Aleatoric Propositions

Reasoning about Coins

Tim French

The University of Western Australia

2023

Overview

- Aleatoric propositions replace the True/False atoms of propositional logic, with aleatoric events, or biased coin flips.
- We present some novel operators to give a language for aleatoric events.
- We examine the correspondence between this language, existing logics, and the set of rational polynomials over $(0, 1)$.

Experiential Logic

The interest in aleatoric logic comes from an AI course I taught, where probabilistic agents out performed logical agents.

It occurred to me that many logical approaches are too brittle for AI applications.

While practical (human) decision processes are not *irrational* they are not entirely deductive either.

They are based on experience, assumption and bias: imagine a Markov decision procedure that is able to simulate the outcomes of your assumptions.

Propositions and Probabilities

The simulation of a Markov decision process, requires aleatoric variable, or coin flips.
Once you have a set of (imagined coins) you can arrange them in coin flipping protocols.

"If I'm *thirsty* or if I'm *tired* drink coffee"

Here, *thirsty* and *tired* are imagined as biased coins, that are flipped (Bernoulli tests) when I consider getting a coffee.

- We have an intensional language, which describes how coin flipping protocols are defined.
- and we have an extensional language, which describes how probabilities are associated with the coins.

This paper serves as a foundation for first order and modal variations of aleatoric propositions.

Propositions as Coins

We consider a propositional atom to represented by a coin, and the interpretation of that proposition is a coin flip, where the coin either lands heads (●) or (○).
From these propositional atoms, we can describe complex propositions using

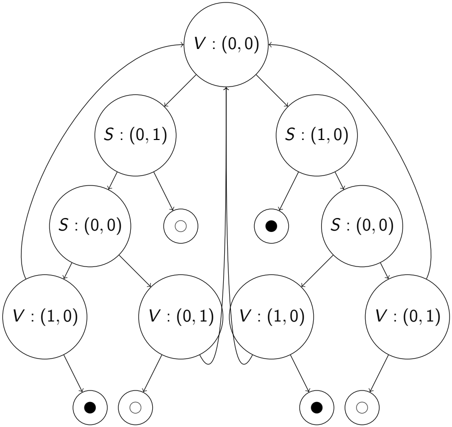
- *negation*: replace heads with tails, and tails with heads.
- *if-then-else*: if proposition A is heads, return of the result of proposition B, otherwise return the result of proposition C.
- *fixed-point operations*: A fixed point operator works as iteration in the protocol: where the quantified variable is substituted the quantifier proposition.

A classic example of an aleatoric proposition is von Neumann's process to define a fair coin flip, given a coin of any bias.

Example 1: Tennis

Suppose that Venus and Serena are in a tennis tennis tie break. Venus serves first, then Serena serves twice, and then Venus serves twice, and so on, until one of them is two points ahead of the other.

We suppose the server has the advantage, and V is the probability Venus wins on her serve, S be the probability that Serena wins on her serve.



Syntax

The *syntax of aleatoric propositions* is given by:

$$\alpha ::= \ \bullet \mid A \mid \neg\alpha \mid (\alpha ? \alpha : \alpha) \mid \mathbb{F}X\alpha$$

where:

- $\bullet$ is heads (or *true*)
- A is an atomic proposition.
- $\neg\alpha$ is *not* α.
- $(\alpha ? \beta : \gamma)$ is *if* α *then* β *else* γ.
- $\mathbb{F}X\alpha$ is *for* X *equal to* α.

For α to be well-formed, we require for every subformula $\mathbb{F}X\beta$, β has no subformulas of the type $(X ? \gamma_1 : \gamma_2)$.

$$\mathit{fairCoin} = \mathbb{F}X(A?(A?X:\bullet):(A?\circ:X)).$$

Semantics

An *interpretation* for propositional aleatoric logic is a function $\mathcal{I} : \mathrm{Atoms} \longrightarrow (0,1)$, and we let $\mathcal{I}[X : p](Y) = \mathcal{I}(Y)$ if $X \neq Y$ and $\mathcal{I}[X : p] = p$.

