IRIF 12 March 2025

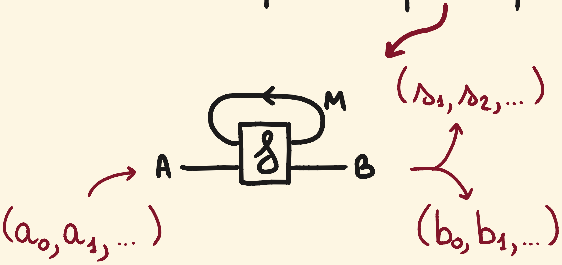
EFFECTFUL STREAMS FOR DATAFLOW PROGRAMMING

Elena Di Lavore
University of Oxford

j.w.w. Mario Román, Filippo Bonchi, Giovanni de Felice
U. of Oxford, U. of Pisa, Quantinuum

MOTIVATION

Semantics of dataflow programs with effects.



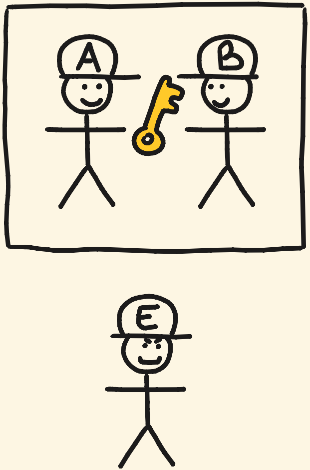
probability
non-determinism
shared memory
⋮

[EDL, Giovanni de Felice and Mario Román
Monoidal streams for dataflow programming (2022) LICS]

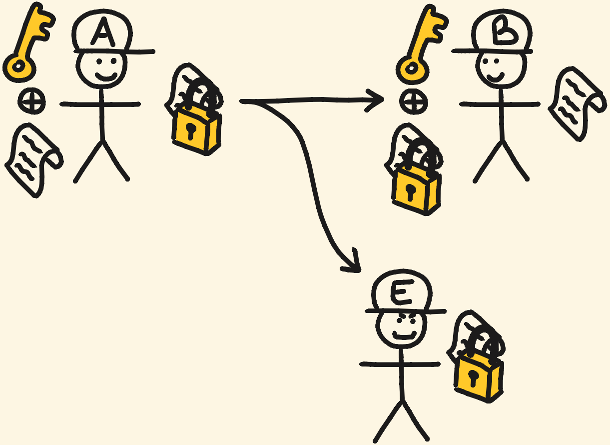
[Filippo Bonchi, EDL, Mario Román
Effectful Mealy machines: bisimulation and trace (2024) preprint]

ONE-TIME PAD PROTOCOL

1. share a key through a secure channel



2. send an encrypted message through a public channel



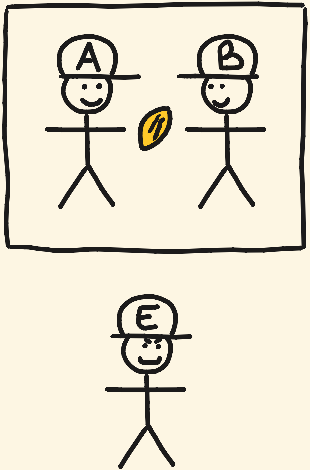
REPEATING THE ONE-TIME PAD

Sending n messages securely requires n private keys
↳ not very useful

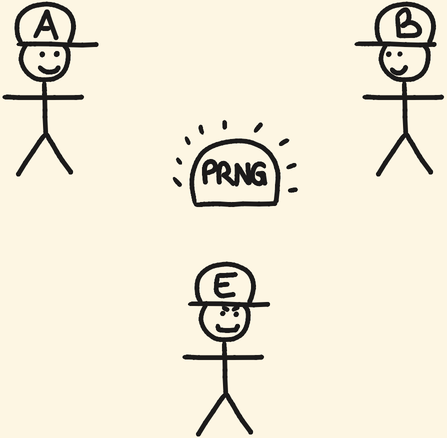
⇒ - privately share a seed
- use identical pseudorandom number generator to obtain a new key for each message

STREAM CIPHER PROTOCOL (1)

