Evidence-Tracked Tape Semantics for Probabilistic Computation

Liron Cohen
Ben-Gurion University, Beer-Sheva, Israel

Tomer Samara
Ben-Gurion University, Beer-Sheva, Israel

Abstract

A standard intensional account of probabilistic computation represents a randomized program as a deterministic computation that consumes an explicit random tape. This yields a two-layer perspective: an intensional layer that makes reuse of randomness and correlation visible, and an extensional layer obtained by interpreting tapes under a chosen probability measure. We develop an evidence-tracked tape semantics using the monadic-core-to-evidenced-frame pipeline (and its induced realizability tripos), obtaining a higher-order logic in which entailments are witnessed by uniform evidence transformers. Quantitative statements are recovered by interpretation: once a tape measure is fixed, probabilities and expectations arise by extracting numerical summaries from tape-indexed predicates, and entailments yield sound inequalities, with an almost-sure quotient supporting probability-one reasoning. We also study intensional principles that are lost at the level of laws, including proof-relevant transport along realizable tape-rewiring maps and a canonical splitting discipline for stream tapes enforcing independent draws. Finally, we relate tape-based reasoning to an extensional law semantics via pushforward, isolating a probability-one must abstraction as a sound summary of tape-based proofs.

2012 ACM Subject Classification Theory of computation → Semantics and reasoning; Theory of computation → Probabilistic computation

Keywords and phrases probabilistic programming, random tapes, realizability, evidenced frames, program logic, monadic combinatory algebras

Digital Object Identifier 10.4230/LIPIcs.FSCD.2026.13

Funding This work was supported by Grant No. 2876/25 from the Israeli Science Foundation (ISF).

1 Introduction

Randomized computation is now a standard part of programming practice, from cryptographic constructions and randomized algorithms to probabilistic programming, e.g., [5, 17, 26, 28]. This has led to a large body of semantic and proof theoretic techniques for specifying and verifying probabilistic programs. A recurring distinction in this landscape is between reasoning extensionally, by the output law (that is, output distribution) of a program, and reasoning intensionally, by how a program consumes randomness.

Most probabilistic program logics internalize quantitative specifications: assertions speak about event probabilities, expectations, or real-valued preconditions, and proof rules are given in weakest-preexpectation or quantitative Hoare style (including relational variants), e.g., [4, 9, 20, 21, 22]. These approaches are well suited for bounds and naturally align with law-based semantics. A recent example is Probabilistic Outcome Logic [31, 32], which reasons about properties of outcome sets and supports algebraic reasoning for probabilistic and randomized nondeterministic constructs.

A different, equally classical viewpoint models randomness as an explicit input. A probabilistic program is a deterministic function that consumes a random seed, a stream of bits, or a random tape [11, 12, 28]. This viewpoint is convenient for modelling implementations,

© Liron Cohen and Tomer Samara;
licensed under Creative Commons License CC-BY 4.0

reuse of randomness, correlation, and deterministic transformations of the random source. For example, drawing one random bit once and reusing it twice is observably different from drawing two independent bits, even if both programs may induce the same marginal distribution on a single output [3]. Such distinctions are invisible once one quotients a program by its law (i.e., the output distribution obtained by pushing forward a fixed tape measure through the program), but they are central to compilation and transformation arguments where the random source is reshaped.

Working with an explicit random tape yields a built-in separation of concerns: the operational semantics is measure-agnostic, and quantitative statements arise only after fixing a measure. Much of the verification literature focuses on internal quantitative reasoning over laws, abstracting away intensional structure of randomness consumption [22, 28]. Conversely, tape-based semantics is often used operationally, while the link from tape-indexed reasoning to probabilistic statements is handled outside the logic. Our goal is a clean evidence-tracked account of tape-based computation with principled bridge results that make this link explicit.

Concretely, the paper develops an evidence-tracked higher-order logic whose entailments are uniform tape indexed refinements. A proof of an entailment is witnessed by a single evidence transformer that works uniformly for every tape. Tracking evidence makes entailment proof-relevant: an entailment is not just a relation between predicates, but is witnessed by a concrete code transformer mapping realizers of the premise to realizers of the conclusion. This is essential here because key reasoning principles, such as tape rewiring and tape splitting, correspond to transformations of tape access. At the semantic level, these principles act by reindexing computations and predicates, and at the evidenced-frame level, they require compatible translations of evidence codes. In stronger concrete language instances, such translations may further be internalized by compiler-like codes. Probabilistic claims are obtained by an external extraction step, after fixing a probability measure on tapes. This is a deliberate design choice: it keeps internal reasoning intensional and measure agnostic, while allowing the same tape level proof to be instantiated with different probabilistic readings.

Technically, we build on an established pipeline for obtaining models of higher-order intuitionistic logic via monadic cores, evidenced frames and realizability triposes [6, 8]. An evidenced frame [8] packages a notion of computation together with a preorder of predicates and a notion of evidence that tracks entailment. Applying a uniform construction to an evidenced frame yields a tripos, and hence a model of higher-order logic whose entailments carry explicit evidence. In turn, the monadic combinatory algebra (MCA) framework [6] obtains evidenced frames from monadic cores, built from a monadic combinatory algebra together with a choice of truth values and a suitable modality. In this paper we take this pipeline as a tool. Our novelty is not the generic construction, but its probabilistic instantiation for tape consuming computation and the resulting bridge theorems that connect uniform entailments to ordinary probability statements after interpretation.

In our tape setting, computations with outputs in X are measurable functions $m : R \to X_\perp$ over a measurable tape space (R, Σ), where $X_\perp := X \uplus \{\perp\}$ (i.e., X with an added divergence value). Predicates are tape indexed truth values, that is, functions $R \to [0, 1]$, ordered pointwise. Entailment is therefore tape by tape: an entailment is witnessed by a single evidence transformer that works uniformly for every tape. This makes the logic intensional and correlation aware: sequential composition threads the same tape through the continuation, so reuse of randomness is visible rather than erased.

To recover numeric probabilistic statements, we fix a probability measure ρ on (R, Σ) and apply extraction maps. For instance, expectation turns a tape-indexed truth value $f : R \to [0, 1]$ into a number $\mathbb{E}_\rho[f]$, and indicator predicates yield event probabilities. We use

the standard expectation interpretation to read quantitative consequences of our tape-level proof principles once a tape measure is fixed. We also introduce ana almost-sure quotient on predicates that identifies functions agreeing outside a null set, yielding a probability-one fragment that avoids null-set artefacts while remaining compatible with our logical structure.

The tape level supports proof principles about randomness as a resource that are invisible at the level of laws. We study two representative forms of intensional structure. First, realizable tape maps let one rewrite the random source while transporting entailments with associated evidence translations, enabling proof reuse across changes of representation. Second, for stream tapes we develop a canonical splitting construction that provides a disciplined notion of independent draws. We show how splitting-based reasoning can be translated back into single-tape evidence. These principles target transformation arguments, where preserving and reshaping the random source is the point.

In many settings one also wants to forget correlations and retain only an extensional law [16, 28]. We therefore study a distribution-level abstraction of the tape semantics. The most robust fragment that aligns directly with tape reasoning without invoking expectation is the almost-sure, must-style fragment, capturing probability-1 properties such as almost-sure termination and support-based safety. We make precise what information is retained and what is lost by passing from tapes to laws.

Contributions

- We develop an evidence-tracked, tape-based instantiation of the monadic-core-to-evidenced-frame pipeline, with tape-consuming computations and tape-indexed truth values, where entailments are uniform tape-by-tape refinements witnessed by evidence transformers.
- We introduce realizable tape maps and show that they transport evidenced entailments together with corresponding translations on evidence, enabling proof reuse across changes of random-source representation.
- For bit-stream tapes we study a canonical splitting construction that supports disciplined independent draws, and we show how splitting-based arguments can be transported back to the single-tape setting.
- After fixing a probability measure on tapes, we quotient tape truth values by almost-sure equality, yielding a probability-one interface that avoids null-set artefacts and supports robust reasoning up to measure-zero differences.
- With a tape measure in place, we use expectation and event-probability extraction to interpret tape-indexed truth values numerically, and we show that tape-level entailments yield the corresponding quantitative inequalities under this interpretation.

Outline. Section 2 recalls the monadic core to evidenced frame to tripos construction used in the paper. Section 3 develops the tape model, extraction principles, and the almost sure quotient. Section 4 treats streams, splitting, and transport along realizable tape maps. Section 5 relates tape computations to laws and formulates the probability-one must abstraction. Then, Section 6 concludes and discusses future work.

2 Background

Our goal is to obtain semantic models of higher-order intuitionistic logic in which specifications relate to probabilistic computations. We do so via an established two-step route from effectful computation to higher-order logic: (i) from a *monadic core* (an effectful combinatory structure

plus a truth-value modality [6]) one builds an *evidenced frame* (a minimal proof-relevant structure abstracting the entailment rules needed for realizability [8]), and (ii) from an evidenced frame one obtains a *realizability tripos*, a standard categorical presentation of higher-order intuitionistic logic [19, 25]. In this paper we use evidenced frames as a lightweight interface: they expose *proof-relevant* entailments while staying close to program semantics. The tripos construction mainly serves as semantic motivation, explaining why evidenced-frame entailments are implications in a higher-order logic and why evidence terms matter. This section reviews the key components from the monadic-core-to-evidenced-frame pipeline.

2.1 Monadic cores

Let M be a monad on **Set** [23]. We write $\eta_{\mathsf{A}} : \mathsf{A} \to M(\mathsf{A})$ for the unit and $(-) \gg\!= (-) : M(\mathsf{A}) \to (\mathsf{A} \to M(B)) \to M(B)$ for Kleisli composition (bind), often written $m \gg\!= f$ for $m \in M(\mathsf{A})$ and $f : \mathsf{A} \to M(B)$. An *M-combinatory algebra* (MCA) consists of a set of codes A equipped with a monadic application operation $\cdot : \mathsf{A} \times \mathsf{A} \to M(\mathsf{A}_\perp)$ satisfying a combinatory completeness requirement that ensures enough expressivity to interpret the untyped λ-calculus. (Equivalently, this can formulated via the existence of combinators witnessing the usual S/K structure internalized through M (see [6, Prop. 7]).

