Proving Differential Privacy via Probabilistic Couplings

Gilles Barthe[⋆] Marco Gaboardi[‡] Benjamin Grégoire[$] Justin Hsu[#] Pierre-Yves Strub[⋆]

[⋆] IMDEA Software [‡] University at Buffalo, SUNY [$] Inria [#] University of Pennsylvania

Abstract

Over the last decade, *differential privacy* has achieved widespread adoption within the privacy community. Moreover, it has attracted significant attention from the verification community, resulting in several successful tools for formally proving differential privacy. Although their technical approaches vary greatly, all existing tools rely on reasoning principles derived from the *composition theorem* of differential privacy. While this suffices to verify most common private algorithms, there are several important algorithms whose privacy analysis does not rely solely on the composition theorem. Their proofs are significantly more complex, and are currently beyond the reach of verification tools.

In this paper, we develop compositional methods for formally verifying differential privacy for algorithms whose analysis goes beyond the composition theorem. Our methods are based on the observation that differential privacy has deep connections with a generalization of *probabilistic couplings*, an established mathematical tool for reasoning about stochastic processes. Even when the composition theorem is not helpful, we can often prove privacy by a coupling argument.

We demonstrate our methods on two algorithms: the *Exponential mechanism* and the *Above Threshold* algorithm, the critical component of the famous *Sparse Vector* algorithm. We verify these examples in a relational program logic apRHL^+, which can construct approximate couplings. This logic extends the existing apRHL logic with more general rules for the Laplace mechanism and the one-sided Laplace mechanism, and new structural rules enabling pointwise reasoning about privacy; all the rules are inspired by the connection with coupling. While our paper is presented from a formal verification perspective, we believe that its main insight is of independent interest for the differential privacy community.

1. Introduction

Differential privacy is a rigorous definition of statistical privacy proposed by Dwork, McSherry, Nissim and Smith [12], and considered to be the gold standard for privacy-preserving computations. Most differentially private computations are built from two fundamental tools: private primitives and composition theorems . However, there are several important examples whose privacy proofs go beyond these tools, for instance:

- The *Above Threshold* algorithm, which takes a list of numerical queries as input and privately outputs the first query whose answer is above a certain threshold. Above Threshold is the critical component of the Sparse Vector technique. (See, e.g., Dwork and Roth [11].)
- The *Report-noisy-max* algorithm, which takes a list of numerical queries as input and privately selects the query with the highest answer. (See, e.g., Dwork and Roth [11].)
- The *Exponential mechanism* [16], which privately returns the element of a (possibly non-numeric) range with the highest score; this algorithm can be implemented as a variant of the Report-noisy-max algorithm with a different noise distribution.

Unfortunately, existing pen-and-paper proofs of these algorithms use ad hoc manipulations of probabilities, and as a consequence are difficult to understand and error-prone.

This raises a natural question: can we develop *compositional proof methods* for verifying differential privacy of these algorithms, even though their proofs appear non-compositional? Surprisingly, the answer is yes. Our method builds on two key insights.

1. A connection between probabilistic liftings and probabilistic couplings [6]. Although the two concepts are tightly connected, their relationship has been little explored.
2. A view of differential privacy as a form of approximate probabilistic liftings [2, 4], a generalization of probabilistic liftings used in probabilistic process algebra [13].

We elaborate on these points, and then present our contributions.

Probabilistic liftings and couplings

Relation lifting is a well-studied construction in mathematics and computer science. Abstractly, relation lifting transforms relations $R \subseteq A \times B$ into relations $R^\sharp \subseteq TA \times TB$, where T is a functor over sets [1]. Relation lifting satisfies a type of composition, so it is a natural foundation for compositional proof methods.

Relation lifting has historically been an important tool in the study of probabilistic systems. For example, *probabilistic lifting* specializes the notion of relation lifting for the probability monad, and appears in standard definitions of probabilistic bisimulation. Over the last 25 years, researchers have developed a wide variety of tools for reasoning about probabilistic liftings, explored applications in numerous areas including security and biology, and uncovered deep connections with the Kantorovich metric and the theory of optimal transport [10].

