Tallinn 19 March 2024

EFFECTFUL TRACE SEMANTICS VIA EFFECTFUL STREAMS

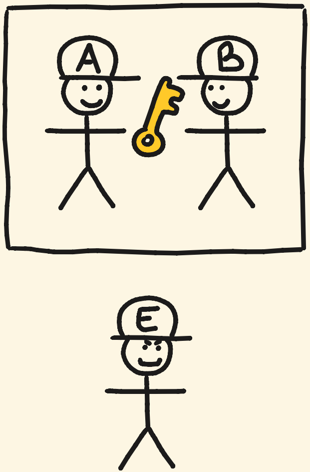
Filippo Bonchi
Università di Pisa

Elena Di Lavore
Università di Pisa

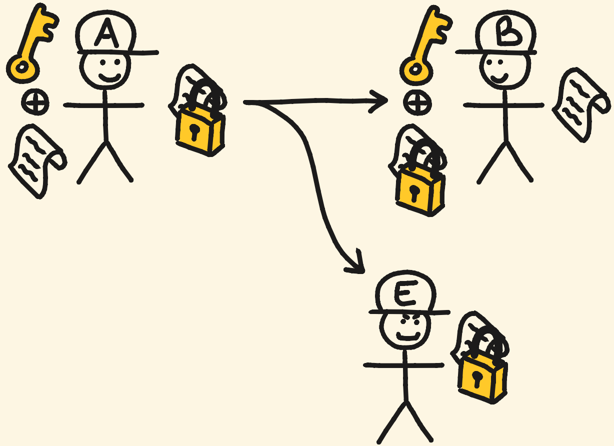
Mario Román
University of Oxford

ONE-TIME PAD PROTOCOL

1. share a key through a secure channel



2. send an encrypted message through a public channel

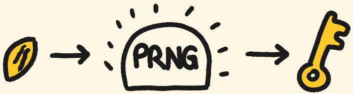


[Broadbent & Karvonen 2023]

REPEATING THE ONE-TIME PAD

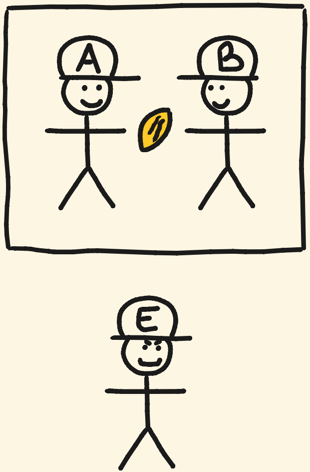
Sending n messages securely requires n private keys
↳ not very useful

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

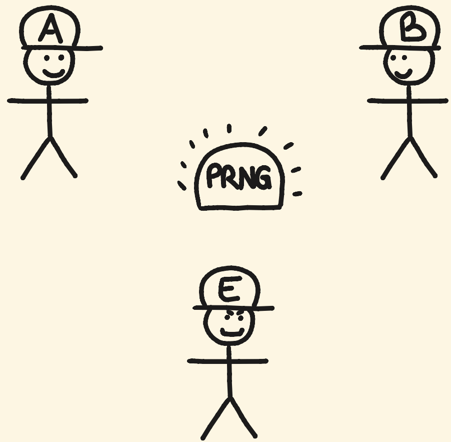


STREAM CIPHER PROTOCOL (1)

1. share a seed through a secure channel

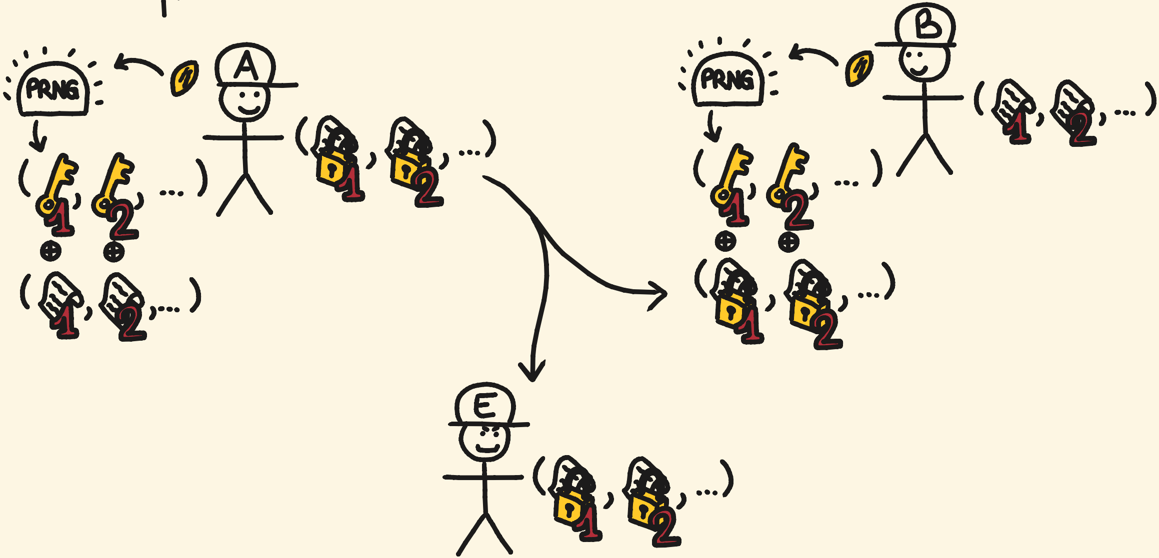


2. share a pseudorandom number generator



STREAM CIPHER PROTOCOL (2)