To turn an MCA into a proof-relevant calculus one has to provide additional ingredients:

- a truth-value object $(\Omega, \leq)$ (a complete Heyting prealgebra), and
- an *M-modality* $\Diamond$-valued in Ω that internalizes effectful sequencing at the level of truth values.[1]

▶ **Definition 1** (M-modality). *An M-modality over Ω is a natural transformation in X,*

$$\Diamond_X : M(X) \to (X \to \Omega) \to \Omega, \qquad \textit{and we note} \quad \langle\!\langle x \leftarrow m \rangle\!\rangle \, \varphi\,(x) := \Diamond_X\,(m)\,(\varphi)$$

satisfying, for all sets A, B, all predicates $\varphi, \varphi_1, \varphi_2 : A \to \Omega$, all Kleisli maps $f : A \to \mathsf{M}(B)$, all $a \in A$, and all $m \in \mathsf{M}(A)$:

1. ***After-Return.*** $\varphi(a) \;\leq\; \langle\!\langle x \leftarrow \mu(a) \rangle\!\rangle \, \varphi(x)$.
2. ***After-Bind.*** $\langle\!\langle x \leftarrow m \rangle\!\rangle \, (\langle\!\langle y \leftarrow f(x) \rangle\!\rangle \, \varphi(y)) \leq \langle\!\langle y \leftarrow (m \gg\!= f) \rangle\!\rangle \, \varphi(y)$
3. ***Internal monotonicity.***

$$\Big(\bigwedge_{x \in A} (\varphi_1(x) \Rightarrow \varphi_2(x)) \Big) \;\leq\; \Big(\langle\!\langle x \leftarrow m \rangle\!\rangle \, \varphi_1(x) \Rightarrow \langle\!\langle x \leftarrow m \rangle\!\rangle \, \varphi_2(x) \Big)$$

Intuitively, $\langle\!\langle x \leftarrow m \rangle\!\rangle \, \varphi\,(x)$ should be read as 'run the computation m, obtain a value x, and then evaluate the post-condition $\varphi(x)$'. Thus $\Diamond$ acts as a weakest-precondition style predicate transformer for effectful computations. The axioms express sound reasoning principles for such sequencing: *After-Return* states that pure values satisfy their post-conditions, *After-Bind* ensures compatibility with sequential composition (Kleisli bind), and *Internal monotonicity* guarantees that strengthening post-conditions preserves validity.

Finally, effectful realizability typically restricts which codes are allowed to serve as *evidence* for entailments through the notion of a separator. In the context of this paper we take the separator to be the full set C. A *monadic core* is then a tuple $\mathsf{MC} \;=\; (\mathsf{A}, \Omega, \Diamond)$.

[1] We use the terminology of [24, 6]. A monad M together with an M-modality induces a monad over the appropriate fibration [1], so an M-modality is also a modality in the "monad as a modality" sense.

2.2 From monadic cores to evidenced frames

Evidenced frames refine Heyting style semantics by recording *proof relevant* entailments [8]. Intuitively, an entailment $\varphi \xrightarrow{e} \psi$ is witnessed by a concrete evidence term e that uniformly converts any realizer of φ into a realizer of ψ. More generally, an evidenced frame can be regarded as a frame, or (more accurately) a complete Heyting algebra, in which one provides evidence that an element is smaller than another, and there can be multiple forms of evidence for the same fact. This proof relevance is what supports realizability style program extraction and enables comparing different effectful semantics at the level of uniform evidence.

Concretely, an evidenced frame is a tuple $(\Phi, \mathsf{E}, \cdot \dot{\to} \cdot)$ where Φ is a set of *propositions*, E is a set of *evidence terms*, and $\phi_1 \xrightarrow{e} \phi_2$ is a ternary *evidence relation* on $\Phi \times E \times \Phi$, together with a designated set of proposition-formers and evidence constructors. Given a monadic core $\mathsf{MC} = (\mathsf{A}, \Omega, \Diamond)$ one obtains an evidenced frame $\mathcal{EF}_{\mathsf{MC}} = (\Phi_{\mathsf{MC}}, \mathsf{E}_{\mathsf{MC}}, \cdot \dot{\to} \cdot)$ as follows: propositions are Ω-valued predicates on codes (i.e. $\Phi_{\mathsf{MC}} := \mathsf{A} \to \Omega$), evidence are codes (i.e. $\mathsf{E}_{\mathsf{MC}} := \mathsf{A}$), and entailment is the tracked refinement condition

$$\varphi \xrightarrow{e} \psi \quad \Longleftrightarrow \quad \forall c \in \mathsf{A}.\ \varphi(c) \leq \langle\!\langle x \leftarrow e \cdot c \rangle\!\rangle_{\Diamond}\, \psi(a).$$

Intuitively, e is a uniform evidence transformer: given any realizer c satisfying φ, the computation $(e \cdot c)$ produces outputs that satisfy ψ under the modality $\Diamond$.

2.3 From evidenced frames to realizability triposes

Every evidenced frame $\mathcal{EF} = (\Phi, E, \cdot \dot{\to} \cdot)$ induces a tripos $P_{\mathcal{EF}} : \mathbf{Set}^{op} \to \mathbf{HeytPreord}$ by taking, for each set X, the fiber $P_{\mathcal{EF}}(X)$ to consist of predicates $X \to \Phi$ ordered by *pointwise evidenced entailment*: $\alpha \leq \beta$ iff $\exists e \in E.\ \forall x \in X.\ \alpha(x) \xrightarrow{e} \beta(x)$. Reindexing is by precomposition, and the generic tripos structure (Heyting operations, quantifiers) is defined uniformly from the evidenced-frame structure [8].

A closed proposition in the induced logic corresponds to an element of Φ (take $X = 1$). Thus, if $\varphi \xrightarrow{e} \psi$ holds in the evidenced frame, then the induced tripos validates the implication $\varphi \Rightarrow \psi$, and the same code e is precisely the *realizer* witnessing this implication. More generally, a single evidence term e establishing $\alpha(x) \Rightarrow \beta(x)$ uniformly for all $x \in X$ is exactly what makes the induced higher-order reading nontrivial. Thus, it is uniform proof-relevant structure, not merely pointwise semantic validity.

The tripos is the model of higher-order logic and the evidenced frame is the proof-relevant layer that remembers *how* entailments are witnessed. This is what lets us state and prove compilation and transport results (one evidence term that works uniformly across all realizers), and later read them as sound principles in the induced logic.

2.4 Measure-theoretic conventions

This section recalls the basic probabilistic notions used in the paper. We keep the presentation lightweight and oriented toward discrete and tape-based semantics. A σ-algebra on a set X is a nonempty collection $\Sigma_X \subseteq \mathcal{P}(X)$ that is closed under complements and countable unions. A *measurable space* is a pair (X, Σ_X) where X is a set and Σ_X is a σ-algebra on X. For (X, Σ_X) and (Y, Σ_Y) measurable spaces, a function $f : X \to Y$ is *measurable* if for every $A \in \Sigma_Y$, $f^{-1}(A) \in \Sigma_X$. A *measure* on (X, Σ_X) is a function $\mu : \Sigma_X \to [0, \infty]$ such that $\mu(\emptyset) = 0$, and for every countable family of pairwise disjoint sets $(E_i)_{i \in \mathbb{N}}$, $\mu\Big(\bigsqcup_i E_i\Big) = \sum_i \mu(E_i)$. A *probability measure* is a measure μ with $\mu(X) = 1$, and a *probability space* is a triple (X, Σ_X, μ). In the discrete setting used throughout this paper, a probability distribution on a countable set A is a function $\mu : A \to [0, 1]$ such that $\sum_{a \in A} \mu(a) = 1$. This presentation is sufficient for all tape-based constructions in this work.

Let $f : X \to Y$ be a measurable function and μ a probability measure on X. The *pushforward measure* (or *law*) of f is the measure $f_*\mu$ on Y defined by $f_*\mu(B) := \mu(f^{-1}(B))$. The law captures the observable output distribution of a computation: it forgets how randomness is consumed internally and retains only the distribution of results. Crisp predicates correspond to characteristic functions. For a subset $P \subseteq X$, its indicator is the function

$$\mathbf{1}_P(x) = \begin{cases} 1 & x \in P, \\ 0 & x \notin P. \end{cases}$$

3 Random tapes via a reader monad

This section develops our main probabilistic model based on *random tapes*. The central idea is to represent probabilistic computation *extensionally* as deterministic computation that reads from an explicit source of randomness. Concretely, a probabilistic program is modeled as a (possibly partial) function that consumes a random tape and produces an observable output. This is the familiar "random variable" view of probabilistic computation, and it is standard in denotational semantics (see, e.g., [12, 13, 15, 28]). The advantage for our purposes is conceptual and proof-theoretic: randomness is made explicit and can be manipulated by program transformations (such as tape reindexing and tape splitting), while the underlying program dynamics remains deterministic.

3.1 The random tape evidenced frame

Fix a set R of random tapes equipped with a σ-algebra Σ. A random tape is an abstract representation of a complete supply of random choices for a run of a program. Formally, we model tapes by a set R whose elements are the possible realizations of randomness. The canonical example is the space of infinite bit streams $R := 2^{\mathbb{N}}$, where the program reads bits from the stream as needed. More generally, one may take R to be any space of infinite streams over a finite alphabet, or a product space encoding multiple sources of randomness. When extracting numeric probabilities, we additionally fix a probability measure ρ on (R, Σ) specifying how tapes are sampled (e.g. the fair-coin product measure on $2^{\mathbb{N}}$). Operationally, a deterministic evaluator consumes a tape $r \in R$ by reading a finite prefix (or more, as the computation proceeds), so sampling $r \sim \rho$ induces the usual probabilistic behavior.

For a countable X, $X_\perp := X \uplus \{\perp\}$, i.e., X with an added divergence value. We use this to "totalize" the output set, and we equip $X_\perp$ with the discrete σ-algebra. A tape computation with outputs in X is a measurable function $m : R \to X_\perp$. Define the **Set**-monad M by

$$\mathsf{M}(X) := \{\, m : R \to X_\perp \mid m \text{ is } \Sigma\text{-measurable} \,\}.$$

We emphasize that M depends only on the measurable structure (R, Σ) (not on any choice of measure on R). We sometimes write M_R to make this dependence explicit.

This is the (partial) *reader monad* on R (restricted to measurable functions), familiar from semantics and functional programming: a computation is a function that reads from a shared environment, here the environment is the random tape [21, 11]. The crucial point is that randomness is not internalized as probabilistic choice in the monad. Instead, the monad exposes randomness as an explicit parameter $r \in R$. Reusing the same tape r across sequential composition makes correlations explicit, and operations such as tape reindexing and splitting can be expressed as deterministic transformations on tapes.

