Asymmetric Distances for Approximate Differential Privacy

Dmitry Chistikov
Centre for Discrete Mathematics and its Applications (DIMAP) & Department of Computer Science, University of Warwick, UK

Andrzej S. Murawski
Department of Computer Science, University of Oxford, UK

David Purser
Centre for Discrete Mathematics and its Applications (DIMAP) & Department of Computer Science, University of Warwick, UK

Abstract
Differential privacy is a widely studied notion of privacy for various models of computation, based on measuring differences between probability distributions. We consider $(\epsilon, \delta)$-differential privacy in the setting of labelled Markov chains. For a given $\epsilon$, the parameter $\delta$ can be captured by a variant of the total variation distance, which we call $lv_{\alpha}$ (where $\alpha = e^\epsilon$).

First we study $lv_{\alpha}$ directly, showing that it cannot be computed exactly. However, the associated approximation problem turns out to be in PSPACE and $\#P$-hard. Next we introduce a new bisimilarity distance for bounding $lv_{\alpha}$ from above, which provides a tighter bound than previously known distances while remaining computable with the same complexity (polynomial time with an NP oracle). We also propose an alternative bound that can be computed in polynomial time. Finally, we illustrate the distances on case studies.

2012 ACM Subject Classification Theory of computation → Probabilistic computation

Keywords and phrases Bisimilarity distances, Differential privacy, Labelled Markov chains

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2019.10

Funding Andrzej S. Murawski: Royal Society Leverhulme Trust Senior Research Fellowship and the International Exchanges Scheme (IE161701)
David Purser: UK EPSRC Centre for Doctoral Training in Urban Science (EP/L016400/1)

Acknowledgements The authors would like to thank the reviewers for their helpful comments.

1 Introduction
Differential privacy [14] is a security property that ensures that a small perturbation of the input leads to only a small perturbation in the output, so that observing the output makes it difficult to discern whether a particular piece of information was present in the input. It has been shown that various bisimilarity distances can bound the differential privacy of a labelled Markov chain, by bounding for example the $\epsilon$ [6, 31] and $\delta$ [9] privacy parameters. Bisimilarity distances [17, 11] were introduced as a metric analogue of probabilistic bisimulation [23], to overcome the problem that bisimilarity is too sensitive to minor changes in probabilities.

We further the study of bounds to $\delta$ by defining new bisimilarity distances. The bisimilarity distance of [9], inspired by the work of [31], transpired to be computable in polynomial time with an NP oracle. The work of [31] defined distances using the Kantorovich metric and the associated bisimilarity distance based on a fixed point; and considered the effect of replacing the absolute value function with another metric. For the purposes of $(\epsilon, \delta)$-differential privacy the distance required is not a metric, nor even a pseudometric, so their methods are adapted in [9] to account for this; resulting in a distance function $bd_{\alpha}$ which can be used to bound...
Asymmetric Distances for Approximate Differential Privacy

The $\delta$ parameter in differential privacy from above. The function, however, retained the symmetry property that $bd_\alpha(s, s') = bd_\alpha(s', s)$. In this paper we further study distances to bound differential privacy in labelled Markov chains, but drop this symmetry property and discover a tighter bound, which can be computed with the same cost. We also define a weaker bisimilarity distance for bounding $\delta$ that can be computed in polynomial time.

The privacy parameter in question, $\delta$, can be expressed as a variant of the total variation distance $tv_\alpha$. In particular we define $lv_\alpha$ as a single component of $tv_\alpha$ (which is a maximum over two functions). This distance is a way of measuring the maximum difference of probabilities between any two states. Total variation distance is usually expressed using absolute difference, but for differential privacy a skew is introduced into this distance. These exact distances transpire to be very difficult to compute: we confirm that the threshold distance problem, which asks whether the distance is below a given threshold, is undecidable and approximating it is $\#P$-hard. We also show that for finite words it can be approximated in $\text{PSPACE}$. These results match the results of [22] for standard total variation distances.

We then bound the distance $lv_\alpha$ from above by a distance $ld_\alpha$ which will turn out to be computable, in a similar manner to how $bd_\alpha$ bounds $tv_\alpha$ in [9]. We show that $ld_\alpha$ can be computed in polynomial time with an $\text{NP}$ oracle (that is, with the same complexity as $bd_\alpha$). We further generalise $ld_\alpha$ to a new distance $lgd_\alpha$, computable in polynomial time. This new distance, is no smaller than $ld_\alpha$, and we conjecture it might be equal. We can then take $\max\{ld_\alpha(s, s'), ld_\alpha(s', s)\}$ and $\max\{lgd_\alpha(s, s'), lgd_\alpha(s', s)\}$ as sound upper bounds on $\delta$. Thus we have defined the first non-trivial estimate of the $\delta$ parameter that can be computed in polynomial time (trivially, always returning 1 is technically correct). Our results show that taking the maximum over two $ld_\alpha$ is a better approximation than $bd_\alpha$ from [9]. We confirm this using several case studies, where we also demonstrate, on a randomised response mechanism, that the estimates based on $ld_\alpha$ can beat standard differential privacy composition theorems. The relationships between distances are summarised in Figure 1.

Research into behavioural pseudometrics has a long history going back to Giacalone et al. [17]. Our work lies in the tradition of bisimulation pseudometrics based on the Kantorovich distance started by Desharnais et al. [11, 12], and builds upon subsequent work on computing.
them [29]. Chatzikokolakis et al. [6] generalised the pseudometric framework to handle $\epsilon$-differential privacy, and indeed arbitrary metrics, but did not consider the complexity of calculating the distances. We introduced a distance in [9] for $(\epsilon, \delta)$-differential privacy, which is improved upon in this paper. As concerns approximation, we are not aware of any related work on distances other than the total variation distance [8, 22].

2 Preliminaries

Given a finite set $X$, let $\text{Dist}(X)$ be the set of all stochastic vectors in $\mathbb{R}^X$. If $X$ is a set of symbols then $X^*$ is the set of all sequences of symbols in $X$, $X^+$ all sequences of length at least one, and $X^\omega$ all infinite sequences.

Definition 1 (labelled Markov chains (LMC’s)). A labelled Markov chain $\mathcal{M}$ is a tuple $(S, \Sigma, \mu, \ell)$, where $S$ is a finite set of states, $\Sigma$ is a finite alphabet, $\mu : S \to \text{Dist}(S)$ is the transition function and $\ell : S \to \Sigma$ is the labelling function.

We assume that all transition probabilities are rational, represented as a pair of binary integers. size$(\mathcal{M})$ is the number of bits required to represent $\langle S, \Sigma, \mu, \ell \rangle$, including the bit size of the probabilities. We will write $\mu_s$ for $\mu(s)$.

In what follows, we study probabilities associated with infinite sequences of labels. However, Markov chains with labelled transitions can also be described in the framework of that definition.

Definition 2. A subset $C \subseteq \Sigma^\omega$ is a cylinder set if there exists $u \in \Sigma^*$ such that $C$ consists of all infinite sequences from $\Sigma^\omega$ whose prefix is $u$. We then write $C_u$ to refer to $C$.