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



STREAM CIPHER PROTOCOL COINDUCTIVELY

$\mathrm{seedGen}()^{\circ} = \mathrm{do}$
- $\mathrm{rand}() \to s$
- $\mathrm{setSeed}(s) \rightsquigarrow ()$
- $\mathrm{return}\ ()$

$\mathrm{seedGen}()^{+\circ} = \mathrm{do}$
- $\mathrm{return}\ ()$

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

$\mathrm{alice}(m)^{\circ} = \mathrm{do}$
- $\mathrm{getSeedA}() \rightsquigarrow (s)$
- $\mathrm{prng}(s) \to (s', k)$
- $\mathrm{return}\ (s', m \oplus k)$

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

$\mathrm{alice}(s, m)^{++} = \mathrm{alice}(s, m)^{+}$

STREAM CIPHER PROTOCOL COINDUCTIVELY

$$
\begin{aligned}
&\mathrm{bob}(m)^{\circ} = \mathrm{do} \\
&\quad \mathrm{getSeedB}() \rightsquigarrow (s) \\
&\quad \mathrm{prng}(s) \to (s', k) \\
&\quad \mathrm{return}\ (s', m \oplus k)
\end{aligned}
$$

$$
\begin{aligned}
&\mathrm{bob}(s, m)^{+\circ} = \mathrm{do} \\
&\quad \mathrm{prng}(s) \to (s', k) \\
&\quad \mathrm{return}\ (s', m \oplus k)
\end{aligned}
$$

$$\mathrm{bob}(s, m)^{++} = \mathrm{bob}(s, m)^{+}$$

$$
\begin{aligned}
&\mathrm{eve}(m)^{\circ} = \mathrm{do} \\
&\quad \mathrm{return}\ (m)
\end{aligned}
$$

$$\mathrm{eve}(m)^{+} = \mathrm{eve}(m)$$

OUTLINE

- effectful categories
- effectful streams
- effectful trace semantics
- causal processes

COMPUTATIONS WITH EFFECTS

- Stochastic effects: the distribution monad

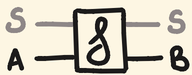
  $\mathcal{D} : \mathbf{Set} \to \mathbf{Set}$

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

- Global state: the state promonad

  $\mathcal{S}t : \mathcal{C}^{op} \times \mathcal{C} \to \mathbf{Set}$

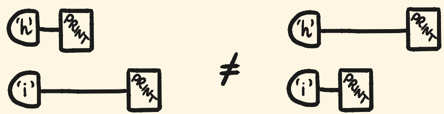
  $\mathcal{S}t(A,B) := \mathcal{C}(S \otimes A, S \otimes B)$



PREMONOIDAL CATEGORIES

Some effects do not interchange.

```
printHI() = do                    printIH() = do
  print('h') ~> ()                  print('i') ~> ()
  print('i') ~> ()        ≠         print('h') ~> ()
  return ()                         return ()
```



prints "hi"   prints "ih"

ex state promonads, IO monad

STREAM CIPHER PROTOCOL (AGAIN)

$\text{seedGen}()^{\circ} = \text{do}$

- $\text{rand}() \rightarrow s$
- $\text{setSeed}(s) \rightsquigarrow ()$
- $\text{return}\ ()$

$\text{seedGen}()^{+\circ} = \text{do}$

- $\text{return}\ ()$

$\text{seedGen}^{++} = \text{seedGen}^{+}$

$\text{alice}(m)^{\circ} = \text{do}$

- $\text{getSeedA}() \rightsquigarrow (s)$
- $\text{prng}(s) \rightarrow (s', k)$
- $\text{return}\ (s', m \oplus k)$

$\text{alice}(s, m)^{+\circ} = \text{do}$

- $\text{prng}(s) \rightarrow (s', k)$
- $\text{return}\ (s', m \oplus k)$

$\text{alice}(s, m)^{++} = \text{alice}(s, m)^{+}$

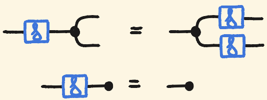
$\text{eve}(m)^{\circ} = \text{do}$

- $\text{return}\ (m)$

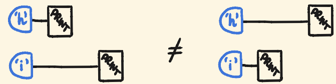
$\text{eve}(m)^{+} = \text{eve}(m)$

EFFECTFUL COPY-DISCARD CATEGORIES

Values can be copied and discarded (cartesian)

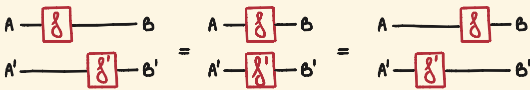


Effectful computations may have global effects (premonoidal)



$$\mathcal{V} \hookrightarrow \mathcal{L} \hookrightarrow \mathcal{C}$$

Local computations interchange (monoidal)



ex (Set, Stoch, State)

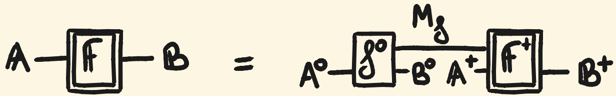
OUTLINE

- effectful categories
- effectful streams
- effectful trace semantics
- causal processes