1. share a seed through a secure channel

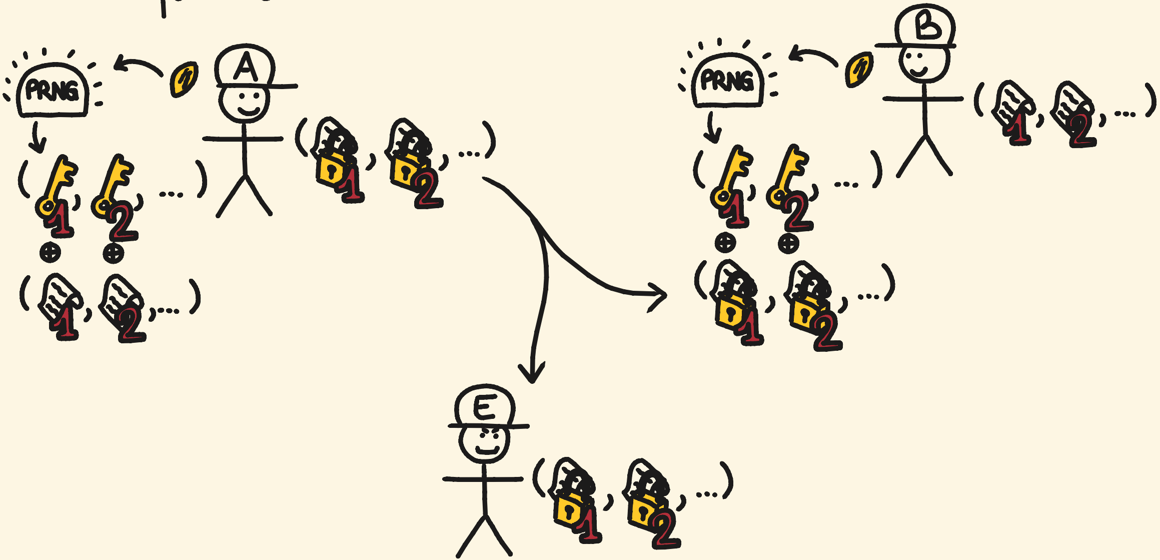


2. share a pseudorandom number generator



STREAM CIPHER PROTOCOL (2)

3. send a stream of encrypted messages through a public channel



STREAM CIPHER PROTOCOL

```
$\text{alice}^{\circ}(m)$ = do
    $\text{seedgen}() \rightsquigarrow ()$
    $\text{get}_A() \rightsquigarrow s$
    $\text{prng}(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$\text{alice}^{+\circ}(s, m)$ = do
    $\text{prng}(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$\text{alice}^{++} = \text{alice}^{+}$
```

```
$\text{bob}^{\circ}(m)$ = do
    $\text{get}_B() \rightsquigarrow s$
    $\text{prng}(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$\text{bob}^{+\circ}(s, m)$ = do
    $\text{prng}(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$\text{bob}^{++} = \text{bob}^{+}$
```

```
$\text{cipher}^{\circ}(m)$ = do
    $\text{alice}^{\circ}(m) \rightsquigarrow (s, m')$
    $\text{bob}^{\circ}(m') \rightsquigarrow (s', m'')$
    return $(s, s', m', m'')$

$\text{cipher}^{+\circ}(s_0, s_0', m)$ = do
    $\text{alice}^{+\circ}(s_0, m) \rightsquigarrow (s, m')$
    $\text{bob}^{+\circ}(s_0', m') \rightsquigarrow (s', m'')$
    return $(s, s', m', m'')$

$\text{cipher}^{++} = \text{cipher}^{+}$
```

COINDUCTION VS FEEDBACK

Streams are coinductive.

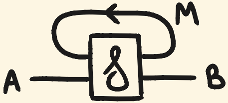
$$\mathrm{Stream}\ A = A \times \mathrm{Stream}\ A$$

Coalgebraic semantics.

$$\begin{tikzcd}
M \arrow[r] \arrow[d, dashed] & (M \times B)^A \arrow[d, dashed] \\
\Omega \arrow[r] & (\Omega \times B)^A
\end{tikzcd}$$

Feedback (pre)monoidal categories.

$$(M \mid f : M \otimes A \to M \otimes B)$$

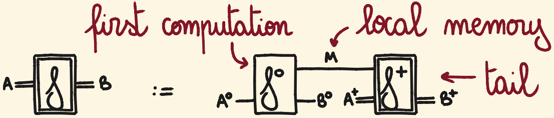


Sequential and parallel compositions.

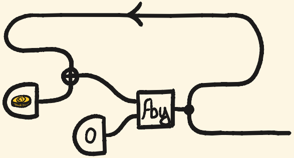
EFFECTFUL STREAMS: COINDUCTION AND FEEDBACK

Semantics of effectful reactive programs

- formal
- coinductive



- compositional



a random walk.

- effect agnostic

Stream : EffCat → EffCat

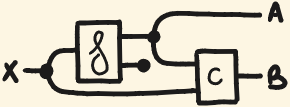
OUTLINE

- [Effectful categories]
- Effectful streams
- Causal processes
- Mealy machines, bisimulation and traces

STRING DIAGRAMS & DO-NOTATION

- Symmetric monoidal categories are theories of processes
- String diagrams and do-notation are convenient syntax

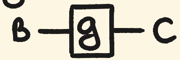
$$\nu_X ; ((f ; (\nu_A \otimes \varepsilon_B)) \otimes \mathbb{1}_X) ; (\mathbb{1}_A \otimes c)$$



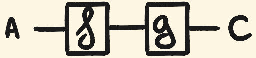
```
cond(x) = do
  f(x) → (a, b)
  c(a, x) → b'
  return (a, b')
```