Cylinder sets play a prominent role in measure theory in that their finite unions can be used as a generating family (an algebra) for the set $\mathcal{F}_\Sigma$ of measurable subsets of $\Sigma^\omega$ (the cylindrical $\sigma$-algebra). Where clear from context we will omit $\Sigma$ in the subscript of $\mathcal{F}$. What will be important for us is that any measure $\nu$ on $(\Sigma^\omega, \mathcal{F}_\Sigma)$ is uniquely determined by its values on cylinder sets [5, Chapter 1, Section 2][2, Section 10.1]. Next we show how to assign a measure $\nu_s$ on $(\Sigma^\omega, \mathcal{F}_\Sigma)$ to an arbitrary state of an LMC $\mathcal{M}$.

Definition 3. Given $\mathcal{M} = \langle S, \Sigma, \mu, \ell \rangle$, let $\mu^+ : S^+ \to [0, 1]$ and $\ell^+ : S^+ \to \Sigma^+$ be the natural extensions of the functions $\mu$ and $\ell$ to $S^+$, i.e. $\mu^+(s_0 \cdots s_k) = \prod_{i=0}^{k-1} \mu_{s_i}(s_{i+1})$ and $\ell^+(s_0 \cdots s_k) = \ell(s_0) \cdots \ell(s_k)$, where $k \geq 0$ and $s_i \in S$ ($0 \leq i \leq k$). Note that, for any $s \in S$, we have $\mu^+(s) = 1$. Given $s \in S$, let $\text{Paths}_s(\mathcal{M})$ be the subset of $S^+$ consisting of all sequences that start with $s$.

Definition 4. Let $\mathcal{M} = \langle S, \Sigma, \mu, \ell \rangle$ and $s \in S$. We define $\nu_s : \mathcal{F}_\Sigma \to [0, 1]$ to be the unique measure on $(\Sigma^\omega, \mathcal{F}_\Sigma)$ such that for any cylinder $C_u$ we have $\nu_s(C_u) = \sum \mu^+(p)$ where the summation is over $p \in \text{Paths}_s(\mathcal{M})$ such that $\ell^+(p) = u$.

Example 5 (transition-labelled LMC’s). Like in [29, 7, 1, 27, 9], Definition 1 features labelled states. However, Markov chains with labelled transitions can also be described in the framework of that definition.

In particular, suppose we are given a chain $\mathcal{M}$ of the form $(S, \Sigma, T)$, where $S$ is a finite set of states, $\Sigma$ is a finite alphabet and $T : S \to \text{Dist}(S \times \Sigma)$ is the transition function. We write each transition as $q \xrightarrow{a} q'$, meaning that $T(q)(q', a) = p$. From this transition-labelled LMC, we create an equivalent state-labelled Markov chain $\mathcal{M}'$: for each state and each label, add new state $(q, a)$ labelled with $a$, such that, when $q \xrightarrow{b} q'$, we have $\mu_{(q,a)}((q', b)) = p$ for every
a ∈ Σ. Technically, this delays reading of the first character until the second state visited. To account for this, introduce an additional character, say ∗, so that νs(Cw) = νs′(Cw), where ν and ν′ refer to the measures associated with M and M′ respectively (Definition 4).

▶ Example 6 (finite-word LMC’s). We can also describe labelled Markov chains over finite words. These chains have a set of final states F, which have no outgoing transitions. We require positive probability of reaching a final state from every reachable state. We define the function νs(w) = ∑ µ+(p), where the summation is over p ∈ Paths+(M) such that ℓ+(p) = w and p[w] ∈ F, so that we only consider paths which end in a final state. The function can be extended to sets of words E ⊆ Σ* (which are countable) by νs(E) = ∑w∈E νs(w).

Such machines can also be represented by infinite-word Markov chains. One can simulate the end of the word by an additional character, say $ such that, for q ∈ F, µq(q) = 1 and ℓ(q) = $, so that the only trace that can be observed from q is $$. Then, for a word w ∈ Σ*, we rather study w$$…, corresponding to the cylinder Cw$$. In the translated infinite-word model, the event Cw corresponds to the event \{w ∈ Σ* | prefix(w) = u\} in the original finite-word model. Some of our arguments will be carried out in the finite-word setting, as hardness results that apply to these chains also apply to infinite-word Markov chains. Other arguments will only be possible in the finite-word setting.

Let us return to the general definition of Markov chains (Definition 1). Our aim will be to compare states from the point of view of differential privacy. Any two states s, s′ can be viewed as indistinguishable if νs(E) = νs′(E) for every E ∈ F. More generally, the difference between them can be quantified using the total variation distance, defined by tv(ν, ν′) = supE∈F |ν(E) − ν’(E)|. Given M = (S, Σ, μ, ℓ) and s, s′ ∈ S, we shall write tv(s, s′) to refer to tv(νs, νs′). Ensuring such pairs of measures (νs, νs′) are “similar” is essential for privacy, so that it is difficult to observe which of the states was the originating position. To measure probabilities relevant to differential privacy, we will need to study a more general variant hv for the above distance, which we introduce shortly.

3 (ε, δ)-Differential Privacy

Differential privacy is a mathematically rigorous definition of privacy due to Dwork et al [14]; the aim is to ensure that inputs which are related in some sense lead to very similar outputs. Formally it requires that for two related states there only ever be a small change in output probabilities, and therefore discerning which of the two states was actually used is difficult, maintaining their privacy. We rely on the definition of approximate differential privacy in the context of labelled Markov chains, as per [9].

▶ Definition 7. Let M = (S, Σ, μ, ℓ) be a labelled Markov chain and let R ⊆ S × S be a symmetric relation. Given ε ≥ 0 and δ ∈ [0, 1], we say that M is (ε, δ)-differentially private w.r.t. R if, for every s, s′ ∈ S such that (s, s′) ∈ R, we have νs(E) ≤ eε · νs′(E) + δ for every measurable set E ∈ F.

What it means for two states to be related, as specified by R, is to a large extent domain-specific. In general, R makes it possible to spell out which states should not appear too different and, consequently, should enjoy a quantitative amount of privacy.

Note that each state s ∈ S can be viewed as defining a random variable Xs with outcomes from Σω such that P[Xs ∈ E] = νs(E). Then the above can be rewritten as

\[ P[Xs ∈ E] ≤ e^{\epsilon} P[Xs′ ∈ E] + \delta, \]

which matches the definition from [14], where one would consider Xs, Xs′ neighbouring in some natural sense. In the typical database scenario, one
would relate database states that differ by exactly one entry. In our setting, we refer to states of a machine, for which we would like it to be indiscernible as to which was the start state, assuming that the states are hidden and the traces are observable.

When \( \delta = 0 \), we use the term \( \epsilon \)-differential privacy, which amounts to measuring the ratio between the probabilities of possible outcomes. When one cannot expect to achieve this pure \( \epsilon \)-differential privacy, the relaxed approximate differential privacy is used [22]. When \( \epsilon = 0 \), \( \delta \) is captured exactly by the statistical distance (total variation distance) \( tv \).