While research in this area has traditionally focused on probabilistic liftings for partial equivalence relations, recent works investigate liftings for more general relations. Applications include formalizing reduction-based cryptographic proofs [3], and modeling stochastic dominance and convergence of probabilistic processes [6]. Seeking to explain the power of liftings, Barthe et al. [6] establish a tight connection between probabilistic liftings and *probabilistic couplings*, a basic tool in probability theory [15, 17]. Roughly, a probabilistic coupling places two distributions in the same probabilistic space, by exhibiting a suitable witness distribution over pairs. Not only does this observation open new avenues for applying probabilistic liftings, it offers an opportunity to revisit existing applications from a fresh perspective.

Differential privacy via approximate probabilistic liftings

Relational program logics [2, 4] and relational refinement type systems [7] are the most flexible techniques known for reasoning formally about differentially private computations. Their expressive power stems from their use of approximate probabilistic liftings, a generalization of probabilistic liftings based on a notion of distance between distributions; differential privacy is a consequence of a particular form of approximate lifting.

These approaches have successfully verified differential privacy for many algorithms. However, they are unsuccessful when privacy does not follow from standard tools and composition properties. In fact, the present authors had long believed that the verification of such examples was beyond the capabilities of lifting-based methods.

Contributions

In this paper, we propose the first formal analysis of differentially private algorithms whose proof does not (exclusively) rely on the basic tools of differential privacy. We make three broad contributions.

New proof principles for approximate liftings We take inspiration from the connection between liftings and coupling to develop new proof principles for approximate liftings.

First, we introduce a principle for decomposing proofs of differential privacy "pointwise", supporting a common pattern of proving privacy separately for each possible output value. This principle is used in pen-and-paper proofs, but is new to formal approaches.

Second, we provide new proof principles for the Laplace mechanism. Informally speaking, existing proof principles capture the intuition that different inputs can be made to "look equal" by the Laplace mechanism, provided that one pays sufficient privacy. Our first new proof principle for the Laplace mechanism is dual, and captures the idea that equal inputs can be made to look arbitrarily *different* by the Laplace mechanism, provided that one pays sufficient privacy. Our second new proof principle for the Laplace mechanism states that if we add the same noise in two runs of the Laplace mechanism, the distance between the two values is preserved and there is no privacy cost. As far as we know, these proof principles are new to the differential privacy literature, and provide the key to proving examples such as Sparse Vector using compositional proof methods.

We also propose approximate probabilistic liftings for the one-sided Laplace mechanism, which can be used to implement the Exponential mechanism, but has been little-studied in the differential privacy literature. The one-sided Laplace mechanism nicely illustrates the benefits of our approach: although it is not differentially private, its properties can be captured formally by approximate probabilistic liftings. These properties can be combined to show privacy for a larger program. These discussions are deferred to the extended version.

An extended probabilistic relational program logic To demonstrate our techniques, we take the relational program logic apRHL [4] as our starting point. Conceived as a probabilistic variant of Benton's relational Hoare logic [9], apRHL has been used to verify differential privacy for examples using the standard composition theorems. Most importantly, the semantics of apRHL is in terms of approximate liftings. We introduce new proof rules representing our new proof principles, and call the resulting logic apRHL$^+$.

New privacy proofs While the extensions amount to just a handful of rules, they significantly increase the power of apRHL: We provide the first formal verification of two algorithms whose privacy proof use tools beyond the composition theorems.

- The *Exponential mechanism*. The standard private algorithm when the output is non-numeric, this construction is typically taken as a primitive in systems verifying privacy. In contrast, we prove its privacy within our logic.
- The *Sparse Vector* algorithm. Perhaps the most famous example not covered by existing techniques, the proof of this mechanism is quite involved; some of its variants are not provably private. We also prove the privacy of its core subroutine in our logic.

The proofs are based on coupling ideas, which avoid reasoning about probabilities explicitly. As a consequence, proofs are clean, concise, and, we believe, appealing to researchers from both the differential privacy and the formal verification communities.

We have formalized the proofs of these algorithms in an experimental branch of the EasyCrypt proof assistant supporting approximate probabilistic liftings. An extended version of this paper [8] is available at http://arxiv.org/abs/1601.05047.

2. Generalized probabilistic liftings

To verify advanced algorithms like AboveT, we will leverage the power of *approximate probabilistic liftings*. In a sentence, our proofs will replace the sequential composition theorem of differential privacy—which is not strong enough to verify our target examples—with the more general composition principle of liftings. This section reviews existing notions of (approximate) probabilistic liftings and introduces proof principles for establishing their existence. Most of these proof principles are new, including those for equality (Proposition 2), differential privacy (Proposition 6), the Laplace mechanism (Propositions 8 and 9), and the one-sided Laplace mechanism .

