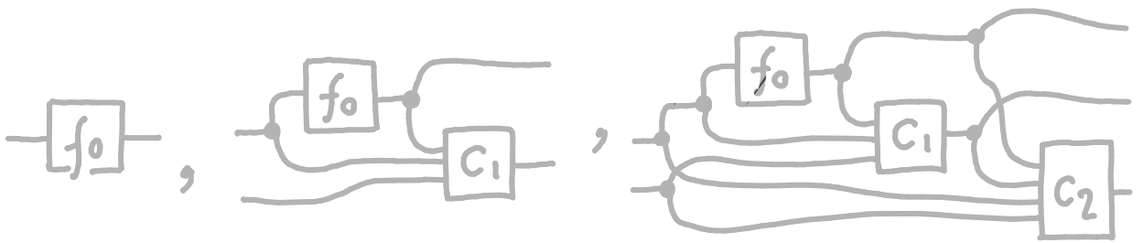
Causal Processes

Usually, controlled traces are simpler than streams: sliding equivalence may be difficult.

THEOREM (Raney, 1958). Causal functions are the executions of deterministic Mealy machines,

$$f_0 : X_0 \to Y_0,$$
$$f_1 : X_0 \times X_1 \to Y_1,$$
$$f_2 : X_0 \times X_1 \times X_2 \to Y_2,$$
$$\cdots$$

Generalizing requires some care: e.g. joint distributions cannot be split into marginals.



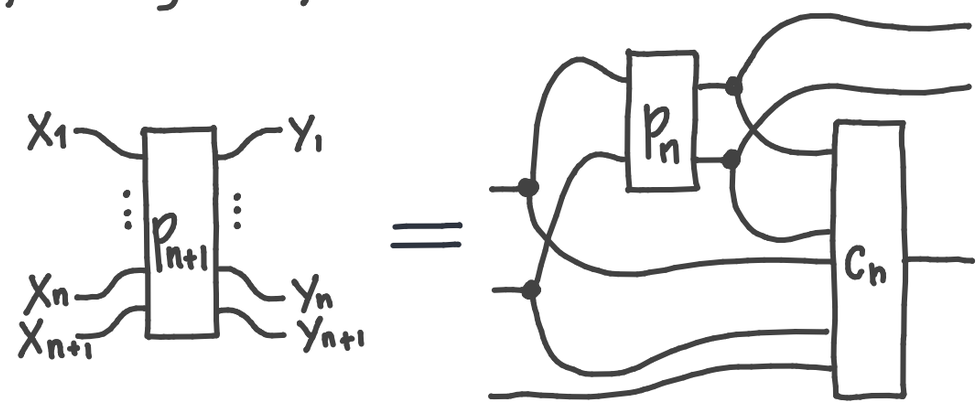
DEFINITION. In a copy-discard category, a causal process,

$$p : X_1, X_2, X_3, \ldots \to Y_1, Y_2, Y_3, \ldots$$

is a family of morphisms,

$$p_n : X_1 \otimes \cdots \otimes X_n \to Y_1 \otimes \cdots \otimes Y_n$$

factoring as follows



for some morphisms,

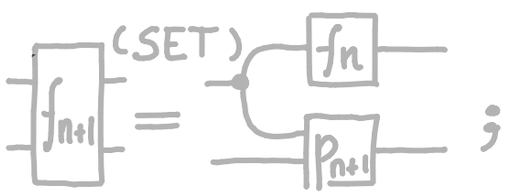
$$C_{n+1} : Y_0 \otimes \cdots \otimes Y_n \otimes X_0 \otimes \cdots \otimes X_n \otimes X_{n+1} \to Y_{n+1}.$$

CAUSAL PROCESSES (Raney's theorem)

THEOREM. For a copy-discard category $\mathbb{C}$, with conditionals and ranges, consider $(\mathit{fun}(\mathbb{C}), \mathit{tot}(\mathbb{C}), \mathbb{C})$; causal processes coincide with streams.