Our aim is to capture the value of \( \delta \) required to satisfy the differential privacy property for a given \( \epsilon \). That is, given a LMC \( M \), a symmetric relation \( R \) and \( \alpha = \epsilon \geq 1 \), we want to determine the smallest \( \delta \) such that \( M \) is \((\epsilon, \delta)\)-differentially private with respect to \( R \). We can measure the difference between two measures \( \nu, \nu' \) on \( (\Sigma^\omega, F) \) as follows:

\[
\Delta_{\alpha}(\nu, \nu') = \sup_{E \in F} \Delta_{\alpha}(\nu(E), \nu'(E)) \quad \text{where} \quad \Delta_{\alpha}(a, b) = \max\{a - \alpha b, b - \alpha a, 0\} \quad [3].
\]

When used on \( \nu_s, \nu_{s'} \) and \( \alpha = \epsilon' \), \( \Delta_{\alpha}(s, s') \) gives the required \( \delta \) between states \( s, s' \) [9].

In this paper we observe that significant simplification occurs by splitting the two main parts of the maximum, taking only the “left variant”. Whilst \( \Delta_{\alpha} \) is symmetric, we break this property to introduce a new distance function \( \Lambda_{\alpha} \) (similarly to [4]). Then we define an analogous total variation distance \( lv_{\alpha} \), which will be our main object of study.

**Definition 8** (Asymmetric skewed total variation distance). Let \( \alpha \geq 1 \). Given two measures \( \nu, \nu' \) on \( (\Sigma^\omega, F) \), let \( lv_{\alpha}(\nu, \nu') = \sup_{E \in F} \Lambda_{\alpha}(\nu(E), \nu'(E)) \), where \( \Lambda_{\alpha}(a, b) = \max\{a - \alpha b, b - \alpha a, 0\} \).

We will write \( lv_{\alpha}(s, s') \) for \( lv_{\alpha}(\nu_s, \nu_{s'}) \). Note that it is not required to take the maximum with zero, that is \( lv_{\alpha}(\nu, \nu') = \sup_{E \in F} \nu(E) - \alpha \nu'(E) \), since there is always an event such that \( \nu'(E) = 0 \), in particular \( \nu(0) = 0 \). Observe that \( \Delta_{\alpha} \) and \( \Lambda_{\alpha} \) are not metrics as \( \Delta_{\alpha}(a, b) = 0 \implies a = b \), and in fact not even pseudometrics as the triangle inequality does not hold. Our new distance \( \Lambda_{\alpha} \) (and \( lv_{\alpha} \)) is not symmetric, while \( \Delta_{\alpha} \) and \( tv_{\alpha} \) are.

If \( \alpha = 1 \), then \( tv_1 = tv_1 = tv \), since if \( \nu, \nu' \) are probability measures and we have \( \nu(E) = 1 - \nu(\bar{E}) \) then \( \sup_{E \in F} |\nu(E) - \nu'(E)| = \sup_{E \in F} \nu(E) - \nu'(E) = \sup_{E \in F} \nu'(E) - \nu(E) \), i.e., despite the use of the absolute value in the definition of \( tv \), it is not required.

We can reformulate differential privacy in terms of \( tv_{\alpha} \) and \( lv_{\alpha} \).

**Proposition 9.** Given a labelled Markov chain \( M \) and a symmetric relation \( R \subseteq S \times S \), the following properties are equivalent for \( \alpha = \epsilon' \):

\[
\begin{align*}
M & \text{ is } (\epsilon, \delta)\text{-differentially private w.r.t. } R, \\
\max_{(s, s') \in R} lv_{\alpha}(s, s') & \leq \delta, \text{ and} \\
\max_{(s, s') \in R} lv_{\alpha}(s, s') & \leq \delta.
\end{align*}
\]

We now focus on computing \( lv_{\alpha} \), since this will allow us to determine the “level” of differential privacy for a given \( \epsilon \). Henceforth we will refer to \( \epsilon' \) as \( \alpha \). For the purposes of our complexity arguments, we will only use rational \( \alpha \) with \( O(\text{size}(M)) \)-bit representation.

Section 4 \( lv_{\alpha} \) is not computable

\( tv(s, s') \) turns out to be surprisingly difficult to compute: the threshold distance problem (whether the distance is strictly greater than a given threshold) is undecidable, and the non-strict variant of the problem (“greater or equal”) is not known to be decidable [22]. The undecidability result is shown by reduction from the emptiness problem for probabilistic automata to the threshold distance problem for finite-word transition-labelled Markov chains. Recall that such chains are a special case of our more general definition of infinite-word state-labelled Markov chains. Thus, the problem is undecidable in this case also.
Since $\ell v = lv_1$, we know that $lv_1(s, s') > \theta$ is undecidable. We show that this is not special, that is, the problem remains undecidable for any fixed $\alpha > 1$. In other words, no value of the privacy parameter $\epsilon$ makes it possible to compute the optimal $\delta$ exactly.

**Theorem 10.** Finding a value of $tv$ reduces in polynomial time to finding a value of $lv_\alpha$.

**Proof.** Given a labelled Markov chain $\mathcal{M} = (Q, \Sigma, \mu, \ell)$, and states $q, q'$ for which we require the answer $tv(q, q')$, we construct a new labelled Markov chain $\mathcal{M}'$, for which $lv_\alpha(s, s') = tv(q, q')$.

We define $\mathcal{M}' = (Q \cup \{s', \perp\}, \Sigma', \mu', \ell')$, with $\ell'(s) = \ell'(s') = \triangleright$, $\ell'(<) = <$, $\ell'(x) = \ell(x)$ for all $x \in Q$, $\Sigma' = \Sigma \cup \{\triangleright, <\}$,

$$
\mu'_x(q) = 1, \quad \mu'_\ell(q') = \frac{1}{\alpha}, \quad \mu'_\ell(\perp) = \frac{\alpha - 1}{\alpha}, \quad \text{and} \quad \mu'_x(y) = \mu_x(y) \text{ for all } x, y \in Q.
$$

The reduction, sketched in Figure 2, adds three new states, so can be done in polynomial time. We claim $lv_\alpha(s, s') = tv(q, q')$.

Consider $E \in \mathcal{F}_\Sigma$, observe that $\nu_q(E) = \nu_s(E')$ and $\nu_{q'}(E) = \alpha \nu_{s'}(E')$, where $E' = \{w \mid \triangleright w \in E \} \in \mathcal{F}_{\Sigma'}$. Then $\nu_q(E) - \nu_{q'}(E) = \nu_s(E') - \alpha \nu_{s'}(E')$ and $lv_\alpha(s, s') \geq tv(q, q')$.

