Bluebell

An alliance of Relational Lifting and Independence for Probabilistic Reasoning

Emanuele D’Osualdo • Uni Konstanz
Jialu Bao • Cornell
Azadeh Farzan • Uni Toronto

[DRAFT ON ARXIV]
GOAL

Unify and generalize Proof principles for Unary & Relational Probabilistic Reasoning.

Long Term:

Build an "Iris Core Logic" for Probabilistic Reasoning.
We consider a simple programming language:

- Sequential & First Order
- Imperative with mutable variable store (no heap)
- Bounded Coops: everything terminates
- Normal assignments $x := e$
- Sampling assignments $x :: \mu$

\[ D(\text{Val}) = \text{Probability distribution over values} \]

**Big Step Semantics**

\[
\llbracket t \rrbracket : D(\text{Store}) \rightarrow D(\text{Store})
\]

Program term
We consider a simple programming language:

- Sequential & First Order
- Imperative with mutable variable store (no heap)
- Bounded Coops: everything terminates
- Normal assignments \( x := e \)
  - Sampling assignments \( x := \mu \)

\[ \Delta(\text{Val}) = \text{Probability distribution over values} \]

Simple? Yes, but already hard enough to keep us busy for a while!
REASONING STYLES

UNARY

- Goal involves one program t
- Example properties:
  - Output distribution of x is μ
  - Probability of x ≥ 10 is 1/2
  - Expected value of x is 1/3
  - By the end, m and c are probabilistically independent
    - m could be a plaintext message
    - c its cyphertext

RELATIONAL

- Goal involves two programs [1:t₁, 2:t₂]
- Example properties:
  - t₁ and t₂ induce the same distribution on x
  - t₂ could be an optimization of t₁
  - t₁ could be a cryptographic protocol
    and t₂ its idealized perfect version
  - Starting from similar input,
    t₁ and t₂ will produce “similar” distributions
    - differential privacy
// Encryption of 1 bit

k \sim \text{Ber}(\frac{1}{2}) \quad // \text{New random key (1 bit)}

m \sim \text{Ber}(\rho) \quad // \text{Message to encrypt (arbitrary bias } \rho) 

c := m \oplus k \quad // \text{Compute ciphertext}
// Encryption of 1 bit
k := Ber(\(\frac{1}{2}\))
m := Ber(p)
c := m \text{ xor } k
\{ c \sim Ber(\(\frac{1}{2}\)) \}

Reasoning (informally)

1. K and m are independent:
   \[ P(k=v, m=w) = P(k=v) \cdot P(m=w) \]

2. Conditioning on m:
   - if \(m=0\) then \(c = k\) so \(c \sim Ber(\frac{1}{2})\)
   - if \(m=1\) then \(c = \overline{k}\) so \(c \sim Ber(\frac{1}{2})\)

\[ \Rightarrow c \sim p \cdot Ber(\frac{1}{2}) + (1-p) \cdot Ber(\frac{1}{2}) \]
\[ = Ber(\frac{1}{2}) \]
// Encryption of 1 bit

k ≈ Ber(1/2)
m ≈ Ber(p)
c := m \oplus k

{ c \approx Ber(1/2) }^3
\land c \text{ and } m \text{ are independent!}

Reasoning (informally)

1. K and m are independent:
   \[ P(k=v, m=w) = P(k=v) \cdot P(m=w) \]

2. Conditioning on m:
   - if \( m=0 \) then \( c=k \) so \( c\sim Ber(1/2) \)
   - if \( m=1 \) then \( c=1k \) so \( c\sim Ber(1/2) \)

\[ \Rightarrow c \sim p \cdot Ber(1/2) + (1-p) \cdot Ber(1/2) = Ber(1/2) \]
UNARY EXAMPLE

// Encryption of 1 bit

k \sim \mathsf{Ber}(\frac{1}{2})

