Coalgebraic Tools for Randomness-Conserving Protocols

Matvey Soloviev (Cornell University)

RAMiCS 2018, Groningen

joint work with Dexter Kozen

This Talk

- A coalgebraic model for constructing and reasoning about state-based protocols that implement efficient reductions among random processes (efficient = conserve randomness)
- Basic tools that allow efficient protocols to be constructed compositionally
- Tradeoffs between latency and efficiency
- Several examples of efficient reductions
- Toward a general coalgebraic semantics of reductions

Randomness as a Computational Resource

Randomness is a resource to be conserved

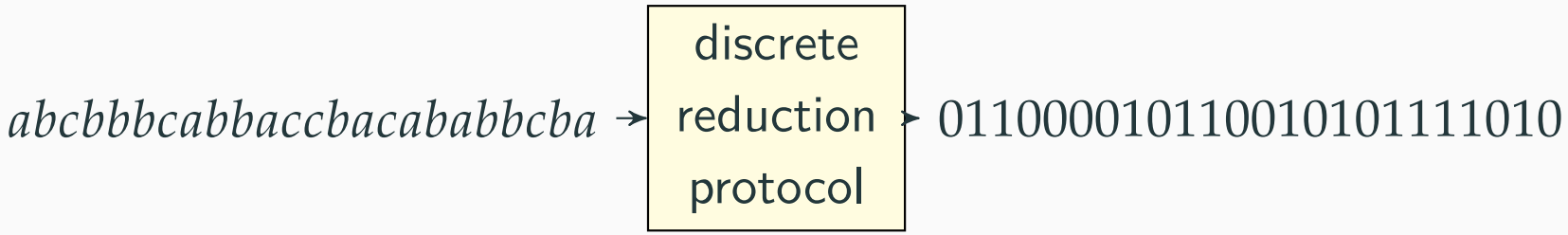
- Information and coding [Shannon]
- Probabilistic complexity and derandomization [Luby]
- Pseudo-random number generation [Yao, Nisan, Wigderson]
- Extracting strong randomness from weak sources [von Neumann, Elias, Blum]

A recent application: Routing in networks

- Randomized routing, gossip protocols, load balancing
- Desirable to minimize local state to achieve high throughput

Measuring Randomness

Discrete reduction protocol
A procedure that maps an input stream to an output stream



- If the input sequence comes from a random process, then the statistical properties of the input stream impart statistical properties to the output stream
- We can think of the process as a reduction between random sources
- But randomness can be lost ...

Shannon Entropy

Entropy of a discrete distribution $\mu = p_1, \ldots, p_n$

$$H(\mu) = -\sum_i p_i \log p_i$$

- Usually described as a measure of uncertainty or information content
- Represents an absolute limit on lossless compression (Shannon source coding theorem, 1948)

Shannon Entropy

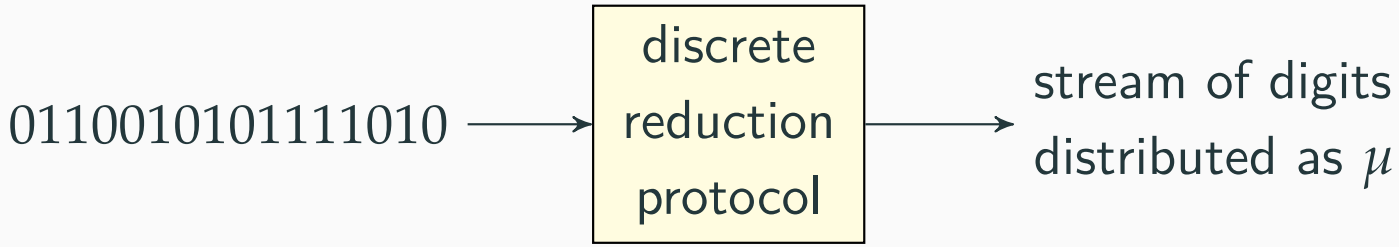
Entropy of a discrete distribution $\mu = p_1, \ldots, p_n$

$$H(\mu) = -\sum_i p_i \log p_i$$

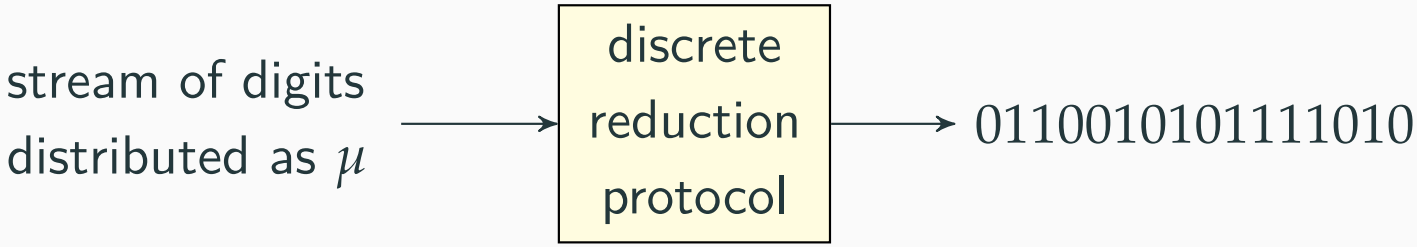
- Usually described as a measure of uncertainty or information content
- Represents an absolute limit on lossless compression (Shannon source coding theorem, 1948)
- $H(\mu) =$ the number of fair coin flips μ is worth