The unit $\eta_X : X \to \mathsf{M}(X)$ is the constant function $\eta_X(x)(r) := x$. Given $f : X \to \mathsf{M}(Y)$ and $m \in \mathsf{M}(X)$, the Kleisli bind threads the *same* tape through sequential composition:

$$(m \gg\!= f)(r) := \begin{cases} \bot & \text{if } m(r) = \bot, \\ f(x)(r) & \text{if } m(r) = x \in X. \end{cases}$$

▶ **Lemma 2.** M *forms a monad.*

Fix a set of codes A with an MCA application over M, $\cdot : \mathsf{A} \times \mathsf{A} \to \mathsf{M}(\mathsf{A}) \subseteq (R \to \mathsf{A}_\bot)$. Thus for $c_1, c_2 \in \mathsf{A}$, $c_1 \cdot c_2$ denotes a measurable tape computation $r \mapsto (c_1 \cdot c_2)(r)$ in $\mathsf{A}_\bot$. From this, one can construct a complete Heyting prealgebra as follow. Take $[0,1]$ with the usual order $\leq$ and Gödel implication ($I_G(a,b) = 1$ if $a \leq b$ and b otherwise). We take truth values to be tape-indexed quantitative values, $\Omega_R := (R \to [0,1])$, ordered pointwise. Intuitively, a truth value $\alpha \in \Omega_R$ is a *random variable of success levels*: given a tape r, the number $\alpha(r)$ is the success level achieved on that particular random tape. Pointwise order then corresponds to tape-by-tape improvement of success. Because meets, joins, and implication are computed pointwise, Ω_R forms a complete Heyting prealgebra.

▶ **Definition 3.** *Define an operation $\Diamond$ valued in Ω_R as follows. For every countable set X, $m \in \mathsf{M}(X)$, and predicate $\psi : X \to \Omega_R$, define $\langle\!\langle x \leftarrow m \rangle\!\rangle \psi(x) \in \Omega_R$ by: for each tape $r \in R$,*

$$\Big(\langle\!\langle x \leftarrow m \rangle\!\rangle \psi(x)\Big)(r) := \begin{cases} 0 & \text{if } m(r) = \bot, \\ \psi(x)(r) & \text{if } m(r) = x \in X. \end{cases}$$

That is, on tape r the modality $\Diamond$ either diverges (yielding success level 0), or returns a value x and then evaluates the post-condition at the same tape r. This is the standard partial-correctness (weakest-precondition) modality for partial computations, defined here pointwise in the tape (e.g., [21, 30]).

▶ **Lemma 4.** *The operator $\Diamond$ is an M-modality.*

▶ **Definition 5** (The random tape evidenced frame)**.** *Let $(\mathsf{A}, \cdot)$ be an MCA over M and let $\Diamond$ be the above modality valued in Ω_R. The induced evidenced frame is denoted $\mathcal{EF}_{\mathsf{tape}}$.*

The propositions in $\mathcal{EF}_{\mathsf{tape}}$ are tape-indexed quantitative predicates

$$\Phi_{\mathsf{tape}} := \mathsf{A} \to \Omega_R = \mathsf{A} \to (R \to [0,1]).$$

For $\varphi \in \Phi_{\mathsf{tape}}$ and $c \in \mathsf{A}$, the value $\varphi(c)(r)$ is the success level asserted for running c on tape r. A postcondition is a predicate $\psi : \mathsf{A} \to (R \to [0,1])$. For instance, if $P \subseteq \mathsf{A}$ is a crisp property of outputs (i.e., that takes only Boolean truth values), we may take the tape-invariant postcondition $\psi_P(a)(r) := \mathbf{1}_P(a)$ Then for $c_1, c_2 \in \mathsf{A}$, the truth value

$$\langle\!\langle a \leftarrow (c_1 \cdot c_2) \rangle\!\rangle \psi_P(a) \in (R \to [0,1])$$

maps each tape r to 1 exactly when $(c_1 \cdot c_2)(r) \in P$, and to 0 otherwise (including divergence). An entailment $\varphi \xrightarrow{e} \psi$ unfolds to the pointwise condition

$$\forall c \in A,\ \forall r \in R. \quad \varphi(c)(r) \ \leq\ \Big(\langle\!\langle a \leftarrow (e \cdot c) \rangle\!\rangle \psi(a)\Big)(r).$$

Thus entailment is a uniform tracked refinement principle parameterized by the tape: the same evidence e works for all inputs c, and the refinement holds tape-by-tape. For example, fix a crisp postcondition ψ_P as above and take φ to be the constant predicate $\top(c)(r) := 1$. Then $\top \xrightarrow{e} \psi_P$ asserts that for every input code c and every tape r, if $e \cdot c$ terminates on r then its output lies in P (and divergence is treated as failure with score 0).

Entailments in $\mathcal{EF}_{\mathsf{tape}}$ are tape-indexed refinements, the next subsection explains how to extract numeric probability bounds from them once a measure ρ on (R, Σ) is fixed.

3.2 From tape-indexed truth values to numeric probabilities

In this section we fix a probability measure ρ on (R, Σ). The tape evidenced frame is built over $\Omega_R := (R \to [0,1])$ to keep completeness and avoid measurability side conditions. Measurability is needed only for probabilistic interpretation. Accordingly, let $\Omega_{\mathrm{meas}} \subseteq \Omega_R$ be the measurable functions. All expressions to which we apply expectation or almost-sure reasoning lie in Ω_{meas} since Ω_{meas} is closed under pullback along measurable tape maps and under the tape modality.

▶ **Lemma 6.** *Let X be countable and let $m : R \to X_\perp$ be measurable. If $\psi : X \to \Omega_{\mathrm{meas}}$ then $\langle \diamond x \leftarrow m \rangle \psi(x) \in \Omega_{\mathrm{meas}}$.*

To recover ordinary numeric probability statements from random-tape evidenced frame entailments, we aggregate tape-indexed truth values by expectation with respect to ρ. This is a standard monotone interpretation that yields sound numeric probability bounds, which we record because it is the bridge we use later to read probabilistic consequences of evidence transport and splitting. For $\alpha \in \Omega^{\mathrm{meas}}$ define its expectation by

$$E_\rho(\alpha) := \int_R \alpha(r)\, d\rho(r) \ \in [0,1].$$

For a tape-indexed predicate $\varphi : \mathsf{A} \to \Omega^{\mathrm{meas}}$ define its extracted numeric predicate $E_\rho(\varphi) : \mathsf{A} \to [0,1]$ by $(E_\rho(\varphi))(c) := E_\rho(\varphi(c))$. Note that expectation is monotonic, i.e., if $\alpha, \beta \in \Omega^{\mathrm{meas}}$ and $\alpha(r) \le \beta(r)$ for all $r \in R$, then $E_\rho(\alpha) \le E_\rho(\beta)$.

Since expectation is monotone, entailments between measurable predicates yield inequalities after extraction: if $\varphi, \psi : A \to \Omega_{\mathrm{meas}}$ and $\mathcal{EF}_{\mathsf{tape}} \vdash \varphi \xrightarrow{e} \psi$, then for every $c \in A$,

$$E_\rho(\varphi(c)) \ \le \ E_\rho\Big(\langle\!\langle\, a \leftarrow (e \cdot c) \,\rangle\!\rangle\, \psi(a) \Big),$$

(where the right-hand side is well defined by Lemma 6). We will use this repeatedly to turn the tape-level theorems (transport, splitting) into numeric bounds.

But while extracting probability bounds is sound, it is not complete. That is, numeric inequalities after extraction do not, in general, determine the underlying pointwise order and therefore do not entail an evidenced-frame entailment. This is because an inequality may hold after applying E_ρ even if it fails on a set of tapes of small (but nonzero) ρ-measure.

For $c', c \in \mathsf{A}$, since $(c' \cdot c) \in \mathsf{M}(\mathsf{A})$ is a measurable function $R \to \mathsf{A}\perp$ (with $\mathsf{A}_\perp$ discrete), it induces a probability measure on $\mathsf{A}_\perp$ by pushforward: for $B \subseteq \mathsf{A}_\perp$, $\mu_{c',c}(B) := \rho\big((c' \cdot c)^{-1}(B)\big)$. In particular, for each $a \in \mathsf{A}_\perp$, $\mu_{c',c}(\{a\}) \ = \ \rho\big(\{\, r \in R \mid (c' \cdot c)(r) = a \,\}\big)$. Restricting $\mu_{c',c}$ to A yields a subdistribution (the missing mass is the divergence probability $\mu_{c',c}(\{\perp\})$).

As standard, $(e \cdot c)$ induces a subdistribution $\mu_{e,c}$ by pushforward. For any numeric postcondition $\psi_0 : A \to [0,1]$, the extracted value $E_\rho\Big(\langle\!\langle\, a \leftarrow (e \cdot c) \,\rangle\!\rangle\, \psi_0^\sharp(a) \Big)$ coincides with the integral $\int \psi_0 \, d\mu_{e,c}$ (and with the corresponding sum when A is countable). For a crisp postcondition $\psi_0 = \mathbf{1}_P$, this is simply the event probability $\mu_{e,c}(P)$.

3.3 Almost-sure extensionality for tape truth values

To connect this intensional layer to probabilistic semantics, we move to a second, extensional layer. Once a measure ρ on (R, Σ) is fixed, changing a computation or a truth value on a ρ-null set should not change its probabilistic meaning. We therefore quotient tape truth values by almost-sure equality, obtaining a Heyting structure $\Omega_{a.s.}$ and an *almost-sure* evidenced frame. From this extensional layer there are two systematic ways to read probabilistic content: a

numeric extraction map (expectation and event probabilities), and a Boolean *probability-one collapse* that records whether a truth value holds almost surely. We use the probability-one collapse as the interface to the later must-style, law-level abstraction.

▶ **Definition 7.** *Let* $\alpha, \beta \in \Omega_{\text{meas}}$. *Define* $\alpha \equiv_{\text{a.s.}} \beta$ *iff* $\rho(\{\, r \in R \mid \alpha(r) \neq \beta(r) \,\}) = 0$. *Write* $[\alpha]$ *for the* $\equiv_{\text{a.s.}}$*-equivalence class of* α, *and let* $\Omega_{a.s.} := \Omega_{\text{meas}}/\equiv_{a.s.}$. *Define an order on* $\Omega_{\text{a.s.}}$ *by*

$$[\alpha] \leq_{\text{a.s.}} [\beta] \quad \Longleftrightarrow \quad \rho(\{\, r \in R \mid \alpha(r) \leq \beta(r) \,\}) = 1.$$

Intuitively, $[\alpha]$ records the truth value α *up to null sets of tapes*, and $[\alpha] \leq_{\text{a.s.}} [\beta]$ means that $\alpha(r) \leq \beta(r)$ holds for ρ-almost every tape r. There is a canonical collapse from $\Omega_{a.s.}$ to Boolean truth values, sending a class $[\alpha]$ to 1 exactly when $\alpha(r) = 1$ for ρ-almost every tape r. Intuitively, this forgets quantitative information and keeps only the probability-one content. This collapse is the bridge point between tape-based reasoning and the must-style, distribution-level abstraction in Section 5.