2.1 Probabilistic couplings and liftings

Probabilistic couplings and liftings are standard tools in probability theory, and semantics and verification, respectively. We present their definitions to highlight their similarities before discussing some useful consequences.

Definition 1 (Coupling)**.** *There is a* coupling *between two sub-distributions* $\mu_1 \in \mathbf{Distr}(B_1)$ *and* $\mu_2 \in \mathbf{Distr}(B_2)$ *if there exists a sub-distribution (called the* witness*)* $\mu \in \mathbf{Distr}(B_1 \times B_2)$ *s.t.* $\pi_1(\mu) = \mu_1$ *and* $\pi_2(\mu) = \mu_2$.

Probabilistic liftings are a special class of couplings.

Definition 2 (Lifting)**.** *Two sub-distributions* $\mu_1 \in \mathbf{Distr}(B_1)$ *and* $\mu_2 \in \mathbf{Distr}(B_2)$ *are related by the* (probabilistic) lifting *of* $\Psi \subseteq B_1 \times B_2$*, written* $\mu_1 \Psi^\sharp \mu_2$*, if there exists a coupling* $\mu \in \mathbf{Distr}(B_1 \times B_2)$ *of* μ_1 *and* μ_2 *such that* $\mathsf{supp}(\mu) \subseteq \Psi$.

Probabilistic liftings have many useful consequences. For example, $\mu_1 =^\sharp \mu_2$ holds exactly when the sub-distributions μ_1 and μ_2 are equal. Less trivially, liftings can bound the probability of one event by the probability of another event. This observation is useful for formalizing reduction-based cryptographic proofs.

Proposition 1 (Barthe et al. [3])**.** *Let* $E_1 \subseteq B_1$, $E_2 \subseteq B_2$, $\mu_1 \in \mathbf{Distr}(B_1)$ *and* $\mu_2 \in \mathbf{Distr}(B_2)$*. Define*

$$\Psi = \{(x_1, x_2) \in B_1 \times B_2 \mid x_1 \in E_1 \Rightarrow x_2 \in E_2\}.$$

If $\mu_1 \Psi^\sharp \mu_2$*, then*

$$\Pr_{x_1 \leftarrow \mu_1}[x_1 \in E_1] \leq \Pr_{x_2 \leftarrow \mu_2}[x_2 \in E_2].$$

One key observation for our approach is that this result can also be used to prove equality between distributions in a pointwise style.

Proposition 2 (Equality by pointwise lifting)**.**

- *Let* $\mu_1, \mu_2 \in \mathbf{SDistr}(B)$*. For every* $b \in B$*, define*
$$\Psi_b = \{(x_1, x_2) \in B \times B \mid x_1 = b \Rightarrow x_2 = b\}.$$
If $\mu_1 \Psi_b^\sharp \mu_2$ *for all* $b \in B$*, then* $\mu_1 = \mu_2$.
- *Let* $\mu_1, \mu_2 \in \mathbf{Distr}(B)$*. For every* $b \in B$*, define*
$$\Psi_b = \{(x_1, x_2) \in B \times B \mid x_1 = b \Leftrightarrow x_2 = b\}.$$

If $\mu_1 \ \Psi_b^\sharp \ \mu_2$ for all $b \in B$, then $\mu_1 = \mu_2$.

2.2 Approximate liftings

It has previously been shown that differential privacy follows from an approximate version of liftings [4]. Our presentation follows subsequent refinements by Barthe and Olmedo [2]. We start by defining a notion of distance between sub-distributions.

Definition 3 (Barthe et al. [4])**.** *Let $\epsilon \geq 0$. The* ϵ-DP divergence *$\Delta_\epsilon(\mu_1, \mu_2)$ between two sub-distributions $\mu_1 \in \mathbf{Distr}(B)$ and $\mu_2 \in \mathbf{Distr}(B)$ is defined as*

$$\sup_{E \subseteq B} \left(\Pr_{x \leftarrow \mu_1}[x \in E] - \exp(\epsilon) \Pr_{x \leftarrow \mu_2}[x \in E] \right)$$

The following proposition relates ϵ-DP divergence with (ϵ, δ)-differential privacy.