STRING DIAGRAMS & DO-NOTATION

$f : A \longrightarrow B \,,\quad g : B \longrightarrow C$



- Composition. $f ; g : A \longrightarrow C$

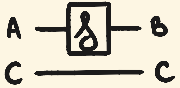


```
comp_fg(a) = do
  f(a) ~> b
  g(b) ~> c
  return(c)
```

- Identity. $id_A : A \longrightarrow A$

```
id_A(a) = do
  return(a)
```

- Whiskering. $w_C f : A \otimes C \longrightarrow B \otimes C$

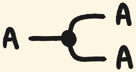


```
wh_fc(a,c) = do
  f(a) ~> b
  return(b,c)
```

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

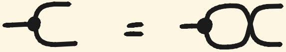
```
sym_AB(a,b) = do
  return(b,a)
```

COPY AND DISCARD

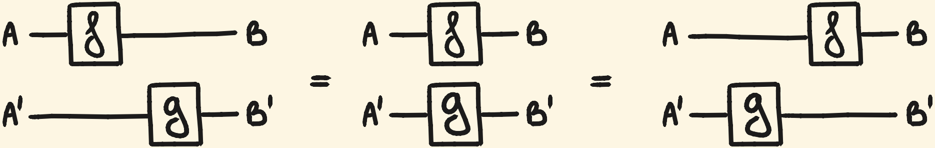


```
copy(a) = do
  return(a,a)
```

```
discard(a) = do
  return ()
```



THE INTERCHANGE LAW



```
par f g (a, a') = do
  f(a) → b
  g(a') → b'
  return (b, b')
```

=

```
par f g (a, a') = do
  g(a') → b'
  f(a) → b
  return (b, b')
```

⇝ holds in monoidal categories

COMPUTATIONS WITH EFFECTS

- Stochastic effects: generating the seed

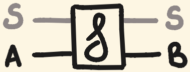
  $\mathcal{D} : \mathbf{Set} \to \mathbf{Set}$ distribution monad

  $\mathcal{D}(A) := \{ \sigma : A \to [0,1] \mid \operatorname{supp} \sigma \text{ is finite} \wedge \sum_{a \in A} \sigma(a) = 1 \}$

- Global state: sharing the seed

  $\mathrm{State}_S : \mathcal{C}^{op} \times \mathcal{C} \to \mathbf{Set}$ state promonad

  $\mathrm{State}_S(A, B) := \mathcal{C}(S \otimes A, S \otimes B)$



VALUES

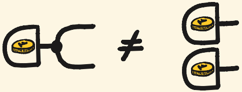
Values are both :

- deterministic

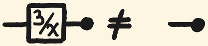
- total

ex $(3 \cdot -) : R \to \mathbb{R}$

NON-ex Flip : $1 \to \mathcal{D}(\{H,T\})$

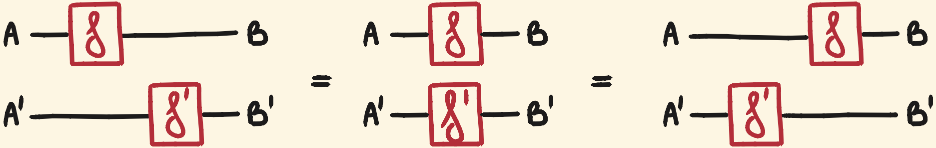


$(3/-) : R \to \mathbb{R}$



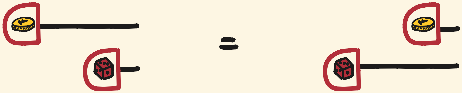
LOCAL COMPUTATIONS

Local computations interchange,



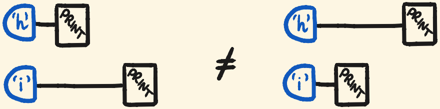
```
localF(a,a') = do                 localF(a,a') = do
  f(a) → b                            f'(a') → b'
  f'(a') → b'          =              f(a) → b
  return(b,b')                        return(b,b')
```

ex Stoch



EFFECTFUL COMPUTATIONS

Effectful computations may have global effects.



```
printHI() = do
  'h'() → c_1
  'i'() → c_2
  print(c_1) ⇝ ()
  print(c_2) ⇝ ()
  return ()
```

≠

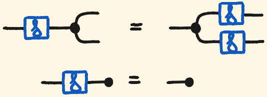
```
printIH() = do
  'h'() → c_1
  'i'() → c_2
  print(c_2) ⇝ ()
  print(c_1) ⇝ ()
  return ()
```

ex state promonads, IO monad

[Power and Robinson (1997)]

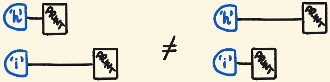
EFFECTFUL COPY-DISCARD CATEGORIES

Values can be copied and discarded (cartesian)

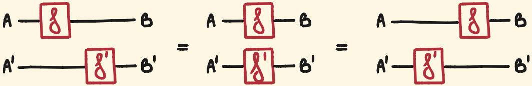


Effectful computations may have global effects (premonoidal)

$$\begin{tikzcd}\mathcal{V} \arrow[r] & \mathcal{L} \arrow[r] & \mathcal{C}\end{tikzcd}$$



