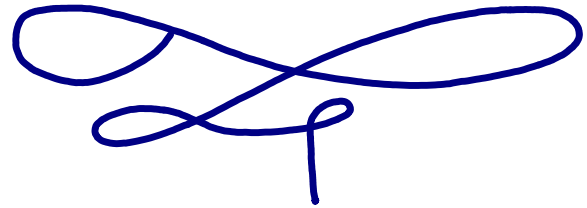
Bluebell



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]

THE JOINT CONDITIONING MODALITY

GOAL

Unify and generalize Proof principles for Unary & Relational Probabilistic Reasoning

Long Term:

Build an "Iris Core Logic" for Probabilistic Reasoning

PROBABILISTIC PROGRAMS

We consider a simple programming language:

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

  Sampling assignments $x :\approx \mu$

  μ ← $\mathbb{D}(Val)$ = Probability distribution over values

BIG STEP SEMANTICS

$$[\![t]\!] : \mathbb{D}(Store) \to \mathbb{D}(Store)$$

t ↑ Program term

PROBABILISTIC PROGRAMS

We consider a simple programming language:

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

  Sampling assignments $x :\approx \mu$

  μ ← $\mathbb{D}(Val)$ = 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 $x \geq 10$ 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 $[1:t_1, 2:t_2]$
- Example properties:
  - t_1 and t_2 induce the same distribution on x

    ↳ t_2 could be an optimization of t_1

    ↳ t_1 could be a cryptographic protocol and t_2 its idealized perfect version
  - Starting from similar input, t_1 and t_2 will produce "similar" distributions

    ↳ differential privacy

UNARY EXAMPLE

// Encryption of 1 bit

$k :\approx \mathrm{Ber}(1/2)$ // New random key (1 bit)

$m :\approx \mathrm{Ber}(p)$ // Message to encrypt (arbitrary bias p)

$c := m \text{ XOR } k$ // Compute cyphertext

UNARY EXAMPLE

// Encryption of 1 bit

$k :\approx \text{Ber}(1/2)$

$m :\approx \text{Ber}(p)$

$c := m \text{ XOR } k$

$\{c \sim \text{Ber}(1/2)\}$

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 $c \sim \text{Ber}(1/2)$
- if $m=1$ then $c=\neg k$ so $c \sim \text{Ber}(1/2)$

$$\Rightarrow c \sim p \cdot \text{Ber}(1/2) + (1-p)\,\text{Ber}(1/2)$$
$$= \text{Ber}(1/2)$$

UNARY EXAMPLE

// Encryption of 1 bit

$k :\approx \mathrm{Ber}(1/2)$

$m :\approx \mathrm{Ber}(p)$

$c := m \text{ XOR } k$

$\{c \sim \mathrm{Ber}(1/2)\}$

$\wedge$ c and m 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 $c \sim \mathrm{Ber}(1/2)$
- if $m=1$ then $c=\neg k$ so $c \sim \mathrm{Ber}(1/2)$

$$\Rightarrow c \sim p \cdot \mathrm{Ber}(1/2) + (1-p)\,\mathrm{Ber}(1/2)$$
$$= \mathrm{Ber}(1/2)$$

UNARY EXAMPLE

// Encryption of 1 bit

$k :\approx \mathrm{Ber}(1/2)$

$\{k \sim \mathrm{Ber}(1/2)\}$

$m :\approx \mathrm{Ber}(p)$

$\{k \sim \mathrm{Ber}(1/2) * m \sim \mathrm{Ber}(p)\}$

$c := m \text{ xor } k$

$\{c \sim \mathrm{Ber}(1/2) * m \sim \mathrm{Ber}(p)\}$

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 $c \sim \mathrm{Ber}(1/2)$
- if $m=1$ then $c=\neg k$ so $c \sim \mathrm{Ber}(1/2)$

$\Rightarrow c \sim p \cdot \mathrm{Ber}(1/2) + (1-p)\,\mathrm{Ber}(1/2)$

$= \mathrm{Ber}(1/2)$

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

UNARY EXAMPLE

// Encryption of 1 bit

$k :\approx \mathrm{Ber}(1/2)$

$\{k \sim \mathrm{Ber}(1/2)\}$

$m :\approx \mathrm{Ber}(p)$

$\{k \sim \mathrm{Ber}(1/2) * m \sim \mathrm{Ber}(p)\}$

$c := m \text{ xor } k$

$\{c \sim \mathrm{Ber}(1/2) * m \sim \mathrm{Ber}(p)\}$

Reasoning (informally)