▶ **Definition 8.** *Let* $\mathcal{A} \subseteq \Omega_{\text{meas}}$ *be any family of measurable functions. An* essential upper bound *of* $\mathcal{A}$ *is a function* $u \in \Omega$ *such that* $\rho(\{r \mid \alpha(r) \leq u(r)\}) = 1$ *for all* $\alpha \in \mathcal{A}$. *An* essential supremum *of* $\mathcal{A}$ *is an essential upper bound* u *such that for every essential upper bound* v *of* $\mathcal{A}$ *we have* $u \leq_{\text{a.s.}} v$. *We write* $u = \operatorname{ess\,sup} \mathcal{A}$. *Essential infimum* $\operatorname{ess\,inf} \mathcal{A}$ *is defined dually.*

Essential suprema/infima exist in Ω for arbitrary families of $[0,1]$-valued measurable functions, and are unique up to $\equiv_{\text{a.s.}}$. (Equivalently: $\Omega_{\text{a.s.}}$ is a complete lattice, with arbitrary joins/meets represented by ess sup and ess inf.)

We next show that $\Omega_{\text{a.s.}}$ is a complete Heyting prealgebra.

▶ **Proposition 9.** $\Omega_{\text{a.s.}}$ *is a complete Heyting prealgebra. For* $[\alpha], [\beta] \in \Omega_{\text{a.s.}}$, *define:*

$$[\alpha] \wedge [\beta] := \big[\, \min(\alpha, \beta) \,\big], \qquad [\alpha] \vee [\beta] := \big[\, \max(\alpha, \beta) \,\big], \qquad [\alpha] \Rightarrow [\beta] := \big[\, \alpha \Rightarrow \beta \,\big],$$

where $\min, \max$ *are pointwise, and* $\Rightarrow$ *is the pointwise Gödel implication on* $[0,1]$. *For any family* $\{[\alpha_i]\}_{i \in I} \subseteq \Omega_{\text{a.s.}}$, *define:* $\bigvee_{i \in I} [\alpha_i] := \big[\operatorname{ess\,sup}_{i \in I} \alpha_i\big]$, *and* $\bigwedge_{i \in I} [\alpha_i] := \big[\operatorname{ess\,inf}_{i \in I} \alpha_i\big]$.

Proof. Well-definedness: if $\alpha \equiv_{\text{a.s.}} \alpha'$ and $\beta \equiv_{\text{a.s.}} \beta'$ then $\min(\alpha, \beta) \equiv_{\text{a.s.}} \min(\alpha', \beta')$, similarly for max and for Gödel implication, because these operators are defined pointwise and are continuous on $[0,1]$. Completeness: for any family $\{\alpha_i\}$, essential sup/inf exist and are unique up to $\equiv_{\text{a.s.}}$, hence define joins/meets in the quotient. Heyting adjunction: for $\alpha, \beta, \gamma \in \Omega$, the pointwise Gödel implication satisfies $\min(\alpha(r), \gamma(r)) \leq \beta(r)$ iff $\gamma(r) \leq (\alpha(r) \Rightarrow \beta(r))$ for each r, therefore $\alpha \wedge \gamma \leq_{\text{a.s.}} \beta$ iff $\gamma \leq_{\text{a.s.}} (\alpha \Rightarrow \beta)$. ◀

▶ **Lemma 10.** *Let* X *be countable and let* $m : R \to X_\perp$ *be measurable. If* $\psi, \psi' : X \to \Omega_{\text{meas}}$ *satisfy* $\psi(x) \equiv_{\text{a.s.}} \psi'(x)$ *for all* $x \in X$, *then* $\langle\!\diamond\!\rangle\, x \leftarrow m \,\rangle\!\diamond\!\rangle\, \psi(x) \;\equiv_{\text{a.s.}}\; \langle\!\diamond\!\rangle\, x \leftarrow m \,\rangle\!\diamond\!\rangle\, \psi'(x)$.

For a postcondition $\bar{\psi} : X \to \Omega_{\text{a.s.}}$, choose measurable representatives $\psi : X \to \Omega_{\text{meas}}$ with $[\psi(x)] = \bar{\psi}(x)$ for all x. By Lemma 10 we can define

$$\langle\!\diamond\!\rangle\, x \leftarrow m \,\rangle\!\diamond\!\rangle^{a.s}\, \bar{\psi}(x) \;:=\; \Big[\langle\!\diamond\!\rangle\, x \leftarrow m \,\rangle\!\diamond\!\rangle\, \psi(x)\Big] \;\in\; \Omega_{\text{a.s.}}.$$

▶ **Lemma 11.** *The operation* $\diamond^{\text{a.s.}}$ *is an* M*-modality.*

Therefore, via the monadic-core-to-evidenced-frame construction using $\Omega_{\text{a.s.}}$ and $\diamond^{\text{a.s.}\cdot}$, we obtain an almost-sure tape evidenced frame $\mathcal{EF}_R^{\text{a.s.}\cdot}$. Concretely, define the extensional propositions to be $\Phi_{\text{a.s.}} := \mathsf{A} \to \Omega_{\text{a.s.}}$. Entailment in the extensional tape evidenced frame is:

$$\varphi \xrightarrow{e} \psi \;\text{ holds iff }\; \forall c \in \mathsf{A}.\ \varphi(c) \leq_{\text{a.s.}} \langle\!\diamond a \leftarrow (e \cdot c)\rangle\!\diamond\, \psi(a).$$

Equivalently, for each fixed input c the required inequality holds for ρ-almost every tape r. This quotient therefore removes null-set artefacts while preserving the full higher-order (Heyting) logical structure needed by the evidenced-frame-to-tripos pipeline.

▶ **Proposition 12.**

$$\mathcal{EF}_R \vdash \varphi \xrightarrow{e} \psi \quad\Longrightarrow\quad \mathcal{EF}_R^{\text{a.s.}\cdot} \vdash [\varphi] \xrightarrow{e} [\psi].$$

Proof. Unfolding $\mathcal{EF}_R \vdash \varphi \xrightarrow{e} \psi$ we get that for every $c \in \mathsf{A}$ and $r \in R$, $\varphi(c)(r) \leq \Big(\langle\!\diamond a \leftarrow (e \cdot c)\rangle\!\diamond\, \psi(a)\Big)(r)$. To show $\mathcal{EF}_R^{\text{a.s.}\cdot} \vdash [\varphi] \xrightarrow{e} [\psi]$, we need to show that for every $c \in \mathsf{A}$, $\varphi(c) \leq_{\mathsf{a.s}} \langle\!\diamond a \leftarrow (e \cdot c)\rangle\!\diamond\, \psi(a)$. By definition of $\leq_{\mathsf{a.s}}$, it suffices to show $\rho(\{r \in R \mid \varphi(c)(r) \leq \Big(\langle\!\diamond a \leftarrow (e \cdot c)\rangle\!\diamond\, \psi(a)\Big)(r)\}) = 1$, which holds because $\{r \in R \mid \varphi(c)(r) \leq \Big(\langle\!\diamond a \leftarrow (e \cdot c)\rangle\!\diamond\, \psi(a)\Big)(r)\} = R$. ◀

The implication of Proposition 12 is strict in general. Let $R = 2^{\mathbb{N}}$ with the fair product measure μ, and let $r_0 := 000\cdots$. Define measurable truth values $\alpha', \beta' : R \to [0,1]$ by

$$\alpha'(r) := 1, \qquad \beta'(r) := \begin{cases} 0 & \text{if } r = r_0, \\ 1 & \text{if } r \neq r_0. \end{cases}$$

Then $\alpha' \leq_{\text{a.s.}} \beta'$ because $\mu(\{r_0\}) = 0$ and $\alpha'(r) \leq \beta'(r)$ holds for μ-almost every r, but $\alpha' \not\leq \beta'$ pointwise since $\alpha'(r_0) = 1 > \beta'(r_0) = 0$. Thus the almost-sure quotient removes null-set artefacts that are invisible to probabilistic semantics but would block entailments in the intensional, pointwise tape logic.

Since expectation identifies almost-surely equal random variables, the extraction map

$$E_\rho : \Omega_{\text{a.s.}} \to [0,1], \qquad E_\rho([\alpha]) := \int_R \alpha(r)\, d\rho(r),$$

is well-defined and monotone. Thus numeric probability bounds can be extracted from entailments in the extensional tape evidenced frame without reintroducing pointwise (tape-by-tape) distinctions.

3.4 Tape maps transport evidenced entailments

In the tape evidenced frame, the probabilistic behaviour of a code is mediated entirely by the tape space: a code does not *sample* randomness internally, but instead deterministically *consumes* a tape. It is therefore natural to compare two tape models by a deterministic map that rewires one tape space into another. We show that a realizable tape map induces a computable translation of entailment evidence.

Let (R, Σ) and (R', Σ') be tape spaces. Given a measurable function $\kappa : (R, \Sigma) \to (R', \Sigma')$, define reindexing (by precomposition) maps

$$\kappa_{\mathsf{A}}^{\mathsf{M}} : \mathsf{M}_{R'}(\mathsf{A}) \to \mathsf{M}_R(\mathsf{A}), \qquad (\kappa_{\mathsf{A}}^{\mathsf{M}}(m))(r) := m(\kappa(r)),$$

$$\kappa^{\Omega} : \Omega_{R'} \to \Omega_R, \qquad (\kappa^{\Omega}(\alpha))(r) := \alpha(\kappa(r)).$$

Intuitively, these maps reinterpret an R'-indexed computation or observable over R by feeding it the rewired tape $\kappa(r)$. Thus a computation or predicate defined over R' can be viewed over R via precomposition. This reindexing operation transports both computations and truth values along the tape map κ. We extend κ^Ω pointwise to tape indexed predicates: for $\varphi' : \mathsf{A} \to \Omega_{R'}$ define $\kappa^*\varphi' : \mathsf{A} \to \Omega_R$ by $(\kappa^*\varphi')(c) := \kappa^\Omega(\varphi'(c))$ and likewise for $\psi' : \mathsf{A} \to \Omega_{R'}$.

We first show that reindexing along κ commutes with the tape modality.

▶ **Lemma 13.** *For all sets X, all $m \in \mathsf{M}_{R'}(X)$, and all $\psi' : X \to \Omega_{R'}$,*

$$\kappa^\Omega\Big(\langle\!\langle\, x \leftarrow m \,\rangle\!\rangle\, \psi'(x)\Big) \;=\; \langle\!\langle\, x \leftarrow \kappa_X^{\mathsf{M}}(m) \,\rangle\!\rangle\, (\kappa^*\psi')(x)$$

A tape map $\kappa : R \to R'$ induces a semantic reindexing operation by precomposition: it rewires an R'-tape computation into an R-tape computation, and it rewires R'-truth values into R-truth values. However, an evidenced-frame entailment is not merely a semantic inequality: it is *tracked* by a code $e \in \mathsf{A}$ that serves as uniform evidence across all inputs. Thus, to transport entailments between tape spaces, it is not enough to reindex computations and predicates semantically, one must also specify how an evidence code over R' is translated into an evidence code over R. We therefore require a code translation compatible with semantic reindexing.