The interpretation assigns the probability $\mathcal{I}(\alpha)$ inductively as follows:

$$
\begin{array}{rcl}
\bullet^{\mathcal{I}} & = & 1 \\
A^{\mathcal{I}} & = & \mathcal{I}(A) \\
(\neg\alpha)^{\mathcal{I}} & = & 1 - \alpha^{\mathcal{I}} \\
(\alpha?\beta:\gamma)^{\mathcal{I}} & = & \alpha^{\mathcal{I}} \cdot \beta^{\mathcal{I}} + (1 - \alpha^{\mathcal{I}}) \cdot \gamma^{\mathcal{I}} \\
(\mathbb{F}X\alpha)^{\mathcal{I}} & = & \begin{cases} 1 \text{ if } \alpha^{\mathcal{I}} = 1 \\ 0 \text{ if } \alpha^{\mathcal{I}} = 0 \\ x \text{ if } x \text{ is the unique value such that } \alpha^{\mathcal{I}[X:x]} = x \\ 1/2 \text{ if } \forall x \in (0,1),\ \alpha^{\mathcal{I}[X:x]} = x \end{cases}
\end{array}
$$

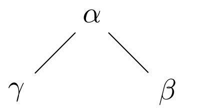
We note the interpretation of $\mathbb{F}XX$ is $1/2$, so the fixed point operator is a *median fixed point*.

If-then-else

The *if-then-else* operator is relatively novel in logic, and is one of the main differences between a fuzzy product logic and aleatoric propositions.
Some of the useful abbreviations that can be defined are:

Abbreviation	Expression
○	$\neg$●
◐	$\mathbb{F}XX$
$\alpha \wedge \beta$	$(\alpha?\beta:$○$)$
$\alpha \vee \beta$	$(\alpha?$●$:\beta)$
$\alpha \rightarrow \beta$	$(\alpha?\beta:$●$)$
$\alpha \leftrightarrow \beta$	$(\alpha?\beta:\neg\beta)$
$\alpha^{\frac{0}{m}}$	●
$\alpha^{\frac{n}{0}}$	○
$\alpha^{\frac{n}{m}}$	$(\alpha?\alpha^{\frac{n-1}{m-1}}:\alpha^{\frac{n}{m-1}})$

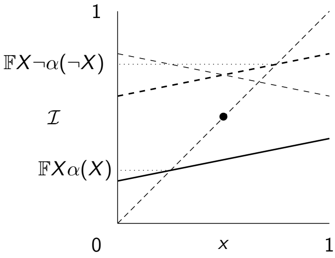
We typically draw the proposition $(\alpha?\beta:\gamma)$ as a tree where the right is α *lands* ● and left is α *lands* ○.



$$(1-\alpha^{\mathcal{I}})\cdot\gamma^{\mathcal{I}} \quad + \quad \alpha^{\mathcal{I}}\cdot\beta^{\mathcal{I}}$$

The Fixed Point Operator

The semantics of the fixed point operator have a closed form, as the well-formed requirement means that in interpretation of $\mathbb{F}X\alpha$, the interpretation of α is always a linear function of the interpretation of X.



The semantic interpretation of $\mathbb{F}X\alpha(X)$, showing how the value of $\mathbb{F}X\alpha$ corresponds to $\alpha^{\mathcal{I}[X:x]}$ with respect to x.