Conversely, consider an event $E' \in \mathcal{F}_{\Sigma'}$. Since the character $<$ can only be reached from $s'$, any word using it contributes negatively to the difference. Hence intersecting the event with $\triangleright \Sigma'$, to remove $<$, can only increase the difference. The character $\triangleright$ must occur (only) as the first character of every (useful) word in $E'$. Let $E = \{w \mid \triangleright w \in E' \cap \triangleright \Sigma''\} \in \mathcal{F}_\Sigma$, then $\nu_q(E) - \nu_{q'}(E) \geq \nu_s(E') - \alpha \nu_{s'}(E')$. Thus $tv(q, q') \geq lv_\alpha(s, s')$.

Since an oracle to solve decision problems for $lv_\alpha$ would solve problems for $tv$, we obtain the following result.

**Corollary 11.** $lv_\alpha(s, s') > \theta$ is undecidable for $\alpha \geq 1$.

It is not clear that $lv_\alpha$ reduces easily to $tv$. Arguments along the lines of the proof of Theorem 10 may not result in a Markov chain due to non-stochastic transitions, or modifications to the $s \rightarrow q$ branch may result in new maximising events.
Approximation of $lv_\alpha$

Given that $lv_\alpha$ cannot be computed exactly, we turn to approximation: the problem, given $\gamma > 0$, of finding some $x$ such that $|x - lv_\alpha(s, s')| \leq \gamma$. For $\alpha = 1$, it is known that approximating $lv = lv_1$ is possible in PSPACE but $\#P$-hard [8, 22]. We show that the case $\alpha = 1$ is not special; that is, when $\alpha > 1$, $lv_\alpha$ can also be approximated and the same complexity bounds apply.

Remark. Typically one might suggest being $\epsilon$ close ($|x - lv_\alpha(s, s')| \leq \epsilon$). To avoid confusion with the differential privacy parameter, we refer to $\gamma$ close.

Theorem 12. For finite-word Markov chains, approximation of $lv_\alpha(s, s')$ within $\gamma$ can be performed in PSPACE and is $\#P$-hard.

Proof (sketch). For the upper bound, we show that the $i$th bit of an $x$ such that $|x - lv_\alpha(s, s')| \leq \gamma$ can be found in PSPACE. The approach, inspired by [22], is to consider the maximising event of $lv_\alpha(s, s') = \sup_{E \subseteq \Sigma} \nu_\alpha(E) - \alpha \nu_{s'}(E)$, which turns out to be $W = \{w \mid \nu_s(w) \geq \alpha \nu_{s'}(w)\}$. Then this choice of the maximising event only applies to finite-word Markov chains, thus the proof does not extend to full generality to infinite-word Markov chains. The shape of the event is the key difference between our proof and [22], which uses events of the form $\{w \mid \nu_s(w) \geq \nu_{s'}(w)\}$.

Let $\overline{W}$ denote the complement of $W$ and let $\nu_s(\overline{W})$ be approximated by a number $X$ and $\nu_{s'}(W)$ by a number $Y$. Normally, one would expect $X$ to be close to $\nu_s(\overline{W})$ and $Y$ to be close to $\nu_{s'}(W)$. Here, the trick is to require only that $\nu_s(\overline{W}) + \alpha \nu_{s'}(W)$ be close to $X + Y$. It is then argued that, for specific $X, Y$ with this property, one can find any bit of $X + Y$.

For the lower bound, we note that approximating $lv$ is $\#P$-hard [22], by a reduction from $\#NFA$, a $\#P$-complete problem [20]. That is, given a non-deterministic finite automaton $A$ and $n \in \mathbb{N}$ in unary, determine $|\Sigma^n \cap L(A)|$, the number of accepted words of $A$ of length $n$. Since $tv$ can be reduced to $lv_\alpha$ (Theorem 10), approximating $lv_\alpha$ is $\#P$-hard as well. The hardness result applies to finite-word transition-labelled Markov chains, thus also to the more general infinite-word labelled Markov chains.

A least fixed point bound $ld_{\alpha}$

We seek to bound $lv_\alpha$ from above by a computable quantity, and will introduce a distance function $ld_{\alpha}$ for this. We first introduce a variant of the Kantorovich lifting as a technique to measure the distance between probability distributions on a set $X$, given a distance function between objects of $X$. We show that $lv_\alpha$ can be reformulated using such a distance over the (infinite) trace distributions $\nu_s, \nu_{s'}$. We then define an alternative distance function between states, $ld_{\alpha}$, as the fixed point of the Kantorovich lifting of distances from individual states to (finite) state distributions. We will observe that it is possible to compute and acts as a sound bound on $lv_\alpha$.

We use this distance to determine $(\epsilon, \delta)$-differential private w.r.t. relation $R$ by bounding $\delta$ with $\max_{(s, s') \in R} ld_{\alpha}(s, s')$. We will show this can be achieved in polynomial time with access to an NP oracle, by computing $ld_{\alpha}(s, s')$ exactly in this time (if $R$ is polynomial with respect to the size of $M$). This suggests a complexity lower than approximation (which is $\#P$-hard by Theorem 12).
Definition 13 (Asymmetric Skewed Kantorovich Lifting). For a set $X$, given $d : X \times X \to [0,1]$ a distance function and measures $\mu, \mu'$, we define
\[
K_\alpha^d(\mu, \mu') = \sup_{f : X \to [0,1]} \Lambda_\alpha \left( \int_X f d\mu, \int_X f d\mu' \right)
\]
where $f$ ranges over functions which are measurable w.r.t. $\mu$ and $\mu'$.

Remark. The (standard) Kantorovich distance lifts a distance function $d$ over the ground objects $X$ to a distance between measures $\mu, \mu'$ on the set $X$. This is equivalent to replacing $\Lambda_\alpha$ with the absolute distance function ($\text{abs}(a, b) = |a - b|$). We note that $K_\alpha^d(\mu)$ is equivalent to the standard Kantorovich distance for $\alpha = 1$ and $d$ symmetric [21, 10]. If $|X| < \infty$ (for example when $X$ is a finite set of states, $S$), we have $\int_X f d\mu = \sum_{x \in X} f(x) \mu(x)$.

The (standard) Kantorovich distance lifts a distance function $K_\alpha^d$ over infinite words ($d : \Sigma^\omega \times \Sigma^\omega \to [0,1]$). We note that the absolute value function was replaced by any metric $d'$. Our lifting $K_\alpha^d$ does not quite fit in this framework, since $\Lambda_\alpha$ is not metric.

The interest in $K_\alpha^d$ is that it allows us to reformulate the definition of the distance function $l_{\alpha_\omega}$. Our goal is to measure the difference between measures over infinite traces $\nu_s, \nu_{s'}$, and so we lift a distance function over infinite words ($d : \Sigma^\omega \times \Sigma^\omega \to [0,1]$). In particular, we lift the discrete metric $1_{\neq}$ (the indicator function over inequality with $1_{\neq}(w, w') = 1$ for $w \neq w'$, and 0 otherwise).

Lemma 14. $l_{\alpha_\omega}(s, s') = K_\alpha^d(1_{\neq})(\nu_s, \nu_{s'})$.