② Conditioning on m:

case analysis

$$\mathbb{C}_{m \to v}\left(k \sim \mathrm{Ber}(1/2) * \begin{cases} \lceil c = k \rceil & \text{if } v = 0 \\ \lceil c = \neg k \rceil & \text{if } v = 1 \end{cases}\right)$$

v: Deterministic value

$\lceil c = \neg k \rceil$: Predicate over stores holds with probability 1

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

IDEA ② : Conditioning via a modality [LILAC]

REASONING TOOLS

- UNARY TRIPLES: $\{P\}\ t\ \{Q\}$ Assertions over $\mathbb{D}(\text{Store})$
- PROBABILISTIC INDEPENDENCE: Separation $*$
- CONDITIONING: via a modality $\mathbb{C}_{X \to v}$

RELATIONAL REASONING

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

GOAL: $y\langle 1\rangle$ is distributed like $y\langle 2\rangle$

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

↳ Can be prohibitively hard to do!

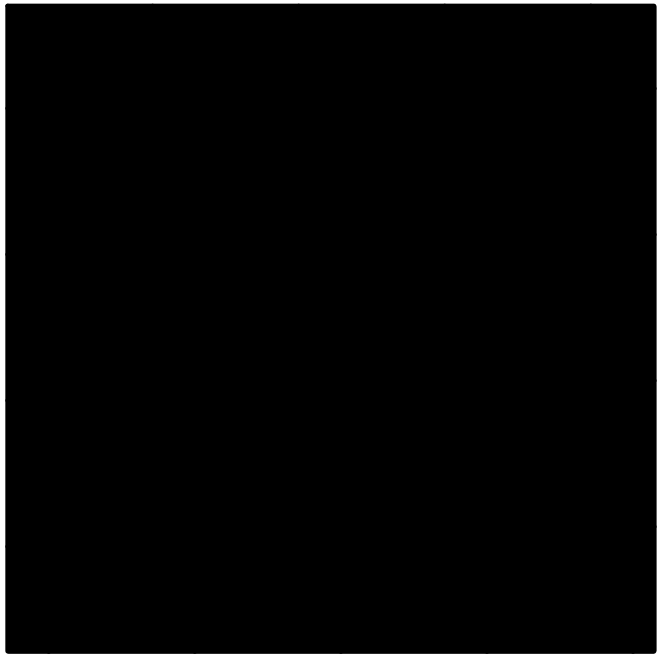
RELATIONAL STRATEGY: Execute programs in lockstep showing that whatever the steps might be computing, the two sides

RELATIONAL REASONING

1: $x :\approx \mu$

$d :\approx \text{unif}(-1,1)$

$y := x - d$



A world of pure imagination

2: $x :\approx \mu$

$d :\approx \text{unif}(-1,1)$

$y := x + d$

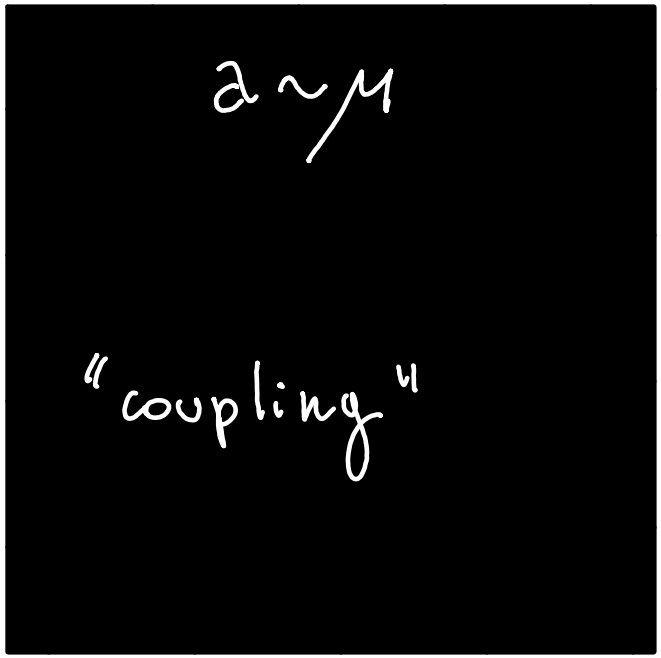
GOAL: $y\langle 1 \rangle$ is distributed like $y\langle 2 \rangle$