Functional Semantics

We can give *functional semantics* for propositional aleatoric logic that assigns a value $h_\alpha \in [0,1]$ and a value i_α^X for each $X \in \mathrm{bnd}(\alpha)$ as follows:

$$\psi = \bullet : \;\; h_\psi = 1 \qquad i_\psi^X = 0$$
$$\psi = A \in \mathrm{free}(alpha) : \;\; h_\psi = \mathcal{I}(A) \qquad i_\psi^X = 0$$
$$\psi = X \in \mathrm{bnd}(\alpha) : \;\; h_\psi = 0 \qquad i_\psi^X = 1$$
$$\psi = Y \in \mathrm{bnd}(\alpha) : \;\; h_\psi = 0 \qquad i_\psi^X = 0$$
$$\psi = (\alpha ? \beta : \gamma) : \;\; h_\psi = h_\alpha \cdot h_\beta + (1 - h_\alpha) \cdot h_\gamma \qquad i_\psi^X = h_\alpha \cdot i_\beta^X + (1 - h_\alpha)$$
$$\psi = \neg\alpha : \;\; h_\psi = 1 - h_\alpha \qquad i_\psi^X = -i_\alpha^X$$
$$\psi = \mathbb{F}X\alpha, \; i_\alpha^X \neq 1 : \;\; h_\psi = \frac{h_\alpha}{1 - i_\alpha^X} \qquad i_\psi^X = 0$$
$$\psi = \mathbb{F}X\alpha, \; i_\alpha^X = 1 : \;\; h_\psi = 1/2 \qquad i_\psi^X = 0$$
$$\psi = \mathbb{F}Y\alpha, \; i_\alpha^Y \neq 1 : \;\; h_\psi = \frac{h_\alpha}{1 - i_\alpha^Y} \qquad i_\psi^X = \frac{i_\alpha^X}{1 - i_\alpha^Y}$$
$$\psi = \mathbb{F}Y\alpha, \; i_\alpha^Y = 1 : \;\; h_\psi = 1/2 \qquad i_\psi^X = 0$$

Lines 7 and 9 show how the fixed point operator is realised as a division of polynomials.

Example: Tennis

Then Venus winning the tie break can be represented as the following aleatoric proposition:

$$\mathbb{F}X(V?(S?(S?(V?X:\bigcirc):(V?\bullet:X)):\bullet):(S?\bigcirc:(S?(V?X:\bigcirc):(V?\bullet:X))))$$

and applying the functional semantics we can reduce this to the equation:

$$VenusWins(V,S) = \frac{V - S\cdot V}{S + V - 2\cdot S\cdot V} \quad (1)$$

with the contour plot given to the right.

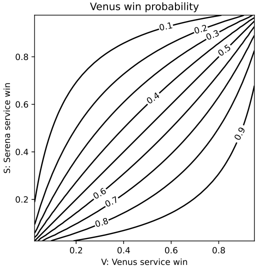


Figure: A contour diagram of the probability of Venus winning a tie break.

A Correspondence Result

The question we seek to address is what is the expressivity of aleatoric propositions, or what kind of functions we can express.

The questions have been examined in the context of "Bernoulli factories", which are essentially infinite coin flipping protocols.

Keane and O'Brien (1994) showed that Bernoulli factories can simulate any continuous polynomial bounded function over $(0, 1)$

Mossel and Peres (2005) showed in the single variable case coin flipping protocols correspond to the set of rational functions over $(0, 1)$, using a result of Pòlya.

The essence of the approach is to show that every aleatoric proposition has a normal form that corresponds to general type of rational function over $(0, 1)$.

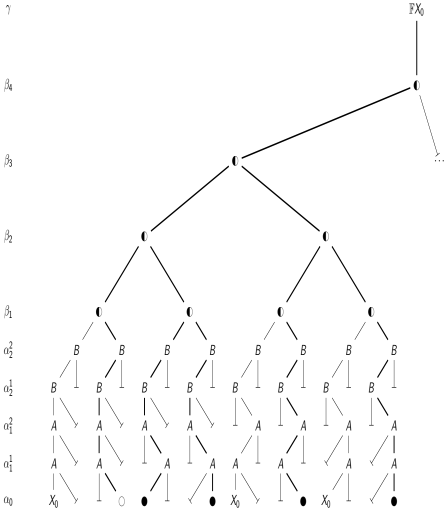
k-block normal form

A formula of aleatoric propositional logic is in *k-block normal form* if it satisfies the following syntax for γ:

$$
\begin{array}{rcl}
\alpha_1^0 & ::= & \bullet \mid \circ \mid X_0 \\
\alpha_i^{j+1} & ::= & (A_i?\alpha_i^j : X_0) \mid (A_i?X_0 : \alpha_i^j) \\
\alpha_{i+1}^1 & ::= & (A_{i+1}?\alpha_i^k : X_0) \mid (A_{i+1}?X_0 : \alpha_i^k) \\
\beta_0 & ::= & \alpha_n^k \\
\beta_{i+1} & ::= & (\LEFTcircle?\beta_i : \beta_i) \\
\gamma & ::= & \mathbb{F}X_0\beta_\ell
\end{array}
$$

A representation of (half) a formula in block form is given where the original formula is

$$\mathbb{F}X(\neg(A \wedge B) \rightarrow (A \wedge X))$$

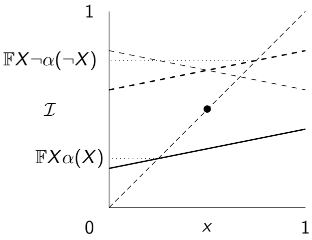


Transformations: Negation

We will give a brief overview of the transformations required to manipulate a formula into k-block normal form.

To move negations to occur only in the context $\bigcirc$, we note:

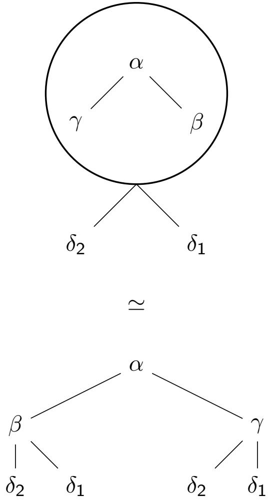
- $\neg \mathbb{F} X \alpha(X) \simeq \mathbb{F} X \neg \alpha(\neg X)$;
- $\neg(\alpha ? \beta : \gamma) \simeq (\alpha ? \neg \beta : \neg \gamma)$;
- $(\neg \alpha ? \beta : \gamma) \simeq (\alpha ? \gamma : \beta)$;
- $(\alpha ? \neg A_i : \beta) \simeq (\alpha ? (A_i ? \bigcirc : \bullet) : \beta)$.



Transformations: If-then-else

To ensure the internal branching nodes are only free variables or instances of ◐, we apply the following transformations:

- $((\alpha ? \beta_1 : \beta_2) ? \gamma_1 : \gamma_2) \simeq (\alpha ? (\beta_1 ? \gamma_1 : \gamma_2) : (\beta_2 ? \gamma_1 : \gamma_2))$, when β_1 and β_2 are not bound variables;
- $((\alpha ? X : \beta) ? \gamma_1 : \gamma_2) \simeq (\alpha ? X : (\beta ? \gamma_1 : \gamma_2))$;
- $((\alpha ? \beta : X) ? \gamma_1 : \gamma_2) \simeq (\alpha ? (\beta ? \gamma_1 : \gamma_2) : X)$;
- $(\mathbb{F} X \alpha(X) ? \beta : \gamma) \simeq \mathbb{F} X (\alpha ? \beta : \gamma)$, under the assumption that X does not appear free in β or γ (or is renamed to a fresh variable if it does).

