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 (ϵ, δ)-differential privacy in the setting of labelled Markov chains. For a given ϵ, the parameter δ can be captured by a variant of the total variation distance, which we call lv_α (where $\alpha = e^\epsilon$).

First we study lv_α 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_α 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.2019.6

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 ϵ [6, 31] and δ [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 δ 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 (ϵ, δ)-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_α which can be used to bound

30th International Conference on Concurrency Theory (CONCUR 2019).
Editors: Wan Fokkink and Rob van Glabbeek; Article No. 6; pp. 6:1–6:17
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

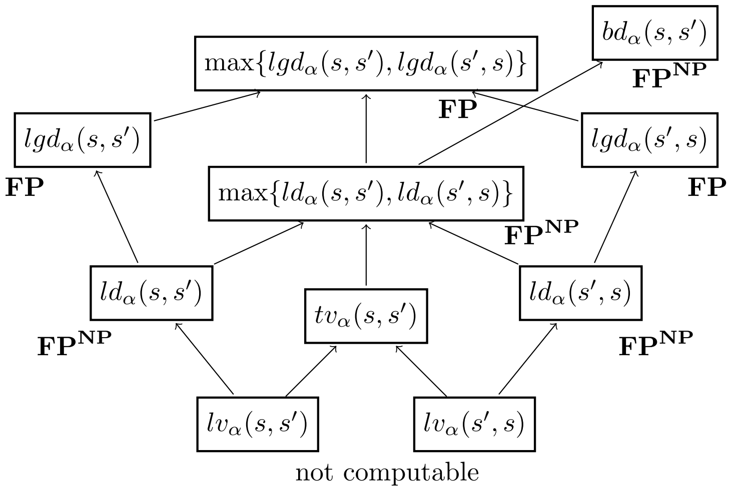


Figure 1 Partial order of distances, such that $a \to b \iff a \le b$. **FP** is the functional counterpart of **P**, where the value of the function can be computed in polynomial time. $\mathbf{FP^{NP}}$ indicates polynomial time with **NP** oracle. tv_α and bd_α are introduced in [9] and recalled in Sections 3 and 6, respectively. The remaining distances are the contribution of this paper.

the δ 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 δ that can be computed in polynomial time.

The privacy parameter in question, δ, can be expressed as a variant of the total variation distance tv_α. In particular we define lv_α as a single component of tv_α (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 **PSPACE**. These results match the results of [22] for standard total variation distances.

We then bound the distance lv_α from above by a distance ld_α which will turn out to be computable, in a similar manner to how bd_α bounds tv_α in [9]. We show that ld_α can be computed in polynomial time with an **NP** oracle (that is, with the same complexity as bd_α). We further generalise ld_α to a new distance lgd_α, computable in polynomial time. This new distance, is no smaller than ld_α, 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 δ. Thus we have defined the first non-trivial estimate of the δ 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_α is a better approximation than bd_α from [9]. We confirm this using several case studies, where we also demonstrate, on a randomised response mechanism, that the estimates based on ld_α 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 ϵ-differential privacy, and indeed arbitrary metrics, but did not consider the complexity of calculating the distances. We introduced a distance in [9] for (ϵ, δ)-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 $Dist(X)$ 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^ω all infinite sequences.

▶ **Definition 1** (labelled Markov chains (LMC's)). *A* labelled Markov chain $\mathcal{M}$ *is a tuple* $\langle S, \Sigma, \mu, \ell \rangle$*, where* S *is a finite set of states,* Σ *is a finite alphabet,* $\mu : S \to Dist(S)$ *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 μ_s for $\mu(s)$.

In what follows, we study probabilities associated with infinite sequences of labels generated by LMC's. We specify the relevant probability spaces next using standard measure theory [5, 2]. Let us start with the definition of cylinder sets.

▶ **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* Σ^ω *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 Σ^ω (the cylindrical σ-algebra). Where clear from context we will omit Σ in the subscript of $\mathcal{F}$. What will be important for us is that any measure ν 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 ν_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* μ *and* ℓ *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* $Paths_s(\mathcal{M})$ *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* $p \in Paths_s(\mathcal{M})$ *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 $\langle S, \Sigma, T \rangle$, where S is a finite set of states, Σ is a finite alphabet and $T : S \to Dist(S \times \Sigma)$ is the transition function. We write each transition as $q \xrightarrow[a]{p} q'$, 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 $q \xrightarrow[b]{p} q'$, we have $\mu_{(q,a)}((q', b)) = p$ for every $a \in \Sigma$. Technically, this delays reading of the first character until the second state visited. To account for this, introduce an additional character, say $\vdash$, so that $\nu_s(C_w) = \nu'_{(s,\vdash)}(C_{\vdash w})$, where ν and ν' refer to the measures associated with $\mathcal{M}$ and $\mathcal{M}'$ 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 $\nu_s(w) = \sum \mu^+(p)$, where the summation is over $p \in \mathit{Paths}_s(\mathcal{M})$ such that $\ell^+(p) = w$ and $p_{|w|} \in F$, so that we only consider paths which end in a final state. The function can be extended to sets of words $E \subseteq \Sigma^*$ (which are countable) by $\nu_s(E) = \sum_{w \in E} \nu_s(w)$.

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 $q \in F$, $\mu_q(q) = 1$ and $\ell(q) = \$$, so that the only trace that can be observed from q is $\$^\omega$. Then, for a word $w \in \Sigma^*$, we rather study $w\$\$\$\ldots$. In an infinite-word model, events are formed from cylinders. A cylinder C_u corresponds to the event $\{w \in \Sigma^* \mid \mathit{prefix}(w) = u\}$ in the finite model. For a specific word in the finite model, we consider the cylinder $C_{w\$}$. Thus any hardness result that applies to finite-word labelled Markov chains also applies to infinite-word Markov chains. Some of our arguments will be simplified, as every event in the finite-word model is countable.