Proposition 3 (Barthe et al. [4])**.** *A probabilistic computation $M : A \to \mathbf{Distr}(B)$ is (ϵ, δ)-differentially private w.r.t. an adjacency relation Φ iff*

$$\Delta_\epsilon(M(a), M(a')) \leq \delta$$

for every two adjacent inputs a and a' (i.e. such that $a \ \Phi \ a'$).

We can use DP-divergence to define an approximate version of probabilistic lifting, called (ϵ, δ)-*lifting*. We adopt the definition by Barthe and Olmedo [2], which extends to a general class of distances called f-divergences.

Definition 4 ((ϵ, δ)-lifting)**.** *Two sub-distributions $\mu_1 \in \mathbf{Distr}(B_1)$ and $\mu_2 \in \mathbf{Distr}(B_2)$ are related by the* (ϵ, δ)-lifting *of $\Psi \subseteq B_1 \times B_2$, written $\mu_1 \Psi^{\sharp(\epsilon,\delta)} \mu_2$, if there exist two witness sub-distributions $\mu_L \in \mathbf{Distr}(B_1 \times B_2)$ and $\mu_R \in \mathbf{Distr}(B_1 \times B_2)$ such that*

1. *$\pi_1(\mu_L) = \mu_1$ and $\pi_2(\mu_R) = \mu_2$;*
2. *$\mathsf{supp}(\mu_L) \subseteq \Psi$ and $\mathsf{supp}(\mu_R) \subseteq \Psi$; and*
3. *$\Delta_\epsilon(\mu_L, \mu_R) \leq \delta$.*

It is relatively easy to see that two sub-distributions μ_1 and μ_2 are related by $=^{\sharp(\epsilon,\delta)}$ iff $\Delta_\epsilon(\mu_1, \mu_2) \leq \delta$. Therefore, a probabilistic computation $M : A \to \mathbf{Distr}(B)$ is (ϵ, δ)-differentially private w.r.t. an adjacency relation Φ iff

$$M(a) \ =^{\sharp(\epsilon,\delta)} \ M(a')$$

for every two adjacent inputs a and a' (i.e. such that $a \ \Phi \ a'$). This fact forms the basis of previous lifting-based approaches for differential privacy [2, 4, 5, 7].

A useful preliminary fact is that approximate liftings generalize probabilistic liftings (which we will sometimes call *exact* liftings).

Proposition 4. *Suppose we are given distributions $\mu_1 \in \mathbf{SDistr}(B_1)$ and $\mu_2 \in \mathbf{SDistr}(B_2)$ and a relation $\Psi \subseteq B_1 \times B_2$. Then, $\mu_1 \Psi^\sharp \mu_2$ if and only if $\mu_1 \Psi^{\sharp(0,0)} \mu_2$.*

The previous results for exact liftings generalize smoothly to approximate liftings. First, we can generalize Proposition 1.

Proposition 5 (Barthe and Olmedo [2])**.** *Let $E_1 \subseteq B_1$, $E_2 \subseteq B_2$, $\mu_1 \in \mathbf{Distr}(B_1)$ and $\mu_2 \in \mathbf{Distr}(B_2)$. Let*

$$\Psi = \{(x_1, x_2) \in B_1 \times B_2 \mid x_1 \in E_1 \Rightarrow x_2 \in E_2\}.$$

If $\mu_1 \Psi^{\sharp(\epsilon,\delta)} \mu_2$, then

$$\Pr_{x_1 \leftarrow \mu_1}[x_1 \in E_1] \leq \exp(\epsilon) \Pr_{x_2 \leftarrow \mu_2}[x_2 \in E_2] + \delta.$$

We can use this proposition to generalize Proposition 2, which provides a way to prove that two distributions μ_1 and μ_2 are equal—equivalently, $\mu_1 \ =^\sharp \ \mu_2$. Generalizing this lifting from exact to approximate yields the following pointwise characterization of differential privacy, a staple technique of pen-and-paper proofs.