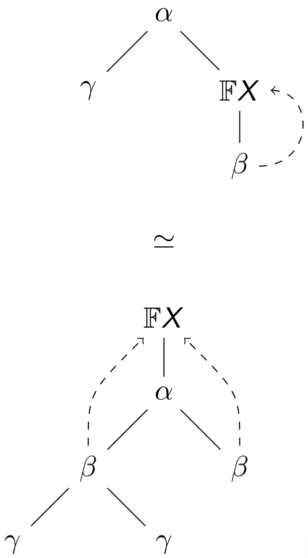


Transformations: Fixed Points

The next defect to address is fixed points appearing anywhere other than the root. To address this we apply the transformations:

- $(\alpha ? \mathbb{F}X\beta : \gamma) \Rightarrow \mathbb{F}X(\alpha ? \beta : \beta[\bullet, \circ\backslash\gamma])$;
- $(\alpha ? \beta : \mathbb{F}X\gamma) \Rightarrow \mathbb{F}X(\alpha ? \gamma[\bullet, \circ\backslash\beta] : \gamma)$.

The idea of this transformation is to move the fixed point operator to the root of the conditional statement, so that when X is encountered (i.e. α was heads, and the evaluation of β was X), the entire statement is re-evaluated from the root, but the γ branch is scaled by β.

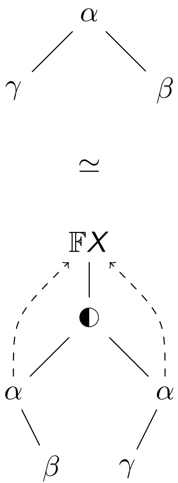


A Bernoulli Race

The next defect to address is, for all conditional statements, $(\alpha ? \beta : \gamma)$, either $\alpha = \LEFTcircle$ or one of β or γ is X_0. To do this, given $\alpha = A_i$ we can apply the following transformation:

$$(A_i ? \beta : \gamma) \Rightarrow \mathbb{F}X_0(\LEFTcircle ?(A_i ? X_0 : \gamma) : (A_i ? \beta : X_0)).$$

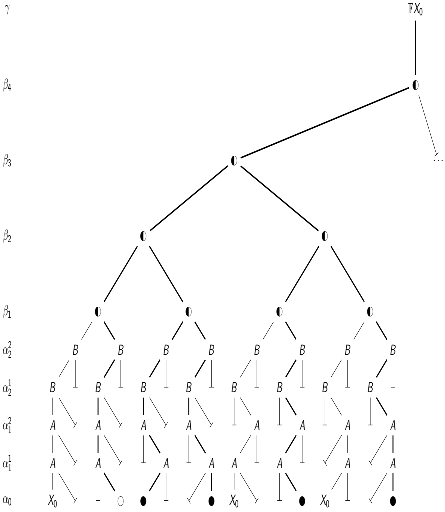
This transformation uses a fair coin flip and fixed point operators to turn the conditional statement into a series of independent tests.
A fair coin is flipped to see whether we test the case where A_i is heads or the case where A_i is tails, each case is tested and we re-evaluate if it fails.



k-Block Normal Form to Rational Functions

We can see that each formula in k-BNF corresponds to a fraction of homogeneous polynomials:

The sum of the products of paths leading to heads, divided by the sum of the products of paths leading to either heads or tails.



The Problem with Rational functions to k-BNF

The aim was to apply Pòlya's theorem to show that every rational function could be reduced to a form corresponding to k-BNF.

> *Given $f : [0,1]^{\mathcal{Y}} \longrightarrow (0,1)$, a homogeneous and positive polynomial, for sufficiently large n, all the coefficients of $(\sum_{y \in \mathcal{Y}} y)^n \cdot f(\mathcal{Y})$ are positive.*

However, the application of Pòlya's theorem to $f : (0,1)^{\mathcal{Y}} \longrightarrow (0,1)$ only holds up to 2 variables. This counter example from Renato Paes Leme shows that we cannot apply the theorem for more than three variables:

$$g(x,y,z) = 3xy^2 + y^3 - 3xyz + 3xz^2 + z \quad g : (0,1)^3 \longrightarrow (01)$$