\{ k \sim \mathsf{Ber}(\frac{1}{2}) \} \\
m \sim \mathsf{Ber}(\rho)

\{ k \sim \mathsf{Ber}(\frac{1}{2}) \land m \sim \mathsf{Ber}(\rho) \} \\
c := m \ XOR \ k

\{ c \sim \mathsf{Ber}(\frac{1}{2}) \land m \sim \mathsf{Ber}(\rho) \}

Reasoning (informally):

(1) k and m are independent:

\[ P(k=\nu, m=\omega) = P(k=\nu) \cdot P(m=\omega) \]

(2) Conditioning on m:

- if \( m=0 \) then \( c = k \) so \( c \sim \mathsf{Ber}(\frac{1}{2}) \)
- if \( m=1 \) then \( c = 1-k \) so \( c \sim \mathsf{Ber}(\frac{1}{2}) \)

\Rightarrow c \sim \rho \cdot \mathsf{Ber}(\frac{1}{2}) + (1-\rho) \mathsf{Ber}(\frac{1}{2})

= \mathsf{Ber}(\frac{1}{2})

IDEA 1: Separation = Independence [PSL] [LILAC]
Unary Example

// Encryption of 1 bit

k \sim \text{Ber}(\frac{1}{2})

\{ k \sim \text{Ber}(\frac{1}{2}) \}

m \sim \text{Ber}(p)

\{ m \sim \text{Ber}(p) \}

c := m \oplus k

\{ c \sim \text{Ber}(\frac{1}{2}) \}

Reasoning (informally)

(2) Conditioning on m:

\[
\begin{align*}
\mathcal{C} &
\left( k \sim \text{Ber}(\frac{1}{2}) \right) \* \left\{ \begin{array}{ll}
\Gamma c = k7 & \text{if } v = 0 \\
\Gamma c = 7k & \text{if } v = 1
\end{array} \right.
\end{align*}
\]

Deterministic value

Predicate over stores holds with probability 1

\textbf{IDEA(1)}: Separation = Independence \text{ } [\text{PSL}] [\text{LILAC}]

\textbf{IDEA(2)}: Conditioning via a modality \text{ } [\text{LILAC}]

\text{case analysis}
REASONING TOOLS

• UNARY TRIPLES: $\{p \downarrow \} + \{q \uparrow \}$  Assertions over $\text{ID}(\text{Store})$

• PROBABILISTIC INDEPENDENCE: Separation *

• CONDITIONING: via a modality $\Box_v$
RELATIONAL REASONING

1: \( x \sim \mu \)
   \( d \sim \text{unif}(-1,1) \)
   \( y := x - d \)

2: \( x \sim \mu \)
   \( d \sim \text{unif}(-1,1) \)
   \( y := x + d \)

GOAL: \( y^{(1)} \) is distributed like \( y^{(2)} \)

UNARY PROOF STRATEGY: Characterize the exact distribution of \( y \)
in the two programs, then compare.

\( \Rightarrow \) Can be prohibitively hard to do!

RELATIONAL STRATEGY: Execute programs in lockstep showing that
whatever the steps might be computing, the two sides remain the same.
RELATIONAL REASONING

1: \( x \sim \mu \)
   \( d \sim \text{unif}(-1,1) \)
   \( y := x - d \)

2: \( x \sim \mu \)
   \( d \sim \text{unif}(-1,1) \)
   \( y := x + d \)

A world of pure imagination

GOAL: \( y^{(1)} \) is distributed like \( y^{(2)} \)
RELATIONAL REASONING

1: \( x := a \)
\( d \sim \text{unif}(-1,1) \)
\( y := x - d \)

2: \( x := a \)
\( d \sim \text{unif}(-1,1) \)
\( y := x + d \)

"coupling"

GOAL: \( y^{(1)} \) is distributed like \( y^{(2)} \)
RELATIONAL REASONING

1: \( x := a \)
   \( d := b \)
   \( y := x - d \)