Since computing $l_{\alpha_\omega}$, or now $K_\alpha^d(1_{\neq})(\nu_s, \nu_{s'})$, is difficult, we introduce an upper bound on $l_{\alpha_\omega}$, inspired by bisimilarity distances, which we will call $ld_{\alpha_\omega}$. This will be the least fixed point of $\Gamma_\alpha^d$, a function which measures (relative to a distance function $d$) the distance between the transition distributions of $s, s'$ where $s, s'$ share a label, or 1 when they do not.

Definition 15. Let $\Gamma_\alpha^d : [0,1]^{S \times S} \to [0,1]^{S \times S}$ be defined as follows.
\[
\Gamma_\alpha^d(d)(s, s') = \begin{cases} 
K_\alpha^d(d)(\mu_s, \mu_{s'}) & \ell(s) = \ell(s') \\
1 & \text{otherwise}
\end{cases}
\]

The utility of this function is that we are not now using the Kantorovich lifting over infinite traces distributions, but rather over finite transition distributions ($\mu_s \in \text{Dist}(S)$).

Note that $[0,1]^{S \times S}$ equipped with the pointwise order, written $\subseteq$, is a complete lattice and that $\Gamma_\alpha^d$ is monotone with respect to that order (larger $d$ permit more functions, thus larger supremum). Consequently, $\Gamma_\alpha^d$ has a least fixed point [28]. We take our distance to be exactly that point.

Definition 16. Let $ld_{\alpha_\omega} : S \times S \to [0,1]$ be the least fixed point of $\Gamma_\alpha^d$.

To provide a guarantee of privacy we require a sound upper bound on $l_{\alpha_\omega}$.

Theorem 17. $l_{\alpha_\omega}(s, s') \leq ld_{\alpha_\omega}(s, s')$ for every $s, s' \in S$.
When we refer to distance between states ($\gamma_i$), we can use a similar formula to approximate bisimilarity distances. The problem can be solved in polynomial time with access to an $\text{NP}$ oracle.

The standard variant of the Kantorovich metric is often presented in its dual formulation. Consider a matrix $A$, let $A^T$ be its transpose. Consider $bd_{\alpha}$ and $ld_{\alpha}$ as matrices. $bd_{\alpha}$ is the least fixed point of $\Gamma_{\alpha}^d$, so $\Gamma_{\alpha}^d(bd_{\alpha})(s,s') = bd_{\alpha}(s,s')$. Also notice that $\Gamma_{\alpha}^d(bd_{\alpha})(s,s') \subseteq K_{\alpha}^d(bd_{\alpha})$, since $K_{\alpha}^d(bd_{\alpha}) \subseteq K_{\alpha}^d(bd_{\alpha})$. To see this, note that, because $bd_{\alpha} = bd_{\alpha}^T$, the relevant set of functions is the same, but the objective function in the supremum is smaller.

Hence $\Gamma_{\alpha}^d(bd_{\alpha}) \subseteq ld_{\alpha}$, i.e. $bd_{\alpha}$ is also a pre-fixed point of $\Gamma_{\alpha}^d$. Since $ld_{\alpha}$ is the least pre-fixed point of $\Gamma_{\alpha}^d$, then we know $ld_{\alpha} \subseteq bd_{\alpha}$. By symmetry, $bd_{\alpha} = bd_{\alpha}^T$, giving $ld_{\alpha} \subseteq bd_{\alpha}^T$ and then $bd_{\alpha}^T \subseteq bd_{\alpha}$. We conclude $\max \{ld_{\alpha}(s,s'), ld_{\alpha}(s',s)\} \leq bd_{\alpha}(s,s')$ for every $s,s' \in S$.

Remark. Example 32 on page 13 demonstrates the inequality in Theorem 18 can be strict.

The standard variant of the Kantorovich metric is often presented in its dual formulation. In the case of finite distributions, the asymmetric skewed Kantorovich distance exhibits a dual form. This is obtained through the standard recipe for dualising linear programming. Interestingly, this technique yields a linear optimisation problem over a polytope independent of $d$, and that will prove useful in the computation of $ld_{\alpha}$.

**Theorem 18.** $\max \{ld_{\alpha}(s,s'), ld_{\alpha}(s',s)\} \leq bd_{\alpha}(s,s')$ for every $s,s' \in S$.

**Proof.** Given a matrix $A$, let $A^T$ be its transpose. Consider $bd_{\alpha}$ and $ld_{\alpha}$ as matrices. $bd_{\alpha}$ is the least fixed point of $\Gamma_{\alpha}^d$, so $\Gamma_{\alpha}^d(bd_{\alpha})(s,s') = bd_{\alpha}(s,s')$. Also notice that $\Gamma_{\alpha}^d(bd_{\alpha})(s,s') \subseteq K_{\alpha}^d(bd_{\alpha})$, since $K_{\alpha}^d(bd_{\alpha}) \subseteq K_{\alpha}^d(bd_{\alpha})$. To see this, note that, because $bd_{\alpha} = bd_{\alpha}^T$, the relevant set of functions is the same, but the objective function in the supremum is smaller.

Hence $\Gamma_{\alpha}^d(bd_{\alpha}) \subseteq ld_{\alpha}$, i.e. $bd_{\alpha}$ is also a pre-fixed point of $\Gamma_{\alpha}^d$. Since $ld_{\alpha}$ is the least pre-fixed point of $\Gamma_{\alpha}^d$, then we know $ld_{\alpha} \subseteq bd_{\alpha}$. By symmetry, $bd_{\alpha} = bd_{\alpha}^T$, giving $ld_{\alpha} \subseteq bd_{\alpha}^T$ and then $bd_{\alpha}^T \subseteq bd_{\alpha}$. We conclude $\max \{ld_{\alpha}(s,s'), ld_{\alpha}(s',s)\} \leq bd_{\alpha}(s,s')$ for every $s,s' \in S$.

**Lemma 19.** Let $X$ be finite and given $d : X \times X \to [0,1]$ a distance function, $\mu, \mu' \in \text{Dist}(X)$ we have

\[
K_{\alpha}^d(d)(\mu, \mu') = \min_{(\omega, \eta) \in \mathcal{O}_{\alpha, \mu', \mu}} \left( \sum_{s,s' \in X} \omega_{s,s'} \cdot d(s,s') + \sum_{s \in X} \eta_s \right),
\]

where

\[
\mathcal{O}_{\alpha, \mu', \mu} = \left\{ (\omega, \eta) \in [0,1]^{|X| \times |X|} \times [0,1]^{|X|} \mid \begin{align*}
\exists \gamma, \tau \in [0,1]^{|X|} : & \forall i : \sum_j \omega_{i,j} + \tau_i - \gamma_i + \eta_i = \mu(i) \\
& \forall j : \sum_i \omega_{i,j} + \frac{\tau_j - \gamma_j}{\alpha} \leq \mu'(j)
\end{align*} \right\}.
\]

When we refer to distance between states ($X = S$) we write $\mathcal{O}_{\alpha, s,s'}$ to mean $\mathcal{O}_{\alpha, \mu, \mu'}$. We take $V(\mathcal{O}_{\alpha, s,s'})$ to be the vertices of the polytope.

