Probabilistic couplings for cryptography and privacy

Gilles Barthe
IMDEA Software Institute, Madrid, Spain

October 7, 2016
Relational properties

Properties about two runs of the same program

- Assume inputs are related by $\Psi$
- Want to prove the outputs are related by $\Phi$
Examples

Monotonicity

- $\Psi: in_1 \leq in_2$
- $\Phi: out_1 \leq out_2$
- “Bigger inputs give bigger outputs”
Examples

Monotonicity

- $\Psi : in_1 \leq in_2$
- $\Phi : out_1 \leq out_2$
- “Bigger inputs give bigger outputs”

Stability

- $\Psi : inp_1 \sim inp_2$
- $\Phi : out_1 \sim out_2$
- “If inputs are similar, then outputs are similar”
Examples

Monotonicity

- $\Psi: \text{in}_1 \leq \text{in}_2$
- $\Phi: \text{out}_1 \leq \text{out}_2$
- “Bigger inputs give bigger outputs”

Stability

- $\Psi: \text{inp}_1 \sim \text{inp}_2$
- $\Phi: \text{out}_1 \sim \text{out}_2$
- “If inputs are similar, then outputs are similar”

Non-interference

- $\Psi: \text{lowinp}_1 = \text{lowinp}_2$
- $\Phi: \text{lowout}_1 = \text{lowout}_2$
- “If low inputs are equal, then low outputs are equal”
Probabilistic relational properties

Monotonicity

- \( \psi : \text{in}_1 \leq \text{in}_2 \)
- \( \Phi : \Pr[\text{out}_1 \geq k] \leq \Pr[\text{out}_2 \geq k] \)
Probabilistic relational properties

Monotonicity

- $\Psi: in_1 \leq in_2$
- $\Phi: \Pr[out_1 \geq k] \leq \Pr[out_2 \geq k]$

Stability

- $\Psi: in_1 \sim in_2$
- $\Phi: \Pr[out_1 = k] \sim \Pr[out_2 = k]$
Probabilistic relational properties

Monotonicity

- $\Psi : in_1 \leq in_2$
- $\Phi : Pr[out_1 \geq k] \leq Pr[out_2 \geq k]$

Stability

- $\Psi : in_1 \sim in_2$
- $\Phi : Pr[out_1 = k] \sim Pr[out_2 = k]$

Non-interference

- $\Psi : lowinp_1 = lowinp_2$
- $\Phi : Pr[lowout_1 = k] = Pr[lowout_2 = k]$
Probabilistic relational properties

Monotonicity

- $\Psi: in_1 \leq in_2$
- $\Phi: \Pr[out_1 \geq k] \leq \Pr[out_2 \geq k]$

Stability

- $\Psi: in_1 \sim in_2$
- $\Phi: \Pr[out_1 = k] \sim \Pr[out_2 = k]$

Non-interference

- $\Psi: lowinp_1 = lowinp_2$
- $\Phi: \Pr[lowout_1 = k] = \Pr[lowout_2 = k]$

Richer properties

- Indistinguishability, differential privacy
Probabilistic couplings

- Used by mathematicians for proving relational properties
- Applications: Markov chains, probabilistic processes

Idea

- Place two processes in the same probability space
- Coordinate the sampling
Probabilistic couplings

- Used by mathematicians for proving relational properties
- Applications: Markov chains, probabilistic processes

Idea

- Place two processes in the same probability space
- Coordinate the sampling

Why is this interesting?

- Proving relational probabilistic properties reduced to proving non-relational non-probabilistic properties
- Compositional
Introducing probabilistic couplings

Basic ingredients

- Given: two distributions $X_1, X_2$ over set $A$
- Produce: joint distribution $Y$ over $A \times A$
  - Projection over the first component is $X_1$
  - Projection over the second component is $X_2$
Introducing probabilistic couplings

Basic ingredients

- Given: two distributions $X_1, X_2$ over set $A$
- Produce: joint distribution $Y$ over $A \times A$
  - Projection over the first component is $X_1$
  - Projection over the second component is $X_2$