EFFECTFUL STREAMS

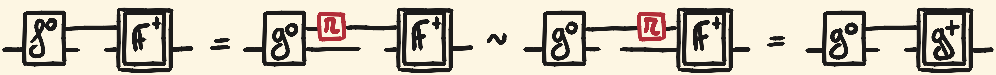
An *effectful stream* $F : A \to B$ on $(\mathcal{V}, \mathcal{L}, \mathcal{C})$ is

- 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 the equivalence relation generated by

$$\begin{cases} f^\circ ; (r \otimes 1) = g^\circ \\ F^+ = r \cdot g^+ \end{cases} \qquad \text{for } r : M_F \to M_G \text{ in } \mathcal{L}$$



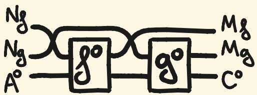
COMPOSITIONAL STRUCTURE OF STREAMS

THEOREM

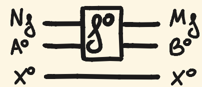
Effectful streams form an effectful category Stream.

- composition and monoidal actions are defined coinductively: for $\mathbb{F} : N_{\mathbb{F}} \cdot \mathbb{A} \to \mathbb{B}$ and $\mathbb{G} : N_{\mathbb{G}} \cdot \mathbb{B} \to \mathbb{C}$,

$$\begin{cases} (\mathbb{F} ;_N \mathbb{G})^\circ := \\ (\mathbb{F} ;_N \mathbb{G})^+ := \mathbb{F}^+ ;_M \mathbb{G}^+ \end{cases}$$



$$\begin{cases} (\mathbb{X} \otimes_N \mathbb{F})^\circ := \\ (\mathbb{X} \otimes_N \mathbb{F})^+ := \mathbb{X}^+ \otimes_M \mathbb{F}^+ \end{cases}$$



SEMANTICS FOR THE STREAM CIPHER PROTOCOL

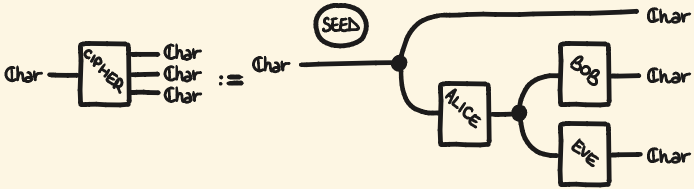
Fix two finite sets Char, Seed
and take the effectful copy-discard category

$$(\mathcal{S}et, \mathcal{S}toch, \mathcal{S}eed\mathcal{S}toch)$$

$\mathcal{S}eed\mathcal{S}toch$ is the Kleisli category of the promonad
that adds the global state $Seed \times Seed$:

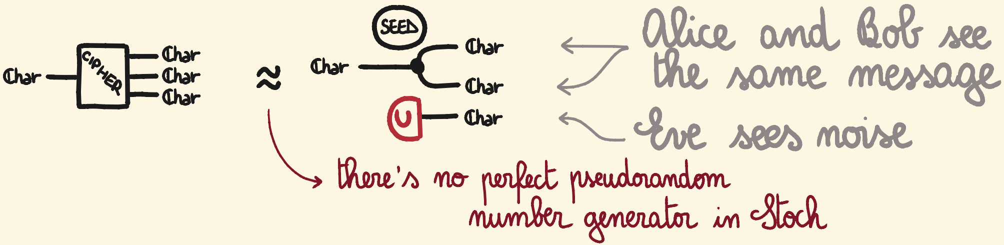
$$\mathcal{S}eed\mathcal{S}toch(A, B) := \mathcal{S}toch(Seed \times Seed \times A, Seed \times Seed \times B)$$

SEMANTICS FOR THE STREAM CIPHER PROTOCOL



THEOREM

The stream cipher protocol is secure.



[cf. Broadbent & Karvonen 2023]

OUTLINE

- effectful categories
- effectful streams
- [effectful trace semantics]
- causal processes

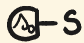
EFFECTFUL MEALY MACHINES

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

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

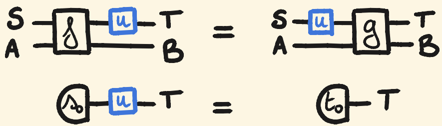
with an initial state

$$s_0 : I \to S$$



A **morphism** of Mealy machines $u : (f, S, s_0) \to (g, T, t_0)$
is a value morphism $u : S \to T$ in $\mathcal{V}$

such that

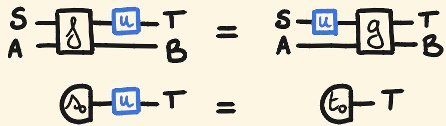


[cf. Katis, Sabadini, Walters 1997 ; EDL, Gianola, Román, Sabadini, Sobociński 2022]