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 $\nu_s(E) = \nu_{s'}(E)$ for every $E \in \mathcal{F}$. More generally, the difference between them can be quantified using the *total variation distance*, defined by $tv(\nu, \nu') = \sup_{E \in \mathcal{F}} |\nu(E) - \nu'(E)|$. Given $\mathcal{M} = \langle S, \Sigma, \mu, \ell \rangle$ and $s, s' \in S$, we shall write $tv(s, s')$ to refer to $tv(\nu_s, \nu_{s'})$. Ensuring such pairs of measures $(\nu_s, \nu_{s'})$ 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 lv_α of 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 $\mathcal{M} = \langle S, \Sigma, \mu, \ell \rangle$ be a labelled Markov chain and let $R \subseteq S \times S$ be a symmetric relation. Given $\epsilon \geq 0$ and $\delta \in [0, 1]$, we say that $\mathcal{M}$ is (ϵ, δ)-differentially private w.r.t. R if, for every $s, s' \in S$ such that $(s, s') \in R$, we have $\nu_s(E) \leq e^\epsilon \cdot \nu_{s'}(E) + \delta$ for every measurable set $E \in \mathcal{F}$.*

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 $s \in S$ can be viewed as defining a random variable X_s with outcomes from Σ^ω such that $\mathbb{P}[X_s \in E] = \nu_s(E)$. Then the above can be rewritten as $\mathbb{P}[X_s \in E] \leq e^\epsilon\, \mathbb{P}[X_{s'} \in E] + \delta$, which matches the definition from [14], where one would consider $X_s, X_{s'}$ 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 *ϵ-differential privacy*, which amounts to measuring the ratio between the probabilities of possible outcomes. When one cannot expect to achieve this pure *ϵ-differential privacy*, the relaxed approximate differential privacy is used [24]. When $\epsilon = 0$, δ is captured exactly by the statistical distance (total variation distance) tv.

Our aim is to capture the value of δ required to satisfy the differential privacy property for a given ϵ. That is, given a LMC $\mathcal{M}$, a symmetric relation R and $\alpha = e^\epsilon \geq 1$, we want to determine the smallest δ such that $\mathcal{M}$ is (ϵ, δ)-differentially private with respect to R. We can measure the difference between two measures ν, ν' on $(\Sigma^\omega, \mathcal{F})$ as follows: $tv_\alpha(\nu, \nu') = \sup_{E \in \mathcal{F}} \Delta_\alpha(\nu(E), \nu'(E))$ where $\Delta_\alpha(a, b) = \max\{a - \alpha b, b - \alpha a, 0\}$ [3]. When used on $\nu_s, \nu_{s'}$ and $\alpha = e^\epsilon$, $tv_\alpha(s, s')$ gives the required δ 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 Δ_α is symmetric, we break this property to introduce a new distance function Λ_α (similarly to [4]). Then we define an analogous total variation distance lv_α, which will be our main object of study.

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

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 $lv_\alpha(\nu, \nu') = \sup_{E \in \mathcal{F}} \nu(E) - \alpha\nu'(E)$, since there is always an event such that $\nu'(E) = 0$, in particular $\nu(\emptyset) = 0$. Observe that Δ_α and Λ_α are not metrics as $\Delta_\alpha(a, b) = 0 \not\Rightarrow a = b$, and in fact not even pseudometrics as the triangle inequality does not hold. Our new distance Λ_α (and lv_α) is not symmetric, while Δ_α and tv_α are.

If $\alpha = 1$, then $lv_1 = tv_1 = tv$, since if ν, ν' are probability measures and we have $\nu(E) = 1 - \nu(\overline{E})$ then $\sup_{E \in \mathcal{F}} |\nu(E) - \nu'(E)| = \sup_{E \in \mathcal{F}} \nu(E) - \nu'(E) = \sup_{E \in \mathcal{F}} \nu'(E) - \nu(E)$, 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_α and lv_α.

▶ **Proposition 9.** *Given a labelled Markov chain $\mathcal{M}$ and a symmetric relation $R \subseteq S \times S$, the following properties are equivalent for $\alpha = e^\epsilon$:*

- *$\mathcal{M}$ is (ϵ, δ)-differentially private w.r.t. R,*
- *$\max_{(s,s') \in R} tv_\alpha(s, s') \leq \delta$, and*
- *$\max_{(s,s') \in R} lv_\alpha(s, s') \leq \delta$.*

We now focus on computing lv_α, since this will allow us to determine the 'level' of differential privacy for a given ϵ. Henceforth we will refer to e^ϵ as α. For the purposes of our complexity arguments, we will only use rational α with $O(size(\mathcal{M}))$-bit representation.

4 lv_α 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.

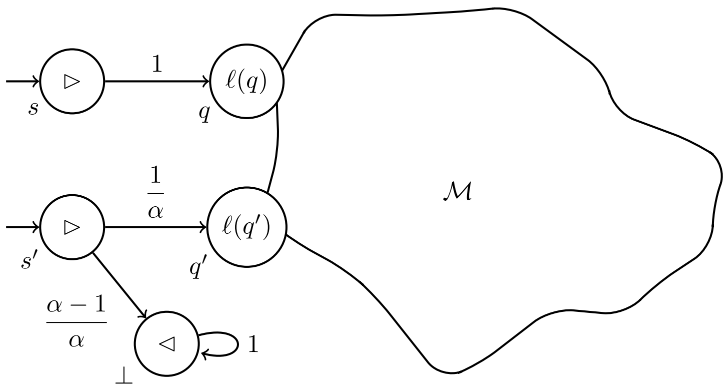


Figure 2 Markov chain $\mathcal{M}'$ in the reduction from $tv(q, q')$ to $lv_\alpha(s, s')$

Since $tv = lv_1$, 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 ϵ makes it possible to compute the optimal δ exactly.