Entropy as a Measure of Randomness

The entropy of μ is the number of fair coin flips it is worth

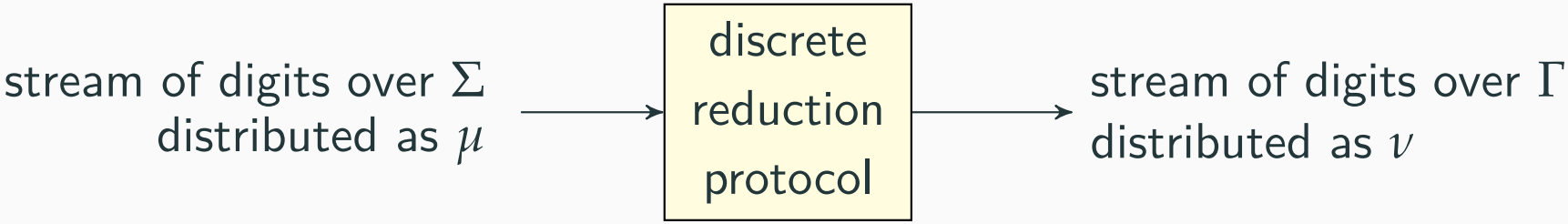


- $1/H(\mu)$ is an upper bound on the rate of production
- achievable asymptotically (requires unbounded latency)



- $H(\mu)$ is an upper bound on the rate of production
- achievable asymptotically (requires unbounded latency)

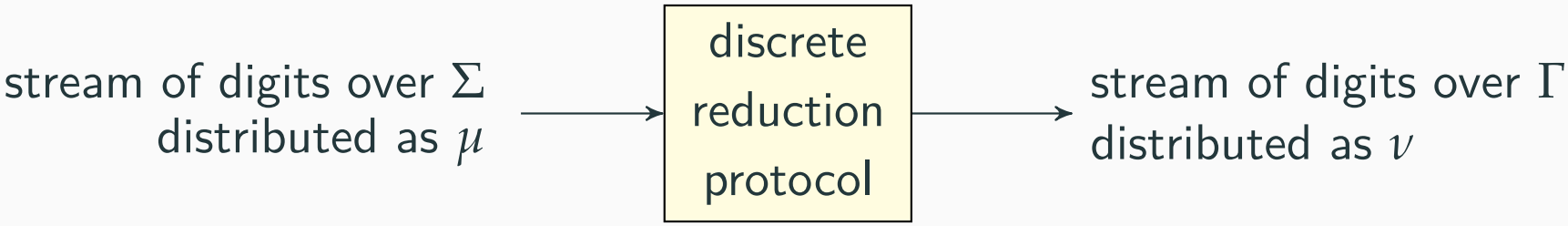
Efficiency of a Simulation



$$\text{Efficiency} = \frac{E_{\text{prod}} \cdot H(\nu)}{E_{\text{cons}} \cdot H(\mu)} \leq 1$$

- E_{cons} = expected number of digits consumed
- E_{prod} = expected number of digits produced
- $H(\mu)$ = entropy of input distribution
- $H(\nu)$ = entropy of output distribution

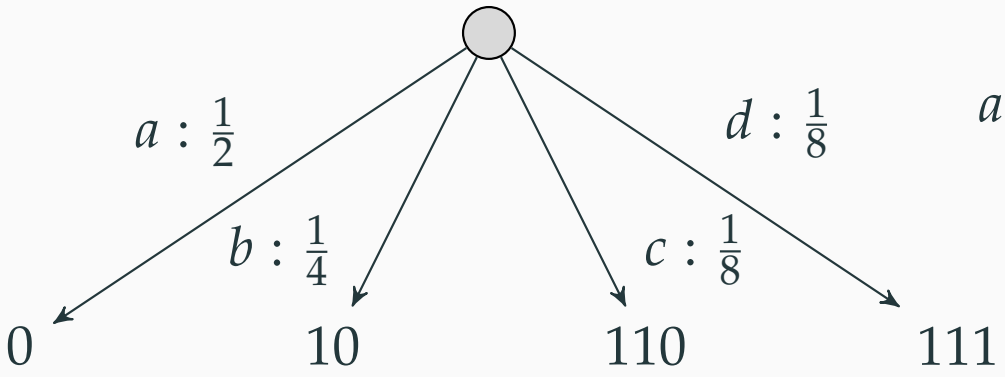
Efficiency of a Simulation

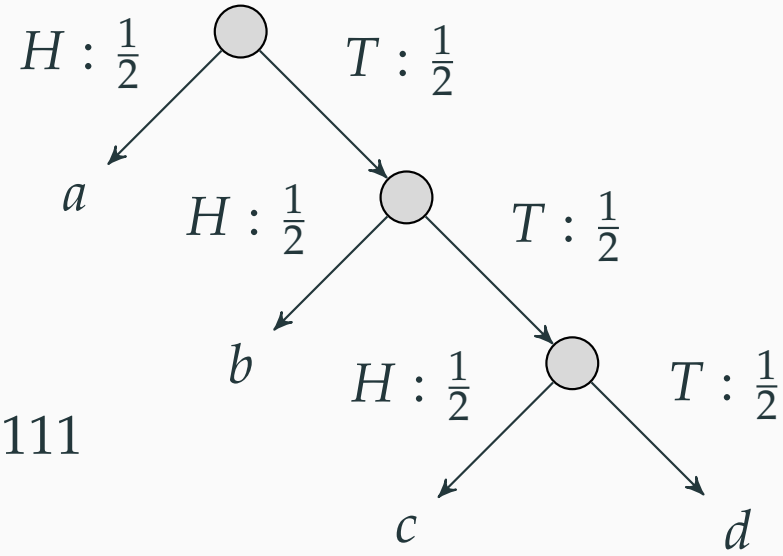


$$\text{Efficiency} = \frac{E_{\text{prod}} \cdot H(\nu)}{E_{\text{cons}} \cdot H(\mu)} \leq 1$$