▶ **Definition 14.** *A measurable $\kappa : (R, \Sigma) \to (R', \Sigma')$ is* realizable *in A if there exists a a function $\mathsf{tr}_\kappa : \mathsf{A} \to \mathsf{A}$ such that for all $e, c \in \mathsf{A}$, $\kappa_{\mathsf{A}}^{\mathsf{M}}(e \cdot_{R'} c) \;=\; \mathsf{tr}_\kappa(e) \cdot_R c$.*

Thus realizability of κ means that semantic reindexing along κ admits a uniform translation of evidence codes: for every evidence code e over the R'-tape model, the translated code $\mathsf{tr}_\kappa(e)$ over the R-tape model has exactly the reindexed behavior. Under this assumption, entailments proved in the R'-tape evidenced frame can be transported to entailments in the R-tape evidenced frame by reindexing predicates and translating evidence.

▶ **Theorem 15.** *Assume $\kappa : (R, \Sigma) \to (R', \Sigma')$ is realizable with witness $\mathsf{tr}_\kappa : \mathsf{A} \to \mathsf{A}$. Then for all predicates $\varphi', \psi' : \mathsf{A} \to \Omega_{R'}$ and all $e \in \mathsf{A}$,*

$$\mathcal{EF}_{R'} \;\vdash\; \varphi' \xrightarrow{e} \psi' \quad\Longrightarrow\quad \mathcal{EF}_R \;\vdash\; \kappa^*\varphi' \xrightarrow{\mathsf{tr}_\kappa(e)} \kappa^*\psi'.$$

Proof. Assume $\mathcal{EF}_{R'} \;\vdash\; \varphi' \xrightarrow{e} \psi'$. Thus for all $c \in \mathsf{A}$ we have $\varphi'(c) \;\le\; \langle\!\langle\, a \leftarrow (e \cdot_{R'} c) \,\rangle\!\rangle\, \psi'(a)$ in $\Omega_{R'}$. Applying κ^Ω to both sides and using monotonicity of κ^Ω (pointwise order) obtains $\kappa^*\varphi'(c) \;= \kappa^\Omega(\varphi'(c)) \;\le\; \kappa^\Omega\Big(\langle\!\langle\, a \leftarrow (e \cdot_{R'} c) \,\rangle\!\rangle\, \psi'(a)\Big)$. Now by Lemma 13 with $m = (e \cdot_{R'} c)$ and ψ', we obtain $\kappa^*\varphi'(c) \;\le\; \langle\!\langle\, a \leftarrow \kappa_{\mathsf{A}}^{\mathsf{M}}(e \cdot_{R'} c) \,\rangle\!\rangle\, (\kappa^*\psi')(a)$. Finally, the realizability of κ (Def. 14) allows rewriting $\kappa_{\mathsf{A}}^{\mathsf{M}}(e \cdot_{R'} c)$ as $(\mathsf{tr}_\kappa(e) \cdot_R c)$, yielding $\kappa^*\varphi'(c) \;\le\; \langle\!\langle\, a \leftarrow (\mathsf{tr}_\kappa(e) \cdot_R c) \,\rangle\!\rangle\, (\kappa^*\psi')(a)$, and thus $\mathcal{EF}_R \;\vdash\; \kappa^*\varphi' \xrightarrow{\mathsf{tr}_\kappa(e)} \kappa^*\psi'$. ◀

In general, Theorem 15 only requires a code translation $\mathsf{tr}_\kappa : \mathsf{A} \to \mathsf{A}$ compatible with semantic reindexing. In concrete language instances with an explicit tape interface, one often expects many computable tape rewiring maps to be realizable in this sense. For stream tapes $R = 2^{\mathbb{N}}$, typical examples include bit flip, prefix dropping, even or odd splitting, and grouping into fixed size blocks. More precisely, if the code language is closed under computable tape transformations, then such maps admit corresponding translations tr_κ satisfying Definition 14. In stronger language instances, for example with a universal evaluator, one may further ask that this translation be internally implemented by a code $\mathsf{tmap}_\kappa \in \mathsf{A}$. In that case, tape transport becomes an internal compiler pass: given a code e, the program tmap_κ computes a rewired code $\mathsf{tr}_\kappa(e)$ whose behavior on R tapes simulates that of e on R' tapes.

Although tape-level entailment is measure-agnostic, the extracted numeric reading depends on the chosen measure. Thus, a tape map interacts with extraction by transporting measures.

▶ **Definition 16.** *Given a measurable function* $\kappa : (R, \Sigma) \to (R', \Sigma')$ *and* ρ*, a probability measure on* (R, Σ)*, i.e* $\rho : \Sigma \to [0, 1]$*, define the pushforward measure on* (R', Σ') *denoted as* $\kappa_* \rho$*, by:* $\forall B \in \Sigma'. \kappa_* \rho(B) = \rho(\kappa^{-1}(B))$.

▶ **Lemma 17** (Extraction commutes with tape reindexing)**.** *Let* $\kappa : (R, \Sigma) \to (R', \Sigma')$ *be measurable,* ρ *be a probability measure on* (R, Σ)*, and* $\rho' := \kappa_* \rho$ *be the pushforward measure on* (R', Σ')*. Then for every measurable* $\alpha' : R' \to [0, 1]$ *we have* $E_\rho(\kappa_\Omega(\alpha')) = E_{\rho'}(\alpha')$.

Combining Lemma 17 with the extraction soundness, the same evidence compilation along κ that transports tape-level entailments also transports the corresponding extracted numeric bounds, provided we push the measure forward along κ.

We next specialize the tape space to infinite bit streams. This fixes a canonical source of unbounded randomness and provides a canonical measure-preserving splitting operation.

4 Stream tapes and unbounded randomness

So far the tape evidenced frame has been developed for an arbitrary measurable tape space. We now instantiate the tape space by infinite bit streams. This serves two purposes: it supports unbounded consumption of randomness, and it provides a canonical, measure-preserving splitting map. The splitting map is available semantically in the stream model, and, when it is realizable in the sense of Definition 14, it yields an independence discipline at the evidenced-frame level. We will later combine this transport principle with the fact that splitting is measure-preserving in order to move between tape-level reasoning about independent draws and standard probabilistic statements.

Let $R = \{0, 1\}^{\mathbb{N}}$. For a finite bit word $w \in \{0, 1\}^{<\mathbb{N}}$ write $[w] \subseteq R$ for the cylinder set of streams with prefix w. Let Σ be the σ-algebra generated by all cylinder sets, and let μ be the fair coin product measure, uniquely determined by $\mu([w]) = 2^{-|w|}$. This section takes the tape evidenced frame $\mathcal{EF}_{\mathsf{tape}}$ specialized to (R, Σ, μ).

4.1 Why unbounded randomness matters: von Neumann unbiasing

Many basic probabilistic procedures do not have a fixed a priori bound on how many random draws they will need. A standard example is von Neumann's procedure for extracting an unbiased coin from a biased coin: read the tape in pairs until seeing 01 or 10 [29]. (Equivalently, repeat until the two bits differ.) The point of this example is not primarily the classical fact that von Neumann's procedure produces a fair coin, but to illustrate a proof pattern that is specific to our evidenced frame: fairness is established by a transport of evidence (a realizable tape map), and only then turned into an ordinary numeric probability statement by extraction.

Let the observable outcomes be $\mathsf{Val} := \{\mathsf{H}, \mathsf{T}\}$. Define an extensional tape computation $\mathsf{vn} : R \to \mathsf{Val}_\perp$ by scanning the tape in consecutive pairs $(r(2k), r(2k+1))_{k \in \mathbb{N}}$: $\mathsf{vn}(r) = \mathsf{H}$ if the first pair $(r(2k), r(2k+1))$ with unequal bits is 01, $\mathsf{vn}(r) = \mathsf{T}$ if that first such pair is 10, and $\mathsf{vn}(r) = \perp$ if no such pair exists. Because $\mathsf{Val}_\perp$ is discrete, vn is measurable iff each fiber is measurable. Indeed, $\mathsf{vn}^{-1}(\{\mathsf{H}\})$ is the disjoint union over $k \in \mathbb{N}$ of the cylinders describing k initial nondecisive pairs in $\{00, 11\}$ followed by the decisive pair 01, and similarly for T and $\perp$.

Let $\mathsf{flip} : R \to R$ be the bit flip map $\mathsf{flip}(r)(n) := 1 - r(n)$. Intuitively, flip exchanges the labels H and T: flipping all bits swaps the decisive patterns 01 and 10 while preserving the nondecisive patterns 00 and 11. Formally, define $\mathsf{swap} : \mathsf{Val}_\perp \to \mathsf{Val}_\perp$ by

$$\mathsf{swap}(\mathsf{H}) := \mathsf{T}, \qquad \mathsf{swap}(\mathsf{T}) := \mathsf{H}, \qquad \mathsf{swap}(\perp) := \perp.$$

Then for all $r \in R$, $\mathsf{vn}(\mathsf{flip}(r)) = \mathsf{swap}(\mathsf{vn}(r))$.

Outcomes are observed using crisp postconditions. Let $P_\mathsf{H} := \{\mathsf{H}\}$ and $P_\mathsf{T} := \{\mathsf{T}\}$, and let $\mathbf{1}_{P_\mathsf{H}}, \mathbf{1}_{P_\mathsf{T}} : \mathsf{Val} \to [0,1]$ be their indicators. Write $(\cdot)^\sharp$ for the tape-invariant lift to predicates $\mathsf{Val} \to (R \to [0,1])$. Consider the tape truth values

$$\alpha_\mathsf{H}(r) := \Big(\langle\!\diamond\!\rangle\, a \leftarrow \mathsf{vn}\, \langle\!\diamond\!\rangle\, (\mathbf{1}_{P_\mathsf{H}})^\sharp(a)\Big)(r), \qquad \alpha_\mathsf{T}(r) := \Big(\langle\!\diamond\!\rangle\, a \leftarrow \mathsf{vn}\, \langle\!\diamond\!\rangle\, (\mathbf{1}_{P_\mathsf{T}})^\sharp(a)\Big)(r),$$

which are simply the crisp tests for whether vn returns H or T on tape r (with divergence contributing 0 by definition of the modality). The symmetry $\mathsf{vn}(\mathsf{flip}(r)) = \mathsf{swap}(\mathsf{vn}(r))$ implies the pointwise identity $\alpha_\mathsf{H}(\mathsf{flip}(r)) = \alpha_\mathsf{T}(r)$ for all $r \in R$. Equivalently, $\mathsf{flip}^\Omega(\alpha_\mathsf{H}) = \alpha_\mathsf{T}$.