Definition

Given two distributions $X_1, X_2$ over a set $A$, a coupling $Y$ is a distribution over $A \times A$ such that $\pi_1(Y) = X_1$ and $\pi_2(Y) = X_2$
Introducing probabilistic couplings

Basic ingredients

- Given: two distributions $X_1, X_2$ over set $A$
- Produce: joint distribution $Y$ over $A \times A$
  - Projection over the first component is $X_1$
  - Projection over the second component is $X_2$

Definition
Given two distributions $X_1, X_2$ over a set $A$, a coupling $Y$ is a distribution over $A \times A$ such that $\pi_1(Y) = X_1$ and $\pi_2(Y) = X_2$ where

$$\pi_1(Y)(a_1) = \sum_{a_2} Y(a_1, a_2)$$
Fair coin toss

- One way to coordinate: require $x_1 = x_2$
- A different way: require $x_1 = \neg x_2$
- Yet another way: product distribution
- Choice of coupling depends on application
- Couplings always exist
Couplings vs liftings

Let $\mu_1, \mu_2 \in \text{Distr}(A)$, $\mu \in \text{Distr}(A \times A)$ and $R \subseteq A \times A$. Then

$$\mu \leftarrow_R \langle \mu_1 \& \mu_2 \rangle \triangleq \pi_1(\mu) = \mu_1 \land \pi_2(\mu) = \mu_2 \land \Pr_{y \leftarrow \mu}[y \in R] = 1$$

Different couplings yield liftings for different relations
Convergence of random walks

Simple random walk on integers

- Start at some position $p$
- Each step, flip coin $x \leftarrow \text{flip}$
- Heads: $p \leftarrow p + 1$
- Tails: $p \leftarrow p - 1$
Convergence of random walks

Simple random walk on integers

- Start at some position $p$
- Each step, flip coin $x \xLeftarrow{\$} \text{flip}$
- Heads: $p \leftarrow p + 1$
- Tails: $p \leftarrow p - 1$
Coupling the walks to meet

Case $p_1 = p_2$: Walks have met

▶ Arrange samplings $x_1 = x_2$
▶ Continue to have $p_1 = p_2$
Coupling the walks to meet

Case $p_1 = p_2$: Walks have met

- Arrange samplings $x_1 = x_2$
- Continue to have $p_1 = p_2$

Case $p_1 \neq p_2$: Walks have not met

- Arrange samplings $x_1 = \neg x_2$
- Walks make mirror moves
Coupling the walks to meet

Case $p_1 = p_2$: Walks have met

- Arrange samplings $x_1 = x_2$
- Continue to have $p_1 = p_2$

Case $p_1 \neq p_2$: Walks have not met

- Arrange samplings $x_1 = \neg x_2$
- Walks make mirror moves

Under coupling, if walks meet, they move together
Why is this interesting?

Memorylessness
Positions converge as we take more steps
Why is this interesting?

**Memorylessness**

Positions converge as we take more steps

**Coupling bounds distance between distributions**

- Once walks meet, they stay equal
- Distance is at most probability walks don’t meet

Theorem

If $Y$ is a coupling of two distributions $(X_1, X_2)$, then

$$\|X_1 - X_2\|_{TV} \leq \Pr(y_1, y_2) \sim Y[y_1 \neq y_2].$$
Why is this interesting?

Memorylessness
Positions converge as we take more steps

Coupling bounds distance between distributions

- Once walks meet, they stay equal
- Distance is at most probability walks don’t meet

Theorem
If \( Y \) is a coupling of two distributions \((X_1, X_2)\), then

\[
\|X_1 - X_2\|_{TV} \triangleq \sum_{a \in A} |X_1(a) - X_2(a)| \leq \Pr_{(y_1, y_2) \sim Y}[y_1 \neq y_2].
\]
probabilistic Relational Hoare Logic

⊢ \{ P \} c_1 \sim c_2 \{ Q \} \text{ iff there exists } \mu \text{ such that}

\begin{equation}
P(m_1 \uplus m_2) \Rightarrow \mu \triangleleft_Q \langle \llbracket c_1 \rrbracket m_1 \& \llbracket c_2 \rrbracket m_2 \rangle
\end{equation}

where

\begin{equation}
\mu \triangleleft_R \langle \mu_1 \& \mu_2 \rangle \triangleq \pi_1(\mu) = \mu_1 \land \pi_2(\mu) = \mu_2 \land \text{supp}(\mu) \subseteq R
\end{equation}