- Measures the amount of randomness lost in the conversion
- May vary with time
- Cannot exceed unity [Shannon]
- Unity is achievable asymptotically [Elias, Cover & Thomas]; requires unbounded latency

Sometimes Perfect Efficiency is Achievable

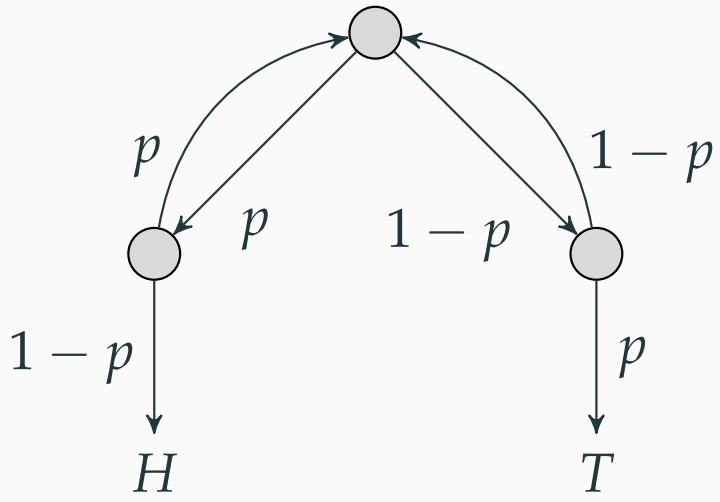




$$H(\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}) = 7/4$$

$$H(\frac{1}{2}, \frac{1}{2}) = 1$$

The von Neumann Trick [1951]



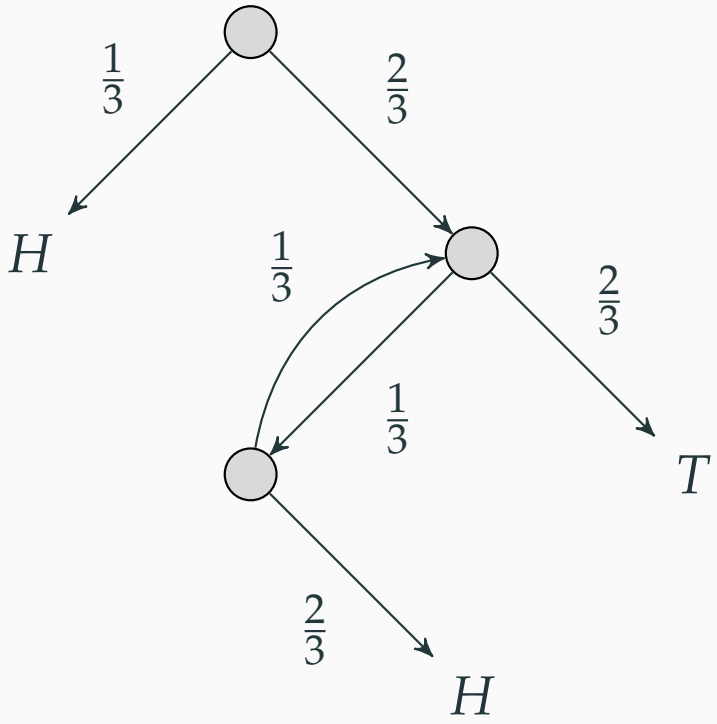
To simulate a fair coin with a bias-p coin:

- flip the bias-p coin twice
- $01 \Rightarrow H$
- $10 \Rightarrow T$
- 00 or $11 \Rightarrow$ flip again

Oblivious to the bias of the input coin, but efficiency is poor:

- for $p = 1/3$, $E_{\text{cons}}/E_{\text{prod}} = 4.5$
- Shannon says
  $1/(-(1/3)\log(1/3) - (2/3)\log(2/3)) \approx 1.083\cdots$

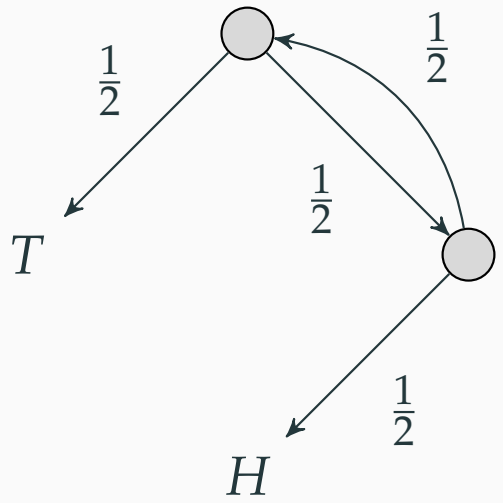
A More Efficient Protocol



Not oblivious to the bias $p = 1/3$, but efficiency is better:

- $E_{\text{cons}}/E_{\text{prod}} = 2$
- This is optimal for single-digit-output protocols