Local computations interchange (monoidal)



ex (Set, Stoch, State_S)

(cart($\mathcal{C}$), $\mathcal{Z}$($\mathcal{C}$), $\mathcal{C}$) for a cd-premonoidal $\mathcal{C}$

[Jeffrey (1997), cf. Levy (2022), Power and Thielecke (1997)]

OUTLINE

- Effectful categories
- [Effectful streams]
- Causal processes
- Mealy machines, bisimulation and traces

STREAMS ARE COINDUCTIVE

A stream of elements of A is

- an element $a^0 \in A$
- a stream a^+ of elements of A

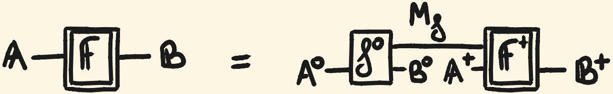
⇝ the set of streams is the final coalgebra of the functor

$$A \times (-) : \mathcal{S}et \to \mathcal{S}et$$

EFFECTFUL STREAMS

An *effectful stream* $F : A \to B$ on $(\mathcal{V}, \mathcal{L}, \mathcal{C})$ is

- a memory $M_f \in \mathcal{L}$
- a first action $f^\circ : A^\circ \to M_f \otimes B^\circ$ in $\mathcal{C}$
- the rest of the action $F^+ : M_f \cdot A^+ \to B^+$



quotiented by sliding

$$\begin{cases} f^\circ ; (r \otimes 1) = g^\circ \\ F^+ = r \cdot g^+ \end{cases}$$

for $r : M_f \to M_g$ in $\mathcal{L}$

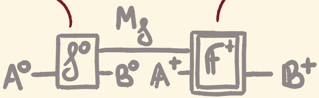
EFFECTFUL STREAMS

The profunctor Stream : $\mathcal{C}^{\mathbb{N}^{op}} \times \mathcal{C}^{\mathbb{N}} \longrightarrow \mathbf{Set}$ is the final coalgebra of the functor

$$F : [\mathcal{C}^{\mathbb{N}^{op}} \times \mathcal{C}^{\mathbb{N}}, \mathbf{Set}] \longrightarrow [\mathcal{C}^{\mathbb{N}^{op}} \times \mathcal{C}^{\mathbb{N}}, \mathbf{Set}]$$

$$F(Q)(A,B) := \int^{M \in \mathcal{C}} \mathcal{C}(A^{\circ}, M \otimes B^{\circ}) \times Q(M \cdot A^{+}, B^{+})$$

quotient by sliding on the memory



POLÝA URN

Consider (Set, Stoch, Stoch).

$\mathrm{pólya} : \mathbb{I} \to \mathbb{B}$

$\mapsto \mathbb{I}^{\circ} = I, \quad \mathbb{I}^{+} = \mathbb{I}$

$\mathbb{B}^{\circ} = \mathrm{Bool}, \quad \mathbb{B}^{+} = \mathbb{B}$

$$\mathrm{pólya}^{\circ}() = \mathrm{do}$$

$$\mathrm{Be}\left(\frac{b_0}{w_0 + b_0}\right) \to x$$

$$\mathrm{return}\,(\underbrace{w_0 + 1 - x,\ b_0 + x}_{\text{memory}:\ \mathbb{N} \times \mathbb{N}},\ x)$$

$\mapsto$ first action

$1 \to \mathbb{N} \times \mathbb{N} \times \mathrm{Bool}$

$$\mathrm{pólya}^{+\circ}(\overbrace{w, b}) = \mathrm{do}$$

$$\mathrm{Be}\left(\frac{b}{w + b}\right) \to x$$

$$\mathrm{return}\,(w + 1 - x,\ b + x,\ x)$$

$$\mathrm{pólya}^{++} = \mathrm{pólya}^{+}$$

$\mapsto$ tail, defined coinductively

$(\mathbb{N} \times \mathbb{N}) \cdot \mathbb{I} \to \mathbb{B}$

SEMANTICS OF STREAM CIPHER

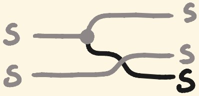
Consider (Set, Stoch, StateStoch),
where $\mathrm{StateStoch}(A,B) := \mathrm{Stoch}(S \otimes S \otimes A, S \otimes S \otimes B)$.

Fix $S = \mathbb{N}$, $C = \mathrm{Bool}^*$.