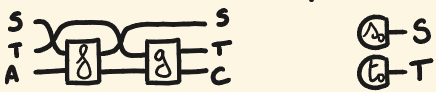
EFFECTFUL CATEGORY OF MEALY MACHINES

Mealy is an effectful category where

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



- composition tensors the state spaces



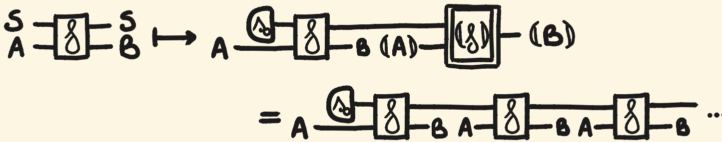
COMPOSITIONAL TRACE SEMANTICS

THEOREM

There is an effectful functor

$$\mathrm{Tr} : \mathsf{Mealy} \to \mathsf{Stream}$$

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



$\rightsquigarrow$ in Rel these traces coincide with the classical traces

BISIMULATION

Two effectful Mealy machines $(f, S, s), (g, T, t) : A \to B$ are <u>bisimilar</u> if they belong to the same connected component in $\mathrm{Mealy}(A, B)$:

$$
\begin{tikzcd}[column sep=small]
 & (h_1, R_1, r_1) \arrow[dl, "u_1"'] \arrow[dr, "v_1"] & & \arrow[dl, "u_2"'] \cdots & (h_n, R_n, r_n) \arrow[dr, "v_n"] & \\
(f, S, s) & & (f_1, S_1, s_1) & & & (g, T, t)
\end{tikzcd}
$$

THEOREM

For Mealy machines in $(\mathcal{V}, \mathcal{L}, \mathcal{C})$, bisimulation implies trace equivalence.

PROOF: By coinduction. $\square$

COALGEBRAIC BISIMULATION

PROPOSITION

When $\mathcal{C} = \mathrm{Kl}(M)$, for a commutative monad preserving weak pullbacks, then (f, S, s) and (g, T, t) are bisimilar iff they have the same bisimulation quotient, i.e. there is (h, Q, q) with morphisms

$$(f, S, s) \xrightarrow{u} (h, Q, q) \xleftarrow{v} (g, T, t) .$$

EXAMPLES

- Set
- Rel
- pStoch

OUTLINE

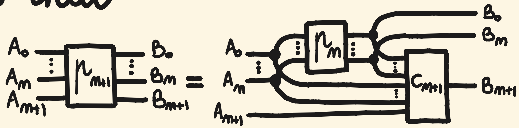
- effectful categories
- effectful streams
- effectful trace semantics
- [causal processes]

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

THEOREM

causal processes form a monoidal category Proc when $\mathcal{C}$ has quasi-total conditionals.

[cf. Raney 1958 ; Sprunger & Katsumata 2019]

CAUSAL PROCESSES ARE STREAMS

THEOREM

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

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

EXAMPLES

- Set
- Rel
- pStoch
- Par
- Stoch

TRACES ARE EFFECTFUL TRACES

Compute the traces of a Mealy machine

$$(f, S, s) : A \to B$$

in some known cases.

$(b_0, \dots, b_n)$ is a trace of $(a_0, \dots, a_n)$

Set if $s_0 = s$ and $\forall k \leq n \ (s_{k+1}, b_k) = f(s_k, a_k)$

Rel if $\exists (s_0, \dots, s_{n+1}) \ \ s_0 \in s$
and $\forall k \leq n \ (s_{k+1}, b_k) \in f(s_k, a_k)$

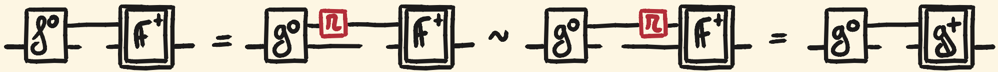
pStoch with probability $\sum_{(s_0, \dots, s_{n+1})} s(s_0 | *) \cdot \prod_{k \leq n} f(s_{k+1}, b_k | s_k, a_k)$

SUMMARY

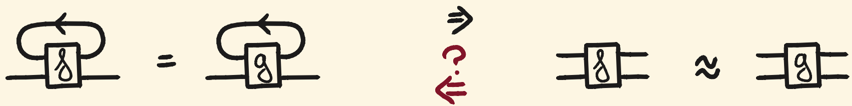
- formal compositional semantics for effectful stream computations
- trace equivalence and bisimulation of effectful Mealy machines
- characterisation as causal stream processes

FUTURE WORK

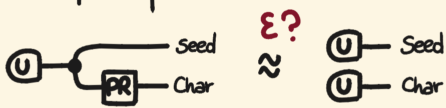
- coinduction up-to dinaturality



- Rel with explicit failure
- equality in St$\mathcal{C}$ implies bisimulation



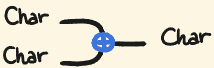
- distance instead of equivalence relation for security



SEMANTICS FOR THE STREAM CIPHER PROTOCOL

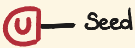
Semantics for values and local computations.

$[\![- \oplus -]\!] := \mathrm{XOR} : \mathrm{Char} \times \mathrm{Char} \to \mathrm{Char}$ ⇝ bitwise XOR



in Set

$[\![\mathrm{rand}]\!] := \mathrm{unif} : 1 \to \mathcal{D}(\mathrm{Seed})$ ⇝ uniform distribution



in Stoch

$[\![\mathrm{prng}]\!] : \mathrm{Seed} \to \mathrm{Seed} \times \mathcal{D}(\mathrm{Char})$ ⇝ use the seed to generate a key

in Stoch

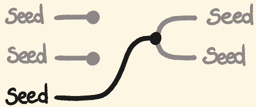
SEMANTICS FOR THE STREAM CIPHER PROTOCOL

Semantics for effectful computations.