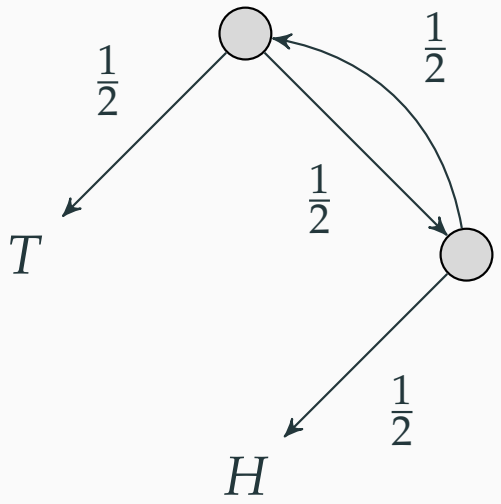
Other Direction ($\frac{1}{2}, \frac{1}{2} \Rightarrow \frac{1}{3}, \frac{2}{3}$)



$$\Pr(H) = \frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \cdots = \frac{1}{3}$$
$$\Pr(T) = \frac{1}{2} + \frac{1}{8} + \frac{1}{32} + \cdots = \frac{2}{3}$$

Q: Is this optimal?

Other Direction ($\frac{1}{2}, \frac{1}{2} \Rightarrow \frac{1}{3}, \frac{2}{3}$)

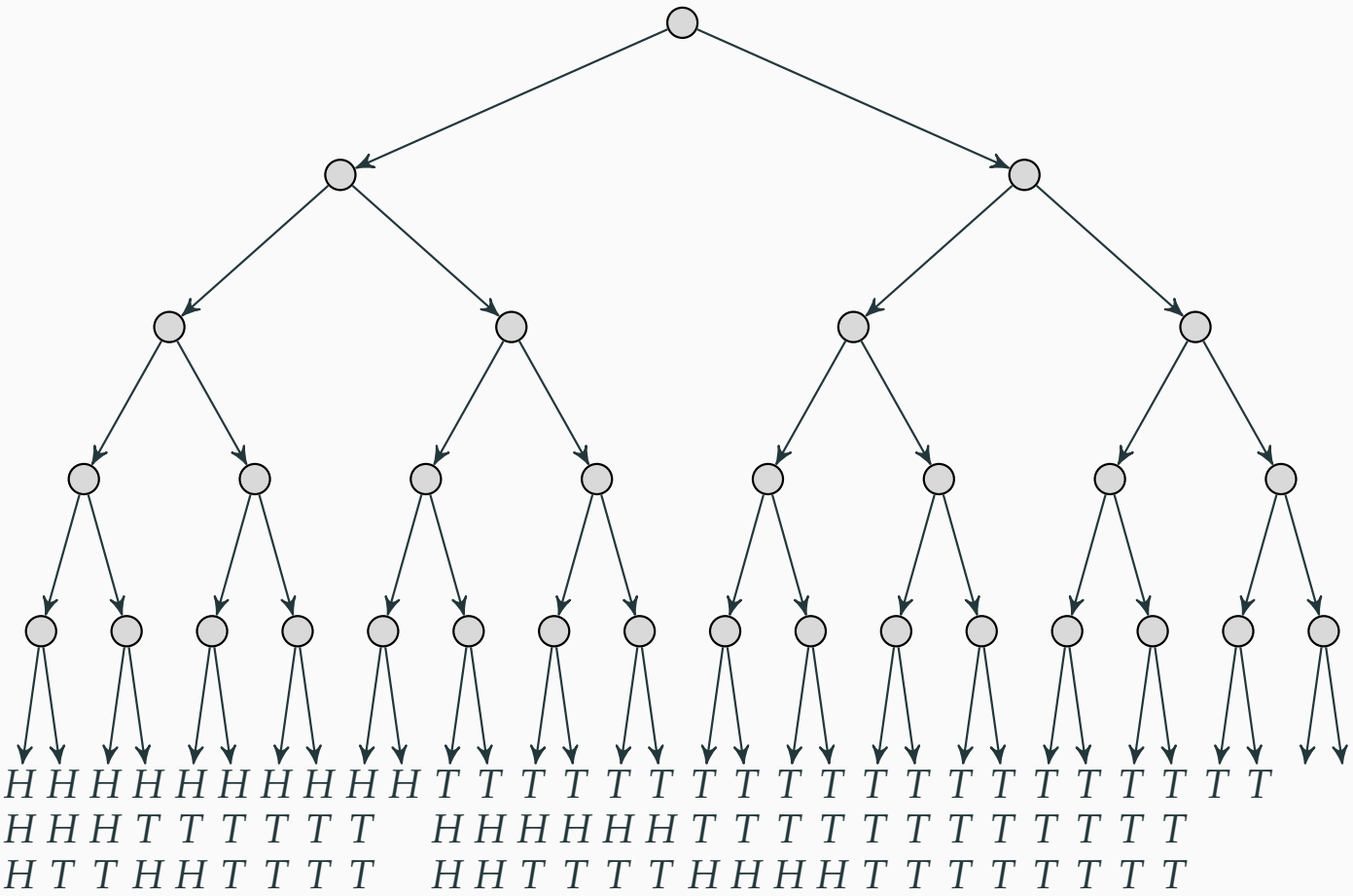


$$\Pr(H) = \tfrac{1}{4} + \tfrac{1}{16} + \tfrac{1}{64} + \cdots = \tfrac{1}{3}$$
$$\Pr(T) = \tfrac{1}{2} + \tfrac{1}{8} + \tfrac{1}{32} + \cdots = \tfrac{2}{3}$$

Q: Is this optimal?