Now assume flip is realizable, witnessed by the translation $\mathsf{tr}_\mathsf{flip} : \mathsf{A} \to \mathsf{A}$. Then Theorem 15 transports any evidenced entailment along this tape symmetry: whenever e witnesses an entailment over the flip-rewired specification, the translated code $\mathsf{tr}_\mathsf{flip}(e)$ witnesses the corresponding entailment over the original tape model. Thus the tape symmetry is reflected not only at the level of truth values, but also at the level of uniform evidence.

Since flip is measure preserving for the fair product measure μ, extraction commutes with reindexing: for measurable $\beta : R \to [0,1]$, $E_\mu(\beta \circ \mathsf{flip}) = E_\mu(\beta)$. Applying this to $\beta = \alpha_\mathsf{H}$ and using $\alpha_\mathsf{H} \circ \mathsf{flip} = \alpha_\mathsf{T}$ gives $E_\mu(\alpha_\mathsf{H}) = E_\mu(\alpha_\mathsf{T})$. If moreover vn terminates μ-almost surely (a standard fact), then $E_\mu(\alpha_\mathsf{H}) + E_\mu(\alpha_\mathsf{T}) = 1$, hence each equals $1/2$. In other words, the classical fairness conclusion is obtained by extracting a numeric statement from an evidenced frame level symmetry, while realizability of flip ensures that the corresponding evidence transports uniformly across the rewiring.

4.2 A splitting discipline for independence

The tape monad threads a single tape through sequential composition. This makes correlation explicit, but it also means that independence is not automatic: two subcomputations executed in sequence will, by default, read from the same tape. To model independent draws, it is convenient to transport the resulting entailments back to one tape along a realizable splitting map, see, e.g., [11].

Fix the stream tape measure μ. Any tape computation $m : R \to X_\perp$ induces an extensional *law* on outcomes by pushforward: for measurable $A \subseteq X_\perp$ define $\mathsf{law}(m)(A) := \mu\big(m^{-1}(A)\big)$. When $X_\perp$ is discrete, we may equivalently write $\mathsf{law}(m)(z) = \Pr_{r\sim\mu}[m(r) = z]$. Intuitively, $\mathsf{law}(m)$ is what an extensional probabilistic semantics remembers about m: it forgets *which* tapes lead to which outcomes and records only their probabilities.

Sequential composition in the tape monad threads the *same* tape through the continuation, so correlation is explicit. As a result, taking laws does not in general commute with sequential composition: running m and then f against the same tape is not the same as running them with *fresh independent* randomness at each step. A standard way to enforce independent sequencing *inside* a tape semantics is therefore to equip tapes with a splitting map, and to use it as an explicit independence discipline.

Define the even/odd splitting map $\mathsf{split} : R \to R \times R$ by

$$\mathsf{split}(r) = (r_\mathsf{even}, r_\mathsf{odd}), \qquad r_\mathsf{even}(n) := r(2n), \qquad r_\mathsf{odd}(n) := r(2n+1).$$

▶ **Lemma 18.** *The pushforward of μ along* split *is the product measure $\mu \times \mu$ on $R \times R$. In particular, r_{even} and r_{odd} are independent and each has distribution μ.*

Using split we can define a derived sequencing operation on tape computations that enforces fresh randomness. Given $m : R \to X_\perp$ and $f : X \to (R \to Y_\perp)$, define

$$(m \gg\!=_{\mathsf{split}} f)(r) \;:=\; \text{let } (r_1, r_2) = \mathsf{split}(r) \text{ in } \begin{cases} \perp & \text{if } m(r_1) = \perp, \\ f(x)(r_2) & \text{if } m(r_1) = x. \end{cases}$$

▶ **Proposition 19.** *For all measurable $B \subseteq Y_\perp$,*

$$\mathsf{law}(m \gg\!=_{\mathsf{split}} f)(B) \;=\; \int_{X_\perp} \mathsf{law}(f(x))(B)\, d\mathsf{law}(m)(x).$$

In particular, split sequencing realizes the intended "fresh independent randomness" interpretation of sequential composition.

Proof. Expand the LHS as $\mu(\{r \mid (m \gg\!=_{\mathsf{split}} f)(r) \in B\})$ and rewrite it using the definition of $\gg\!=_{\mathsf{split}}$. Apply Lemma 18 to replace sampling $r \sim \mu$ with sampling $(r_1, r_2) \sim \mu \times \mu$, and then use the law of total probability (equivalently, conditional expectation) to separate the m-part (on r_1) from the f-part (on r_2). ◀

To use split as an independence discipline inside the evidenced frame, we apply the general tape map transport theorem of Section 3.4. Let $R^{(2)} := R \times R$ with product σ-algebra and product measure $\mu \times \mu$. The two tape evidenced frame uses the same code set A, but application now returns computations over $R^{(2)}$: for $e, c \in \mathsf{A}$, $(e \cdot_{R^{(2)}} c) : R^{(2)} \to \mathsf{A}_\perp$. Precomposition along $\mathsf{split} : R \to R^{(2)}$ then rewires such a two tape computation into a one tape computation by feeding it the pair $(r_{\mathsf{even}}, r_{\mathsf{odd}})$.

The measurable map split is canonical at the semantic level, but its realizability is an additional assumption on the code language rather than a consequence of the abstract tape semantics. Assume therefore that split is realizable, witnessed by a translation $\mathsf{tr}_{\mathsf{split}} : \mathsf{A} \to \mathsf{A}$. Then Theorem 15 yields, for all predicates $\varphi', \psi' : A \to \Omega_{R^{(2)}}$ and all $e \in A$,

$$\mathcal{EF}_{R^{(2)}} \vdash \varphi' \xrightarrow{e} \psi' \quad\Longrightarrow\quad \mathcal{EF}_R \vdash \mathsf{split}^*\varphi' \xrightarrow{\mathsf{tr}_{\mathsf{split}}(e)} \mathsf{split}^*\psi'.$$

Thus, an entailment proved via an interface that assumes access to two independent tapes can be transported into an entailment for a single tape by reindexing predicates along split and translating evidence accordingly. Lemma 18 ensures that, after probabilities extraction, this transport matches the intended semantics of independent randomness under the product measure.

▶ **Example 20** (Error reduction by majority)**.** This example shows how our splitting discipline supports the standard "repeat and take majority" amplification argument. Since tape-based repetition reuses the same tape, independence is not automatic, so we build the majority-of-k verifier on k split sub-tapes (to enforce independence), transport the construction back to a single-tape program using realizability of splitting, and read off the usual probability improvement by expectation extraction.

Fix a tape probability space (R, Σ, ρ) and the induced evidenced frame $\mathcal{EF}_R$. Let $P \subseteq \mathsf{A}_R$ be a crisp set of accepting outputs, and write $\psi_P(c)(r) := \mathbf{1}_P$. Let $e \in \mathsf{A}_R$ be a (tape-dependent) randomized verifier, i.e., a code that may consult the tape when run.

For $k \geq 1$ let $R^{(k)} := R^k$ with the product measure $\rho^{\otimes k}$ and the induced evidenced frame $\mathcal{EF}_{R^{(k)}}$. Let $e_{k,t} \in \mathsf{A}_{R^{(k)}}$ be the code that runs the input code k times (once per tape component) and accepts if at least t of the runs accept. Define the threshold predicate

$$\varphi^{(k)}_{\geq t}(c)(r_1, \ldots, r_k) \;:=\; \mathbf{1}_{\left[\#\{\, i \mid (e\cdot c)(r_i) \in P \,\} \geq t\right]}.$$

By the threshold construction, we have an evidenced entailment in $\mathcal{EF}_{R^{(k)}}$:

$$\mathcal{EF}_{R^{(k)}} \vdash \varphi^{(k)}_{\geq t} \xrightarrow{e_{k,t}} \psi_P,$$

Now let $\kappa := \mathsf{split}_k : R \to R^k$ be the k-ary splitting map, $\mathsf{split}_k(r) := (r_0, ..., r_{k-1})$, where $r_i(n) = r(k \cdot n + i)$. Assume split_k is realizable, with associated evidence translation $\mathsf{tr}_k : \mathsf{A} \to \mathsf{A}$. Then Theorem 15 yields the single-tape entailment

$$\mathcal{EF}_R \vdash \mathsf{split}_k^* \varphi^{(k)}_{\geq t} \xrightarrow{\mathsf{tr}_k(e_{k,t})} \mathsf{split}_k^* \psi_P.$$

The logical entailment is tape-by-tape, and we induce it with extracted probabilistic reading. Soundness of extraction gives $E_\rho\big(\mathsf{split}_k^* \varphi^{(k)}_{\geq t}(c)\big) \;\leq\; E_\rho\Big(\langle \Diamond a \leftarrow (\mathsf{tr}_k(e_{k,t}) \cdot c)\rangle \psi_P(a)\Big)$ for any $c \in \mathsf{A}_R$. Unfolding split_k^* and the fact that split_k pushes ρ forward to $\rho^{\otimes k}$ obtains $E_\rho\big(\mathsf{split}_k^* \varphi^{(k)}_{\geq t}(c)\big) = E_{\rho^{\otimes k}}\big(\varphi^{(k)}_{\geq t}(c)\big)$, hence $E_\rho\Big(\langle\!\langle\, a \leftarrow (\mathsf{tr}_k(e_{k,t}) \cdot c) \,\rangle\!\rangle\, \psi_P(a)\Big) \;\geq\; E_{\rho^{\otimes k}}\big(\varphi^{(k)}_{\geq t}(c)\big)$.

As a concrete instance, consider the majority of three by taking $k = 3$ and $t = 2$. Let $p_c := E_\rho\Big(\langle\!\langle\, a \leftarrow (e \cdot c) \,\rangle\!\rangle\, \psi_P(a)\Big)$ be the one-shot acceptance probability. Then $E_{\rho^{\otimes 3}}\big(\varphi^{(3)}_{\geq 2}(c)\big) = 3p_c^2 - 2p_c^3$ is the usual majority amplification, which strictly improves error when $p_c > \frac{1}{2}$. Thus, by combining splitting (to get independent runs), realizability of the corresponding tape map, and expectation extraction, we obtain an evidence-tracked derivation of error reduction.

5 Finitary distributions as an abstraction of tapes

The tape-based evidenced frame exposes correlations by reusing the same tape and supports evidence compilation along realizable tape maps. It is nevertheless useful to record a coarser view that forgets the tape and keeps only the *law* (output distribution) of a computation. This section packages that view as a finitary distribution semantics and relates it back to the tape evidenced frame. Because the tape model is naturally quotiented by almost-sure equality, the fragment that survives most cleanly without expectation is the probability-one fragment, so we work with Boolean truth values and the corresponding "must" modality.