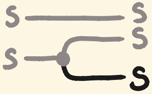
$\oplus : C \otimes C \to C$ ⇝ bitwise XOR

$\mathrm{prng} : S \to S \otimes C$ ⇝ pseudorandom number generator

$\mathrm{get}_A : I \rightsquigarrow S$



$\mathrm{get}_B : I \rightsquigarrow S$



$\mathrm{seedgen} : I \rightsquigarrow I$

STREAM CIPHER PROTOCOL

```
$alice^{\circ}(m)$ = do
    $seedgen() \rightsquigarrow ()$
    $get_A() \rightsquigarrow s$
    $prng(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$alice^{+\circ}(s, m)$ = do
    $prng(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$alice^{++} = alice^{+}$
```

```
$bob^{\circ}(m)$ = do
    $get_B() \rightsquigarrow s$
    $prng(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$bob^{+\circ}(s, m)$ = do
    $prng(s) \rightarrow (k, s')$
    return $(s', k \oplus m)$

$bob^{++} = bob^{+}$
```

```
$cipher^{\circ}(m)$ = do
    $alice^{\circ}(m) \rightsquigarrow (s, m')$
    $bob^{\circ}(m') \rightsquigarrow (s', m'')$
    return $(s, s', m', m'')$

$cipher^{+\circ}(s_0, s'_0, m)$ = do
    $alice^{+\circ}(s_0, m) \rightsquigarrow (s, m')$
    $bob^{+\circ}(s'_0, m') \rightsquigarrow (s', m'')$
    return $(s, s', m', m'')$

$cipher^{++} = cipher^{+}$
```

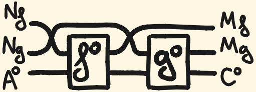
COMPOSITIONAL STRUCTURE OF STREAMS

THEOREM

Effectful streams form an effectful category Stream.

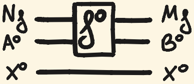
- composition and monoidal actions are defined coinductively: for $f : N_f \cdot A \to B$ and $g : N_g \cdot B \to C$,

$(f ;_N g)^\circ :=$



$(f ;_N g)^+ := f^+ ;_M g^+$

$(X \otimes_N f)^\circ :=$



$(X \otimes_N f)^+ := X^+ \otimes_M f^+$

FEEDBACK ON EFFECTFUL STREAMS

$\partial : \mathrm{Stream} \to \mathrm{Stream}$ $\qquad$ $\partial(A) := (I, A^{0}, A^{1}, \dots)$

THEOREM

Stream has ∂-feedback.

- feedback is defined coinductively

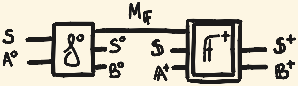
$$\mathbb{F} : (S \cdot \partial \mathbb{S}) \otimes A \to \mathbb{S} \otimes \mathbb{B}$$

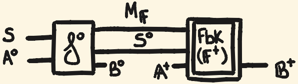
$$\mathrm{Fbk}_{\mathbb{S}}\,\mathbb{F} : S \cdot A \to \mathbb{B}$$

$$M(\mathrm{Fbk}^{S}_{\mathbb{S}}\,\mathbb{F}) := M(\mathbb{F}) \otimes \mathbb{S}^{\circ}$$

$$(\mathrm{Fbk}^{S}_{\mathbb{S}}\,\mathbb{F})^{\circ} := f^{\circ}$$

$$(\mathrm{Fbk}^{S}_{\mathbb{S}}\,\mathbb{F})^{+} := \mathrm{Fbk}^{S^{\circ}}_{\mathbb{S}^{+}}(\mathbb{F}^{+})$$





OUTLINE

- Effectful categories
- Effectful streams
- [Causal processes]
- Mealy machines, bisimulation and traces

STREAM COMPUTATIONS

- Sliding equivalence might be difficult to handle
- Causal stream functions are old :

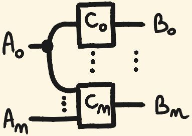
  [Raney (1958)] shows that they are
  the executions of deterministic Mealy machines

⇝ is there a similar explicit form for effectful streams ?

CAUSAL STREAM FUNCTIONS

Stream computations $(c_n)_{n\in\mathbb{N}} : A \to B$ in a cartesian category are families $c_n : A_0 \times \cdots \times A_n \longrightarrow B_n$.

⇝ $c_n(a_0, \dots, a_n) \in B_n$ is the output at time n



reconstructs the outputs until time n

[Sprunger & Katsumata (2019), Uustalu & Vene (2008)]

STOCHASTIC PROCESSES

Stochastic stream computations $(p_n)_{n\in\mathbb{N}} : \mathbb{A} \to \mathbb{B}$ are families $p_n : A_0 \times \dots \times A_n \longrightarrow \mathcal{D}(B_0 \times \dots \times B_n)$ such that $p_n(a_0,\dots,a_n) = \sum_{a\in A_{n+1}} p_{n+1}(a_0,\dots,a_n,a)$.

⇝ $p_n(a_0,\dots,a_n) \in \mathcal{D}(B_0 \times \dots \times B_n)$ is the distribution of the outputs <u>until</u> time n

↳ the outputs may be correlated

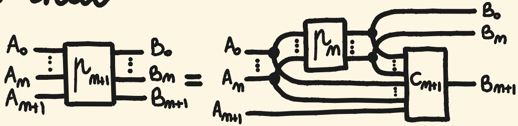
Is there a monoidal version of causal stream functions?