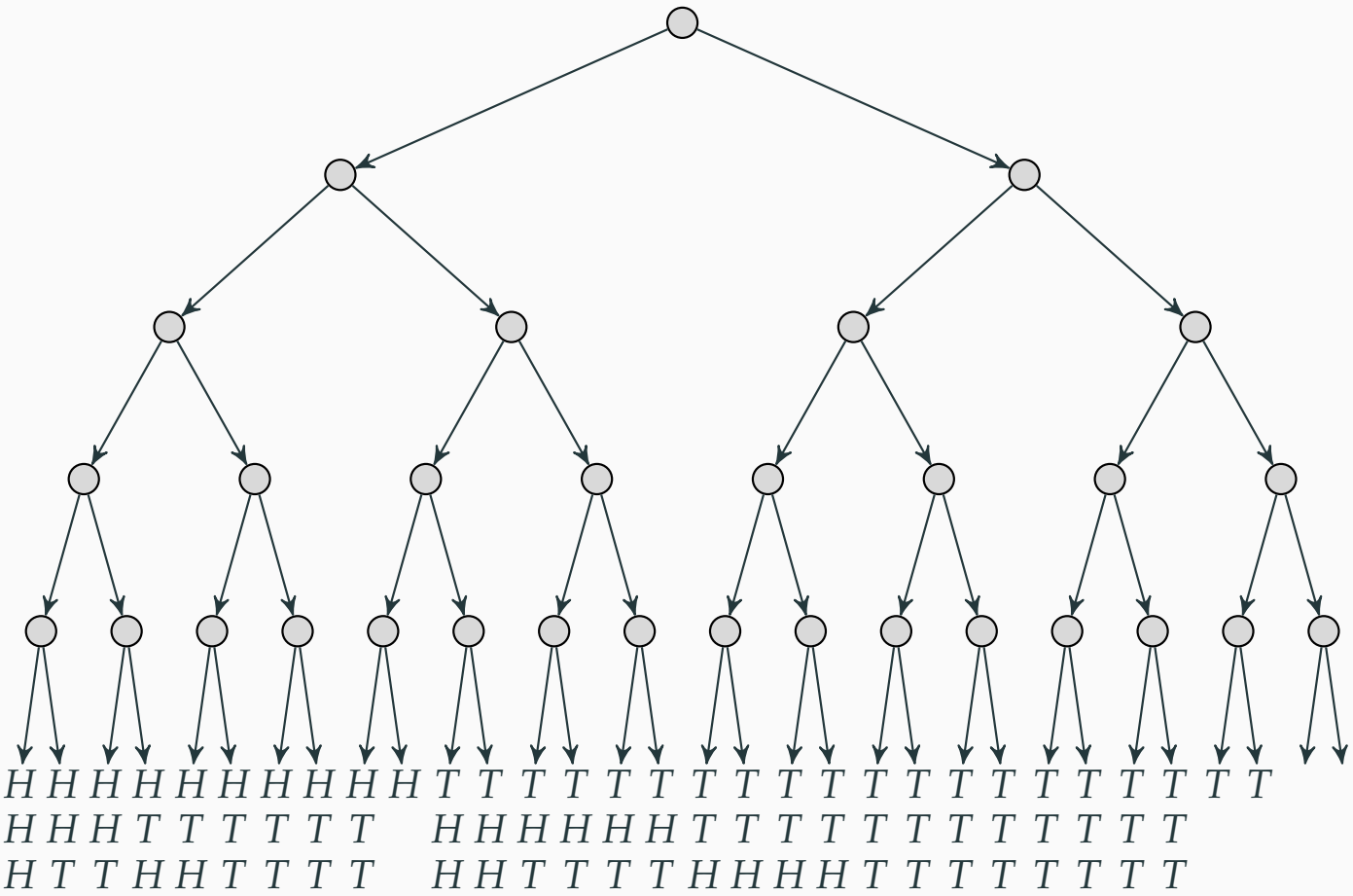
A: No! $E_{\text{cons}} / E_{\text{prod}} = 2$, but $-\tfrac{1}{3}\log\tfrac{1}{3} - \tfrac{2}{3}\log\tfrac{2}{3} \approx .92\cdots$

How to Do Better?



Q: Is this optimal?

How to Do Better?



Q: Is this optimal?

A: No, but better! $E_{\text{cons}}/E_{\text{prod}} = 5/2.625 = 1.905$

Latency

This protocol has many more states, and we'll have to read in at least 4 symbols before we output anything. Define:

$$\text{Latency} = \text{expected consumption before producing at least one output symbol.}$$

Generally, higher latency = longer input = higher-grained probability (sub)space = more leeway to carve it up into "correctly sized" chunks for better efficiency.

This tradeoff is inevitable whenever no perfect protocol exists.

Asymptotic optimality is not everything

It's known that asymptotically optimal families of reductions exist.

Now we can say that some are better than others: it matters whether efficiency $1 - \varepsilon$ would require latency $O(1/\varepsilon)$, $O(1/\varepsilon^2)$ or even worse.