Proposition 6 (Differential privacy from pointwise lifting)**.** *A probabilistic computation $M : A \to \mathbf{Distr}(B)$ is (ϵ, δ)-differentially private w.r.t. an adjacency relation Φ iff there exists $(\delta_b)_{b \in B} \in \mathbb{R}^{\geq 0}$ such that $\sum_{b \in B} \delta_b \leq \delta$, and $M(a) \ \Psi_b^{\sharp(\epsilon,\delta_b)} \ M(a')$ for every $b \in B$ and every two adjacent inputs a and a', where*

$$\Psi_b = \{(x_1, x_2) \in B \times B \mid x_1 = b \Rightarrow x_2 = b\}.$$

2.3 Probabilistic liftings for the Laplace mechanism

So far, we have seen general properties about approximate liftings and differential privacy. Now, we turn to more specific liftings relevant to typical distributions in differential privacy. In terms of approximate liftings, we can state the privacy of the Laplace mechanism in the following form.

Proposition 7. *Let $v_1, v_2 \in \mathbb{Z}$ and $k \in \mathbb{N}$ s.t. $|v_1 - v_2| \leq k$. Then $\mathcal{L}_\epsilon(v_1) \ =^{\sharp(k \cdot \epsilon, 0)} \ \mathcal{L}_\epsilon(v_2)$.*

Proposition 7 is sufficiently general to capture most examples from the literature, but not for the examples of this paper; informally, applying Proposition 7 only allows us to prove privacy using the standard composition theorems. To see how we might generalize the principle, note that privacy from pointwise liftings (Proposition 6) involves liftings of an *asymmetric* relation, rather than equality. This suggests that it could be profitable to consider asymmetric liftings. Indeed, we propose the following generalization of Proposition 7.

Proposition 8. *Let $v_1, v_2, k \in \mathbb{Z}$. Then*

$$\mathcal{L}_\epsilon(v_1) \ \Psi^{\sharp(|k+v_1-v_2| \cdot \epsilon, 0)} \ \mathcal{L}_\epsilon(v_2),$$

where

$$\Psi = \{(x_1, x_2) \in \mathbb{Z} \times \mathbb{Z} \mid x_1 + k = x_2\}.$$

Proposition 8 has several useful consequences. For instance, when $|v_1 - v_2| \leq k$ we have $\mathcal{L}_\epsilon(v_1) \ \Psi^{\sharp(2k \cdot \epsilon, 0)} \ \mathcal{L}_\epsilon(v_2)$ with

$$\Psi = \{(x_1, x_2) \in \mathbb{Z} \times \mathbb{Z} \mid x_1 + k = x_2\}, \tag{1}$$

following from Proposition 8 and the triangle inequality

$$|v_1 - v_2| \leq k \Rightarrow |k + (v_1 - v_2)| \leq k + k = 2k.$$

Informally, this instance of Proposition 8 shows that by "paying" privacy cost ϵ, we can ensure that the samples are a certain distance apart. This stands in contrast to Proposition 7, which ensures that the samples are equal.

Another useful consequence is that adding identical noise to both v_1 and v_2 incurs no privacy cost, and we can assume the difference between the samples is the difference between v_1 and v_2.

Proposition 9. *Let $v_1, v_2 \in \mathbb{Z}$. Then $\mathcal{L}_\epsilon(v_1) \ \Psi^{\sharp(0,0)} \ \mathcal{L}_\epsilon(v_2)$, where*

$$\Psi = \{(x_1, x_2) \in \mathbb{Z} \times \mathbb{Z} \mid x_1 - x_2 = v_1 - v_2\}.$$

3. Formalization in a program logic

In this section we present a new program logic called apRHL^+ for reasoning about differential privacy of programs written in a core programming language with samplings from the Laplace mechanism and the one-sided Laplace Mechanism. Our program logic apRHL^+ extends apRHL, a relational Hoare logic that has been used to verify many examples of differentially private algorithms [4].

We will use a standard imperative language with a sampling command for the Laplace distribution. We omit the grammar here.

The semantics of programs is standard [4, 14]. We first define the set Mem of memories to contain all well-typed functions from variables to values. Then, commands are interpreted as functions from memories to distributions over memories.