RELATIONAL REASONING

1: $x := a$

$d :\approx \text{unif}(-1,1)$

$y := x - d$



2: $x := a$

$d :\approx \text{unif}(-1,1)$

$y := x + d$

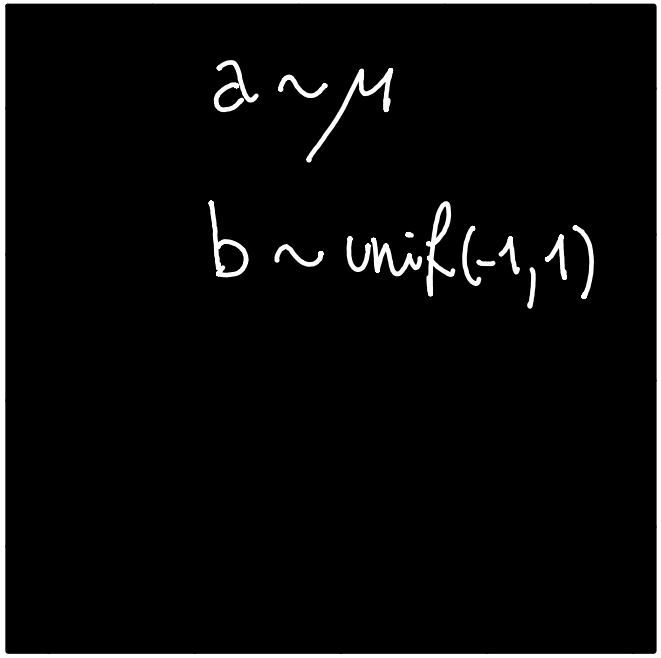
GOAL: $y\langle 1 \rangle$ is distributed like $y\langle 2 \rangle$

RELATIONAL REASONING

1: $x := a$

$d := b$

$y := x - d$



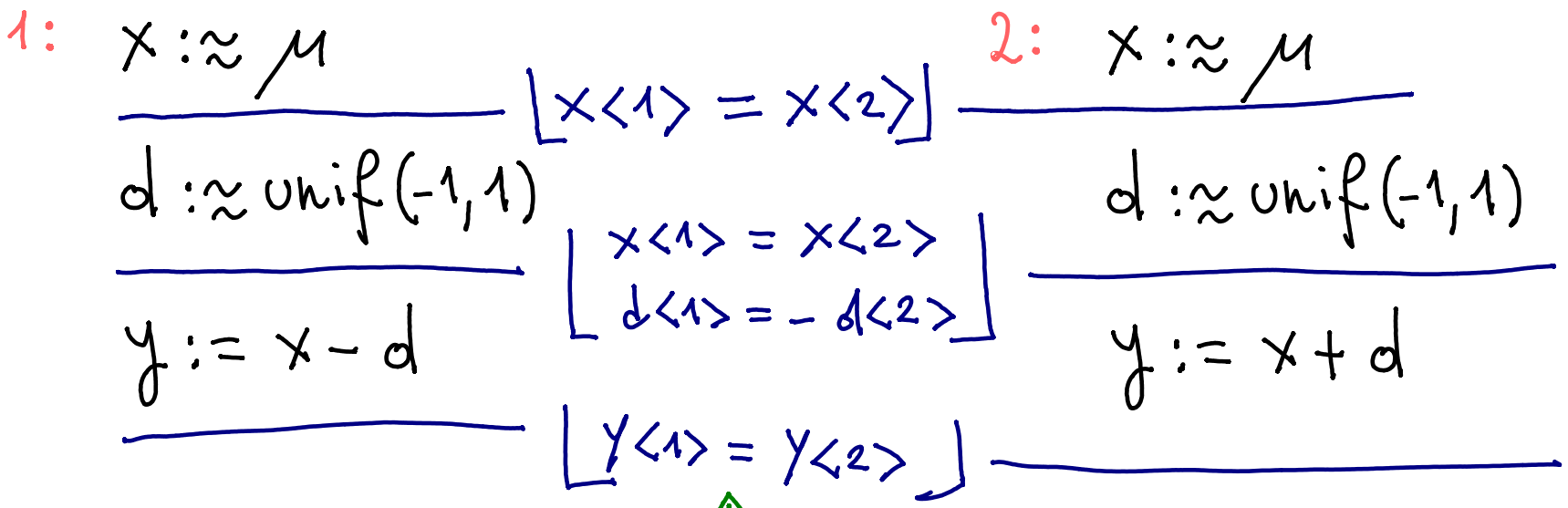
2: $x := a$

$d := -b$

$y := x + d$

GOAL: $y\langle 1 \rangle$ is distributed like $y\langle 2 \rangle$

RELATIONAL REASONING



[pRHL]

Relation over Store × Store
Holding with probability 1
in some "fictional" joint
distribution