Fundamental lemma of pRHL

If \( Q \triangleq E_1 \Rightarrow E_2 \) then \( \Pr(\llbracket c_1 \rrbracket m_1)[E_1] \leq \Pr(\llbracket c_2 \rrbracket m_2)[E_2] \)
Core rules

\[
\begin{align*}
\{\Phi\} c_1 & \sim c_2 \{\Theta\} & \{\Theta\} c'_1 & \sim c'_2 \{\Psi\} \\
\{\Phi\} c_1; c'_1 & \sim c_2; c'_2 \{\Psi\}
\end{align*}
\]

\[
\begin{align*}
\{\Phi \land b_1 \land b_2\} c_1 & \sim c_2 \{\Psi\} & \{\Phi \land \neg b_1 \land \neg b_2\} c'_1 & \sim c'_2 \{\Psi\} \\
\{\Phi \land b_1 = b_2\} \text{if } b_1 \text{ then } c_1 \text{ else } & c'_1 \sim \text{if } b_2 \text{ then } c_2 \text{ else } c'_2 \{\Psi\}
\end{align*}
\]

\[
\begin{align*}
\{\Phi \land b_1 \land b_2\} c_1 & \sim c_2 \{\Phi \land b_1 = b_2\} \\
\{\Phi \land b_1 = b_2\} \text{while } b_1 \text{ do } c_1 \sim \text{while } b_2 \text{ do } c_2 \{\Phi \land \neg b_1 \land \neg b_2\}
\end{align*}
\]
Loops

\[ \psi \implies p_0 \oplus p_1 \oplus p_2 \]

\[ \psi \land p_0 \implies e_1 \land e_2 \quad \psi \land p_1 \implies e_1 \quad \psi \land p_2 \implies e_2 \]

\[ \text{while } e_1 \land p_1 \text{ do } c_1 \Downarrow \text{while } e_2 \land p_2 \text{ do } c_2 \]

\[ \{ \psi \land p_1 \} c_1 \sim \text{skip}\{ \psi \} \quad \{ \psi \land p_2 \} \text{skip} \sim c_2\{ \psi \} \]

\[ \{ \psi \land p_0 \} c_1 \sim c_2\{ \psi \} \]

\[ \{ \psi \} \text{while } e_1 \text{ do } c_1 \sim \text{while } e_2 \text{ do } c_2\{ \psi \land \neg e_1 \land \neg e_2 \} \]
Random assignment

\[\mu \triangleleft_q \langle \mu_1 \& \mu_2 \rangle \]
\[\vdash \{ \top \} x_1 \leftarrow \mu_1 \sim x_2 \leftarrow \mu_2 \{ Q \}\]

Specialized rule

\[f \in T \xrightarrow{1-1} T \quad \forall v \in T. \ d_1(v) = d_2(f \ v) \]
\[\vdash \{ \forall v, Q[v/x_1, f \ v/x_2] \} \ x_1 \leftarrow \mu_1 \sim x_2 \leftarrow \mu_2 \{ Q \}\]

Notes

- Bijection \( f \): specifies how to coordinate the samples
- Side condition: marginals are preserved under \( f \)
- Assume: samples coupled when proving postcondition \( \Phi \)
Applications to cryptography

▶ EasyCrypt: interactive proof assistant (inspired from ssreflect) with back-end to SMT and CAS
▶ applied to encryption, signatures, hash designs, key exchange protocols, zero knowledge protocols, garbled circuits, SHA3, e-voting

Formalizing cryptographic proofs?