Assertions and judgments Assertions in the logic are first-order formulae over generalized expressions. The latter are expressions built from tagged variables $x\langle 1\rangle$ and $x\langle 2\rangle$, where the tag is used to determine whether the interpretation of the variable is taken in the first memory or in the second memory. For instance, $x\langle 1\rangle = x\langle 2\rangle+1$ is the assertion which states that the interpretation of the variable x in the first memory is equal to the interpretation of the variable x in the second memory plus 1. More formally, assertions are interpreted as predicates over pairs of memories. We let $[\![\Phi]\!]$ denote the set of memories (m_1, m_2) that satisfy Φ. The interpretation is standard (besides the use of tagged variables) and is omitted. By abuse of notation, we write $e\langle 1\rangle$ or $e\langle 2\rangle$, where e is a program expression, to denote the generalized expression built according to e, but in which all variables are tagged with a $\langle 1\rangle$ or $\langle 2\rangle$, respectively.

Judgments in both apRHL and apRHL^+ are of the form

$$\vdash c_1 \sim_{\langle \epsilon,\delta\rangle} c_2 : \Phi \Longrightarrow \Psi$$

where c_1 and c_2 are statements, the precondition Φ and postcondition Ψ are relational assertions, and ϵ and δ are non-negative reals.Informally, a judgment of the above form is valid if the two distributions produced by the executions of c_1 and c_2 on any two initial memories satisfying the precondition Φ are related by the (ϵ, δ)-lifting of the postcondition Ψ. Formally, the judgment

$$\vdash c_1 \sim_{\langle \epsilon,\delta\rangle} c_2 : \Phi \Longrightarrow \Psi$$

is *valid* iff for every two memories m_1 and m_2, such that $m_1\ [\![\Phi]\!]\ m_2$, we have

$$([\![c_1]\!]_{m_1})\ [\![\Psi]\!]^{\sharp(\epsilon,\delta)}\ ([\![c_2]\!]_{m_2}).$$

Proof system We defer the presentation of the proof system of apRHL to the extended version.

Figure 1 collects the new rules in apRHL^+, which are all derived from the new proof principles we saw in the previous section. The first rule [FORALL-EQ] allows proving differential privacy via pointwise privacy; this rule reflects Proposition 6.

The next pair of rules, [LAPGEN] and [LAPNULL], reflect the liftings of the distributions of the Laplace mechanism presented in Propositions 8 and 9 respectively. Note that we need a side-condition on the free variables in [LAPNULL]—otherwise, the sample may change e_1 and e_2.

4. Above Threshold algorithm

The *Sparse Vector* algorithm is the canonical example of a program whose privacy proof goes beyond proofs of privacy primitives and composition theorem. The core of the algorithm is the Above Threshold algorithm. In this section, we prove that the latter (as modeled by the program AboveT) is $(\epsilon, 0)$-differentially private; privacy for the full mechanism follows by sequential composition.

Informal proof By Proposition 6, it suffices to show that for every integer i, the output of AboveT on two adjacent databases yields two sub-distributions over Mem that are related by the $(\epsilon, 0)$-lifting of the interpretation of the assertion

$$r\langle 1\rangle = i \Rightarrow r\langle 2\rangle = i.$$

The coupling proof goes as follows. We start by coupling the samplings of the noisy thresholds so that $T\langle 1\rangle + 1 = T\langle 2\rangle$; the cost of this coupling is $(\epsilon/2, 0)$. For the first $i-1$ queries, we couple the samplings of the noisy query outputs using the rule [LAPNULL]. By 1-sensitivity of the queries and adjacency of the two databases, we know $\mathsf{evalQ}(Q[j], d)\langle 2\rangle - \mathsf{evalQ}(Q[j], d)\langle 1\rangle \le 1$, so

$$S\langle 1\rangle < T\langle 1\rangle \Rightarrow S\langle 2\rangle < T\langle 2\rangle.$$

Thus, if side $\langle 1\rangle$ does not change the value of r, neither does side $\langle 2\rangle$. In fact, we have the stronger invariant

$$r\langle 1\rangle = |Q|+1 \Rightarrow r\langle 2\rangle = |Q|+1 \wedge (r\langle 1\rangle = |Q|+1 \vee r\langle 1\rangle < i),$$

where $r = |Q| + 1$ means that the loop has not exceeded the threshold yet.