RELATIONAL REASONING

$$\frac{\begin{array}{c} 1: \quad x :\approx \mu \\ \hline d :\approx \mathrm{unif}(-1,1) \\ \hline y := x - d \end{array} \qquad \lfloor x\langle 1\rangle = x\langle 2\rangle \rfloor \qquad \begin{array}{c} 2: \quad x :\approx \mu \\ \hline d :\approx \mathrm{unif}(-1,1) \\ \hline y := x + d \end{array}}{\lfloor \begin{array}{l} x\langle 1\rangle = x\langle 2\rangle \\ d\langle 1\rangle = -\, d\langle 2\rangle \end{array} \rfloor}$$

$$\frac{}{\lfloor y\langle 1\rangle = y\langle 2\rangle \rfloor}$$

↑ Relation over Store × Store

Holding with probability 1 in some "fictional" joint distribution

[pRHL]

= Relational lifting $\lfloor R \rfloor$

RELATIONAL REASONING

$$\dfrac{\dfrac{x :\approx \mu}{d :\approx \mathrm{unif}(-1,1)}}{y := x - d} \quad \lfloor x\langle 1\rangle = x\langle 2\rangle \rfloor \quad \dfrac{\dfrac{x :\approx \mu}{d :\approx \mathrm{unif}(-1,1)}}{y := x + d}$$

$$\lfloor x\langle 1\rangle = x\langle 2\rangle \;\wedge\; d\langle 1\rangle = -d\langle 2\rangle \rfloor$$

$$\lfloor y\langle 1\rangle = y\langle 2\rangle \rfloor$$

[pRHL]

↑ Relation over Store × Store

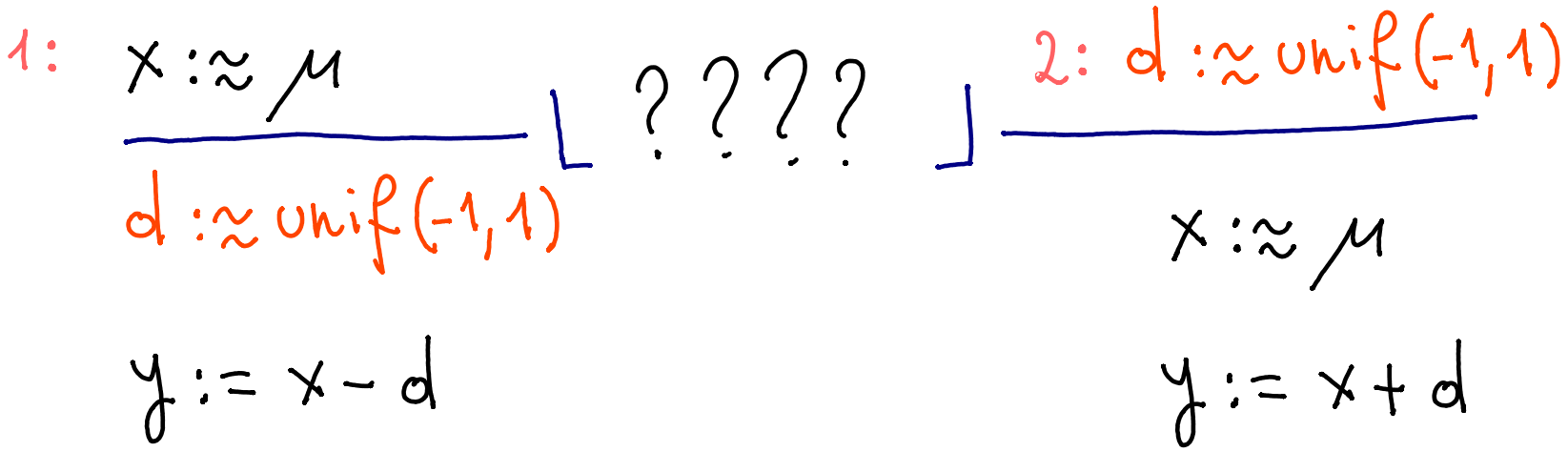
Holding with probability 1 in some "fictional" joint distribution

= Relational lifting $\lfloor R \rfloor$

FUNDAMENTAL THEOREM OF RELATIONAL LIFTING: (Meta)

If $\lfloor y\langle 1\rangle = y\langle 2\rangle \rfloor$ then $y\langle 1\rangle$ is distributed like $y\langle 2\rangle$

RELATIONAL REASONING (LIMITATIONS)



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 $\mathbb{D}$(Store)
- PROBABILISTIC INDEPENDENCE: Separation $*$
- CONDITIONING: via a modality $\mathbb{C}_{X \to v}$

REASONING TOOLS

- UNARY TRIPLES: $\{P\}\ t\ \{Q\}$ — Assertions over $\mathbb{D}(\text{Store})$
- PROBABILISTIC INDEPENDENCE: Separation $*$
- CONDITIONING: via a modality $\mathbb{C}_{X \to v}$
- RELATIONAL TRIPLES: $\lfloor R_1 \rfloor\ [1{:}t_1, 2{:}t_2]\ \lfloor R_2 \rfloor$ — Relations over Store $R_1, R_2 \subseteq \text{Store} \times \text{Store}$