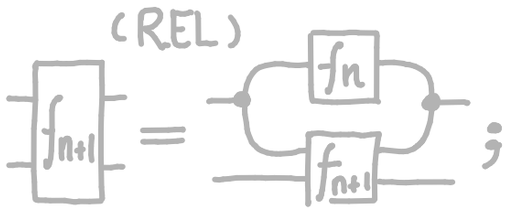
CAUSALITY CONDITION.

(SET)



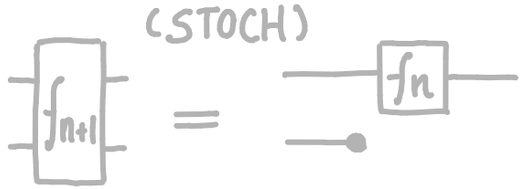
;

(REL)



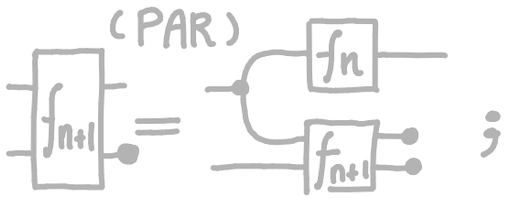
;

(STOCH)



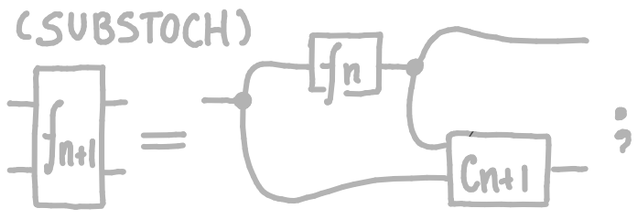
;

(PAR)



;

(SUBSTOCH)



;

TRACE PREDICATES.

SET: $(\underline{s} = s_0) \;\wedge\; \forall i \leq n \,.\, f(s_i, a_i) = (s_{i+1}, b_i)$

PAR: $(\underline{s} = s_0) \;\wedge\; \forall i \leq n \,.\, f(s_i, a_i)\!\downarrow \;\wedge\; f(s_i, a_i) = (s_{i+1}, b_i)$

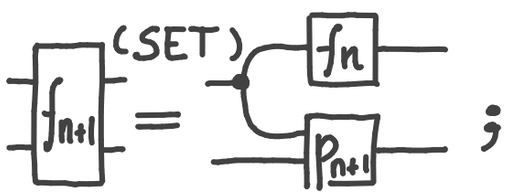
REL: $\exists\, s_0, \ldots, s_{n+1} \,.\, (s_0 \in \underline{s}) \;\wedge\; \forall i \leq n \,.\, f(s_i, a_i) \ni (s_{i+1}, b_i)$

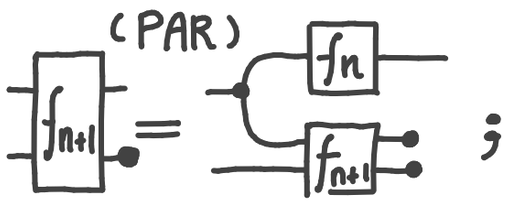
STOCH≤ 1: $\sum s_0, \ldots, s_{n+1} \,.\, \underline{s}(s_0) \cdot \prod_{i \leq n} f(s_{i+1}, b_i \,|\, s_i, a_i)$.

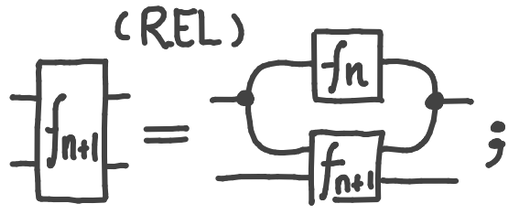
CAUSAL PROCESSES (Raney's theorem)

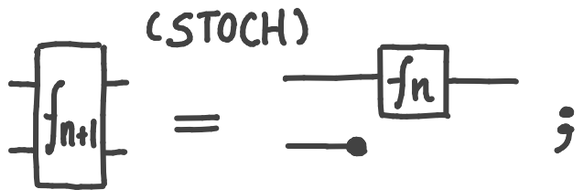
THEOREM. For a copy-discard category $\mathbb{C}$, with conditionals and ranges, consider $(\mathit{fun}(\mathbb{C}), \mathit{tot}(\mathbb{C}), \mathbb{C})$; causal processes coincide with streams.