CAUSAL PROCESSES

A **causal process** $p : A \to B$ in a copy-discard category $\mathcal{C}$ is a family of morphisms

$$p_n : A_0 \otimes \cdots \otimes A_n \longrightarrow B_0 \otimes \cdots \otimes B_n$$

such that



for some $c_{n+1} : B_0 \otimes \cdots \otimes B_n \otimes A_0 \otimes \cdots \otimes A_n \otimes A_{n+1} \longrightarrow B_{n+1}$

⇝ p_n determines the input-output behaviour *until* time n

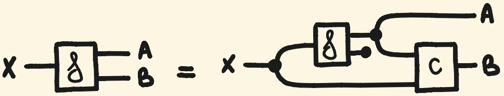
COMPOSING CAUSAL PROCESSES

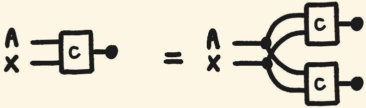
$\mathcal{C}$ copy-discard

CONDITIONALS WITH SHARP DOMAIN

[Cho & Jacobs (2017), Fritz (2020), EDL & Román (2023)]

For all $f : X \to A \otimes B$ there is $c : A \otimes X \to B$ st

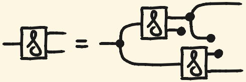
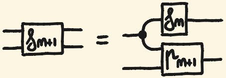
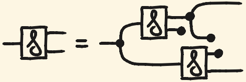
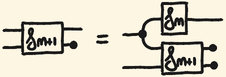
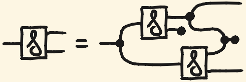
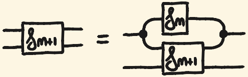
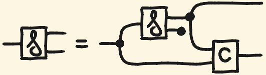
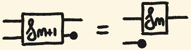
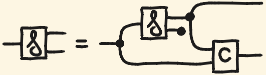




THEOREM

Causal processes form a monoidal category Proc when $\mathcal{C}$ has conditionals with sharp domain.

CAUSAL PROCESSES : EXAMPLES

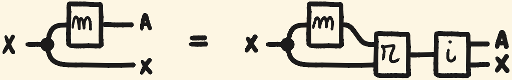
	CONDITIONALS	CAUSALITY CONDITION
Set		
Par		
Rel		
Stoch		
pStoch		

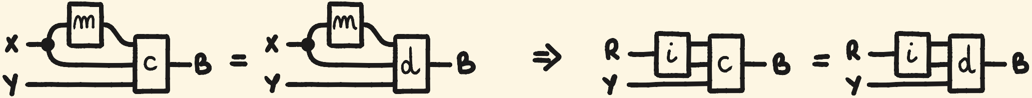
CAUSAL PROCESSES ARE STREAMS

$\mathcal{C}$ copy-discard

RANGES

For all $m : X \to A$ there are $\begin{cases} r : A \otimes X \to R & \text{deterministic} \\ i : R \to A \otimes X & \text{total} \end{cases}$





THEOREM

Consider ($\mathrm{fun}\,\mathcal{C}$, tot $\mathcal{C}$, $\mathcal{C}$).
If $\mathcal{C}$ has quasi-total conditionals and ranges,

$$\mathsf{Proc} \simeq \mathsf{Stream} .$$

OUTLINE

- Effectful categories
- Effectful streams
- Causal processes

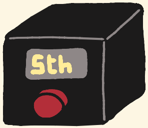
[• Mealy machines, bisimulation and traces]

MEALY MACHINES AND COALGEBRAS

Two faces of Mealy machines

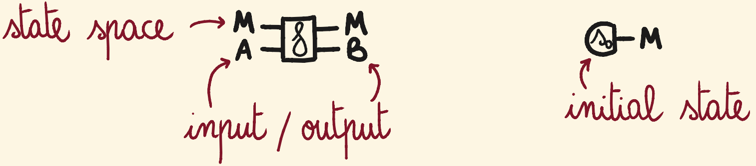
$$M \times A \to \mathcal{D}(M \times B) \qquad M \to \mathcal{D}(M \times B)^A$$

$$M \times A \to \mathcal{D}_c(M \times B) \qquad M \to \mathcal{D}_c(M \times B)^A$$



MEALY MACHINES

Systems are $\delta : M \otimes A \to M \otimes B$ with $s_0 : I \to M$



- native sequential and parallel compositions
- parametric in the underlying process theory
- premonoidal categories for global effects

⤳ what is their behaviour?
when are two of them equivalent?

[cf. Katis, Sabadini, Walters 1997]

COALGEBRAIC SEMANTICS

Systems are coalgebras $f : M \to F(M)$

input/output

$M \to (M \times B)^A$

non-determinism

$M \to \mathcal{P}(M \times B)$

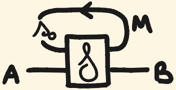
- bisimulation is equality in the final coalgebra

⇝ how do these compose?
how to change the underlying process theory?