▶ **Theorem 10.** *Finding a value of tv reduces in polynomial time to finding a value of lv_α.*

Proof. Given a labelled Markov chain $\mathcal{M} = \langle Q, \Sigma, \mu, \ell \rangle$, 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 $\mathcal{M}' = \langle Q \cup \{s, s', \bot\}, \Sigma', \mu', \ell' \rangle$, with $\ell'(s) = \ell'(s') = \triangleright$, $\ell'(\bot) = \triangleleft$, $\ell'(x) = \ell(x)$ for all $x \in Q$, $\Sigma' = \Sigma \cup \{\triangleright, \triangleleft\}$,

$$\mu'_s(q) = 1, \qquad \mu'_{s'}(q') = \frac{1}{\alpha}, \qquad \mu'_{s'}(\bot) = \frac{\alpha - 1}{\alpha}, \qquad \text{and} \qquad \mu'_x(y) = \mu_x(y) \text{ for all } x, y \in Q.$$

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 $E' = \{\triangleright w \mid w \in E\} \in \mathcal{F}_{\Sigma'}$. 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 $\triangleleft$ can only be reached from s', any word using it contributes negatively to the difference. Hence intersecting the event with $\triangleright\Sigma^\omega$, to remove $\triangleleft$, can only increase the difference. The character $\triangleright$ must occur (only) as the first character of every (useful) word in E'. Let $E = \{w \mid \triangleright w \in E' \cap \triangleright\Sigma^\omega\} \in \mathcal{F}_\Sigma$, 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_α 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_α 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 $s \to q$ branch may result in new maximising events.

5 Approximation of lv_α

Given that lv_α 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 $tv = lv_1$ 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_α can also be approximated and the same complexity bounds apply.

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

▶ **Theorem 12.** *For finite-word Markov chains, approximation of $lv_\alpha(s, s')$ within γ 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 $lv_\alpha(s, s') = \sup_{E \subseteq \Sigma^*} \nu_s(E) - \alpha \nu_{s'}(E)$, which turns out to be $W = \{w \mid \nu_s(w) \geq \alpha \nu_{s'}(w)\}$, so that $lv_\alpha(s, s') = \nu_s(W) - \alpha \nu_{s'}(W)$. This choice of the maximising event only applies to finite-word Markov chains, thus the proof does not extend in 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 $X + \alpha Y$. It is then argued that, for specific X, Y with this property, one can find any bit of $X + \alpha Y$.

For the lower bound, we note that approximating tv is #**P**-hard [22], by a reduction from *#NFA*, a #**P**-complete problem [20]. That is, given a non-deterministic finite automaton $\mathcal{A}$ and $n \in \mathbb{N}$ in unary, determine $|\Sigma^n \cap L(\mathcal{A})|$, the number of accepted words of $\mathcal{A}$ of length n. Since tv can be reduced to lv_α (Theorem 10), approximating lv_α 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. ◀

6 A least fixed point bound ld_α

We seek to bound lv_α from above by a computable quantity, and will introduce a distance function ld_α 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_α 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_α, 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_α.

We determine (ϵ, δ)-differential private w.r.t. relation R by bounding δ by $\max_{(s,s') \in R} ld_\alpha(s, s')$. 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 ($|R|$ is polynomial with respect to the size of $\mathcal{M}$). 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 μ, μ', we define*

$$K_\alpha^\Lambda(d)(\mu, \mu') = \sup_{\substack{f : X \to [0,1] \\ \forall x, x' \in X \;\; \Lambda_\alpha(f(x), f(x')) \leq d(x, x')}} \Lambda_\alpha\Big(\int_X f d\mu, \int_X f d\mu'\Big)$$

where f ranges over functions which are measurable w.r.t. μ and μ'.

▶ Remark. The (standard) Kantorovich distance lifts a distance function d over the ground objects X to a distance between measures μ, μ' on the set X. This is equivalent to replacing Λ_α with the absolute distance function ($abs(a,b) = |a-b|$). We note that $K_\alpha^\Lambda(d)$ 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 $\int_X f d\mu = \sum_{x\in X} f(x)\,\mu(x)$. Chatzikokolakis *et al* [6] considered the case where the absolute value function was replaced by any metric d'. Our lifting K_α^Λ does not quite fit in this framework, since Λ_α is not metric.

The interest in K_α^Λ is that it allows us to reformulate the definition of the distance function lv_α. 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 $\mathbb{1}_{\neq}$ (the indicator function over inequality with $\mathbb{1}_{\neq}(w, w') = 1$ for $w \neq w'$, and 0 otherwise).

▶ **Lemma 14.** $lv_\alpha(s, s') = K_\alpha^\Lambda(\mathbb{1}_{\neq})(\nu_s, \nu_{s'})$.