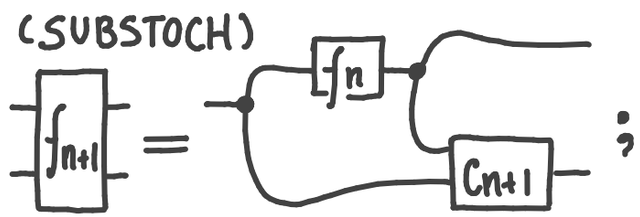
CAUSALITY CONDITION.











TRACE PREDICATES.

SET: $(\underline{s} = s_0) \wedge \forall i \leq n.\ f(s_i, a_i) = (s_{i+1}, b_i)$

PAR: $(\underline{s} = s_0) \wedge \forall i \leq n.\ f(s_i, a_i)\downarrow \wedge f(s_i, a_i) = (s_{i+1}, b_i)$

REL: $\exists s_0, \ldots, s_{n+1}.\ (s_0 \in \underline{s}) \wedge \forall i \leq n.\ f(s_i, a_i) \ni (s_{i+1}, b_i)$

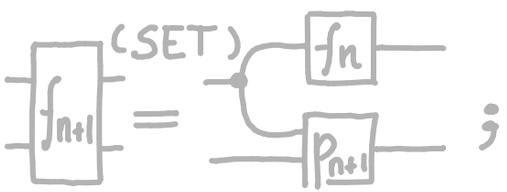
STOCH≤ 1: $\sum s_0, \ldots, s_{n+1}.\ \underline{s}(s_0) \cdot \prod_{i \leq n} f(s_{i+1}, b_i \mid s_i, a_i).$

CAUSAL PROCESSES (Raney's theorem)

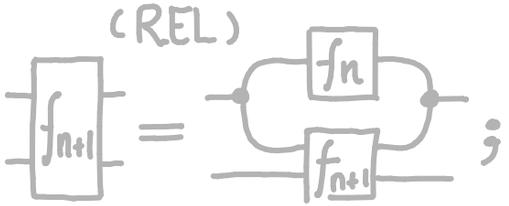
THEOREM. For a copy-discard category $\mathbb{C}$, with conditionals and ranges, consider $(\mathit{fun}(\mathbb{C}), \mathit{tot}(\mathbb{C}), \mathbb{C})$; causal processes coincide with streams.

CAUSALITY CONDITION.

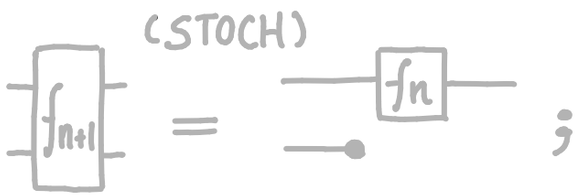
(SET)



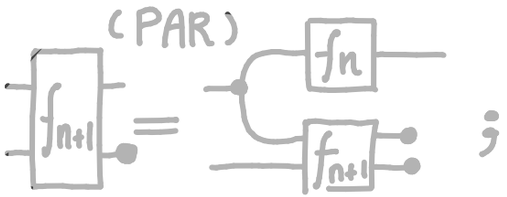
(REL)



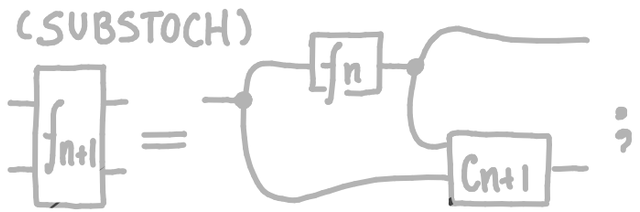
(STOCH)



(PAR)



(SUBSTOCH)



TRACE PREDICATES.