▶ *In our opinion, many proofs in cryptography have become essentially unverifiable. Our field may be approaching a crisis of rigor.* Bellare and Rogaway, 2004-2006
▶ *Do we have a problem with cryptographic proofs? Yes, we do [...] We generate more proofs than we carefully verify (and as a consequence some of our published proofs are incorrect).* Halevi, 2005
approximate probabilistic Relational Hoare Logic

- Quantitative generalization of pRHL \( \vdash_{\epsilon,\delta} \{ P \} c_1 \sim c_2 \{ Q \} \)

- Valid if there exists \( \mu_L, \mu_R \) such that

\[
P(m_1 \uplus m_2) \implies \mu_L, \mu_R \triangleright^\epsilon,\delta_Q \langle \llbracket c_1 \rrbracket m_1 & \llbracket c_2 \rrbracket m_2 \rangle
\]

where

\[
\mu_L, \mu_R \triangleright^\epsilon,\delta_Q \langle \mu_1 & \mu_2 \rangle \triangleq \begin{cases} 
\pi_1(\mu_L) = \mu_1 & \pi_2(\mu_R) = \mu_2 \\
\text{supp}(\mu_L), \text{supp}(\mu_R) \subseteq Q \\
\Delta_\epsilon(\mu_1, \mu_2) \leq \delta
\end{cases}
\]

- Fundamental theorem of apRHL: if \( Q \triangleq E_1 \Rightarrow E_2 \) then

\[
\Pr(\llbracket c_1 \rrbracket m_1)[E_1] \leq \exp(\epsilon) \Pr(\llbracket c_2 \rrbracket m_2)[E_2] + \delta
\]

- Extends to \( f \)-divergences
Application: differential privacy

A randomized algorithm $K$ is $(\epsilon, \delta)$-differentially private w.r.t. $\Phi$ iff for all databases $D_1$ and $D_2$ s.t. $\Phi(D_1, D_2)$:

$\forall S. \Pr[K(D_1) \in S] \leq \exp(\epsilon) \cdot \Pr[K(D_2) \in S] + \delta$

Privacy as approximate couplings

$K$ is $(\epsilon, \delta)$-differentially private wrt $\Phi$ iff $\vdash \epsilon, \delta \{ \Phi \}^K_1 \sim K_2 \{ \equiv \}$
Application: differential privacy

A randomized algorithm $K$ is $(\epsilon, \delta)$-differentially private w.r.t. $\Phi$ iff for all databases $D_1$ and $D_2$ s.t. $\Phi(D_1, D_2)$:

$$\forall S. \Pr[K(D_1) \in S] \leq \exp(\epsilon) \cdot \Pr[K(D_2) \in S] + \delta$$

Privacy as approximate couplings:

$K$ is $(\epsilon, \delta)$-differentially private w.r.t $\Phi$ iff $\vdash \epsilon, \delta \{ \Phi \} K_1 \sim K_2 \{ \equiv \}$
Application: differential privacy

A randomized algorithm $K$ is $(\epsilon, \delta)$-differentially private w.r.t. $\Phi$ iff for all databases $D_1$ and $D_2$ s.t.

$$\forall S. \Pr[K(D_1) \in S] \leq \exp(\epsilon) \cdot \Pr[K(D_2) \in S] + \delta$$

Privacy as approximate couplings

$K$ is $(\epsilon, \delta)$-differentially private wrt $\Phi$ iff

$$\epsilon, \delta \{\Phi\} K \sim_1 K_2 \{\equiv\}$$

Bounded ratio
A randomized algorithm $\mathcal{K}$ is $(\epsilon, \delta)$-differentially private w.r.t. $\Phi$ iff for all databases $D_1$ and $D_2$ s.t. $\Phi(D_1, D_2)$

$$\forall S. \Pr[\mathcal{K}(D_1) \in S] \leq \exp(\epsilon) \cdot \Pr[\mathcal{K}(D_2) \in S] + \delta$$
A randomized algorithm $\mathcal{K}$ is $(\epsilon, \delta)$-differentially private w.r.t. $\Phi$ iff for all databases $D_1$ and $D_2$ s.t. $\Phi(D_1, D_2)$

$$\forall S. \ Pr[\mathcal{K}(D_1) \in S] \leq \exp(\epsilon) \cdot Pr[\mathcal{K}(D_2) \in S] + \delta$$

Privacy as approximate couplings
$\mathcal{K}$ is $(\epsilon, \delta)$-differentially private wrt $\Phi$ iff $\vdash_{\epsilon, \delta} \{\Phi\} \mathcal{K}_1 \sim \mathcal{K}_2\{\equiv\}$
Differential privacy via output perturbation