2: \( x := a \)
   \( d := -b \)
   \( y := x + d \)

Goal: \( y^{(1)} \) is distributed like \( y^{(2)} \)
1: \[ x \sim \mathcal{N} \]
\[ d \sim \text{unif}(-1,1) \]
\[ y := x - d \]

2: \[ x \sim \mathcal{N} \]
\[ d \sim \text{unif}(-1,1) \]
\[ y := x + d \]

\[ [pRHL] \]

Relation over Store x Store
Holding with probability 1
in some “fictional” joint distribution
RELATIONAL REASONING

1: \[ x \sim \mu \]
   \[ d \sim \text{unif}(\text{-}1,1) \]
   \[ y := x - d \]

2: \[ x \sim \mu \]
   \[ d \sim \text{unif}(\text{-}1,1) \]
   \[ y := x + d \]

\[ _{\text{[PRHL]}} \]

Relation over Store \( x \) Store
Holding with probability 1
in some "fictional" joint distribution

= Relational lifting \( LR \)
1: $x \sim \mu$
   \[x^{<1>} = x^{<2>}\]
   $d \sim \text{unif}(-1,1)$
   \[d^{<1>} = -d^{<2>}\]
   \[y^{<1>} = y^{<2>}\]

2: $x \sim \mu$
   \[x^{<1>} = x^{<2>}\]
   $d \sim \text{unif}(-1,1)$
   \[d^{<1>} = -d^{<2>}\]
   \[y^{<1>} = y^{<2>}\]

\[\text{Relation over Store} \times \text{Store}\]
\[\text{Holding with probability 1 in some "fictional" joint distribution}\]

\[\text{Relational Lifting}\ LR\]

**Fundamental Theorem of Relational Lifting:** (Meta)

If $[y^{<1>} = y^{<2>}]$ then $y^{<1>}$ is distributed like $y^{<2>}$
1: \( x \sim \mu \)
\[
\frac{d \sim \text{unif}(-1,1)}{y := x - d}
\]

2: \( d \sim \text{unif}(-1,1) \)
\[
\frac{x \sim \mu}{y := x + d}
\]

Only asserting via Relational lifting is too limiting!

**Goal**: Improve expressivity while retaining the relational “spirit”
REASONING TOOLS

- UNARY TRIPLES: $\{p\} t \{q\}$ Assertions over $\text{ID}($Store$)$

- PROBABILISTIC INDEPENDENCE: Separation $

- CONDITIONING: via a modality $\triangleright$
REASONING TOOLS

- **UNARY TRIPLES:** $\{ p \} t \{ q \}$  
  Assertions over $\mathcal{ID}(\text{Store})$

- **PROBABILISTIC INDEPENDENCE:** Separation $\star$

- **CONDITIONING:** via a modality $\boxtimes$

- **RELATIONAL TRIPLES:** $ LR_1 \upharpoonright [1:t_1, 2:t_2] LR_2 \downharpoonright $  
  Relations over $\text{Store}$
  $R_1, R_2 \subseteq \text{Store} \times \text{Store}$
REASONING TOOLS

• UNARY TRIPLES: $\{ p \} \vdash \{ q \}$  Assertions over $ID(\text{Store})$

• PROBABILISTIC INDEPENDENCE: Separation *

• CONDITIONING: via a modality $\boxtimes$

• RELATIONAL TRIPLES: $LR_1 \vdash [1: t_1, 2: t_2] LR_2$  Relations over Store $R_1, R_2 \subseteq \text{Store} \times \text{Store}$

• RELATIONAL LIFTING: $LR$
**REASONING TOOLS**

- **UNARY TRIPLES**: \{P\} \& \{Q\}  \quad \text{Assertions over ID(Store)}
- **PROBABILISTIC INDEPENDENCE**: Separation \star
- **CONDITIONING**: via a modality \&_{x\in v}
- **RELATIONAL TRIPLES**: \([R_1 \setminus [1:t_1,2:t_2] \& R_2]\)  \quad \text{Relations over Store}
- **RELATIONAL LIFTING**: \([R]\)

Can we unify and generalize?