SET: $(\underline{s} = s_0) \wedge \forall i \leq n \,.\, f(s_i, a_i) = (s_{i+1}, b_i)$

PAR: $(\underline{s} = s_0) \wedge \forall i \leq n \,.\, f(s_i, a_i)\!\downarrow \wedge\, f(s_i, a_i) = (s_{i+1}, b_i)$

REL: $\exists s_0, \ldots, s_{n+1} \,.\, (s_0 \in \underline{s}) \wedge \forall i \leq n \,.\, f(s_i, a_i) \ni (s_{i+1}, b_i)$

STOCH≤ 1: $\sum s_0, \ldots, s_{n+1} \,.\, \underline{s}(s_0) \cdot \prod_{i \leq n} f(s_{i+1}, b_i \mid s_i, a_i)$.

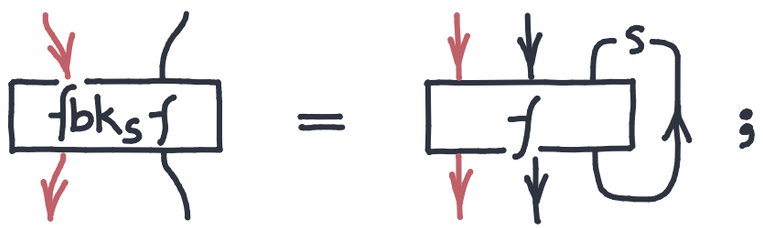
Summary

1. Effectful triples: theories of processes.
2. Machines over an effectful triple: unifying bisimulation.
3. Traces over an effectful triple: effectful streams; causal processes.

END

FEEDBACK STRUCTURE

THEOREM. Effectful machines quotiented by bisimilarity are the free uniform feedback structure over an effectful triple.

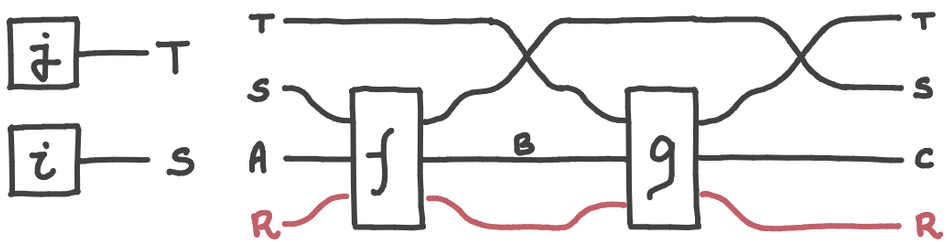


DEFINITION (Uniform feedback structure). A feedback structure over $(\mathbb{V}, \mathbb{P}, \mathbb{C})$ is a premonoidal category, $\mathbb{F}$, with a feedback operator

$$\text{FBK}: \mathbb{P}(I;S) \times \mathbb{F}(S \otimes X; S \otimes Y) \longrightarrow \mathbb{F}(X;Y),$$

satisfying $\mathbb{C}$-uniform feedback axioms.

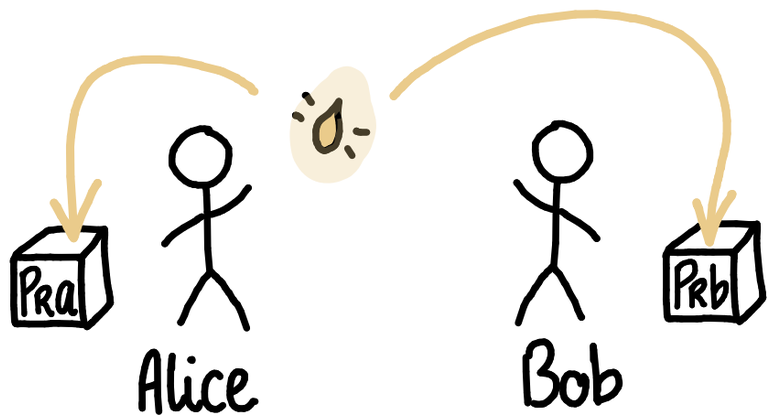
PROPOSITION. Machines over an effectful triple V,P,C, form an effectful triple.