Let $f$ be $k$-sensitive w.r.t. $\Phi$:

$$\Phi(a, a') \implies |f(a) - f(a')| \leq k$$

Then $a \mapsto L_\epsilon(f(a))$ is $(k \cdot \epsilon, 0)$-differentially private w.r.t. $\Phi$. 
Proof principles for Laplace mechanism

Making different things look equal

\[ \Phi \triangleq |e_1 - e_2| \leq k' \]

\[ \vdash_{k', \epsilon, 0} \{ \Phi \} y_1 \overset{\$}{\sim} L_\epsilon(e_1) \sim y_2 \overset{\$}{\sim} L_\epsilon(e_2) \{ y_1 = y_2 \} \]

Making equal things look different

\[ \Phi \triangleq e_1 = e_2 \]

\[ \vdash_{k, \epsilon, 0} \{ \Phi \} y_1 \overset{\$}{\sim} L_\epsilon(e_1) \sim y_2 \overset{\$}{\sim} L_\epsilon(e_2) \{ y_1 + k = y_2 \} \]

Pointwise equality

\[ \forall i. \vdash_{\epsilon, 0} \{ \Phi \} c_1 \sim c_2 \{ x_1 = i \Rightarrow x_2 = i \} \]

\[ \vdash_{\epsilon, 0} \{ \Phi \} c_1 \sim c_2 \{ x_1 = x_2 \} \]
If $\mathcal{K}$ is $(\epsilon, \delta)$-differentially private, and

$\lambda a. \mathcal{K}'(a, b)$ is $(\epsilon', \delta')$-differentially private for every $b \in B$,

then $\lambda a. \mathcal{K}'(a, \mathcal{K}(a))$ is $(\epsilon + \epsilon', \delta + \delta')$-differentially private.
Beyond composition: Sparse Vector Technique

SparseVector\(_{bt}(a, b, M, N, d) := \)
\[
i \leftarrow 0; l \leftarrow []; u \leftarrow \mathcal{L}_\epsilon(0); A \leftarrow a - u; B \leftarrow b + u; \\
\text{while } i < N \text{ do} \\
\quad i \leftarrow i + 1; q \leftarrow A(l); S \leftarrow \mathcal{L}_\epsilon(q(d)); \\
\quad \text{if } (A \leq S \leq B \land |l| < M) \text{ then } l \leftarrow i :: l; \\
\text{return } l
\]

Privacy
If queries are 1-sensitive, then \((\sqrt{M\epsilon}, \delta')\)-diff. private

Tools
- advanced composition
- accuracy-dependent privacy
- optimal subset coupling
Proofs as (products) programs: xpRHL

- Every pRHL derivation yields a product program
- Different derivations yield different programs
- Can be modelled by a proof system
  \[ \vdash \{ \Phi \} c_1 \leadsto c_2 \{ \Psi \} \leadsto c \]

Fundamental lemma of xpRHL

- \[ \vdash \{ \Phi \} c_1 \leadsto c_2 \{ \Psi \} \implies x_1 = x_2 \]
- \[ \{ \Box \Phi \} c \{ \Pr[\neg \Psi] \leq \epsilon \} \]
  implies

\[ m_1 \Phi m_2 \Rightarrow |\Pr([c_1] m_1)[E(x_1)] - \Pr([c_2] m_2)[E(x_2)]| \leq \epsilon \]
Dynkin's card trick (shift coupling)

\[
p \leftarrow s; \quad l \leftarrow [p];
\]

while \( p < N \) do
\[
n \leftarrow \mathcal{U}[1, 10];
\]
\[
p \leftarrow p + n;
\]
\[
l \leftarrow p :: l;
\]
return \( p \)

Convergence

If \( s_1, s_2 \in [1, 10] \), and \( N > 10 \), then
\[
\Delta(p_1^{\text{final}}, p_2^{\text{final}}) \leq \left( \frac{9}{10} \right)^{N/5-2}
\]
Perspectives and further directions

- Program logics for provable security and differential privacy
  - Based on probabilistic couplings

Open questions
- couplings
- applications