(spoiler: YES)
**First observation**  We can harmonize all these features by:

- **Using** \( \text{Assrt} := \text{ID}(\text{Store}) \times \text{ID}(\text{Store}) \rightarrow \text{Prop} \)
  - Unary assertions just ignore one of the two distributions \( x \sim \text{Ber} (\frac{1}{2}) \)
  - Relational lifting as a construct
    \[ R \subseteq \text{Store} \times \text{Store} \Rightarrow LR : \text{ID}(\text{Store}) \times \text{ID}(\text{Store}) \rightarrow \text{Prop} \]

- **Multi-ary wp from [LHC]** : \( \text{wp} \uparrow \{ Q \} \) \( \xrightarrow{\text{partial map Indices \rightarrow Terms}} \)

  \[ \text{wp} [1: t_1, 2: t_2] \{ Q \} \equiv \text{wp} [1: t_1] \{ \text{wp} [2: t_2] \{ Q \} \} \]

Can have unary triples, binary triples, switch back & forth.
1: \( x \sim \mu \)
\( d \sim \text{unif}(-1,1) \)
\( y := x - d \)

2: \( d \sim \text{unif}(-1,1) \)
\( x \sim \mu \)
\( y := x + d \)
SMALL EXAMPLE

1: \( x \sim \mu \)  
   \( d \sim \text{unif}(-1,1) \)

2: \( d \sim \text{unif}(-1,1) \)  
   \( x \sim \mu \)

\[ \exists x, x', x'' \mid x < 1 > \sim \lambda \ast x < 2 > \sim \mu \ast d < 1 > \sim \text{unif}(-1,1) \ast d < 2 > \sim \text{unif}(-1,1) \]

\[ \exists y < 1 > = x < 2 > \land d < 1 > = -d < 2 > \]

\[ y := x - d \]

\[ y := x + d \]

\[ y < 1 > = y < 2 > \]

Questions:

1) Can entailment \( \otimes \) be proven in the logic?

2) Are there useful interactions between \( \ast \), \( \otimes \) and LRL?
BLUEBELL'S KEY INSIGHT

Questions:

1) Can entailment $\otimes$ be proven in the logic?

2) Are there useful interactions between $*$, $\mathcal{C}$ and $\mathcal{LRI}$?
**BLUEBELL'S KEY INSIGHT**

**Questions:**
1) Can entailment be proven in the logic?
2) Are there useful interactions between $\star$, $\mathcal{C}$ and $\mathcal{LR}$?

**Bluebell says YES!**
Bluebell's Key Insight

Questions:
1) Can entailment be proven in the logic?
2) Are there useful interactions between $\ast$, $\mathcal{C}$ and $\mathcal{LR}$?

Bluebell says YES!

Their definitions and laws can be derived

Rich set of core laws
RELATIONAL LIFTING AS CONDITIONING

Usual picture:
\[
\text{DB(Store)} \times \text{DB(Store)} \Rightarrow \text{LR1}
\]
\[
\text{Lift} \uparrow
\]
\[
\text{Store} \times \text{Store} \Rightarrow R
\]

Bluebell's view:
\[
\text{DB(Store)} \times \text{DB(Store)} \Rightarrow \text{LR1}
\]
\[
\text{Conditioning} \downarrow
\]
\[
\text{Store} \times \text{Store} \Rightarrow R
\]

Analog of
\[
\text{DB(Store)} \Rightarrow \text{IP(A) = 1}
\]
\[
\text{Lift} \uparrow
\]
\[
\text{Store} \Rightarrow A
\]

If you condition jointly on the two distributions, you get a pair of stores satisfying R.