**Theorem 20.** $ld_{\alpha}$ can be computed in polynomial time with access to an $\text{NP}$ oracle.

We first show that the LD-threshold problem, which asks if $ld_{\alpha}(s,s') \leq \theta$, is in $\text{NP}$. This is achieved through the formula shown in Figure 3, based on Lemma 19 and [30] which used a similar formula to approximate bisimilarity distances. The problem can be solved in $\text{NP}$.
as each of the variables can be shown to be satisfied in the optimal solution with rational numbers that are of polynomial size (see [9, Theorems 1 and 2]). It suffices to guess these numbers (non-deterministically) and verify the correctness of the formula in polynomial time.

Since the threshold problem can be solved in \( \text{NP} \), we can approximate the value using binary search with polynomial overhead to arbitrary accuracy \( \gamma \), thus we find a value \( x \) such that \( |x - l_{d\alpha}(s, s')| \leq \gamma \). In fact, one can find the exact value of \( l_{d\alpha}(s, s') \) in polynomial time assuming the oracle. We can show the value of \( l_{d\alpha} \) is rational and its size is polynomially bounded, one can find it by approximation to a carefully chosen level of precision and then finding the relevant rational with the continued fraction algorithm [18, Section 5.1][16].

### 7 A greatest fixed point bound \( l_{d\alpha} \)

In the previous section we have used the least fixed point of \( \Gamma_{\alpha}^{A} \), which finds the fixed point closest to our objective \( l_{v\alpha} \). We now consider relaxing this requirement so that we can find a fixed point in polynomial time. We will introduce \( l_{d\alpha} \), expressing the greatest fixed point and represent it as a linear program that can be solved in polynomial time. Relaxing to any fixed point could of course be much worse than \( l_{d\alpha} \), so we first refine our fixed point function (\( \Gamma_{\alpha}^{A} \)) to reduce the potential gap. We do this by characterising the elements which are zero in \( l_{d\alpha} \) and fixing these as such; so that they cannot be larger in the greatest fixed point.

#### Refinement of \( \Gamma_{\alpha}^{A} \)

In the case of standard bisimulation distances the kernel of \( l_{d1} \), that is \( \{(s, s') \mid l_{d1}(s, s') = 0 \} \), is exactly bisimilarity. We consider the kernel for \( l_{d\alpha} \) and define a new relation \( \sim_{\alpha} \), which we call skewed bisimilarity, which captures zero distance.

- **Definition 21.** Let a relation \( R \subseteq S \times S \) have the property

\[
(s, s') \in R \iff \exists (\omega, \eta) \in \Omega_{s, s'}^{\alpha} \text{ s.t. } (\omega_{u, v} > 0 \implies (u, v) \in R) \land \forall u \eta_{u} = 0.
\]

Arbitrary unions of such relations also maintain the property, thus a largest such relation exists. Let \( \sim_{\alpha} \) be the largest relation with this property.

- **Remark.** When \( \alpha = 1 \) the formulation corresponds to an alternative characterisation of bisimilarity [19, 27], so \( \sim_{1} \equiv \sim_{\alpha} \).

- **Lemma 22.** \( l_{d\alpha}(s, s') = 0 \) if and only if \( s \sim_{\alpha} s' \).

Since \( l_{d\alpha}(s, s') = 0 \) implies \( l_{v\alpha}(s, s') = 0 \), this also provides a way to show that \( \delta \) is zero, that is, to show \( \epsilon \)-differential privacy holds. However, note this is not a complete method to do this, and there are bisimilarity distances focused on finding \( \epsilon \) [6].

- **Lemma 23.** If \( s \sim_{\alpha} s' \) then \( l_{v\alpha}(s, s') = 0 \).

We need to be able to quickly and independently compute which pairs of states are related by \( \sim_{\alpha} \). In fact we can do this in polynomial time using a closure procedure, which will terminate after polynomially many rounds.

- **Proposition 24.** \( \sim_{\alpha} \) can be computed in polynomial time in size(\( \mathcal{M} \)).

**Proof.** We present a standard refinement algorithm, let \( A_{0} = S \times S \) and compute \( A_{i+1} = \{(s, s') \in A_{i} \mid \exists (\omega, \eta) \in \Omega_{s, s'}^{\alpha} : \eta = 0 \land \omega_{u, v} > 0 \implies (u, v) \in A_{i}\} \). To find this, define \( \mathbb{1}_{A_{i}} \), a matrix such that \( \mathbb{1}_{A_{i}}(s, s') = 0 \) if \( (s, s') \in A_{i} \) and 1 otherwise. Apply \( \Gamma_{\alpha}^{A} \) to \( \mathbb{1}_{A_{i}} \).
whence amounts to computing \( n^2 \) linear programs. Take \( A_{i+1} \) to be indices of the matrix
where \( \Gamma_{\alpha}^A(\mathbb{1}, A_i) \) is zero. At each step, we remove at least one element, or stabilise so that the
function will not change in subsequent rounds. After \( n^2 \) steps it is either stable or empty.

\[ A_{n+2} \subseteq \sim_{\alpha}; \] after convergence we have some set such that \((s, s') \in A_{n+2} \Rightarrow \exists (\omega, \eta) \in \Omega_{s, s'}: \eta = 0 \land (\omega, \nu) > 0 \Rightarrow (u, v) \in A_{n+2} \).

\( \sim_{\alpha} \) is the largest such set, so it contains \( A_{n+2} \).

\[ \sim_{\alpha} \subseteq A_{n+2}; \] by induction we start with \( \sim_{\alpha} \subseteq A_0 \) and only remove pairs not in \( \sim_{\alpha} \).

Recall that \( ld_{\alpha} \) was defined as the least fixed point of \( \Gamma_{\alpha}^A \). Let us refine \( \Gamma_{\alpha}^A \) so the gap
between the least fixed point and the greatest is as small as possible. We do this by fixing
the known values of the least fixed point in the function, in particular the zero cases. We let

\[ \Gamma_{\alpha}^A(d)(s, s') = \begin{cases} 0 & s \sim_{\alpha} s' \\ \Gamma_{\alpha}^A(d)(s, s') & \text{otherwise} \end{cases} \]

and observe that \( ld_{\alpha} \) is also the least fixed point of \( \Gamma_{\alpha}^A \).

**Definition 26.** We let \( lgd_{\alpha} \) be the greatest fixed point of \( \Gamma_{\alpha}^A \).

**Definition and Computation of \( lgd_{\alpha} \)**

Towards a more efficiently computable function, we now study the greatest fixed point.

**Lemma 25.** \( ld_{\alpha} \) is the least fixed point of \( \Gamma_{\alpha}^A \).

**Lemma 27.** Each \((\omega, \eta) \in V(\Omega^A_{s, s'})\) are rational numbers requiring a number of bits polynomial in \( \text{size}(\mathcal{M}) \).