Notation

- Σ, Γ finite alphabets
- Σ^* = finite words over Σ $\quad x, y, \ldots \in \Sigma^*, \Gamma^*$
- Σ^ω = ω-words (streams) over Σ $\quad \alpha, \beta, \ldots \in \Sigma^\omega, \Gamma^\omega$
- $\preceq$ prefix, $\prec$ proper prefix
- μ is a probability measure on Σ, endow Σ^ω with the product measure – each symbol independent and distributed as μ
- The measurable sets of Σ^ω are the Borel sets of the Cantor space topology whose basic open sets are the intervals $\{\alpha \in \Sigma^\omega \mid x \prec \alpha\}$ for $x \in \Sigma^*$
- $\mu(a_1 a_2 \cdots a_n) = \mu(a_1)\mu(a_2)\cdots\mu(a_n)$
- $\mu(\{\alpha \in \Sigma^\omega \mid x \prec \alpha\}) = \mu(x)$

Protocols

A protocol is a coalgebra (S, δ) where $\delta : S \times \Sigma \to S \times \Gamma^*$ (a form of Mealy automaton)

Extend δ to domain $S \times \Sigma^*$ by coinduction:

$$\delta(s, \varepsilon) = (s, \varepsilon)$$
$$\delta(s, ax) = \text{let } (t, y) = \delta(s, a) \text{ in let } (u, z) = \delta(t, x) \text{ in } (u, yz)$$

It follows that

$$\delta(s, xy) = \text{let } (t, z) = \delta(s, x) \text{ in let } (u, w) = \delta(t, y) \text{ in } (u, zw)$$

Extension to Streams

A protocol δ also induces a partial map $\delta^\omega : S \times \Sigma^\omega \rightharpoonup \Gamma^\omega$ by coinduction:

$$\delta^\omega(s, a\alpha) = \text{let } (t, z) = \delta(s, a) \text{ in } z \cdot \delta^\omega(t, \alpha)$$

It follows that

$$\delta^\omega(s, x\alpha) = \text{let } (t, z) = \delta(s, x) \text{ in } z \cdot \delta^\omega(t, \alpha)$$

Given $\alpha \in \Sigma^\omega$, this defines a unique infinite string in $\delta^\omega(s, \alpha) \in \Gamma^\omega$ except in the degenerate case in which only finitely many output letters are ever produced

A protocol is productive (wrt a given probability measure on input streams) if, starting in any state, an output symbol is produced within finite expected time (therefore w.p. 1)

Reductions

Let ν be a probability measure on Γ, $\nu(a_1 \cdots a_n) = \nu(a_1) \cdots \nu(a_n)$

(S, δ, s) with start state $s \in S$ is a reduction from μ to ν if

$$\forall y \in \Gamma^* \quad \mu(\{\alpha \mid y \preceq \delta^\omega(s, \alpha)\}) = \nu(y)$$

This implies that the symbols of $\delta^\omega(s, \alpha)$ are independent and distrinbuted as ν

Advantages of the Coalgebraic View

- Many constructions in the information theory literature are expressed in terms of trees – but
- Protocols are coalgebras $\delta : S \times \Sigma \to S \times \Gamma^*$, a form of Mealy automata, i.e. not trees
- This class admits a final coalgebra $D : (\Gamma^*)^{\Sigma^+} \times \Sigma \to (\Gamma^*)^{\Sigma^+} \times \Gamma^*$, where

$$D(f, a) = (f@a, f(a)) \quad f@a(x) = f(ax),\ a \in \Sigma,\ x \in \Sigma^+$$

- Extension to streams $D^\omega : (\Gamma^*)^{\Sigma^+} \times \Sigma^\omega \rightharpoonup \Gamma^\omega$

$$D^\omega(f, a\alpha) = f(a) \cdot D^\omega(f@a, \alpha)$$

Advantages of the Coalgebraic View

- A state $f : \Sigma^+ \to \Gamma^*$ can be viewed as a labeled tree with nodes Σ^* and edge labels Γ^*
- The nodes xa are the children of x for $x \in \Sigma^*$ and $a \in \Sigma$
- The label on the edge (x, xa) is $f(xa)$
- The tree $f@x$ is the subtree rooted at $x \in \Sigma^*$, where $f@x(y) = f(xy)$
- For any coalgebra (S, δ), there is a unique coalgebra morphism $h : (S, \delta) \to ((\Gamma^*)^{\Sigma^+}, D)$ defined coinductively by

$$(h(s)@a, h(s)(a)) = \text{let } (t, z) = \delta(s, a) \text{ in } (h(t), z)$$

Protocols can inherit structure from the final coalgebra under h^{-1}, thereby providing a mechanism for transferring results on trees to state transition systems

Restart Protocols

- A prefix code is a subset $A \subseteq \Sigma^*$ such that every element of Σ^ω has at most one prefix in A
- The elements of a prefix code are $\preceq$-incomparable
- A prefix code is exhaustive (wrt μ) if $\alpha \in \Sigma^\omega$ has a prefix in A w.p. 1

A restart protocol (S, δ, s) is determined by a function $f : A \rightharpoonup \Gamma^*$, where A is an exhaustive prefix code

Intuitively, starting in s, read symbols of Σ from the input stream until encountering a string $x \in A$, output $f(x)$, repeat

Restart Protocols

Formally,

$$S = \{u \in \Sigma^* \mid x \not\preceq u \text{ for any } x \in A\}$$

$$\delta(u,a) = \begin{cases} (ua, \varepsilon), & ua \notin A, \\ (\varepsilon, z), & ua \in A \text{ and } f(ua) = z \end{cases}$$

with start state ε.