5.1 A finitary distribution monadic core

For countable X, let $\mathcal{D}_{\mathrm{fin}}(X\bot)$ be the set of finitely supported distributions on $X_\bot$:

$$\mathcal{D}_{\mathrm{fin}}(X_\bot) \;=\; \Big\{ \mu : X_\bot \to [0,1] \;\Big|\; \mathrm{supp}(\mu) \text{ finite and } \sum_{z \in X_\bot} \mu(z) = 1 \Big\}.$$

The unit is the Dirac distribution, and Kleisli bind is mixture: $(\mu \gg\!= f)(w) := \sum_{z \in X_\bot} \mu(z) \cdot f(z)(w)$. We take $\Omega_{\mathrm{bool}} := \{0, 1\}$ with the usual order, hence a complete Heyting algebra. Predicates $\varphi : X_\bot \to \Omega_{\mathrm{bool}}$ are characteristic functions of subsets of $X_\bot$.

▶ **Definition 21.** *Define* $\Diamond_{must}$ *by*

$$\langle\!\langle\, x \leftarrow \mu \,\rangle\!\rangle^{must} \varphi(x) = 1 \quad\Longleftrightarrow\quad \forall z \in X_\bot.\; \mu(z) > 0 \;\Rightarrow\; \varphi(z) = 1.$$

Equivalently, $\langle\!\!\diamond x \leftarrow \mu \rangle\!\!\rangle^{must} \varphi(x) = 1$ iff $\mu(\{z \mid \varphi(z) = 1\}) = 1$. Thus must expresses support-based safety: a postcondition holds iff it holds for all outcomes that may occur with nonzero probability. In particular, excluding $\bot$ expresses almost-sure termination.

▶ **Lemma 22.** *The operator $\Diamond_{must}$ is an M-modality.*

Note that we can replace Ω_{bool} by the complete Heyting chain $\Omega = [0, 1]$ (with Gödel implication) and keep $\mathcal{D}_{\mathrm{fin}}$. The induced must modality is then the support-infimum transformer $\langle\!\!\diamond x \leftarrow \mu \rangle\!\!\rangle^{must} \varphi(x) := \inf\{\, \varphi(z) \mid \mu(z) > 0 \,\}$. This is an M-modality, but it is best read as demonic or safety-style quantitative refinement (a worst-case score across possible outcomes), not as a probability bound. Since our tape model focuses on almost-sure extensionality, we use the Boolean must fragment as the canonical distribution-level abstraction and leave other quantitative choices to future work.

▶ **Definition 23.** *Let $\mathcal{EF}_{\mathrm{dist}}$ denote the evidenced frame induced by the monadic core $\mathsf{MC} = (\mathsf{A}, \Omega_{\mathrm{bool}}, \Diamond_{must})$ over the monad $\mathcal{D}_{\mathrm{fin}}$.*

$\mathcal{EF}_{\mathrm{dist}}$ provides a compact law-level evidenced frame whose propositions range over deterministic codes and whose entailments express support-based (probability-one) properties. Thus, the proposition fiber is $\Phi_{\mathrm{dist}} := \mathsf{A} \to \Omega_{\mathrm{bool}}$, so predicates range over deterministic codes. Entailment has the usual uniform refinement form:

$$\varphi \xrightarrow{e} \psi \quad \Longleftrightarrow \quad \forall c \in \mathsf{A}.\ \varphi(c) = 1 \Rightarrow \langle\!\!\diamond a \leftarrow (e \cdot_0 c) \rangle\!\!\rangle^{must} \psi(a) = 1.$$

Thus predicates express support-based properties such as safety, invariants over possible outputs, and probability-one termination (by excluding $\bot$).

Note that one can also take codes themselves to be finitary distributions. Let $\mathsf{A}^* := \mathcal{D}_{\mathrm{fin}}(\mathsf{A})$ and suppose we have an MCA structure with application

$$\cdot_* : \mathsf{A}^* \times \mathsf{A}^* \to \mathcal{D}_{\mathrm{fin}}(\mathsf{A}^*_\bot) \;=\; \mathcal{D}_{\mathrm{fin}}\big(\mathcal{D}_{\mathrm{fin}}(\mathsf{A})_\bot\big).$$

Equipping this MCA with the same Ω_{bool} and $\Diamond_{must}$ yields an alternative evidenced frame whose propositions are predicates $\mathsf{A} \to \Omega_{\mathrm{bool}}$ on internal laws. Although truth values remain Boolean, such predicates can express numerical constraints on the internal distribution carried by a code. For $E \subseteq \mathsf{A}$ and threshold $p \in [0, 1]$, define $P_{E,p}(\nu) = 1 \quad \Longleftrightarrow \quad \sum_{x \in E} \nu(x) \geq p$.

5.2 From tapes to finitary laws

We now relate the finitary distribution view to the tape model. Fix a tape probability space (R, Σ, ρ). For any measurable tape computation $m : R \to X_\bot$, define its law, i.e., its pushforward output distribution under ρ, $\mathsf{push}(m) : X_\bot \to [0, 1]$ by

$$\mathsf{push}(m)(z) := \rho(\{\, r \in R \mid m(r) = z \,\}).$$

This forgets which tapes lead to which outcomes, keeping only their probabilities. If $\mathsf{push}(m)$ has finite support then $\mathsf{push}(m) \in \mathcal{D}_{\mathrm{fin}}(X_\bot)$.

If $m, n : R \to X_\bot$ are equal ρ-almost surely, then $\mathsf{push}(m) = \mathsf{push}(n)$. Equality in law is strictly coarser than almost-sure equality: it identifies computations with the same output distribution, even when they differ on how they consume and reuse randomness.

For finitary laws, probability-one validity coincides with a must-style, support-based reading. Given $P \subseteq X_\bot$ and $\mu \in \mathcal{D}_{\mathrm{fin}}(X_\bot)$, write

$$\mu \models_{\mathsf{must}} P \quad \Longleftrightarrow \quad \forall z \in X_\bot.\ \mu(z) > 0 \Rightarrow z \in P.$$

Equivalently, $\mu \models_{\mathsf{must}} P$ iff $\sum_{z \in P} \mu(z) = 1$.

This gives a clean interface from tape-based reasoning to the finitary law view: whenever a tape-level argument establishes that a crisp postcondition P holds with probability 1 (for instance via the almost-sure quotient and extraction), the induced finitary law $\mathsf{push}(m)$ satisfies the must judgment $\mathsf{push}(m) \models_{\mathsf{must}} P$. Conversely, must judgments about $\mathsf{push}(m)$ are stable under the almost-sure quotient.

6 Conclusion

We developed an evidence-tracked, tape-based semantics for probabilistic computation, where randomized programs consume an explicit random tape and sequencing reuses it, making correlation explicit. Entailments are uniform tape-by-tape refinements witnessed by evidence transformers. After fixing a tape measure, probabilistic content is recovered by extraction and soundness, and we quotient truth values by almost-sure equality to obtain a probability-one fragment. We also identify two intensional principles lost at the level of laws: proof-relevant transport along realizable tape rewiring maps and a canonical splitting discipline for stream tapes. Finally, we relate tape reasoning to a probability-one must abstraction via pushforward.

Several lines of work make randomness explicit through streams, tapes, or random variables, including deterministic stream sampling [10] and tape-based denotational accounts of probabilistic programming [27]. We share this randomness-as-input perspective, which is particularly useful for higher-order probabilistic computation where measure-theoretic models face well-known structural difficulties, such as the failure of the category **Meas** to be cartesian closed and the resulting use of quasi-Borel spaces and measurable cones [18, 14]. However, our goal and interface are different: we use realizability and evidenced frames to obtain proof-relevant entailments, so reasoning steps carry evidence and support transport along realizable tape maps and splitting principles for independence. Extensional probabilistic claims are extracted only later, after fixing a tape measure and passing to laws. Compared with probabilistic Hoare, expectation-based, relational, and outcome-oriented logics [4, 22, 31, 32], our approach is more intensional: outcome-style reasoning appears as a Boolean fragment, while richer truth values support quantitative evidence-tracked reasoning.

Several extensions are natural. On the semantic side, we currently restrict to countable outcome types so that measurability of Kleisli composition follows from countable unions. Extending the tape monad to categories of measurable spaces such as **Meas** or standard Borel spaces [2, 18, 14] would allow richer outputs while preserving extraction and almost sure reasoning. Our transport theorems isolate realizability of a tape map as the key condition ensuring that entailments and evidence compile along tape rewiring, and it would be valuable to characterize this condition in concrete language instances. At the extensional level, we singled out the probability one must fragment as the abstraction commuting with the almost sure quotient. A natural next step is to identify the largest law-level fragment soundly obtainable from tape proofs, possibly via extraction, and to understand when such law-level principles can be reflected back into tape refinements under additional structure. Finally, although the present paper is semantic in focus, the induced tripos also points toward a proof relevant higher-order proof theory: the non-tape fragment should follow existing evidence tracked systems for evidenced frames, e.g. [7], while the tape-specific content would appear as additional admissible principles for transport and splitting. Developing such a syntactic account for a concrete tape access language, together with soundness and completeness results, is left for future work.

References

1 Alejandro Aguirre and Shin-ya Katsumata. Weakest Preconditions in Fibrations. *Electronic Notes in Theoretical Computer Science*, 352:5–27, 2020. doi:10.1016/j.entcs.2020.09.002.

2 Robert J. Aumann. Borel Structures for Function Spaces. *Illinois Journal of Mathematics*, 5(4):614–630, December 1961. doi:10.1215/ijm/1255631584.

3 Gilles Barthe, Thomas Espitau, Benjamin Grégoire, Justin Hsu, Léo Stefanesco, and Pierre-Yves Strub. *Relational Reasoning via Probabilistic Coupling*, pages 387–401. Springer Berlin Heidelberg, 2015. doi:10.1007/978-3-662-48899-7_27.

4 Gilles Barthe, Boris Köpf, Federico Olmedo, and Santiago Zanella-Béguelin. Probabilistic Relational Reasoning for Differential Privacy. *ACM Transactions on Programming Languages and Systems*, 35(3):1–49, November 2013. doi:10.1145/2492061.

5 Dan Boneh and Victor Shoup. A graduate course in applied cryptography, 2015.

6 Liron Cohen, Ariel Grunfeld, Dominik Kirst, and Étienne Miquey. From Partial to Monadic: Combinatory Algebra with Effects. In Maribel Fernández, editor, *10th International Conference on Formal Structures for Computation and Deduction (FSCD 2025)*, volume 337 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 14:1–14:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.FSCD.2025.14.