So, what is "joint conditioning"?
**Joint Conditioning**

**Def.** Given \( \mu : \text{D(A)} \) and \( k : A \to \text{D(Store)} \) define

\[
\text{bind}(\mu, k) := \lambda s. \sum_{a \in A} \mu(a) k(a)(s)
\]

**Example.** \( A = \{0, 1\} \), \( \mu = \text{Ber}(\frac{1}{3}) \)

\[
\text{bind}(\mu, k) = \frac{1}{3} k(0) + \frac{2}{3} k(1)
\]

[This is actually the bind of the monad \( \text{D}(. \cdot) \) !]
**Joint Conditioning**

**Def**: Given \( \mu : D(A) \) and \( P : A \rightarrow \text{Ass} \)

define \( C_A \text{ v. } P(v) : \text{Ass} \) by

\[
(\mu_1, \mu_2) \vdash C_A \text{ v. } P(v) \quad \text{iff} \quad \left\{ \begin{array}{l}
\exists K_1, K_2 : A \rightarrow D(\text{store}). \\
\mu_1 = \text{bind}(\mu_1, K_1) \land \\
\mu_2 = \text{bind}(\mu_1, K_2) \land \\
\forall a \in \text{supp}(\mu). \\
(K_1(a), k_2(a)) \vdash P(a) \end{array} \right. 
\]
JOINT CONDITIONING

\[
\left( \langle \mu_1, \mu_2 \rangle \vdash C \_{\mu} . P(\nu) \right) \quad \text{iff} \quad \begin{cases}
\exists K_1, K_2 : A \rightarrow \mathcal{D}(\text{store}) . \\
M_1 = \text{bind}(\mu_1, K_1) \land \\
M_2 = \text{bind}(\mu_1, K_2) \land \\
\forall \alpha \in \text{supp}(\mu) . \\
(\mu_1(\alpha), \mu_2(\alpha)) \vdash P(\alpha)
\end{cases}
\]
JOINT CONDITIONING

\[(\frac{\lambda_1}{\lambda_2}) \models C_{\lambda} \forall \cdot P(v) \quad \text{iff} \quad \begin{cases} \exists K_1, K_2 : A \rightarrow \mathbb{D} (\text{Store}) . \\ M_1 = \text{bind}(\lambda_1, K_1) \land \\ M_2 = \text{bind}(\lambda_2, K_2) \land \\ \forall x \in \text{supp}(\lambda_1) . \\ (K_1(x), K_2(x)) \models P(x) \end{cases} \]

Example: \( A = \{0, 1, 2\} \quad M = \text{Ber}(\frac{1}{3}) \)

\[
\begin{align*}
M_1 &= \frac{1}{3} K_1(0) + \frac{2}{3} K_1(1) \\
M_2 &= \frac{1}{3} K_2(0) + \frac{2}{3} K_2(1) \\
\end{align*}
\]

\[
\begin{array}{c}
P(0) \\
P(1) \\
\end{array}
\]

\[
\begin{array}{c}
\{x | x \in \text{Ber}(\frac{1}{3}) \} \\
\Gamma_{x \in \{0\}} = 0 \quad \Gamma_{x \in \{1\}} = 1 \\
\end{array}
\]
JOINT CONDITIONING

\[
\left( \frac{\mu_1}{\mu_2} \right) \vdash C_\mu \cup P(\nu) \quad \text{iff} \quad \exists K_1, K_2 : A \rightarrow D(\text{Store}) .
\]

\[
\begin{align*}
M_1 &= \text{bind}(\mu_1, K_1) \\
M_2 &= \text{bind}(\mu_1, K_2) \\
\forall a \in \text{supp}(\mu). \\
(\nu(a), k_2(a)) &\vdash P(a)
\end{align*}
\]