Convergence

We say that the sequence X_n converges to X in probability and write $X_n \xrightarrow{\Pr} X$ if

$$\forall \varepsilon > 0 \quad \Pr(|X_n - X| > \varepsilon) = o(1).$$

Efficiency, revisited

Efficiency = the long-term ratio of entropy production to entropy consumption

Formally,

$$E_n(\alpha) = \frac{|\delta(s, \alpha_n)|}{n} \cdot \frac{H(\nu)}{H(\mu)}$$

where H is the Shannon entropy

$$H(p_1, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log p_i$$

Intuitively, E_n measures the ratio of entropy production to consumption after n steps of δ starting in state s

Efficiency, revisited

In most cases of interest, E_n converges to a unique constant value

$$E_n \xrightarrow{\Pr} \mathsf{Eff}_\delta$$

independent of start state and history

- Well-defined for finite-state protocols
- For restart protocols, it is enough to measure the ratio for one iteration of the protocol.

Properties of δ^ω

Theorem

(i) *The partial function* $\delta^\omega(s,-) : \Sigma^\omega \rightharpoonup \Gamma^\omega$ *is continuous, thus Borel measurable*

(ii) *If* δ *is productive, then* $\delta^\omega(s,\alpha)$ *is almost surely infinite; that is,* $\mu(\operatorname{dom}\delta^\omega(s,-)) = 1$

(iii) *The measure* ν *on* Γ^ω *is the push-forward measure* $\nu = \mu \circ \delta^\omega(s,-)^{-1}$

Properties of E_n

Theorem

If δ is a reduction from μ to ν, then the random variables E_n are continuous and uniformly bounded by an absolute constant $R > 0$ depending only on μ and ν

Sequential Composition

Given

$$\delta_1 : S \times \Sigma \to S \times \Gamma^* \qquad \delta_2 : T \times \Gamma \to T \times \Delta^*$$

define

$$(\delta_1 \,;\, \delta_2) : S \times T \times \Sigma \to S \times T \times \Delta^*$$

$$\begin{aligned}(\delta_1 \,;\, \delta_2)((s,t),a) = \quad & \text{let } (u,y) = \delta_1(s,a) \text{ in} \\ & \text{let } (v,z) = \delta_2(t,y) \text{ in} \\ & ((u,v),z)\end{aligned}$$

Run δ_1 for one step, then run δ_2 on the output of δ_1

Correctness of Sequential Composition

Theorem
The partial maps

$$(\delta_1 \,;\, \delta_2)^\omega((s,t),-) \qquad\qquad \delta_2^\omega(t, \delta_1^\omega(s,-))$$

of type $\Sigma^\omega \rightharpoonup \Delta^\omega$ *are defined and agree on all but a* μ*-nullset*

The map on infinite strings induced by the sequential composition of protocols is almost everywhere equal to the functional composition of the induced maps of the component protocols

Correctness of Sequential Composition

Proof idea
Show that the binary relation

$$\begin{aligned} \beta R \gamma \;\Leftrightarrow\; & \exists \alpha \in \Sigma^{\omega} \;\; \exists s \in S \;\; \exists t \in T \\ & \beta = (\delta_1 \,;\, \delta_2)^{\omega}((s,t),\alpha) \;\wedge \\ & \gamma = \delta_2^{\omega}(t, \delta_1^{\omega}(s,\alpha)) \end{aligned}$$

on Δ^{ω} is a bisimulation

Reductions Compose

Theorem
If $\delta_1(s,-)$ is a reduction from μ to ν and $\delta_2(t,-)$ is a reduction from ν to o, then $(\delta_1 \,;\, \delta_2)((s,t),-)$ is a reduction from μ to o

Proof.
Follows from the previous theorem and

$$\nu = \mu \circ \delta_1^{\omega}(s,-)^{-1} \qquad\qquad o = \nu \circ \delta_2^{\omega}(t,-)^{-1}$$

□

Theorem

If $\delta_1(s, -)$ *is a reduction from* μ *to* ν *and* $\delta_2(t, -)$ *is a reduction from* ν *to* o*, and if* Eff_{δ_1} *and* Eff_{δ_2} *exist, then* $\mathsf{Eff}_{\delta_1;\delta_2}$ *exists and*

$$\mathsf{Eff}_{\delta_1;\delta_2} = \mathsf{Eff}_{\delta_1} \cdot \mathsf{Eff}_{\delta_2}$$

Serial Protocols

Consider a sequence (S_0, δ_0, s_0), $(S_1, \delta_1, s_1), \ldots$ of positive recurrent restart protocols defined in terms of maps $f_k : A_k \to \Gamma^*$, where the A_k are exhaustive prefix codes

These can be combined into a single serial protocol δ that executes one iteration of each δ_k, then goes on to the next

Formally, the states of δ are the disjoint union of the S_k, and δ is defined so that $\delta(s_k, x) = (s_{k+1}, f_k(x))$ for $x \in A_k$, and within S_k behaves like δ_k.

Theorem
Let δ be a serial protocol with finite-state components $\delta_0, \delta_1, \ldots$ having subexponential growth:

$$\max_{x \in A_n} |x| = o(\sum_{i=0}^{n-1} c_i)$$

Let c_n and p_n be the expected consumption and production, respectively, of one iteration of δ_n. If the limit

$$\ell = \lim_n \frac{\sum_{i=0}^{n} p_i}{\sum_{i=0}^{n} c_i}$$

exists, then the efficiency of the serial protocol exists and is equal to ℓ.