EFFECTFUL TRACE SEMANTICS

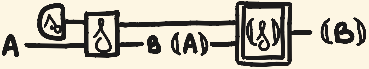
Fix a theory of programs.

effectful Mealy machines



↓

effectful streams



free construction
~ syntax

↓

coalgebraic construction
~ semantics

EFFECTFUL MEALY MACHINES

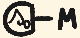
a **Mealy machine** $(f, M, s_0) : A \to B$ in $(\mathcal{V}, \mathcal{L}, \mathcal{C})$

is a morphism

$$f : M \otimes A \to M \otimes B$$

with an initial state

$$s_0 : I \to M$$



ex $(\mathbf{Set}, \mathbf{Rel}_{\mathrm{TOT}}, \mathbf{Rel})$

$$\begin{cases} f : M \times A \to \mathcal{P}(M \times B) \\ s_0 \subseteq M \end{cases}$$

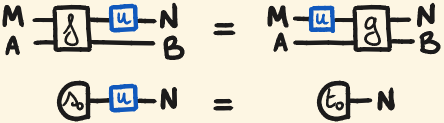
ex $(\mathbf{Set}, \mathbf{Stoch}, \mathbf{Stoch})$

$$\begin{cases} f : M \times A \to \mathcal{D}(M \times B) \\ s_0 \in \mathcal{D}(M) \end{cases}$$

[cf. Katis, Sabadini, Walters (1997); EDL, Gianola, Román, Sabadini, Sobociński (2022)]

MORPHISMS OF MEALY MACHINES

A <u>morphism</u> of Mealy machines $u : (f, M, s_0) \to (g, N, t_0)$ is a value morphism $u : M \to N$ in $\mathcal{V}$ such that



<u>ex</u> (Set, $\mathsf{Rel}_{\mathsf{TOT}}$, Rel)

$$\forall s \in M \ \forall a \in A \ \forall t \in N \ \forall b \in B$$

$$(t, b) \in g(u(s), a) \Leftrightarrow \exists s' \in M \ u(s') = t \wedge (s', b) \in f(s, a)$$

<u>ex</u> (Set, Stoch, Stoch)

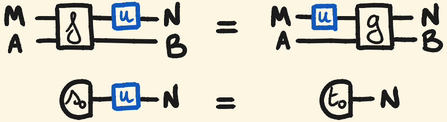
$$\forall s \in M \ \forall a \in A \ \forall t \in N \ \forall b \in B$$

$$g(t, b \mid u(s), a) = \sum_{s' : u(s') = t} f(s', b \mid s, a)$$

EFFECTFUL CATEGORY OF MEALY MACHINES

$\mathbf{Mealy}$ is an effectful category where

- objects are the objects of $\mathcal{C}$
- morphisms $(f, M, s_0) : A \to B$ are Mealy machines quotiented by value isomorphisms $u : M \xrightarrow{\cong} N$



- composition tensors the state spaces ⇝ local states

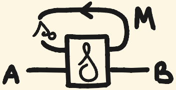
MEALY MACHINES ARE FREE

$\mathcal{S} := pt\mathcal{C}_{iso}$

THEOREM

Mealy is the free pointed-feedback category over $\mathcal{C}$.

$$\mathrm{Mealy}(A,B) = \int^{(s_0,M)\in pt\mathcal{C}_{iso}} \mathcal{C}(M\otimes A, M\otimes B)$$



[cf. Katis, Sabadini, Walters (1997); EDL, Gianola, Román, Sabadini, Sobociński (2022)]

COALGEBRAIC BISIMULATION

A bisimulation is a span of coalgebras.

$$\begin{tikzcd}
M \arrow[d, swap, "f"] & R \arrow[l, swap, "\pi_1"] \arrow[r, "\pi_2"] \arrow[d, "\alpha"] & N \arrow[d, "g"] \\
F(M) & F(R) \arrow[l, "F(\pi_1)"] \arrow[r, swap, "F(\pi_2)"] & F(N)
\end{tikzcd}$$

THEOREM [Rutten (2000)]

When $F : \mathbf{Set} \to \mathbf{Set}$ preserves weak pullbacks, bisimularity is an equivalence relation.

[Aczel & Mendler (1989), Rutten (2000)]

BISIMULATION

For two effectful Mealy machines $(f, M, s), (g, N, t) : A \to B$, a **bisimulation** is a sequence of spans of morphisms.