Since computing lv_α, or now $K_\alpha^\Lambda(\mathbb{1}_{\neq})(\nu_s, \nu_{s'})$, is difficult, we introduce an upper bound on lv_α, inspired by bisimilarity distances, which we will call ld_α. This will be the least fixed point of Γ_α^Λ, 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* $\Gamma_\alpha^\Lambda : [0,1]^{S\times S} \to [0,1]^{S\times S}$ *be defined as follows.*

$$\Gamma_\alpha^\Lambda(d)(s, s') = \begin{cases} K_\alpha^\Lambda(d)(\mu_s, \mu_{s'}) & \ell(s) = \ell(s') \\ 1 & \textit{otherwise} \end{cases}$$

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

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

▶ **Definition 16.** *Let* $ld_\alpha : S \times S \to [0,1]$ *be the least fixed point of* Γ_α^Λ.

To provide a guarantee of privacy we require a sound upper bound on lv_α.

▶ **Theorem 17.** $lv_\alpha(s, s') \leq ld_\alpha(s, s')$ *for every* $s, s' \in S$.

The proof of Theorem 17 proceeds similarly to Lemma 2 in [9]. We will see, however, that this upper bound on lv_α is stronger (or at least no worse) than the bound obtained in [9]. Recall from [9] that bd_α is defined as the least fixed point of

$$\Gamma_\alpha^\Delta(d)(s, s') = \begin{cases} K_\alpha^\Delta(d)(\mu_s, \mu_{s'}) & \ell(s) = \ell(s') \\ 1 & \text{otherwise} \end{cases}$$

where $K_\alpha^\Delta(d)$ behaves as $K_\alpha^\Lambda(d)$, but uses $\Delta_\alpha(a, b) = \max\{a - \alpha b, b - \alpha a, 0\}$ rather than $\Lambda_\alpha(a, b) = \max\{a - \alpha b, 0\}$.

▶ **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 $A^\intercal$ be its transpose. Consider bd_α and ld_α as matrices. bd_α is the least fixed point of Γ_α^Δ so $\Gamma_\alpha^\Delta(bd_\alpha)(s, s') = bd_\alpha(s, s')$. Also notice that $\Gamma_\alpha^\Lambda(bd_\alpha)(s, s') \leq$

$$\text{LD-THRESHOLD}(s,s',\theta) = \exists (d_{i,j})_{i,j\in S} \bigwedge_{i,j\in S} (0\le d_{i.j}\le 1) \wedge d_{s,s'}\le\theta$$
$$\wedge \bigwedge_{q,q'\in S} \begin{cases} d_{q,q'}=1 & \ell(q)\neq\ell(q') \\ \mathit{couplingConstraint}(d,q,q') & \ell(q)=\ell(q') \end{cases}$$
$$\mathit{couplingConstraint}(d,q,q') = \exists(\omega_{i,j})_{i,j\in S} \quad \exists(\gamma_i)_{i\in S} \quad \exists(\tau_i)_{i\in S} \quad \exists(\eta_i)_{i\in S}$$
$$\sum_{i,j\in S}\omega_{i,j}\cdot d_{i,j}+\sum_i \eta_i\le d_{q,q'} \quad\wedge\quad \bigwedge_{i,j\in S}(0\le\omega_{i,j}\le 1) \quad\wedge\quad \bigwedge_{i\in S}\begin{cases}0\le\gamma_i\le 1\\0\le\tau_i\le 1\\0\le\eta_i\le 1\end{cases}$$
$$\wedge\bigwedge_{i\in S}(\sum_{j\in S}\omega_{i,j}-\gamma_i+\tau_i+\eta_i=\mu_q(i)) \quad \wedge \bigwedge_{j\in S}(\sum_{i\in S}\omega_{i,j}+\frac{\tau_j-\gamma_j}{\alpha}\le\mu_{q'}(j))$$

Figure 3 **NP** Formula for LD-THRESHOLD

$\Gamma^{\Delta}_{\alpha}(bd_\alpha)(s,s')$, since $K^{\Lambda}_{\alpha}(bd_\alpha)\sqsubseteq K^{\Delta}_{\alpha}(bd_\alpha)$. To see this, note that, because $bd_\alpha = bd^{\mathsf{T}}_\alpha$, the relevant set of functions is the same, but the objective function in the supremum is smaller.

Hence $\Gamma^{\Lambda}_{\alpha}(bd_\alpha)\sqsubseteq bd_\alpha$, i.e. bd_α is also a pre-fixed point of $\Gamma^{\Lambda}_{\alpha}$. Since ld_α is the least pre-fixed point of $\Gamma^{\Lambda}_{\alpha}$ then we know $ld_\alpha\sqsubseteq bd_\alpha$. By symmetry, $bd_\alpha = bd^{\mathsf{T}}_\alpha$ giving $ld_\alpha\sqsubseteq bd^{\mathsf{T}}_\alpha$ and then $ld^{\mathsf{T}}_\alpha\sqsubseteq bd_\alpha$. We conclude $\max\{ld_\alpha(s,s'), ld_\alpha(s',s)\}\le bd_\alpha(s,s')$ 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_α.

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

$$K^{\Lambda}_{\alpha}(d)(\mu,\mu') = \min_{(\omega,\eta)\in\Omega^{\alpha}_{\mu,\mu'}}\Big(\sum_{s,s'\in X}\omega_{s,s'}\cdot d(s,s')+\sum_{s\in X}\eta_s\Big), \qquad \textit{where}$$

$$\Omega^{\alpha}_{\mu,\mu'} = \left\{ (\omega,\eta)\in[0,1]^{X\times X}\times[0,1]^X \quad\middle|\quad \begin{array}{l}\exists\gamma,\tau\in[0,1]^X\\ \forall i: \sum_j\omega_{i,j}+\tau_i-\gamma_i+\eta_i=\mu(i)\\ \forall j:\sum_i\omega_{i,j}+\frac{\tau_j-\gamma_j}{\alpha}\le\mu'(j)\end{array}\right\}.$$