REASONING TOOLS

- UNARY TRIPLES: $\{P\}\ t\ \{Q\}$ — Assertions over $\mathbb{D}(\text{Store})$
- PROBABILISTIC INDEPENDENCE: Separation $*$
- CONDITIONING: via a modality $\mathbb{C}_{X \to v}$
- RELATIONAL TRIPLES: $\lfloor R_1 \rfloor\ [1{:}t_1, 2{:}t_2]\ \lfloor R_2 \rfloor$ — Relations over Store, $R_1, R_2 \subseteq \text{Store} \times \text{Store}$
- RELATIONAL LIFTING: $\lfloor R \rfloor$

REASONING TOOLS

- UNARY TRIPLES: $\{P\}\ t\ \{Q\}$ — Assertions over $\mathbb{D}(\text{Store})$
- PROBABILISTIC INDEPENDENCE: Separation $*$
- CONDITIONING: via a modality $\mathbb{C}_{X \to v}$
- RELATIONAL TRIPLES: $\lfloor R_1 \rfloor\ [1{:}t_1, 2{:}t_2]\ \lfloor R_2 \rfloor$ — Relations over Store $R_1, R_2 \subseteq \text{Store} \times \text{Store}$
- RELATIONAL LIFTING: $\lfloor R \rfloor$

Can we unify and generalize?

(spoiler: YES)

BLUEBELL

First observation We can harmonize all these features by:

- Using $\mathsf{Assrt} := \mathbb{D}(\mathsf{Store}) \times \mathbb{D}(\mathsf{Store}) \to \mathsf{Prop}$
  - Unary assertions just ignore one of the two distributions $x\langle 1\rangle \sim \mathrm{Ber}(1/2)$
  - Relational lifting as a construct

$$R \subseteq \mathsf{Store} \times \mathsf{Store} \Rightarrow \lfloor R \rfloor : \mathbb{D}(\mathsf{Store}) \times \mathbb{D}(\mathsf{Store}) \to \mathsf{Prop}$$

- Multi-ary wp from [LHC]: $\mathsf{wp}\ t\ \{Q\}$ — t: partial map Indices → Terms

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

Can have unary triples, binary triples, switch back & forth.

SMALL EXAMPLE

1: $x :\approx \mu$

$d :\approx \text{unif}(-1,1)$

$y := x - d$

2: $d :\approx \text{unif}(-1,1)$

$x :\approx \mu$

$y := x + d$

SMALL EXAMPLE

1: $x :\approx \mu$; $d :\approx \mathrm{unif}(-1,1)$

2: $d :\approx \mathrm{unif}(-1,1)$; $x :\approx \mu$

$$\{ x\langle 1\rangle \sim \mu * x\langle 2\rangle \sim \mu * d\langle 1\rangle \sim \mathrm{unif}(-1,1) * d\langle 2\rangle \sim \mathrm{unif}(-1,1) \}$$
$$\{ \lfloor x\langle 1\rangle = x\langle 2\rangle \wedge d\langle 1\rangle = -d\langle 2\rangle \rfloor \} \circledast$$

1: $y := x - d$; 2: $y := x + d$

$$\{ \lfloor y\langle 1\rangle = y\langle 2\rangle \rfloor \}$$

QUESTIONS:

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

2) Are there useful interactions between $*$, $\mathbb{C}$ and $\lfloor R \rfloor$?

BLUEBELL'S KEY INSIGHT

QUESTIONS:

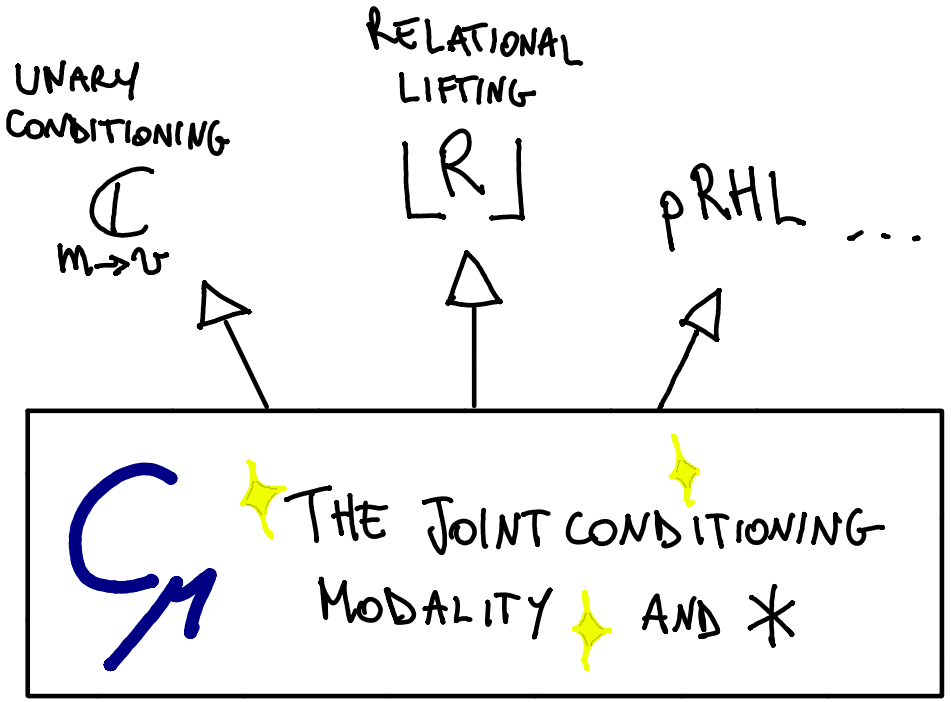
1) Can entailment $\circledast$ be proven in the logic?
2) Are there useful interactions between $*$, $\mathbb{I}$ and $\lfloor R \rfloor$?

BLUEBELL'S KEY INSIGHT

QUESTIONS:

1) Can entailment $\circledast$ be proven in the logic?
2) Are there useful interactions between $*$, $\mathbb{C}$ and $\lfloor R \rfloor$?

Bluebell says YES!

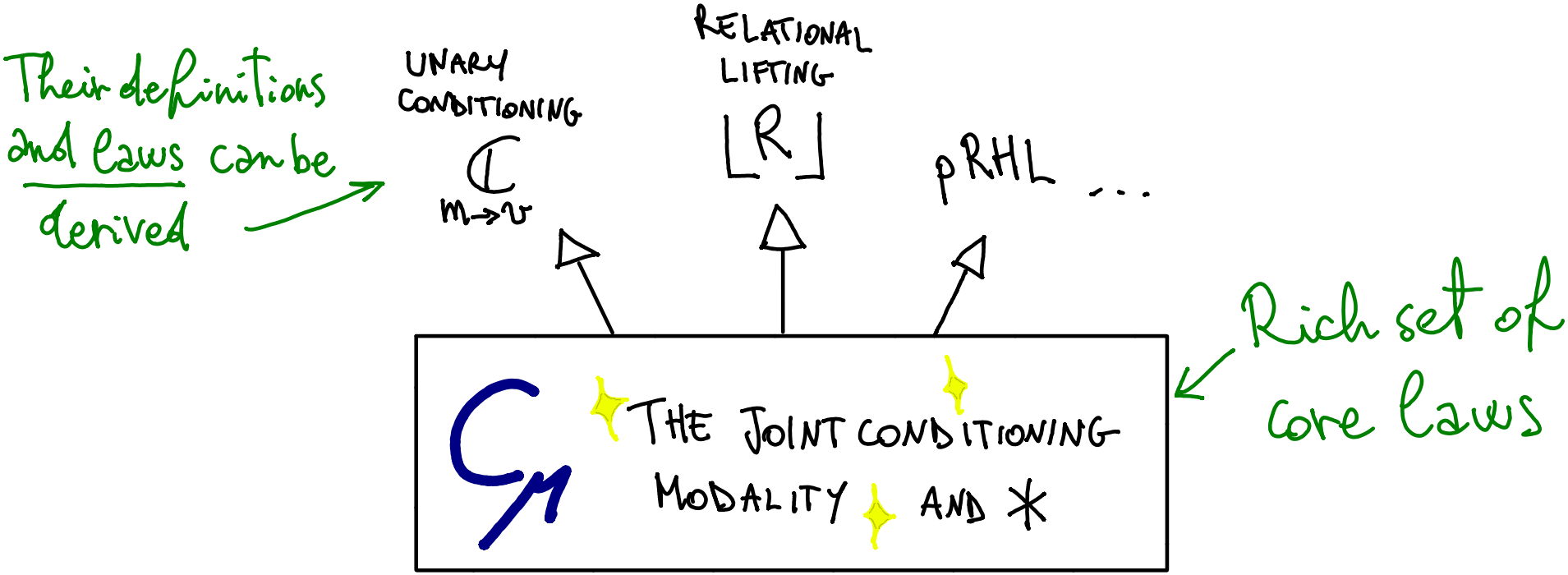


BLUEBELL'S KEY INSIGHT

QUESTIONS:

1) Can entailment $\circledast$ be proven in the logic?
2) Are there useful interactions between $*$, $\mathbb{C}$ and $\lfloor R \rfloor$?