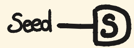
$[\![\mathrm{setSeed}]\!] : \mathrm{Seed}^3 \to \mathrm{Seed}^2$ $\rightsquigarrow$ copy the seed to the global state

$\mathrm{Seed} \rightsquigarrow 1$

$[\![\mathrm{setSeed}]\!] :=$



$=$

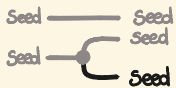


in SeedStoch

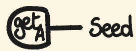
$[\![\mathrm{getSeedA}]\!], [\![\mathrm{getSeedB}]\!] : \mathrm{Seed}^2 \to \mathrm{Seed}^3$ $\rightsquigarrow$ alice and bob get their seeds

$1 \rightsquigarrow \mathrm{Seed}$

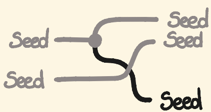
$[\![\mathrm{getSeedA}]\!] :=$



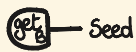
$=$



$[\![\mathrm{getSeedB}]\!] :=$



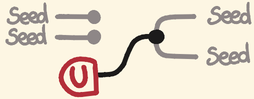
$=$



SEMANTICS FOR THE STREAM CIPHER PROTOCOL

- seedGen = (SEED) : $\mathbb{I} \to \mathbb{I}$ in Stream

seedGen° :=



=

seedGen$^{+\circ}$:=

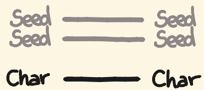
=

seedGen^{++} = seedGen^{+}

- eve =

: Char $\to$ Char in Stream

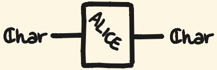
eve° :=

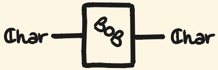


=

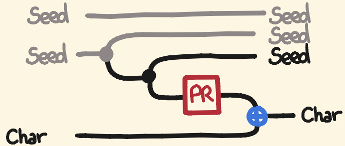
eve^{+} = eve

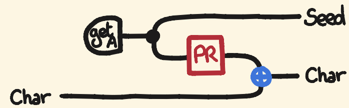
SEMANTICS FOR THE STREAM CIPHER PROTOCOL

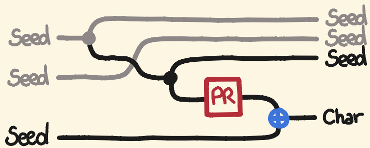
- alice = 

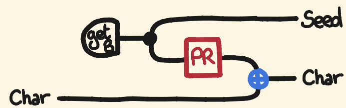
: Char → Char in Stream
- bob = 

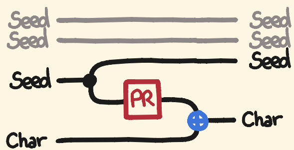
: Char → Char in Stream

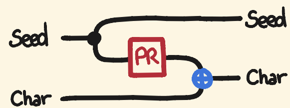
alice° := 

= 

bob° := 

= 

$\text{alice}^{+\circ} = \text{bob}^{+\circ}$:= 

= 

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

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

STREAM CIPHER IS SECURE

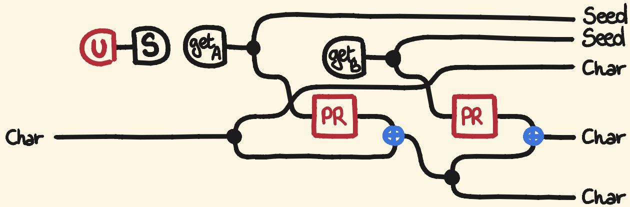
Proceed by coinduction.

cipher$^\circ$

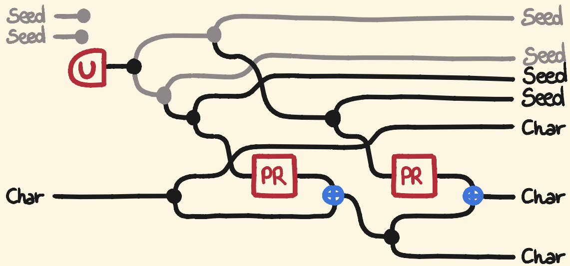
$$
\text{cipher}(m)^\circ = \textbf{do} \\
= \left|\begin{array}{l}
\text{rand}() \to s \\
\text{setSeed}(s) \rightsquigarrow () \\
\text{getSeedA}() \rightsquigarrow s_A \\
\text{prng}(s_A) \to (s'_A, k_A) \\
\text{getSeedB}() \rightsquigarrow s_B \\
\text{prng}(s_B) \to (s'_B, k_B) \\
\textbf{return}\ (m, s'_A, m \oplus k_A \oplus k_B, s'_B, m \oplus k_A)
\end{array}\right.
$$