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

▶ **Theorem 20.** *ld_α can be computed in polynomial time with access to an* **NP** *oracle.*

We first show that the LD-THRESHOLD problem, which asks if $ld_\alpha(s,s')\le\theta$, is in **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 **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 **NP**, we can approximate the value using binary search with polynomial overhead to arbitrary accuracy γ, thus we find a value x such that $|x - ld_\alpha(s, s')| \leq \gamma$. In fact, one can find the exact value of $ld_\alpha(s, s')$ in polynomial time assuming the oracle. We can show the value of ld_α 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 lgd_α

In the previous section we have used the least fixed point of Γ_α^Λ, which finds the fixed point closest to our objective lv_α. We now consider relaxing this requirement so that we can find a fixed point in polynomial time. We will introduce lgd_α, 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 ld_α, so we first refine our fixed point function (Γ_α^Λ) to reduce the potential gap. We do this by characterising the elements which are zero in ld_α and fixing these as such; so that they cannot be larger in the greatest fixed point.

Refinement of Γ_α^Λ

In the case of standard bisimulation distances the kernel of ld_1, that is $\{(s, s') \mid ld_1(s, s') = 0\}$, is exactly bisimilarity. We consider the kernel for ld_α 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*

$$(s, s') \in R \iff \exists\ (\omega, \eta) \in \Omega^\alpha_{s,s'}\ s.t.\ (\omega_{u,v} > 0 \implies (u, v) \in\ R) \quad \wedge \quad \forall u\ \eta_u = 0.$$

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 $\sim_1 = \sim$.

▶ **Lemma 22.** $ld_\alpha(s, s') = 0$ *if and only if* $s \sim_\alpha s'$.

Since $ld_\alpha(s, s') = 0$ implies $lv_\alpha(s, s') = 0$, this also provides a way to show that δ is zero, that is, to show ϵ-differential privacy holds. However, note this is not a complete method to do this, and there are bisimilarity distances focused on finding ϵ [6].

▶ **Lemma 23.** *If* $s \sim_\alpha s'$ *then* $lv_\alpha(s, s') = 0$.

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 $A_{i+1} = \{(s, s') \in A_i \mid \exists(\omega, \eta) \in \Omega^\alpha_{s,s'} : \eta = \mathbf{0}\ \wedge\ (\omega_{u,v} > 0 \implies (u, v) \in A_i)\}$. 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 Γ_α^Λ to $\mathbb{1}_{!A_i}$, which amounts to computing n^2 linear programs. Take A_{i+1} to be indices of the matrix where $\Gamma_\alpha^\Lambda(\mathbb{1}_{!A_i})$ is zero. At each step, we remove at least one element, or stabilise so that the set 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 $(s,s') \in A_{n^2} \implies \exists(\omega,\eta) \in \Omega^\alpha_{s,s'} : \eta = \mathbf{0} \wedge (\omega_{u,v} > 0 \implies (u,v) \in A_{n^2})$. $\sim_\alpha$ 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_α was defined as the least fixed point of Γ^Λ_α. Let us refine Γ^Λ_α 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

$$\Gamma'^\Lambda_\alpha(d)(s,s') = \begin{cases} 0 & s \sim_\alpha s' \\ \Gamma^\Lambda_\alpha(d)(s,s') & \text{otherwise} \end{cases}$$

and observe that ld_α is also the least fixed point of Γ'^Λ_α.

▶ **Lemma 25.** *ld_α is the least fixed point of Γ'^Λ_α.*

Definition and Computation of lgd_α

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

▶ **Definition 26.** *We let lgd_α to be the greatest fixed point of Γ'^Λ_α.*

It is equivalent to consider the greatest *post*-fixed point. It turns out that when $\alpha = 1$, $lgd_1 = ld_1$ [7]. We do not know if this holds for $\alpha > 1$, although conjecture that it might. Whilst it may not necessarily be as tight a bound on lv_α as ld_α, we can also use lgd_α to bound lv_α, thus the δ parameter of (ϵ,δ)-differential privacy. Because $ld_\alpha(s,s') \le lgd_\alpha(s,s')$ for every $s,s' \in S$, then Theorem 17 implies that $lv_\alpha(s,s') \le lgd_\alpha(s,s')$, for every $s,s' \in S$.

We will show that lgd_α can be computed in polynomial time using the ellipsoid method for solving a linear program of exponential size, matching the result of [7] for standard bisimilarity distances. Whilst we will not need to express the entire linear program in one go, we may need any one constraint at a time, so we need to be able to express each constraint, in polynomially many bits. We show that the representation of vertices of $\Omega^\alpha_{s,s'}$ is small.

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

Proof. Consider the polytope:

$$\Omega'^\alpha_{\mu,\mu'} = \left\{ (\omega,\tau,\gamma,\eta) \in [0,1]^{S\times S} \times ([0,1]^S)^3 \quad \middle| \quad \begin{array}{l} \forall i : \sum_j \omega_{i,j} + \tau_i - \gamma_i + \eta_i = \mu(i) \\ \forall j : \sum_i \omega_{i,j} + \frac{\tau_j - \gamma_j}{\alpha} \le \mu'(j) \end{array} \right\}$$

Each vertex is the intersection of hyperplanes defined in terms of μ,μ' (rationals given in the input $\mathcal{M}$), thus vertices of $\Omega'^\alpha_{\mu,\mu'}$ are rationals with representation size polynomial in the input. Vertices of $\Omega^\alpha_{\mu,\mu'} = \{(\omega,\eta) \mid \exists\tau,\gamma\ (\omega,\tau,\gamma,\eta) \in \Omega'^\alpha_{\mu,\mu'}\}$ 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 $\omega \in V(\Omega^\alpha_{s,s'})$). 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.