Bluebell says YES!



RELATIONAL LIFTING AS CONDITIONING

Usual picture:

$$\begin{array}{ccc} \mathbb{D}(\text{Store}) \times \mathbb{D}(\text{Store}) & \geq & \lfloor R \rfloor \\ \uparrow \text{Lift} & & \uparrow \\ \text{Store} \times \text{Store} & \geq & R \end{array}$$

Analog of

$$\begin{array}{ccc} \mathbb{D}(\text{Store}) & \geq & \mathbb{P}(A) = 1 \\ \uparrow & & \uparrow \\ \text{Store} & \supseteq & A \end{array}$$

Bluebell's view:

$$\begin{array}{ccc} \mathbb{D}(\text{Store}) \times \mathbb{D}(\text{Store}) & \geq & \lfloor R \rfloor \\ \downarrow \text{Conditioning} & & \downarrow \\ \text{Store} \times \text{Store} & \geq & R \end{array}$$

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 $\mu : \mathbb{D}(A)$ and $k : A \to \mathbb{D}(\text{Store})$ define

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

Example $A = \{0,1\}$ $\mu = \text{Ber}(1/3)$

$$\text{bind}(\mu, k) = 1/3\, k(0) + 2/3\, k(1)$$

[This is actually the bind of the monad $\mathbb{D}(\cdot)$!]

JOINT CONDITIONING

Def Given $\mu : \mathbb{D}(A)$ and $P : A \to \mathrm{Assrt}$

define $\mathbf{C}_{\mu} v.\, P(v) : \mathrm{Assrt}$ by

$$(\mu_1, \mu_2) \models \mathbf{C}_{\mu} v.\, P(v) \quad \text{iff} \quad \left[\begin{array}{l} \exists\, K_1, K_2 : A \to \mathbb{D}(\mathrm{store}). \\ \quad \mu_1 = \mathrm{bind}(\mu, K_1) \;\wedge \\ \quad \mu_2 = \mathrm{bind}(\mu, K_2) \;\wedge \\ \quad \forall a \in \mathrm{supp}(\mu). \\ \quad\quad (K_1(a), K_2(a)) \models P(a) \end{array} \right]$$

JOINT CONDITIONING

$$(\mu_1, \mu_2) \models \mathbf{C}_{\mu} v.\, P(v) \quad \text{iff} \quad \left[\begin{array}{l} \exists K_1, K_2 : A \to \mathbb{D}(\text{Store}). \\ \quad \mu_1 = \text{bind}(\mu, K_1) \wedge \\ \quad \mu_2 = \text{bind}(\mu, K_2) \wedge \\ \quad \forall a \in \text{supp}(\mu). \\ \quad\quad (K_1(a), K_2(a)) \models P(a) \end{array}\right]$$

JOINT CONDITIONING

$$(\mu_1, \mu_2) \models C_\mu v.\, P(v) \quad \text{iff} \quad \left[\begin{array}{l} \exists K_1, K_2 : A \to \mathbb{D}(\text{Store}). \\ \quad \mu_1 = \text{bind}(\mu, K_1) \wedge \\ \quad \mu_2 = \text{bind}(\mu, K_2) \wedge \\ \quad \forall a \in \text{supp}(\mu). \\ \quad\quad (K_1(a), K_2(a)) \models P(a) \end{array}\right]$$

Example: $A = \{0,1\}$ $\mu = \text{Ber}(1/3)$

$$\mu_1 = 1/3\, K_1(0) + 2/3\, K_1(1)$$
$$\mu_2 = 1/3\, K_2(0) + 2/3\, K_2(1)$$

$P(0)$ $P(1)$

$X\langle 1\rangle \sim \text{Ber}(1/3)$ $\quad \lceil X\langle 1\rangle = 0 \rceil$ $\quad \lceil X\langle 1\rangle = 1 \rceil$

$$C_\mu v.\, \left(\lceil X\langle 1\rangle = v \rceil \mathbin{*} P(v)\right)$$

JOINT CONDITIONING

$$(\mu_1, \mu_2) \models \mathbf{C}_{\mu}\, v.\, P(v) \quad \text{iff} \quad \left[\begin{array}{l} \exists\, k_1, k_2 : A \to \mathbb{D}(\text{Store}). \\ \quad \mu_1 = \text{bind}(\mu, k_1) \wedge \\ \quad \mu_2 = \text{bind}(\mu, k_2) \wedge \\ \quad \forall a \in \text{supp}(\mu). \\ \quad\quad (k_1(a), k_2(a)) \models P(a) \end{array}\right]$$