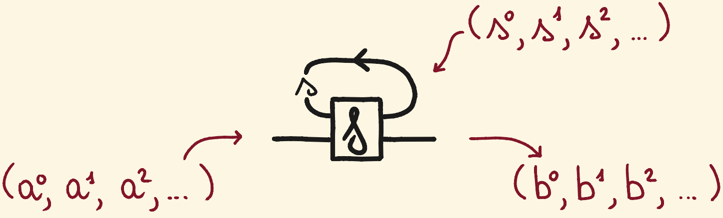
$$\begin{tikzcd}
 & (h_1, R_1, r_1) \arrow[dl, "u_1"'] \arrow[dr, "v_1"] & & \cdots \arrow[dl, "u_2"'] & (h_n, R_n, r_n) \arrow[dr, "v_n"] & \\
(f, M, s) & & (f_1, M_1, s_1) & & & (g, N, t)
\end{tikzcd}$$

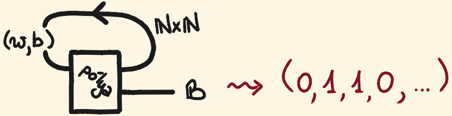
PROPOSITION

When $\mathcal{C} = Kl(T)$, for a commutative monad T preserving weak pullbacks, effectful bisimulation coincides with coalgebraic bisimulation.

ex Set, Par, Rel, Stoch, pStoch

EXECUTING MEALY MACHINES





⇝ what should the semantic universe be?

when do two Mealy machines have the same executions?

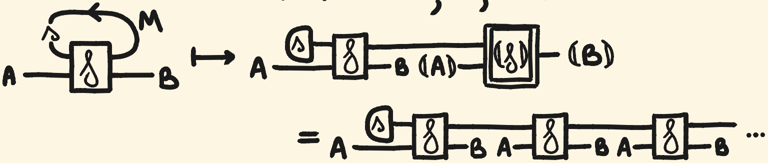
COMPOSITIONAL TRACE SEMANTICS

THEOREM

There is an effectful functor

$$Tr : Mealy \to Stream$$

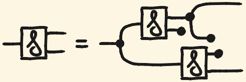
$$A \mapsto (\!|A|\!) = (A, A, \ldots)$$

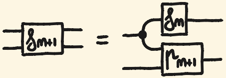


TRACES ARE EFFECTFUL TRACES

CONDITIONALS — CAUSALITY CONDITION — TRACE PREDICATE

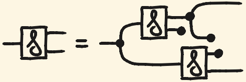
Set

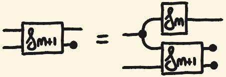




$s_0 = s \wedge \forall i \leq n \; (s_{i+1}, b_i) = f(s_i, a_i)$

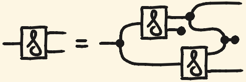
Par

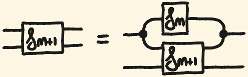




$s_0 = s \wedge \forall i \leq n \; (s_{i+1}, b_i) = f(s_i, a_i)$

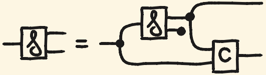
Rel

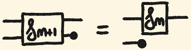




$\exists s_0, \dots, s_{n+1} \; s_0 \in s \wedge \forall i \leq n \; (s_{i+1}, b_i) \in f(s_i, a_i)$

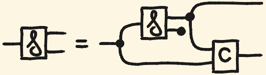
Stoch





$\sum_{s_0, \dots, s_{n+1}} s(s_0) \cdot \prod_{i \leq n} f(s_{i+1}, b_i \mid s_i, a_i)$

nStoch



$\sum_{s_0, \dots, s_{n+1}} s(s_0) \cdot \prod_{i \leq n} f(s_{i+1}, b_i \mid s_i, a_i)$

TRACE IS UNIVERSAL

THEOREM

The trace functor $\mathsf{Tr} : \mathsf{Mealy} \to \mathsf{Stream}$ is the unique feedback effectful functor determined by freeness of Mealy.

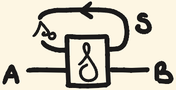
$$\begin{tikzcd}
\mathsf{FbkEffCat} \arrow[d, bend left] \\
\mathsf{EffCat} \arrow[u, bend left, "\mathsf{Mealy}"] \arrow[u, phantom, "\dashv"]
\end{tikzcd}
\quad \Rightarrow \quad
\exists!\ \mathsf{Tr} :
\begin{tikzcd}
\mathsf{Mealy}_{(\mathcal{V},\mathcal{L},\mathcal{C})} \arrow[rr] \arrow[dr] & {} \arrow[d, phantom, "\shortparallel"] & \mathsf{Stream}_{(\mathcal{V},\mathcal{L},\mathcal{C})} \\
& \mathsf{Mealy}_{(\mathcal{V},\mathcal{L},\mathcal{C})}/\mathsf{bisim} \arrow[ur] &
\end{tikzcd}$$

COROLLARY

Bisimilarity implies trace equivalence.

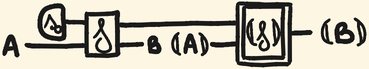
SUMMARY

effectful Mealy machines



↓ trace

effectful streams



≃ causal processes

free construction

~ syntax

↓ ∃!

coalgebraic construction

~ semantics

FUTURE WORK

- Adding choice, iteration and higher-order

- Coinduction up-to dinaturality
- Distributive law ?

- Linear temporal logic
- Behavioural metrics