- Kleisli categories of strong (pro)monads: $(\mathcal{V}, \mathcal{Z}(Kl(T)), Kl(T))$.

Example: Stream Cipher

```
alice(m)° =
    seed() ⇝ ()
    rand_a() ⇝ k_a
    return (m ⊕ k_a);
alice(m)^+° =
    rand_a() ⇝ k_a
    return (m ⊕ k_a);
alice(m)^++ = alice(m)^+;
```

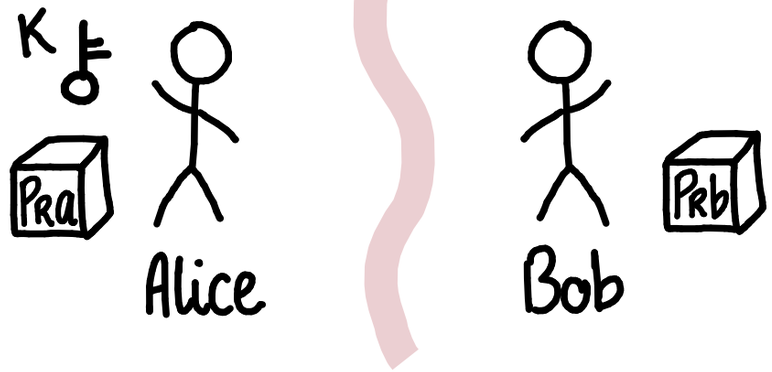



```
bob(n)° =
    rand_b() ⇝ k_b
    return (n ⊕ k_b);

bob(n)^+ = bob(n);
```

Example: Stream Cipher

$alice(m)^{\circ} =$
  $seed() \rightsquigarrow ()$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$
$alice(m)^{+\circ} =$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$
$alice(m)^{++} = alice(m)^{+};$

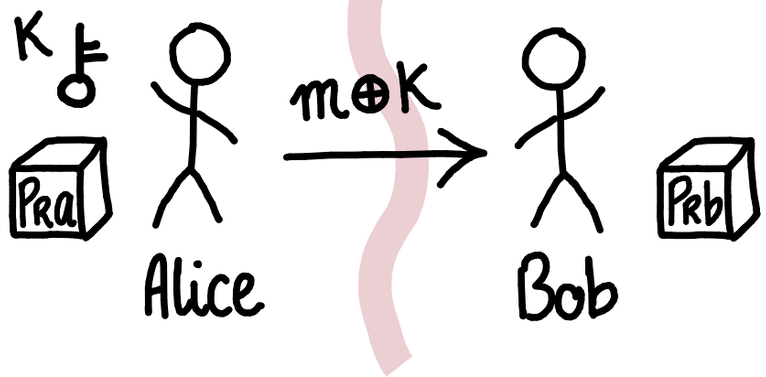


$bob(n)^{\circ} =$
  $rand_b() \rightsquigarrow k_b$
  $return(n \oplus k_b);$

$bob(n)^{+} = bob(n);$

Example: Stream Cipher

$alice(m)^{\circ} =$
  $seed() \rightsquigarrow ()$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$
$alice(m)^{+\circ} =$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$
$alice(m)^{++} = alice(m)^{+};$