Example: $A = \{0,1\}$ $\mu = \text{Ber}(1/3)$

$$\mu_1 = 1/3\, k_1(0) + 2/3\, k_1(1)$$
$$\mu_2 = 1/3\, k_2(0) + 2/3\, k_2(1)$$

$P(0)$ $P(1)$

$X\langle 1\rangle \sim \text{Ber}(1/3)$ $\lceil X\langle 1\rangle = 0 \rceil$ $\lceil X\langle 1\rangle = 1 \rceil$

$$\mathbf{C}_{\mu}\, v.\, (\lceil X\langle 1\rangle = v \rceil \not\models P(v))$$

[C-UNIT-R]

$$X\langle 1\rangle \sim \mu \dashv\vdash \mathbf{C}_{\mu}\, v.\, \lceil X\langle 1\rangle = v \rceil$$

[This reflects the right unit law of the underlying monad!]

ENCODING LIFTING AS CONDITIONING

Unary Conditioning: $C_{\mu}\, v.\left(\lceil x = v \rceil * P(v)\right)$

Relational Lifting:

$$\lfloor R(x\langle 1\rangle, x\langle 2\rangle) \rfloor :=$$

$$\exists \mu : \mathbb{D}(Val \times Val).\ C_{\mu}\,(v_1, v_2).\left(\lceil x\langle 1\rangle = v_1 \rceil * \lceil x\langle 2\rangle = v_2 \rceil * \underset{\text{pure}}{R(v_1, v_2)}\right)$$

JOINT COND. RULES

[C-UNIT-R]

$$x\langle i\rangle \sim \mu \dashv\vdash \mathbf{C}_\mu v.\ \lceil x\langle i\rangle = v\rceil$$

[C-FRAME]

$$P * \mathbf{C}_\mu v.\ Q(v) \vdash \mathbf{C}_\mu v.\ (P * Q(v))$$

[C-CONS]

$$\frac{\forall v \in \mathrm{supp}(\mu).\ P(v) \vdash P'(v)}{\mathbf{C}_\mu v.\ P(v) \vdash \mathbf{C}_\mu v.\ P'(v)}$$

$$\begin{aligned}
& x\langle 1\rangle \sim \mu * y\langle 1\rangle \sim \mu' * \lceil z = x+y\rceil \\
\vdash & \left(\mathbf{C}_\mu v.\ \lceil x\langle 1\rangle = v\rceil\right) * y\langle 1\rangle \sim \mu' * \lceil z = x+y\rceil \\
\vdash & \ \mathbf{C}_\mu v.\left(\lceil x\langle 1\rangle = v\rceil * y\langle 1\rangle \sim \mu' * \lceil z = x+y\rceil\right) \\
\vdash & \ \mathbf{C}_\mu v.\left(\mathbf{C}_{\mu'} v'.\left(\lceil x\langle 1\rangle = v\rceil * \lceil y\langle 1\rangle = v'\rceil * \lceil z = x+y\rceil\right)\right) \\
\vdash & \ \mathbf{C}_\mu v.\ \mathbf{C}_{\mu'} v'.\ \lceil x\langle 1\rangle = v \wedge y\langle 1\rangle = v' \wedge z = x+y\rceil
\end{aligned}$$

[C-ASSOC]

$$C_{\mu}\, v.\ C_{K(v)}\, v'.\ P(v,v') \vdash C_{bind'(\mu,K)}.\ P(v,v')$$

$$bind'(\mu,K) = \text{do } v \leftarrow \mu\ ;\ v' \leftarrow K(v)\ ;\ \text{return } (v,v')$$

[C-UNASSOC]

$$C_{bind(\mu,K)}\, v'.\ P(v') \vdash C_{\mu}\, v.\ C_{K(v)}\, v'.\ P(v')$$

SOME DERIVABLE RULES

$C_{/r}$ v. $\lfloor R \rfloor \vdash \lfloor R \rfloor$ (convexity of Rel Liftings)

$\lfloor R_1 \rfloor * \lfloor R_2 \rfloor \vdash \lfloor R_1 \wedge R_2 \rfloor$

NOTE

$\lfloor R_1 \rfloor \wedge \lfloor R_2 \rfloor \not\vdash \lfloor R_1 \wedge R_2 \rfloor$

CHALLENGES

- Generalization to Iris-style user-defined ghost resources
- [C-wp-SWAP]

$$\text{ownVars} \wedge C_{/\mu}\, v.\ \mathbf{wp}\, t\, \{Q(v)\} \vdash \mathbf{wp}\, t\, \{C_{/\mu}\, v.\ Q(v)\}$$

↑ Bluebell needs this for soundness

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


Thanks