7 Liron Cohen, Ariel Grunfeld, Dominik Kirst, and Étienne Miquey. Syntactic Effectful Realizability in Higher-Order Logic. In *40th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, 2025. doi:10.1109/lics65433.2025.00009.

8 Liron Cohen, Etienne Miquey, and Ross Tate. Evidenced Frames: A Unifying Framework Broadening Realizability Models. In *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13. IEEE, June 2021. doi:10.1109/lics52264.2021.9470514.

9 Ricardo Corin and Jerry den Hartog. *A Probabilistic Hoare-Style Logic for Game-Based Cryptographic Proofs*, pages 252–263. Springer Berlin Heidelberg, 2006. doi:10.1007/11787006_22.

10 Fredrik Dahlqvist, Alexandra Silva, and William Smith. Deterministic stream-sampling for probabilistic programming: semantics and verification . In *2023 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–13, Los Alamitos, CA, USA, June 2023. IEEE Computer Society. doi:10.1109/LICS56636.2023.10175773.

11 Fredrik Dahlqvist, Alexandra Silva, and William Smith. Deterministic Stream-Sampling for Probabilistic Programming: Semantics and Verification, June 2023. doi:10.1109/lics56636.2023.10175773.

12 Pietro Di Gianantonio and Abbas Edalat. A Cartesian Closed Category for Random Variables. In *Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '24, pages 1–14. ACM, July 2024. doi:10.1145/3661814.3662126.

13 Thomas Ehrhard. Differentials and Distances in Probabilistic Coherence Spaces. *LIPIcs, Volume 131, FSCD 2019*, 131:17:1–17:17, 2019. doi:10.4230/LIPIcs.FSCD.2019.17.

14 Thomas Ehrhard, Michele Pagani, and Christine Tasson. Measurable Cones and Stable, Measurable Functions: a Model for Probabilistic Higher-Order Programming. *Proceedings of the ACM on Programming Languages*, 2(POPL):1–28, December 2017. doi:10.1145/3158147.

15 John T. Gill. Computational Complexity of Probabilistic Turing Machines. In *Proceedings of the sixth annual ACM symposium on Theory of computing - STOC '74*, STOC '74, pages 91–95. ACM Press, 1974. doi:10.1145/800119.803889.

16 Michèle Giry. A Categorical Approach to Probability Theory. In *Categorical Aspects of Topology and Analysis*, pages 68–85. Springer Berlin Heidelberg, 1982. doi:10.1007/bfb0092872.

17 Andrew D. Gordon, Thomas A. Henzinger, Aditya V. Nori, and Sriram K. Rajamani. Probabilistic Programming. In *Future of Software Engineering Proceedings*, ICSE '14, pages 167–181. ACM, May 2014. doi:10.1145/2593882.2593900.

18 Chris Heunen, Ohad Kammar, Sam Staton, and Hongseok Yang. A Convenient Category for Higher-Order Probability Theory. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12. IEEE, June 2017. doi:10.1109/lics.2017.8005137.

19 J. M. E. Hyland, P. T. Johnstone, and A. M. Pitts. Tripos Theory. *Mathematical Proceedings of the Cambridge Philosophical Society*, 88(2):205–232, September 1980. doi:10.1017/s0305004100057534.

20 Benjamin Lucien Kaminski, Joost-Pieter Katoen, Christoph Matheja, and Federico Olmedo. *Weakest Precondition Reasoning for Expected Run–Times of Probabilistic Programs*, pages 364–389. Springer Berlin Heidelberg, 2016. doi:10.1007/978-3-662-49498-1_15.

21 Dexter Kozen. Semantics of Probabilistic Programs. *Journal of Computer and System Sciences*, 22(3):328–350, June 1981. doi:10.1016/0022-0000(81)90036-2.

22 Annabelle McIver and Carroll Morgan. *Abstraction, Refinement and Proof for Probabilistic Systems*. Springer, Springer Nature, United States, 2005. doi:10.1007/b138392.

23 Eugenio Moggi. Notions of Computation and Monads. *Information and Computation*, 93(1):55–92, July 1991. doi:10.1016/0890-5401(91)90052-4.

24 Andrew M Pitts. Evaluation logic. In *IV Higher Order Workshop, Banff 1990: Proceedings of the IV Higher Order Workshop, Banff, Alberta, Canada 10–14 September 1990*, pages 162–189. Springer, 1991. doi:10.1007/978-1-4471-3182-3_11.

25 Andrew M. Pitts. Tripos Theory in Retrospect. *Mathematical Structures in Computer Science*, 12(3):265–279, June 2002. doi:10.1017/s096012950200364x.

26 Anish Saini, Athanasios Tsokanos, and Raimund Kirner. Quantum Randomness in Cryptography—A Survey of Cryptosystems, RNG-Based Ciphers, and QRNGs. *Information*, 13(8):358, July 2022. doi:10.3390/info13080358.

27 Sam Staton. Commutative semantics for probabilistic programming. In *Programming Languages and Systems: 26th European Symposium on Programming, ESOP 2017, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2017, Uppsala, Sweden, April 22–29, 2017, Proceedings*, pages 855–879, Berlin, Heidelberg, 2017. Springer-Verlag. doi:10.1007/978-3-662-54434-1_32.

28 Sam Staton, Hongseok Yang, Frank Wood, Chris Heunen, and Ohad Kammar. Semantics for Probabilistic Programming: Higher-Order Functions, Continuous Distributions, and Soft Constraints. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '16, pages 525–534. ACM, July 2016. doi:10.1145/2933575.2935313.

29 John von Neumann. Various Techniques Used in Connection with Random Digits. In *Notes by G. E. Forsythe, in National Bureau of Standards, Applied Mathematics Series*, volume 12, pages 36–38. National Bureau of Standards, Washington, D. C., 1951.

30 Noam Zilberstein, Derek Dreyer, and Alexandra Silva. Outcome Logic: A Unifying Foundation for Correctness and Incorrectness Reasoning. *Proceedings of the ACM on Programming Languages*, 7(OOPSLA1):522–550, April 2023. doi:10.1145/3586045.

31 Noam Zilberstein, Dexter Kozen, Alexandra Silva, and Joseph Tassarotti. A demonic Outcome Logic for Randomized Nondeterminism. *Proceedings of the ACM on Programming Languages*, 9(POPL):539–568, January 2025. doi:10.1145/3704855.

32 Noam Zilberstein, Alexandra Silva, and Joseph Tassarotti. Probabilistic Concurrent Reasoning in Outcome Logic: Independence, Conditioning, and Invariants. *Proc. ACM Program. Lang.*, 10(POPL), January 2026. doi:10.1145/3776651.

A Appendix

This appendix contains proofs for the paper.

Proof of Lemma 2. $\eta_X(x)$ is constant, hence measurable. If m and all $f(x)$ are measurable, then $(m \gg\!= f)^{-1}(\{y\})$ is a union of sets of the form $m^{-1}(\{x\}) \cap f(x)^{-1}(\{y\})$, hence measurable. The monad laws are the usual reader-monad equalities. ◀

Proof of Lemma 4. All three axioms hold pointwise in $r \in R$. After-Return holds because $\eta_X(x)(r) = x$. After-Bind holds because both sides case split on whether $m(r) = \bot$ or $m(r) = x$. Internal Monotonicity holds because the Heyting structure on Ω is pointwise: it reduces to the corresponding inequality in $\Omega_0 = [0, 1]$ for each fixed tape r. ◀

Proof of Lemma 6. For $q \in \mathbb{Q}$,

$$\{r \mid \llparenthesis x \leftarrow m \rrparenthesis\, \psi(x)(r) > q\} = \bigcup_{x \in X} \Big(m^{-1}(\{x\}) \cap \{r \mid \psi(x)(r) > q\} \Big),$$

a countable union of measurable sets. ◀

Proof of Lemma 10. For each $x \in X$ let $N_x := \{\, r \in R \mid \psi(x)(r) \neq \psi'(x)(r) \,\}$, so $\rho(N_x) = 0$. Since X is countable, $N := \bigcup_{x \in X} N_x$ is measurable and $\rho(N) = 0$. Fix $r \notin N$. If $m(r) = \bot$, then both sides evaluate to 0 at r. If $m(r) = x$, then $\psi(x)(r) = \psi'(x)(r)$ by choice of N, hence the two sides agree at r. Thus the two resulting truth values agree outside a ρ-null set, i.e. are $\equiv_{\text{a.s.}}$. ◀

Proof of Lemma 11. Choose measurable representatives throughout, and use that the corresponding laws hold *pointwise in* r for $\diamond$ on Ω_{meas}. Monotonicity follows by collecting the countably many null sets witnessing $\psi(x) \leq \chi(x)$ and arguing as in Lemma 10. Unit and bind are equalities of measurable functions $R \to [0,1]$ holding for every r, hence their classes are equal in $\Omega_{\text{a.s.}}$. ◀

Proof of Lemma 13. Both sides are equal as functions $R \to [0,1]$. If $m(\kappa(r)) = \bot$ then both sides are 0 and if $m(\kappa(r)) = x \in X$ then both sides are $\psi'(x)(\kappa(r))$. ◀

Proof of Lemma 17. Unfolding LHS results in

$$E_\rho(\kappa_\Omega(\alpha')) = \int_R \kappa_\Omega(\alpha'(r) d\rho(r) = \int_R \alpha'(\kappa(r)) d\rho(r)$$

By definition of $\rho' = \kappa_* \rho$ (i.e $\rho'(B) = \rho(\kappa^{-1}(B)$ for all $B \in \Sigma'$) and by change-of-variables formula for pushforward measures (equivalently, the defining property of $\kappa_* \rho$), the above equals to

$$\int_{R'} \alpha'(r')\, d\rho'(r') = E_{\rho'}(\alpha')$$

Hence $E_\rho(\kappa_\Omega(\alpha')) = E_{\rho'}(\alpha')$. as required. ◀

Proof of Lemma 18. It suffices to check equality on cylinder rectangles $[u] \times [v]$. The preimage $\mathsf{split}^{-1}([u] \times [v])$ constrains $|u|$ even positions and $|v|$ odd positions of r, so it is a cylinder of length $|u| + |v|$ and has measure $2^{-(|u|+|v|)}$, which equals $\mu([u])\mu([v])$. ◀

Proof of Lemma 22. After-Return: δ_x has support $\{x\}$, so the must condition reduces to $\varphi(x) = 1$. After-Bind: an outcome w has positive probability under $\mu \gg\!= f$ iff there exists z with $\mu(z) > 0$ and $f(z)(w) > 0$, so the must condition for $\mu \gg\!= f$ is equivalent to: for all z in the support of μ, the must condition holds for $f(z)$. Internal Monotonicity is immediate for Boolean truth values. ◀