Example: \( A = \{0, 1, \frac{2}{3} \} \) \( M = \text{Ber}(\frac{2}{3}) \)

\[
\begin{align*}
M_1 &= \frac{1}{3} K_1(0) + \frac{2}{3} K_1(1) \\
M_2 &= \frac{1}{3} K_2(0) + \frac{2}{3} K_2(1) \\
\text{P(0)} &\quad \text{P(1)}
\end{align*}
\]

\[
C_\mu \cup (\Gamma x \leftarrow \nu \vdash P(\nu))
\]

\[
\begin{align*}
\text{[C-unit-R]} \\
x \leftarrow \nu \vdash C_\mu \cup (\Gamma x \leftarrow \nu \\
\text{This reflects the right unit law of} \\
\text{the underlying monoid!}
\end{align*}
\]
Encoding Lifting as Conditioning

Unary Conditioning: \( C_{\mu,v}([x=v] \cdot p(v)) \)

Relational Lifting:

\[
LR(x\langle 1 \rangle, x\langle 2 \rangle) := \exists M : D(Val \times Val). C_M(u_1, u_2). (\Gamma x\langle 1 \rangle = u_1) \cdot (\Gamma x\langle 2 \rangle = u_2) \cdot R(u_1, u_2)
\]
JOINT COND. RULES

[C-UNIT-R]
\[
\x\langle i \rangle \sim m \vdash \text{C}_m \text{v}. [\x\langle i \rangle = v]
\]

[C-FRAME]
\[
P \ast \text{C}_m \text{v}. Q(v) \vdash \text{C}_m \text{v}. (P \ast Q(v))
\]

[C-CONS]
\[
\forall u \in \text{supp}(m). \ P(u) \vdash P'(u)
\]
\[
\text{C}_m \text{v}. P(u) \vdash \text{C}_m \text{v}. P'(u)
\]
\[
\x\langle i \rangle \sim m \ast \ y\langle i \rangle \sim m' \ast \ [z = x + y]
\]
\[
\vdash (\text{C}_m \text{v}. [\x\langle i \rangle = v]) \ast \ y\langle i \rangle \sim m' \ast \ [z = x + y]
\]
\[
\vdash \text{C}_m \text{v}. ([\x\langle i \rangle = v] \ast \ [y\langle i \rangle = v'] \ast \ [z = x + y])
\]
\[
\vdash \text{C}_m \text{v}. (\text{C}_m \text{v}.1. ([\x\langle i \rangle = v] \ast \ [y\langle i \rangle = v'] \ast \ [z = x + y]))
\]
\[
\vdash \text{C}_m \text{v}. \text{C}_m \text{v}.1. \ [\x\langle i \rangle = v \land y\langle i \rangle = v' \land z = x + y]
\]
\[ \text{[c-assoc]} \]

\[ C_\mu v . C_\kappa(v') P(v, v') \vdash C_{\text{bind'}}(\mu, \kappa) . P(v, v') \]

\[ \text{bind'}(\mu, \kappa) = \text{do } v \leftarrow \mu ; v' \leftarrow \kappa(v) ; \text{return } (v, v') \]

\[ \text{[c-unassoc]} \]

\[ C_{\text{bind}(\mu, \kappa)} v' P(v') \vdash C_\mu v . C_\kappa(v') P(v') \]
Some Derivable Rules

\[ C \vee LR \models LR \] (Convexity of Rel Lifting)

\[ LR_1 \ast LR_2 \models LR_1 \land LR_2 \]

Note:

\[ LR_1 \land LR_2 \nRightarrow LR_1 \land LR_2 \]
CHALLENGES

- Generalization to Iris-style user-defined ghost resources
- \([c \text{-wp-swAP}]\)

\[ \text{ownVars} \land \mathcal{C}_{m,v} \text{ wp t } \{ \mathbf{Q}(v) \} \models \text{ wp t } \{ \mathcal{C}_{m,v} . \mathbf{Q}(v) \} \]

↑ Bluebell needs this for soundness

OPEN QUESTION: Can we find a model that validates the rule without ownVars?
Thanks