When we reach the ith iteration and $i < |Q| + 1$, we couple the samplings of S so that $S\langle 1\rangle + 1 = S\langle 2\rangle$; the cost of this coupling is $(\epsilon/2, 0)$. Because $T\langle 1\rangle + 1 = T\langle 2\rangle$ and $S\langle 1\rangle + 1 = S\langle 2\rangle$, we enter the conditional in the second execution as soon as we enter the conditional in the first execution. For the remaining iterations $r > i$, it is easy to prove

$$r\langle 1\rangle = i \Rightarrow r\langle 2\rangle = i.$$

Formal proof We prove the following apRHL^+ judgment, which entails $(\epsilon, 0)$-differential privacy:

$$\vdash \mathsf{AboveT} \sim_{\langle \epsilon,0\rangle} \mathsf{AboveT} : \Phi \Longrightarrow r\langle 1\rangle = r\langle 2\rangle,$$

where Φ denotes the precondition

$$\begin{array}{ll} & \mathsf{adj}(d\langle 1\rangle, d\langle 2\rangle) \\ \wedge & t\langle 1\rangle = t\langle 2\rangle \\ \wedge & Q\langle 1\rangle = Q\langle 2\rangle \\ \wedge & \forall j.\ |\mathsf{evalQ}(Q\langle 1\rangle[j], d\langle 1\rangle) - \mathsf{evalQ}(Q\langle 2\rangle[j], d\langle 2\rangle)| \le 1. \end{array}$$

The conjuncts of the precondition are straightforward: the first states that the two databases are adjacent, the second and third state that Q and t coincide in both runs, and the last states that all queries are 1-sensitive. By the rule [FORALL-EQ], it suffices to prove

$$\vdash \mathsf{AboveT} \sim_{\langle \epsilon,0\rangle} \mathsf{AboveT} : \Phi \Longrightarrow (r\langle 1\rangle\ = i) \Rightarrow (r\langle 2\rangle\ = i).$$

for every $i \in \mathbb{Z}$.

We begin with the three initializations:

$$\begin{array}{l} j \leftarrow 1; \\ r \leftarrow |Q| + 1; \\ T \xleftarrow{\$} \mathcal{L}_\epsilon(t); \end{array}$$

This command c_0 computes a noisy version of the threshold t. We use the rule [LAPGEN] with $\epsilon = \epsilon/2$, $k = 1$ and $k' = k$, noticing that t is the same value in both sides. This proves the judgment

$$\vdash c_0 \sim_{\epsilon/2} c_0 : \Phi \Longrightarrow T\langle 1\rangle + 1 = T\langle 2\rangle.$$

Notice that the $\epsilon/2$ we are paying here is *not* for the privacy of the threshold—which is not private information!—but rather for ensuring that the noisy thresholds are *one apart* in the two runs.

Next, we consider the main loop c_1:

$$\begin{array}{l} \mathsf{while}\ j < |Q|\ \mathsf{do} \\ \quad S \xleftarrow{\$} \mathcal{L}_{\epsilon/4}(\mathsf{evalQ}(Q[j], d)); \\ \quad \mathsf{if}\ (T \le S\ \wedge r = |Q| + 1)\ \mathsf{then}\ r \leftarrow j; \\ \quad j \leftarrow j + 1; \end{array}$$

and prove the judgment

$$\vdash c_1 \sim_{\epsilon/2} c_1 : \Phi \wedge T\langle 1\rangle + 1 = T\langle 2\rangle \Longrightarrow (r\langle 1\rangle\ = i) \Rightarrow (r\langle 2\rangle\ = i)$$

with the [WHILEEXT] rule.

References

[1] M. Barr. Relational algebras. In S. Mac Lane, editor, *Reports of the Midwest Category Seminar, IV*, volume 137 of *Lecture Notes in Mathematics*, page 39–55. Springer-Verlag, 1970.

[2] G. Barthe and F. Olmedo. Beyond differential privacy: Composition theorems and relational logic for f-divergences between probabilistic programs. In *International Colloquium on Automata, Languages and Programming (ICALP), Riga, Latvia*, volume 7966 of *Lecture Notes in Computer Science*, pages 49–60. Springer, 2013.