STREAM CIPHER IS SECURE

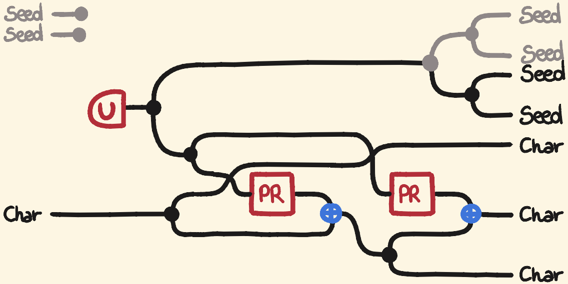
=



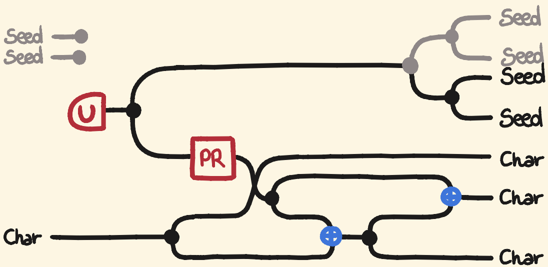
=



STREAM CIPHER IS SECURE

= 

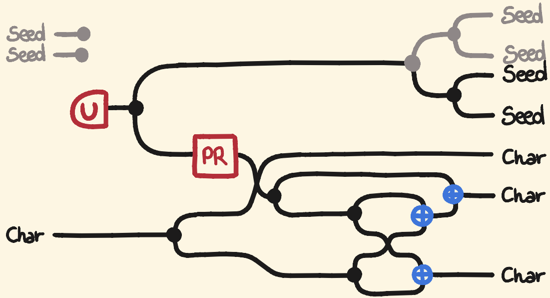
by associativity of copy

= 

pseudorandom is deterministic

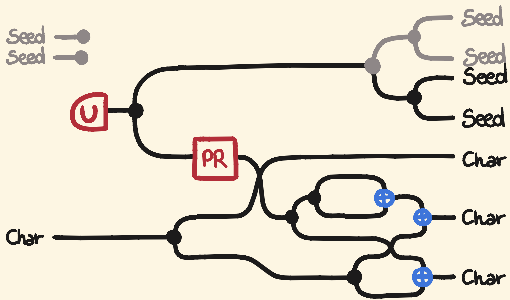
STREAM CIPHER IS SECURE

=



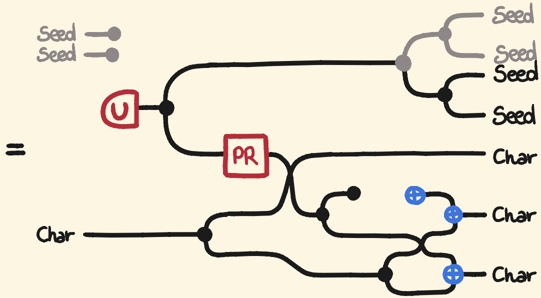
xor is deterministic

=

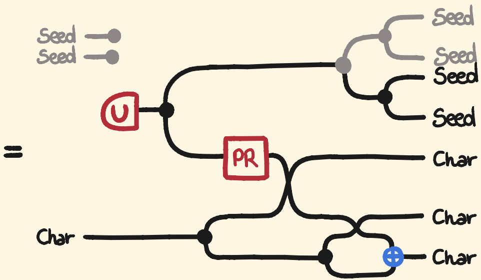


by associativity of copy and xor

STREAM CIPHER IS SECURE



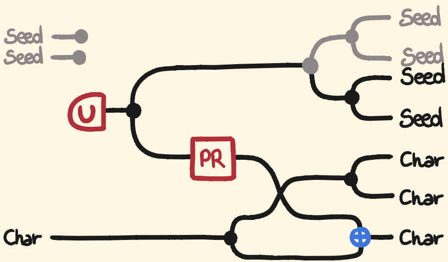
xor is nihilpotent



by unitality of copy and xor

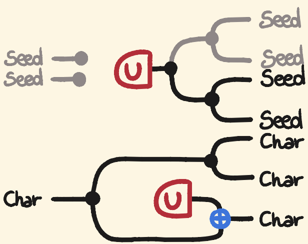
STREAM CIPHER IS SECURE

=



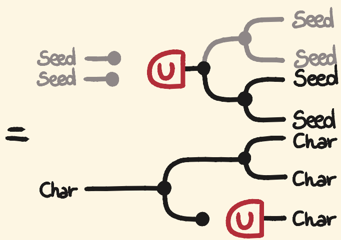
by associativity of copy

≈

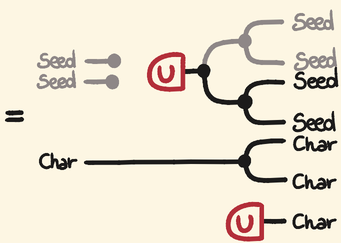


by assumption

STREAM CIPHER IS SECURE



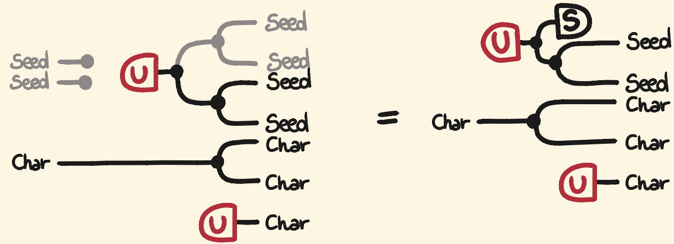
the uniform distribution is a Sweedler integral for XOR