**Proof.** Consider the polytope:

\[ \Omega_{\mu, \mu'}^{\alpha} = \left\{ (\omega, \tau, \gamma, \eta) \in [0, 1]^S \times [0, 1] \times (\mathbb{1}, 1)^S \mid \begin{align*} &\forall i: \sum_j \omega_{i, j} + \tau_i - \gamma_i + \eta_i = \mu(i) \\ &\forall j: \sum_i \omega_{i, j} + \frac{\tau_j - \gamma_j}{\alpha} \leq \mu'(j) \end{align*} \right\} \]

Each vertex is the intersection of hyperplanes defined in terms of \( \mu, \mu' \) (rationals given in
the input \( \mathcal{M} \)), thus vertices of \( \Omega_{\mu, \mu'}^{\alpha} \) are rationals with representation size polynomial in the
input. Vertices of \( \Omega_{\mu, \mu'}^{\alpha} = \{ (\omega, \eta) \mid \exists r, \gamma (\omega, r, \gamma, \eta) \in \Omega_{\mu, \mu'}^{\alpha} \} \) require only fewer bits.

The following linear program (LP) expresses the greatest post-fixed point. It has polynomially
many variables but exponentially many constraints (for each \( s, s' \) one constraint for each \( \omega \in V(\Omega_{s, s'}) \)). Since linear programs can be solved in polynomial time, the greatest
fixed point can be found in exponential time using the exponential size linear program.
Asymmetric Distances for Approximate Differential Privacy

Proposition 28. \( \text{lgd}_\alpha \) is the optimal solution, \( d \in [0, 1]^{S \times S} \) of the following linear program:

\[
\max_{d \in [0, 1]^{S \times S}} \sum_{(u,v) \in S \times S} d_{u,v} \text{ subject to: for all } s, s' \in S:\n\]

\[
d_{s, s'} = 0 \\
d_{s, s'} = 1 \\
d_{s, s'} \leq \sum_{(u,v) \in S \times S} \omega_{u,v}d_{u,v} + \sum_{u \in S} \eta_u \text{ for all } (\omega, \eta) \in V(\Omega_\alpha^{s,s'}) \text{ whenever } s \sim_\alpha s', \\
\text{ otherwise.}
\]

Proof. The \( s \sim_\alpha s' \) and \( \ell(s) \neq \ell(s') \) cases follow by definition. Observe that by the definition of \( \text{lgd}_\alpha \) as a post-fixed point it is required that \( d(s, s') \leq \Gamma\alpha(d)(s, s') = K\alpha(d)(s, s') = \min_{(\omega, \eta) \in \Omega_\alpha^{s,s'}} \sum_{(u,v) \in S \times S} \omega_{u,v}d_{u,v} + \sum_{u \in S} \eta_u \) or equivalently, for all \( (\omega, \eta) \in \Omega_\alpha^{s,s'}: d(s, s') \leq \sum_{(u,v) \in S \times S} \omega_{u,v}d_{u,v} + \sum_{u \in S} \eta_u \)

In the spirit of [7], we can solve the exponential-size linear program given in Proposition 28 using the ellipsoid method, in polynomial time. Whilst the linear program has exponentially many constraints, it has only polynomially many variables. Therefore, the ellipsoid method can be used to solve the linear program in polynomial time, provided a polynomial-time separation oracle can be given [26, Chapter 14]. Separation oracle takes as argument \( d \in [0, 1]^{S \times S} \), a proposed solution to the linear program and must decide whether \( d \) satisfies the constraints or not. If not then it must provide \( \theta \in Q^{S \times S} \) as a separating hyperplane such that, for every \( d' \) that does satisfy the constraints, \( \sum_{u,v} d_{u,v}\theta_{u,v} < \sum_{u,v} d'_{u,v}\theta_{u,v} \).

Our separation oracle will perform the following: for every \( s, s' \in S \) check that \( d(s, s') \leq \min_{(\omega, \eta) \in \Omega_\alpha^{s,s'}} \omega \cdot d + \eta \cdot 1 \). This is done by solving \( \min_{(\omega, \eta) \in \Omega_\alpha^{s,s'}} \omega \cdot d + \eta \cdot 1 \) using linear programming. If every check succeeds, return YES. If some check fails for \( s, s' \) return NO and

\[
\theta_{u,v} = \begin{cases} 
\omega_{u,v} - 1 & (u,v) = (s, s') \\
\omega_{u,v} & \text{otherwise}
\end{cases}
\]

where \( (\omega, \eta) = \arg\min_{(\omega, \eta) \in V(\Omega_\alpha^{s,s'})} d \cdot \omega + \eta \cdot 1 \).

Lemma 29. \( \theta \) is a separating hyperplane, i.e., it separates the unsatisfying \( d \) and all satisfying \( d' \).

Theorem 30. \( \text{lgd}_\alpha \) can be found in polynomial time in the size of \( \mathcal{M} \).

Proof. Checking \( d(s, s') \leq \min_{(\omega, \eta) \in \Omega_\alpha^{s,s'}} \omega \cdot d + \eta \cdot 1 \) is polynomial time. The linear program is of polynomial size, so runs in polynomial time in the size of the encoding of the linear program. Similarly finding \( \theta \) is polynomial time by running essentially the same linear program and reading off the minimising result.

Because pairs \( (\omega, \eta) \) are in \( V(\Omega_\alpha^{s,s'}) \), they are polynomial size in the size of \( \mathcal{M} \), independent of \( d \), by Lemma 27. Note that, unlike in Chen et al. [7], the oracle procedure is not strongly polynomial, so the time to find \( \theta \) may depend on the size of \( d \), but the output \( \theta \) and \( d \) remain polynomial in the size of the initial system.

We conclude there is a procedure for computing \( \text{lgd}_\alpha \) running in polynomial time [26, Theorem 14.1, Page 173]. There exists a polynomial \( \psi \) where the ellipsoid algorithm solves the linear program in time \( T \cdot \psi(\text{size}(\mathcal{M})) \), where \( T \) is the time the separation algorithm takes on inputs of size \( \psi(\text{size}(\mathcal{M})) \). Since the \( T \in \text{poly}(\psi(\text{size}(\mathcal{M}))) \) and \( \psi(\text{size}(\mathcal{M})) \in \text{poly}(\text{size}(\mathcal{M})) \) then \( T \in \text{poly}(\text{size}(\mathcal{M})) \). Overall we have \( T \cdot \psi(\text{size}(\mathcal{M})) \in \text{poly}(\text{size}(\mathcal{M})) \). \( \blacksquare \)
We take as an example, in Figure 4, a PIN checking system from [32, 31]. Intuitively, the labelled Markov chain. with some degree of plausible deniability. This is achieved by flipping a biased coin and to each independently to be combined with no additional loss in privacy. Let us consider the datasets, differential privacy allows the results of \((\epsilon, \delta)-\)differentially private mechanism applied to each independently to be combined with no additional loss in privacy. Let us consider the

\[
\delta \leq e^\epsilon \quad \text{where \(\delta\) is the other privacy parameter).}
\]

In the blue line (●) we see the estimate \(bd_\alpha\) as defined in [9]; which correctly bounds the true privacy, but is unresponsive to \(\alpha\). Using the methods introduced in this paper we compute \(ld_\alpha\) on the red line (■) and \(lgd_\alpha\) on the black line (●), which coincide. We observe that this is an improvement and is within approximately 1.5 times the true privacy for \(\alpha \leq 1.035\). In this example observe that \(ld_\alpha = lgd_\alpha\); suggesting \(lgd_\alpha\), which can be computed in polynomial time is as good as \(ld_\alpha\). Our results do eventually suffer, as increasing \(\alpha\) cannot find a better \(\delta\), despite a lower value existing.