Correspondence Theorem

- For every rational function $f : (0,1)^2 \longrightarrow (0,1)$ there is an aleatoric proposition α where $f_\alpha = f$.
- For every rational function $f : [0,1]^{\mathcal{Y}} \longrightarrow (0,1)$ there is an aleatoric proposition α where $f_\alpha = f$ over $(0,1)^{\mathcal{Y}}$.
- The set of aleatoric propositions corresponds to the set of functions $\ell(\mathcal{Y})/m(\mathcal{Y})$ where

$$\ell(\mathcal{Y}) = \sum_{a \in \rho^k_{\mathcal{Y}}} \ell_a \prod_{x \in \mathcal{Y}} x^{a_{(x,+)}} \cdot (1-x)^{a_{(x,-)}}$$
$$m(\mathcal{Y}) = \sum_{a \in \rho^k_{\mathcal{Y}}} m_a \prod_{x \in \mathcal{Y}} x^{a_{(x,+)}} \cdot (1-x)^{a_{(x,-)}}$$

such that
$\rho^k_{\mathcal{Y}} = \{\bar{a} \in \{0,\ldots,k\}^{\mathcal{Y}\times\{+,-\}} \mid \sum_{x\in\mathcal{Y}} a_{(x,+)} + a_{(x,-)} = k\}$,
and for all $a \in \rho^k_{\mathcal{Y}}$, ℓ_a and m_a are integers such that $\ell_a < m_a$.

Future work: Axioms and Complexity

We are investigating aleatoric propositions as a theoretical foundation for reasoning under uncertainty in AI.

To further support this foundation we are interested in being able to axiomatise intensional equivalence, and give complexity bounds for reasoning.

k-BNF may be very useful in both these cases.

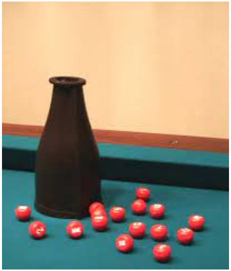
$$\alpha \simeq \alpha$$
$$(\alpha?\beta:\beta) \simeq \alpha$$
$$(\alpha?(\beta?\gamma_1:\gamma_2):(\beta?\delta_1:\delta_2))$$
$$\simeq (\beta?(\alpha?\gamma_1:\delta_1):(\alpha?\gamma_2:\delta_2))$$
$$((\alpha?\beta:\gamma)?\delta_1:\delta_2)$$
$$\simeq (\alpha?(\beta?\delta_1:\delta_2):(\gamma?\delta_1:\delta_2))$$
$$(\bullet?\alpha:\beta) \simeq \bullet$$
$$(\neg\alpha?\beta:\gamma) \simeq (\alpha?\gamma:\beta)$$
$$\mathbb{F}X\alpha(X) \simeq \alpha(\mathbb{F}X\alpha(X))$$

Future work: Reasoning about marbles

We are also very interested in first order extensions of aleatoric propositions, or *aleatoric predicates*.

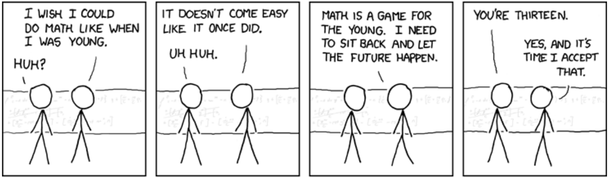
In this case the first order quantifiers (whether something exists) are replaced with expectation operators (whether something is likely).

The analogy or coin flips is replaced by the analogy of an urn of marbles (the domain), and rather then asking if there *is* some marble in the urn that satisfies a predicate, we ask *how likely it is* that we draw a marble satisfying a predicate.



Conclusion

- We have presented a syntax and semantics for aleatoric propositions, including a novel fixed point operator, and shown that they are able to generate a large and interesting subclass of rational functions.
- Aleatoric propositions capture a simple experiential logic, where biases are used to generate simulations.
- First order extensions will allow us to investigate how these biases are formed from observation, and applied in reasoning.



Thank you