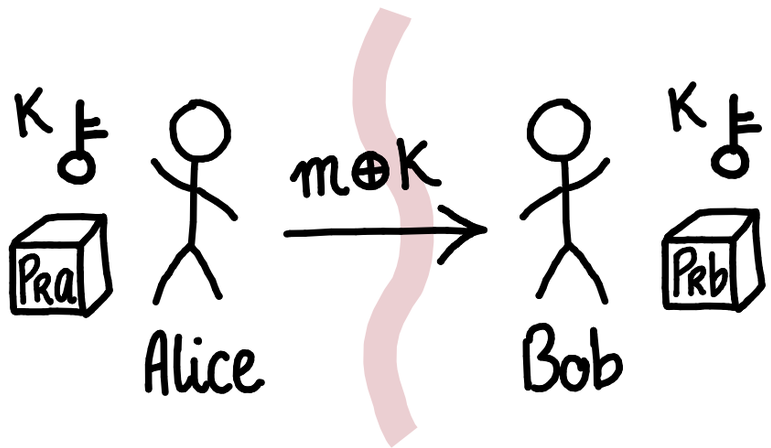


$bob(n)^{\circ} =$
  $rand_b() \rightsquigarrow k_b$
  $return(n \oplus k_b);$

$bob(n)^{+} = bob(n);$

Example: Stream Cipher

```
alice(m)° =
    seed() ⇝ ()
    rand_a() ⇝ k_a
    return(m ⊕ k_a);
alice(m)^+° =
    rand_a() ⇝ k_a
    return(m ⊕ k_a);
alice(m)^++ = alice(m)^+;
```

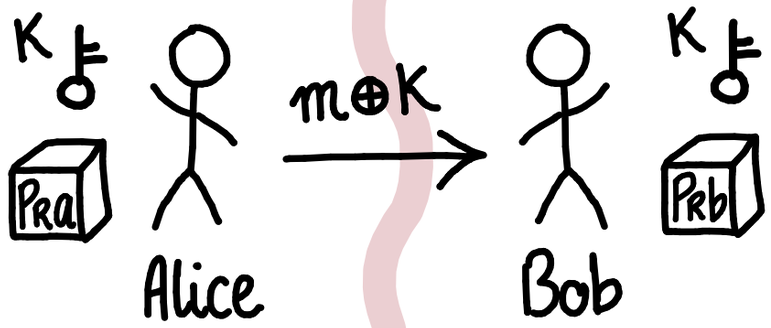



```
bob(n)° =
    rand_b() ⇝ k_b
    return(n ⊕ k_b);

bob(n)^+ = bob(n);
```

Example: Stream Cipher

$alice(m)^{\circ} =$
  $seed() \rightsquigarrow ()$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$
$alice(m)^{+\circ} =$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$
$alice(m)^{++} = alice(m)^{+};$



$bob(n)^{\circ} =$
  $rand_b() \rightsquigarrow k_b$
  $return(n \oplus k_b);$

$bob(n)^{+} = bob(n);$

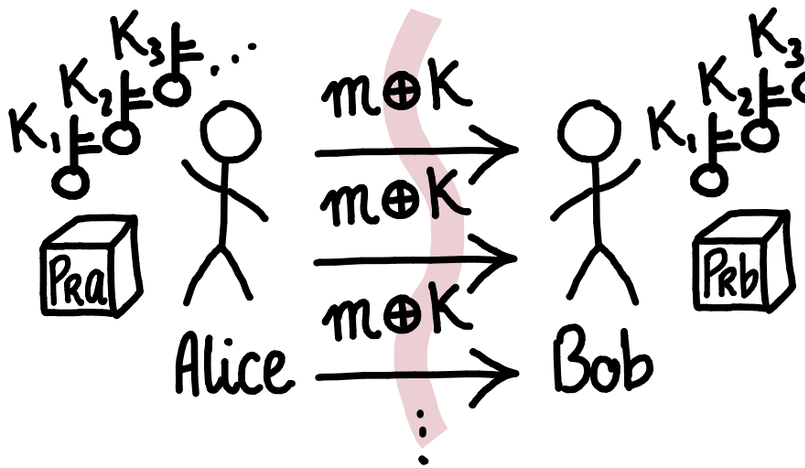
$n \oplus K = (m \oplus K) \oplus K = m$

Example: Stream Cipher

$alice(m)^{\circ} =$
  $seed() \rightsquigarrow ()$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$

$alice(m)^{+\circ} =$
  $rand_a() \rightsquigarrow k_a$
  $return(m \oplus k_a);$

$alice(m)^{++} = alice(m)^{+};$



$bob(n)^{\circ} =$
  $rand_b() \rightsquigarrow k_b$
  $return(n \oplus k_b);$

$bob(n)^{+} = bob(n);$