$$\frac{\forall i.\vdash c_1 \sim_{\langle \epsilon,\delta_i \rangle} c_2 : \Phi \Longrightarrow x\langle 1\rangle = i \Rightarrow x\langle 2\rangle = i \qquad \sum_{i\in I}\delta_i \le \delta}{\vdash c_1 \sim_{\langle \epsilon,\delta \rangle} c_2 : \Phi \Longrightarrow x\langle 1\rangle = x\langle 2\rangle}\,[\textsc{Forall-Eq}]$$

$$\frac{}{\vdash y_1 \xleftarrow{\$} \mathcal{L}_\epsilon(e_1) \sim_{\langle k'\cdot\epsilon,0\rangle} y_2 \xleftarrow{\$} \mathcal{L}_\epsilon(e_2) : |k + e_1\langle 1\rangle - e_2\langle 2\rangle| \le k' \Longrightarrow y_1\langle 1\rangle + k = y_2\langle 2\rangle}\,[\textsc{LapGen}]$$

$$\frac{y_1 \notin FV(e_1) \qquad y_2 \notin FV(e_2)}{\vdash y_1 \xleftarrow{\$} \mathcal{L}_\epsilon(e_1) \sim_{\langle 0,0\rangle} y_2 \xleftarrow{\$} \mathcal{L}_\epsilon(e_2) : \top \Longrightarrow y_1\langle 1\rangle - y_2\langle 2\rangle = e_1\langle 1\rangle - e_2\langle 2\rangle}\,[\textsc{LapNull}]$$

Figure 1. Selected proof rules from apRHL$^+$

[3] G. Barthe, B. Grégoire, and S. Zanella-Béguelin. Formal certification of code-based cryptographic proofs. In *ACM SIGPLAN–SIGACT Symposium on Principles of Programming Languages (POPL), Savannah, Georgia*, pages 90–101, New York, 2009.

[4] G. Barthe, B. Köpf, F. Olmedo, and S. Zanella-Béguelin. Probabilistic relational reasoning for differential privacy. *ACM Transactions on Programming Languages and Systems*, 35(3):9, 2013.

[5] G. Barthe, M. Gaboardi, E. J. Gallego Arias, J. Hsu, C. Kunz, and P.-Y. Strub. Proving differential privacy in Hoare logic. In *IEEE Computer Security Foundations Symposium (CSF), Vienna, Austria*, 2014.

[6] G. Barthe, T. Espitau, B. Grégoire, J. Hsu, L. Stefanesco, and P.-Y. Strub. Relational reasoning via probabilistic coupling. In *International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR), Suva, Fiji*, volume 9450, pages 387–401, 2015.

[7] G. Barthe, M. Gaboardi, E. J. Gallego Arias, J. Hsu, A. Roth, and P.-Y. Strub. Higher-order approximate relational refinement types for mechanism design and differential privacy. In *ACM SIGPLAN–SIGACT Symposium on Principles of Programming Languages (POPL), Mumbai, India*, 2015.

[8] G. Barthe, M. Gaboardi, B. Grégoire, J. Hsu, and P.-Y. Strub. Proving differential privacy via probabilistic couplings. In *IEEE Symposium on Logic in Computer Science (LICS), New York, New York*, 2016. To appear.

[9] N. Benton. Simple relational correctness proofs for static analyses and program transformations. In *ACM SIGPLAN–SIGACT Symposium on Principles of Programming Languages (POPL), Venice, Italy*, pages 14–25, 2004.

[10] Y. Deng and W. Du. Logical, metric, and algorithmic characterisations of probabilistic bisimulation. Technical Report CMU-CS-11-110, Carnegie Mellon University, March 2011.

[11] C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9 (3-4):211–407, 2014.

[12] C. Dwork, F. McSherry, K. Nissim, and A. Smith. Calibrating noise to sensitivity in private data analysis. In *IACR Theory of Cryptography Conference (TCC), New York, New York*, pages 265–284, 2006.

[13] B. Jonsson, W. Yi, and K. G. Larsen. Probabilistic extensions of process algebras. In *Handbook of Process Algebra*, pages 685–710. Elsevier, Amsterdam, 2001.

[14] D. Kozen. Semantics of probabilistic programs. In *IEEE Symposium on Foundations of Computer Science (FOCS), San Juan, Puerto Rico*, pages 101–114, 1979.

[15] T. Lindvall. *Lectures on the coupling method*. Courier Corporation, 2002.

[16] F. McSherry and K. Talwar. Mechanism design via differential privacy. In *IEEE Symposium on Foundations of Computer Science (FOCS), Providence, Rhode Island*, pages 94–103, 2007.

[17] H. Thorisson. *Coupling, Stationarity, and Regeneration*. Springer, 2000.