▶ **Proposition 28.** *lgd_α is the optimal solution, $d \in [0,1]^{S\times S}$ of the following linear program:* $\max_{d\in[0,1]^{S\times S}} \sum_{(u,v)\in S\times S} d_{u,v}$ *subject to: for all $s,s' \in S$:*

$$\begin{array}{lll} d_{s,s'} = 0 & & \textit{whenever } s \sim_\alpha s', \\ d_{s,s'} = 1 & & \textit{whenever } \ell(s) \ne \ell(s'), \\ d_{s,s'} \le \displaystyle\sum_{(u,v)\in S\times S} \omega_{u,v} d_{u,v} + \sum_{u\in S} \eta_u & \textit{for all } (\omega,\eta) \in V(\Omega^\alpha_{s,s'}) & \textit{otherwise.} \end{array}$$

Proof. The $s \sim_\alpha s'$ and $\ell(s) \neq \ell(s')$ cases follow by definition. Observe that by the definition of lgd_α as a post-fixed point it is required that $d(s,s') \leq \Gamma'^{\Lambda}_\alpha(d)(s,s') = K^{\Lambda}_\alpha(d)(s,s') = \min_{(\omega,\eta)\in\Omega^\alpha_{s,s'}} \sum_{(u,v)\in S\times S} \omega_{u,v} d_{u,v} + \sum_{u\in S} \eta_u$ 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 $\theta \in \mathbb{Q}^{|S\times S|}$ 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 $d(s,s') \leq \min_{(\omega,\eta)\in\Omega^\alpha_{s,s'}} \omega \cdot d + \eta \cdot \mathbf{1}$. This is done by solving $\min_{(\omega,\eta)\in\Omega^\alpha_{s,s'}} \omega \cdot d + \eta \cdot \mathbf{1}$ using linear programming. If every check succeeds, return YES. If some check fails for s, s' return NO and

$$\theta_{u,v} = \begin{cases} \omega_{u,v} - 1 & (u,v) = (s,s') \\ \omega_{u,v} & \text{otherwise} \end{cases} \qquad \text{where } (\omega,\eta) = \operatorname*{argmin}_{(\omega,\eta)\in V(\Omega^\alpha_{s,s'})} d \cdot \omega + \eta \cdot \mathbf{1}.$$

▶ **Lemma 29.** *θ is a separating hyperplane, i.e., it separates the unsatisfying d and all satisfying d'.*

▶ **Theorem 30.** *lgd_α can be found in polynomial time in the size of $\mathcal{M}$.*

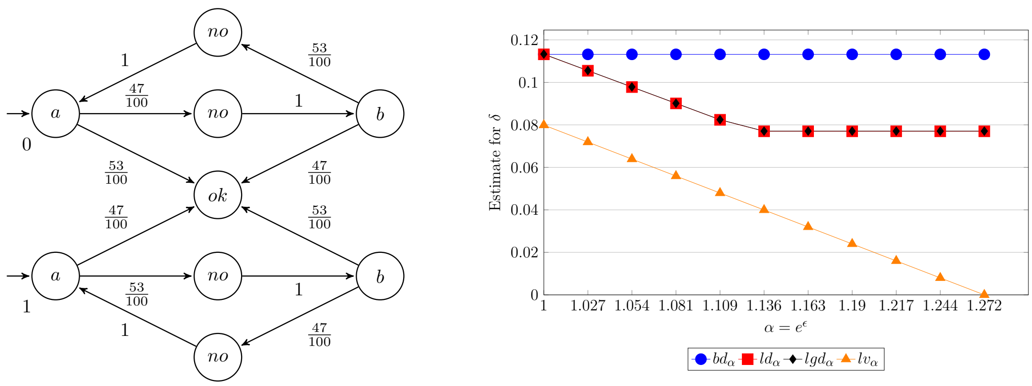
Proof. Checking $d(s,s') \leq \min_{\omega,\eta\in\Omega^\alpha_{s,s'}} \omega \cdot d + \eta \cdot \mathbf{1}$ is polynomial time. The linear program is of polynomial size, so runs in polynomial time in the size of the encoding of the linear program. Similarly finding θ is polynomial time by running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

Because pairs (ω,η) 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 θ may depend on the size of d, but the output θ and d remain polynomial in the size of the initial system.

We conclude there is a procedure for computing lgd_α running in polynomial time [26, Theorem 14.1, Page 173]. There exists a polynomial ψ where the ellipsoid algorithm solves the linear program in time $T \cdot \psi(size(\mathcal{M}))$, where T is the time the separation algorithm takes on inputs of size $\psi(size(\mathcal{M}))$. Since the $T \in poly(\psi(size(\mathcal{M})))$ and $\psi(size(\mathcal{M})) \in poly(size(\mathcal{M}))$ then $T \in poly(size(\mathcal{M}))$. Overall we have $T \cdot \psi(size(\mathcal{M})) \in poly(size(\mathcal{M}))$. ◀

8 Examples

▶ **Example 31** (PIN Checker)**.** We demonstrate our methods are a sound technique for determining the δ privacy parameter (given e^ϵ, where ϵ 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



(a) Labelled Markov chain.

(b) Calculated approximations of δ given ϵ.