A Reduction

d-Uniform $\Rightarrow$ c-Uniform

Let $m = \lfloor k \log_c d \rfloor$. Let the c-ary expansion of d^k be

$$d^k = \sum_{i=0}^{m} a_i c^i$$

Do k calls on the d-uniform distribution. For each $0 \leq i \leq m$, for $a_i c^i$ of the possible outcomes, emit a c-ary string of length i, every possible such string occurring exactly a_i times. For a_0 outcomes, nothing is emitted (and this is lost entropy), but this occurs with probability $a_0 d^{-k}$. Restart.

- latency $= k$, efficiency $= 1 - \Theta(k^{-1})$
- can combine these into a serial protocol with asymptotically optimal efficiency (at the cost of unbounded latency)

Other Reductions

- Uniform $\Rightarrow$ Rational with efficiency $1 - \Theta(k^{-1})$
- Uniform $\Rightarrow$ Arbitrary with efficiency $1 - \Theta(k^{-1})$
- Arbitrary $\Rightarrow$ Uniform with efficiency $1 - \Theta(\log k / k)$
- $(\frac{1}{r}, \frac{r-1}{r}) \Rightarrow (r-1)$-Uniform with efficiency $1 - \Theta(k^{-1})$
  Uses Dirichlet approximation

What We Did and Didn't Do

What we did

- A coalgebraic model for constructing and reasoning about state-based protocols that implement entropy-conserving reductions between random processes
- Provided provide basic tools that allow efficient protocols to be constructed in a compositional way
- analyzed tradeoff between latency and loss of entropy
- illustrated the use of the model in various reductions

What We Did and Didn't Do

What we didn't do

- We have considered only homogeneous measures on Σ^ω and Γ^ω, those induced by Bernoulli processes in which the probabilistic choices are i.i.d., for fixed finite Σ and Γ
- However, the coalgebraic definitions of protocol and reduction make sense even if Σ and Γ are countably infinite and even if the measures are non-homogeneous

Open questions

Further open questions

- Can we do better when converting from non-uniform distributions?
- The notion of latency doesn't feel quite right: e.g. can only guarantee

$$k \leq \text{latency of composition} \leq k \cdot k'.$$

  Can we do better?
- Infinite alphabets? Continuous space?

Non-Homogeneous Processes

- A fixed measure μ on Σ induces a homogeneous measure (the product measure) on Σ^ω
- But in the final coalgebra, we can go the other direction: For an arbitrary μ on Σ^ω and state $f : \Sigma^+ \to \Gamma^*$, there is a unique assignment of transition probabilities on Σ^+ compatible with μ:

$$f(xa) = \frac{\mu(\{\alpha \mid xa \prec \alpha\})}{\mu(\{\alpha \mid x \prec \alpha\})}$$

- This determines the probabilistic behavior of the final coalgebra as a protocol starting in state f when the input stream is distributed as μ
- Any measure μ on Σ^ω induces a push-forward measure $\mu \circ (D^\omega)^{-1}$ on Γ^ω. This gives a notion of reduction even in the non-homogeneous case

Non-Homogeneous Processes

- This behavior would also be reflected in any protocol (S, δ) starting in any state $s \in h^{-1}(f)$ under the same measure on input streams, thus providing a semantics for (S, δ) even under non-homogeneous conditions
- Thus we can lift the entire theory to Mealy automata that operate probabilistically relative to an arbitrary μ on Σ^ω
- These are essentially discrete Markov transition systems with observations in Γ^*.

Continuous Space

- The state set S and alphabets Σ and Γ need not be discrete
- The appropriate generalization would give reductions between discrete-time, continuous-space Markov transition systems [Panangaden 2009, Doberkat 2007]
- Let S, Σ, and Γ be measurable spaces. A reduction protocol is a measurable function $\delta : S \times \Sigma \to S \times \Gamma^*$
- The final coalgebra is

$$D : (\Gamma^*)^{\Sigma^+} \times \Sigma \to (\Gamma^*)^{\Sigma^+} \times \Gamma^*$$
$$D(f, a) = (f@a, f(a))$$

where $f@x$ is the subtree of f at x, $f@x(y) = f(xy)$

Continuous Space

- Given a probability measure μ on Σ^ω, the transition kernel at $x \in \Sigma^*$ is the function

$$K_\mu(x, B) = \mu(\pi_{|x|}^{-1}(B) \mid x)$$

  where B is a measurable subset of Σ
- $\mu(A \mid x)$ is the conditional probability of A given x, $A \subseteq \Sigma^\omega$
- Can be obtained by disintegration from the joint distribution $\theta(B \times A) = \mu(\pi_n^{-1}(B) \cap A)$, $B \subseteq \Sigma^n$, $A \subseteq \Sigma^\omega$
- The partial function $D^\omega : \Sigma^\omega \rightharpoonup \Gamma^\omega$ is measurable and induces a push-forward measure $\mu \circ (D^\omega)^{-1}$ on Γ^ω. This is a reduction from μ to $\mu \circ (D^\omega)^{-1}$

Martingales

- Let $\mathcal{B}$ and $\mathcal{B}_\omega$ be the Borel sets of Σ and Σ^ω, respectively
- For any n and $A \in \mathcal{B}_\omega$, consider the measurable function $X_n(\alpha) = \mu(A \mid \alpha_n)$ and σ-subalgebra $\mathcal{B}_n \subseteq \mathcal{B}_\omega$ generated by $\{\pi_i^{-1}(B) \mid i \leq n,\ B \in \mathcal{B}\}$
- The sequence $(X_n, \mathcal{B}_n)$ forms a martingale that by the Lévy 0,1-law converges almost surely to the characteristic function of A

Thanks!