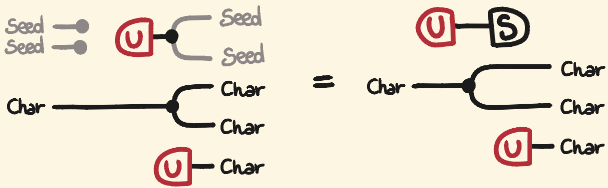
by unitality of copy

STREAM CIPHER IS SECURE

cipher° ≈



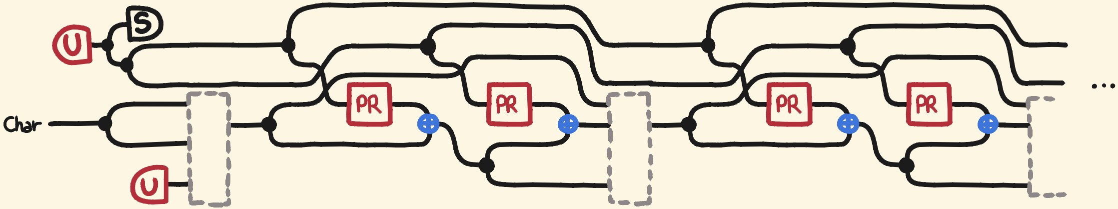
secure° :=



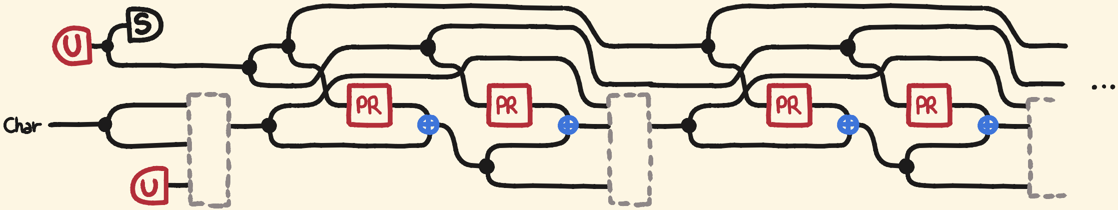
STREAM CIPHER IS SECURE

cipher

=

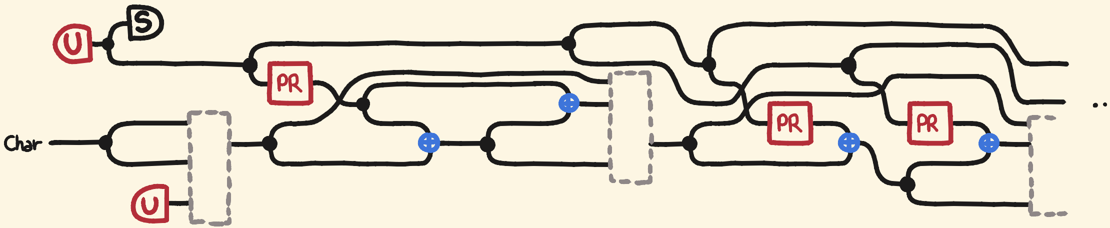


= (by sliding)

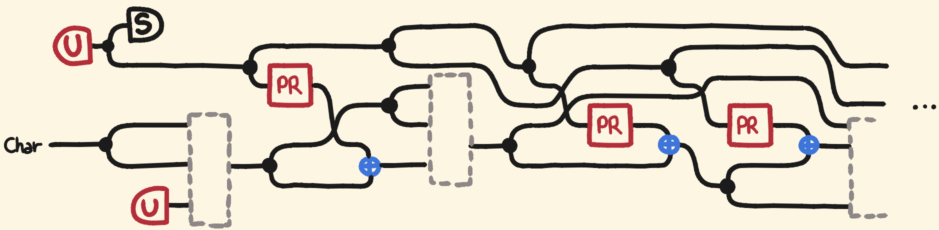


STREAM CIPHER IS SECURE

= (pseudorandom is deterministic)

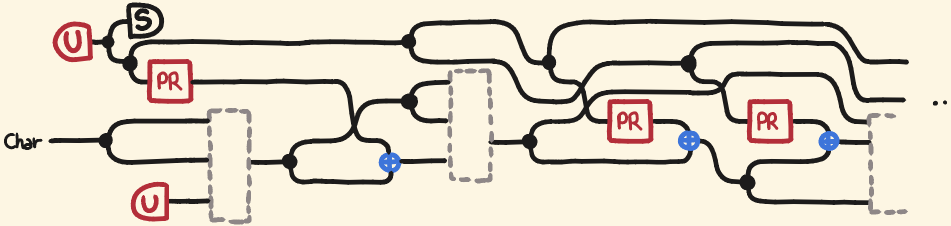


= (XOR is deterministic and nihilpotent)

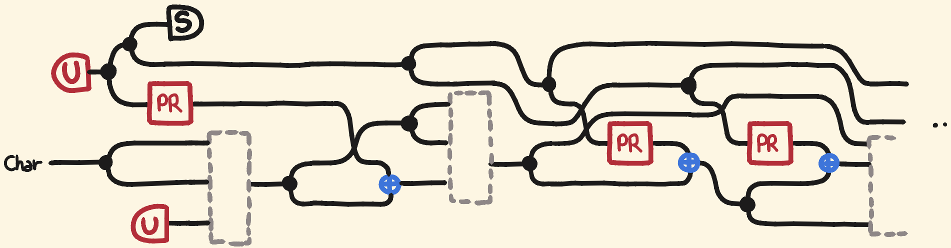


STREAM CIPHER IS SECURE

= (by sliding)