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

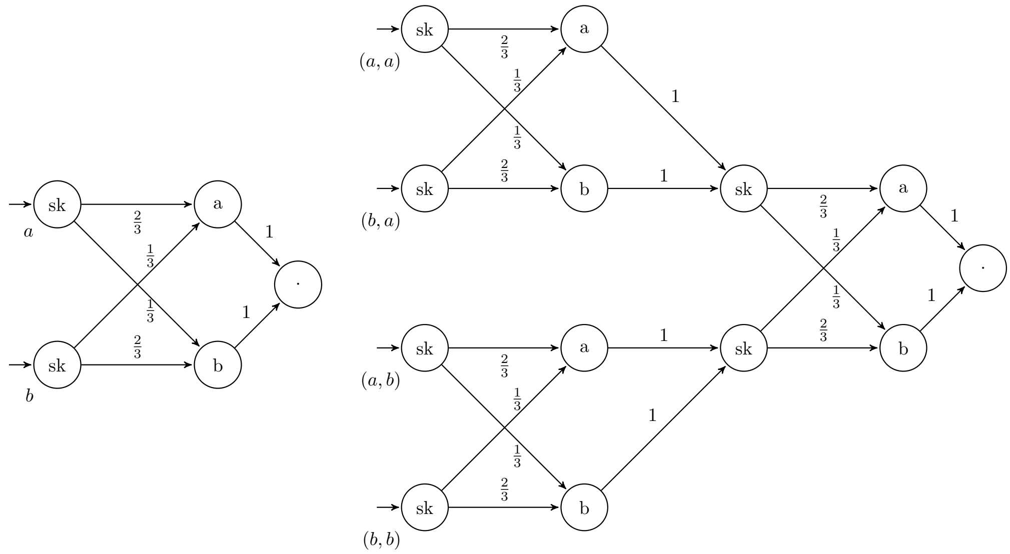
is $(\ln(\frac{2809}{2209}), 0)$-differentially private, or at the other extreme $(0, \frac{200}{2503})$-differentially private ($\frac{2809}{2209} \approx 1.27$, $\frac{200}{2503} \approx 0.0799$). The true privacy, lv_α is shown along the orange line (▲).

In the blue line (●) we see the estimate bd_α as defined in [9]; which correctly bounds the true privacy, but is unresponsive to α. Using the methods introduced in this paper we compute ld_α on the red line (■) and lgd_α 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_α, which can be computed in polynomial time is as good as ld_α. Our results do eventually suffer, as increasing α cannot find a better δ, 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 ϵ-differential privacy where $e^\epsilon = \beta$ and such a bound is tight (there is no smaller ϵ' such that answering in this way gives ϵ'-differential privacy). However, it can be (ϵ', δ)-differentially private for $\epsilon' < \epsilon$ and some δ.

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 $(\ln(\frac{6}{5}), \frac{4}{15})$-differential privacy ($\ln(\frac{6}{5}) \approx \frac{\ln(2)}{4}$). 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 (ϵ, δ)-differentially private mechanism applied to each independently to be combined with no additional loss in privacy. Let us consider the 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 $R = \{((a,a),(a,b)), ((a,a),(b,a)), ((b,b),(a,b)), ((b,b),(b,a))\}$. 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_α 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_α and lgd_α can provide a good approximation.



(a) Single-input, single-output **(b)** Two-input, two-output

Figure 5 Randomised response. Every second label is the outcome of the randomised response mechanism and alternately `sk` (for 'skip'). The left most state represents the sensitive input.

The basic composition theorems suggest that if a mechanism that is (ϵ, δ)-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 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_α, but it is not computable and difficult to approximate. We have defined an upper bound ld_α, showing that it is more accurate than the previously known bound bd_α 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_α, which has the same flavour but can be computed in polynomial time. When considering lv_α directly, we approximate to arbitrary precision in **PSPACE** and show it is #**P**-hard (which generalises a known result on tv). It is open whether the least fixed point bisimilarity distance (or any refinement smaller than lgd_α) can be computed in polynomial time, or even if $lgd_\alpha = ld_\alpha$. It is also open whether approximation can be resolved to be in #**P**, **PSPACE**-hard, or complete for some intermediate class.

References

1 Giorgio Bacci, Giovanni Bacci, Kim G. Larsen, and Radu Mardare. On-the-fly exact computation of bisimilarity distances. In *Tools and Algorithms for the Construction and Analysis of Systems - 19th International Conference, TACAS 2013, Held as Part of the European Joint*

Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013. Proceedings, pages 1–15, 2013. doi:10.1007/978-3-642-36742-7_1.

2 Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008.

3 Gilles Barthe, Boris Köpf, Federico Olmedo, and Santiago Zanella Béguelin. Probabilistic relational reasoning for differential privacy. In *Proceedings of the 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2012, Philadelphia, Pennsylvania, USA, January 22-28, 2012*, pages 97–110, 2012. doi:10.1145/2103656.2103670.

4 Gilles Barthe and Federico Olmedo. Beyond differential privacy: Composition theorems and relational logic for f-divergences between probabilistic programs. In *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part II*, pages 49–60, 2013. doi:10.1007/978-3-642-39212-2_8.

5 Patrick Billingsley. *Probability and Measure*. John Wiley and Sons, 2nd edition, 1986.

6 Konstantinos Chatzikokolakis, Daniel Gebler, Catuscia Palamidessi, and Lili Xu. Generalized bisimulation metrics. In *CONCUR 2014 - Concurrency Theory - 25th International Conference, CONCUR 2014, Rome, Italy, September 2-5, 2014. Proceedings*, pages 32–46, 2014. doi:10.1007/978-3-662-44584-6_4.

7 Di Chen, Franck van Breugel, and James Worrell. On the complexity of computing probabilistic bisimilarity. In *Foundations of Software Science and Computational Structures - 15th International Conference, FOSSACS 2012, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2012, Tallinn, Estonia, March 24 - April 1, 2012. Proceedings*, pages 437–451, 2012. doi:10.1007/978-3-642-28729-9_29.