**Example 32 (Randomised Response).** The randomised response mechanism allows a data subject to reveal a secret answer to a potentially humiliating or sensitive question honestly with some degree of plausible deniability. This is achieved by flipping a biased coin and providing the wrong answer with some probability based on the coin toss. If there are two answers \(a\) or \(b\), answering truthfully with probability \(\frac{\beta}{1+\beta}\) and otherwise with \(\frac{1}{1+\beta}\) leads to \(\epsilon\)-differential privacy where \(\epsilon' = \beta\) and such a bound is tight (there is no smaller \(\epsilon'\) such that answering in this way gives \(\epsilon'\)-differential privacy). However, it can be \((\epsilon', \delta)-\)differentially private for \(\epsilon' < \epsilon\) and some \(\delta\).

Let us consider the single-input, single-output randomised response mechanism shown in Figure 5a with \(\beta = 2\), hence \(\ln(2)\)-differentially private, alternatively it is \(\ln(\frac{e}{2})\), \(\frac{1}{15}\)-differential privacy \((\frac{\ln(2)}{2})\approx \frac{\ln(2)}{15}\). We consider the application of composing automata to determine more complex properties automatically.

Differential privacy enjoys multiple composition theorems [15]. When applied to disjoint datasets, differential privacy allows the results of \((\epsilon, \delta)-\)differentially private mechanism applied to each independently to be combined with no additional loss in privacy. Let us consider the

![Figure 4 PIN Checker example: each state denotes its label, transition probabilities on arrows.]

---

**8 Examples**

**Example 31 (PIN Checker).** We demonstrate our methods are a sound technique for determining the \(\delta\) privacy parameter (given \(\epsilon'\), where \(\epsilon\) is the other privacy parameter). We take as an example, in Figure 4, a PIN checking system from [32, 31]. Intuitively, the machine accepts or rejects a code \((a\) or \(b\)). Instead of accepting a code deterministically, it probabilistically decides whether to accept. The machine allows an attempt with the other code if it is not accepted. We model the system that accepts more often on the the pin-code \(a\), from state 0, and the system that accepts more often from code \(b\), from state 1. The chain simulates attempts to gain access to the system by trying code \(a\) then \(b\) until the system accepts (reaching the “end” state). Pen-and-paper analysis can determine that the system is \((\ln(\frac{2809}{2209}), 0)\)-differentially private, or at the other extreme \((0, \frac{200}{2503})\)-differentially private \((\frac{2809}{2209} \approx 1.27, \frac{200}{2503} \approx 0.0799)\). The true privacy, \(lv_\alpha\) is shown along the orange line (▲).

![Diagram of a labelled Markov chain with transition probabilities labeled on arrows.](image)

![Graph showing calculated approximations of \(\delta\) given \(\epsilon\).](image)

In the blue line (●) we see the estimate \(bd_\alpha\) as defined in [9]; which correctly bounds the true privacy, but is unresponsive to \(\alpha\). Using the methods introduced in this paper we compute \(ld_\alpha\) on the red line (■) and \(lgd_\alpha\) on the black line (●), which coincide. We observe that this is an improvement and is within approximately 1.5 times the true privacy for \(\alpha \leq 1.035\). In this example observe that \(ld_\alpha = lgd_\alpha\); suggesting \(lgd_\alpha\), which can be computed in polynomial time is as good as \(ld_\alpha\). Our results do eventually suffer, as increasing \(\alpha\) cannot find a better \(\delta\), despite a lower value existing.
two-input, two-output labelled Markov chain (Figure 5b), where we consider each input to be from two independent respondents, using our methods verifies that the privacy does not increase on the partitioned data. We consider the adjacency relation as the symmetric closure of $R = \{(a, a), (a, b), (b, a), (b, b), (b, b)\}$. We determine $(\ln(\frac{6}{5}), \frac{4}{15})$-differential privacy by computing $\max_{(s, s') \in R} ld_{6/5}(s, s') = \frac{4}{15}$, verifying there is no privacy loss from composition. Because randomised response is finite we can compute $lv_{\alpha}$ for adjacent inputs in exponential time for comparison. In this instance, our technique provides the optimal solution, in the sense $\max_{(s, s') \in R} ld_{6/5}(s, s') = \max_{(s, s') \in R} lv_{6/5}(s, s')$; indicating that $ld_{\alpha}$ and $lgd_{\alpha}$ can provide a good approximation.

The basic composition theorems suggest that if a mechanism that is $(\epsilon, \delta)$-differentially private is used $k$ times, one achieves $(k\epsilon, k\delta)$-differential privacy [13]. However, this is not necessarily optimal. More advanced composition theorems may enable tighter analysis, although this can be computationally difficult ($\#P$-complete) [25]. Even this may not be exact when allowed to look inside the composed mechanisms. If we assume the responses are from two questions answered by the same respondent and let $R' = R \cup \{((a, a), (b, b))\}$, naively applying basic composition concludes $(\ln(\frac{36}{25}), \frac{8}{15})$-differential privacy. Our methods can find a better bound than basic composition since $\max_{(s, s') \in R} ld_{36/25}(s, s') = \frac{103}{225} < \frac{8}{15}$. However, in this case, our technique is not optimal either.

9 Conclusion

Our results are summarised in Figure 1 on page 2. We are interested in the value of $lv_{\alpha}$, but it is not computable and difficult to approximate. We have defined an upper bound $ld_{\alpha}$, showing that it is more accurate than the previously known bound $bd_{\alpha}$ from [9] and just as easy to compute (in polynomial time with an NP oracle). We also defined a distance based on the greatest fixed point, $lgd_{\alpha}$, which has the same flavour but can be computed in polynomial time. When considering $lv_{\alpha}$ directly, we approximate to arbitrary precision.
in \textit{PSPACE} and show it is \#\textit{P}-hard (which generalises a known result on \textit{tv}). It is open whether the least fixed point bisimilarity distance (or any refinement smaller than \(\text{lgd}_\alpha\)) can be computed in polynomial time, or even if \(\text{lgd}_\alpha = \text{ld}_\alpha\). It is also open whether approximation can be resolved to be in \#\textit{P}, \textit{PSPACE}-hard, or complete for some intermediate class.

References


Asymmetric Distances for Approximate Differential Privacy