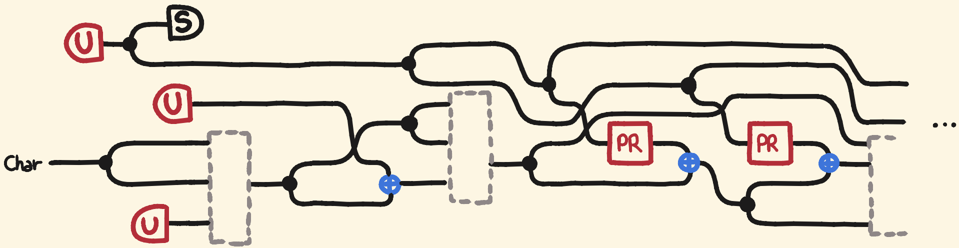


= (by associativity)

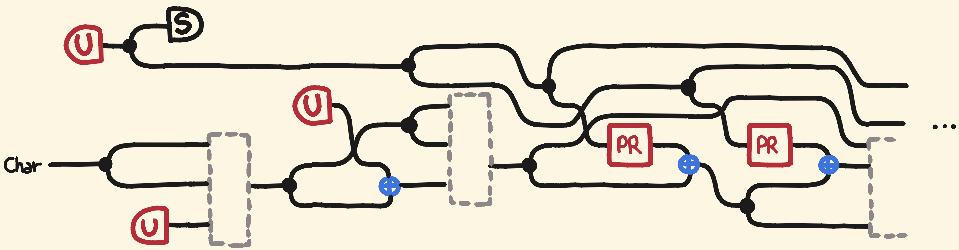


STREAM CIPHER IS SECURE

≈ (by assumption on pseudorandom)

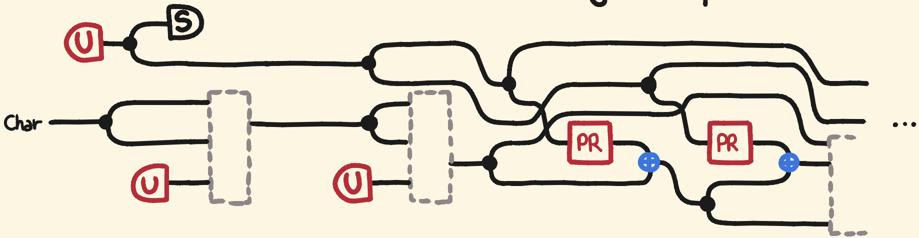


= (by sliding)

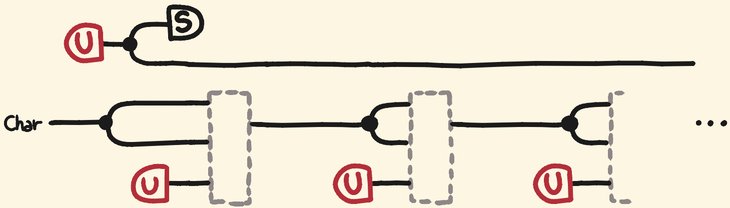


STREAM CIPHER IS SECURE

= (unif is a Sweedler integral for XOR)

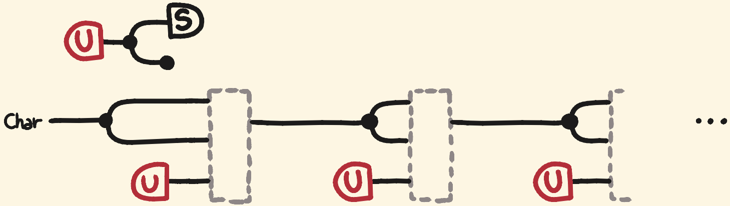


≈ (by coinduction)

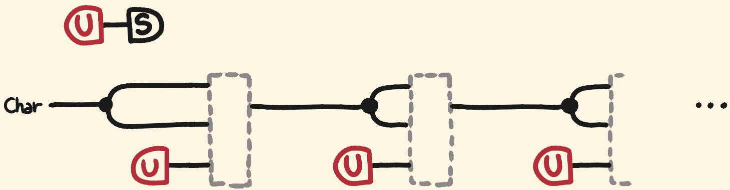


STREAM CIPHER IS SECURE

= (by coinduction)



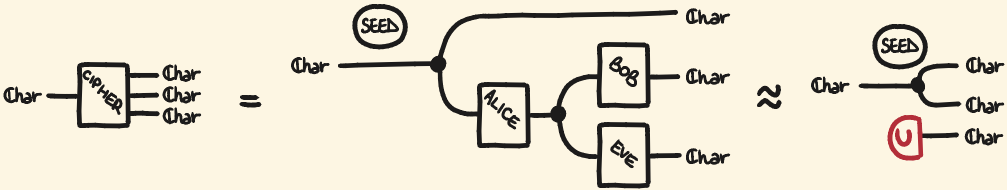
= (by unitality)



= secure

STREAM CIPHER IS SECURE

We have shown



using sliding and coinduction.