8 Taolue Chen and Stefan Kiefer. On the total variation distance of labelled markov chains. In *Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS '14, Vienna, Austria, July 14 - 18, 2014*, pages 33:1–33:10, 2014. doi:10.1145/2603088.2603099.

9 Dmitry Chistikov, Andrzej S. Murawski, and David Purser. Bisimilarity distances for approximate differential privacy. In *Automated Technology for Verification and Analysis - 16th International Symposium, ATVA 2018, Los Angeles, CA, USA, October 7-10, 2018, Proceedings*, pages 194–210, 2018. Full version with proofs can be found at https://arxiv.org/abs/1807.10015. doi:10.1007/978-3-030-01090-4_12.

10 Yuxin Deng and Wenjie Du. The kantorovich metric in computer science: A brief survey. *Electr. Notes Theor. Comput. Sci.*, 253(3):73–82, 2009. doi:10.1016/j.entcs.2009.10.006.

11 Josee Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for labelled markov processes. *Theor. Comput. Sci.*, 318(3):323–354, 2004. doi:10.1016/j.tcs.2003.09.013.

12 Josee Desharnais, Radha Jagadeesan, Vineet Gupta, and Prakash Panangaden. The metric analogue of weak bisimulation for probabilistic processes. In *17th IEEE Symposium on Logic in Computer Science (LICS 2002), 22-25 July 2002, Copenhagen, Denmark, Proceedings*, pages 413–422, 2002. doi:10.1109/LICS.2002.1029849.

13 Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*, pages 486–503, 2006. doi:10.1007/11761679_29.

14 Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings*, pages 265–284, 2006. doi:10.1007/11681878_14.

15 Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014. doi:10.1561/0400000042.

16 Kousha Etessami and Mihalis Yannakakis. On the complexity of nash equilibria and other fixed points. *SIAM J. Comput.*, 39(6):2531–2597, 2010. doi:10.1137/080720826.

17 Alessandro Giacalone, Chi-Chang Jou, and Scott A. Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Programming concepts and methods: Proceedings of the IFIP Working Group 2.2, 2.3 Working Conference on Programming Concepts and Methods, Sea of Galilee, Israel, 2-5 April, 1990*, pages 443–458, 1990.

18 Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics*. Springer, 1988. doi:10.1007/978-3-642-97881-4.

19 Bengt Jonsson and Kim Guldstrand Larsen. Specification and refinement of probabilistic processes. In *Proceedings of the Sixth Annual Symposium on Logic in Computer Science (LICS '91), Amsterdam, The Netherlands, July 15-18, 1991*, pages 266–277, 1991. doi:10.1109/LICS.1991.151651.

20 Sampath Kannan, Z Sweedyk, and Steve Mahaney. Counting and random generation of strings in regular languages. In *Proceedings of the sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 551–557. Society for Industrial and Applied Mathematics, 1995.

21 L. V. Kantorovich. On the translocation of masses. *Doklady Akademii Nauk SSSR*, 37(7-8):227—-229, 1942.

22 Stefan Kiefer. On Computing the Total Variation Distance of Hidden Markov Models. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*, volume 107 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 130:1–130:13, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. Full version with proofs can be found at http://arxiv.org/abs/1804.06170. doi:10.4230/LIPIcs.ICALP.2018.130.

23 Kim Guldstrand Larsen and Arne Skou. Bisimulation through probabilistic testing. *Inf. Comput.*, 94(1):1–28, 1991. doi:10.1016/0890-5401(91)90030-6.

24 Sebastian Meiser. Approximate and probabilistic differential privacy definitions. *IACR Cryptology ePrint Archive*, 2018:277, 2018. URL: https://eprint.iacr.org/2018/277.

25 Jack Murtagh and Salil P. Vadhan. The complexity of computing the optimal composition of differential privacy. In *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, pages 157–175, 2016. doi:10.1007/978-3-662-49096-9_7.

26 Alexander Schrijver. *Theory of linear and integer programming*. Wiley-Interscience series in discrete mathematics and optimization. Wiley, 1999.

27 Qiyi Tang and Franck van Breugel. Computing probabilistic bisimilarity distances via policy iteration. In *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, pages 22:1–22:15, 2016. doi:10.4230/LIPIcs.CONCUR.2016.22.

28 Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285–309, 1955.

29 Franck van Breugel. Probabilistic bisimilarity distances. *SIGLOG News*, 4(4):33–51, 2017. URL: https://dl.acm.org/citation.cfm?id=3157837.

30 Franck van Breugel, Babita Sharma, and James Worrell. Approximating a behavioural pseudometric without discount for probabilistic systems. In *Foundations of Software Science and Computational Structures, 10th International Conference, FOSSACS 2007, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2007, Braga, Portugal, March 24-April 1, 2007, Proceedings*, pages 123–137, 2007. doi:10.1007/978-3-540-71389-0_10.

31 Lili Xu. *Formal Verification of Differential Privacy in Concurrent Systems*. PhD thesis, Ecole Polytechnique (Palaiseau, France), 2015.

32 Lili Xu, Konstantinos Chatzikokolakis, and Huimin Lin. Metrics for differential privacy in concurrent systems. In *Formal Techniques for Distributed Objects, Components, and Systems - 34th IFIP WG 6.1 International Conference, FORTE 2014, Held as Part of the*

9th International Federated Conference on Distributed Computing Techniques, DisCoTec 2014, Berlin, Germany, June 3-5, 2014. Proceedings, pages 199–215, 2014. doi:10.1007/978-3-662-43613-4_13.