Concurrent Composition of Differential Privacy

Salil Vadhan
salil_vadhan@harvard.edu
Harvard University

Tianhao Wang
tianhaowang@princeton.edu
Princeton University

ABSTRACT

We initiate a study of the composition properties of *interactive* differentially private mechanisms.[1] An interactive differentially private mechanism is an algorithm that allows an analyst to adaptively ask queries about a sensitive dataset, with the property that an adversarial analyst's view of the interaction is approximately the same regardless of whether or not any individual's data is in the dataset. Previous studies of composition of differential privacy have focused on non-interactive algorithms, but interactive mechanisms are needed to capture many of the intended applications of differential privacy and a number of the important differentially private primitives.

We focus on *concurrent composition*, where an adversary can arbitrarily interleave its queries to several differentially private mechanisms, which may be feasible when differentially private query systems are deployed in practice. We prove that when the interactive mechanisms being composed are *pure* differentially private, their concurrent composition achieves privacy parameters (with respect to pure or approximate differential privacy) that match the (optimal) composition theorem for noninteractive differential privacy. We also prove a composition theorem for interactive mechanisms that satisfy approximate differential privacy. That bound is weaker than even the basic (suboptimal) composition theorem for noninteractive differential privacy, and we leave closing the gap as a direction for future research, along with understanding concurrent composition for other variants of differential privacy.

CCS CONCEPTS

• **Security and privacy → Information-theoretic techniques**;
• **Theory of computation → Cryptographic primitives**.

KEYWORDS

Interactive Differential Privacy, Concurrent Composition

ACM Reference Format:
Salil Vadhan and Tianhao Wang. 2018. Concurrent Composition of Differential Privacy. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

1.1 Differential Privacy

Differential privacy is a framework for protecting privacy when performing statistical releases on a dataset with sensitive information about individuals. (See the surveys [9, 22].) Specifically, for a differentially private mechanism, the probability distribution of the mechanism's outputs of a dataset should be nearly identical to the distribution of its outputs on the same dataset with any single individual's data replaced. To formalize this, we call two datasets x, x', each multisets over a data universe $\mathcal{X}$, *adjacent* if one can be obtained from the other by adding or removing a single element of $\mathcal{X}$.

Definition 1 (Differential Privacy [7]). *For $\varepsilon, \delta \geq 0$, a randomized algorithm $\mathcal{M}$: $\mathrm{MultiSets}(\mathcal{X}) \to \mathcal{Y}$ is (ε, δ)-differentially private if for every pair of adjacent datasets $x, x' \in \mathrm{MultiSets}(\mathcal{X})$, we have:*

$$\forall\, T \subseteq \mathcal{Y} \;\; \Pr[\mathcal{M}(x) \in T] \leq e^{\varepsilon} \cdot \Pr[\mathcal{M}(x') \in T] + \delta \qquad (1)$$

where the randomness is over the coin flips of the algorithm $\mathcal{M}$.

In the practice of differential privacy, we generally view ε as "privacy-loss budget" that is small but non-negligible (e.g., $\varepsilon = 0.1$), and we view δ as cryptographically negligible (e.g., $\delta = 2^{-60}$). We refer to the case where $\delta = 0$ as *pure differential privacy*, and the case where $\delta > 0$ as *approximate differential privacy.*

1.2 Composition of Differential Privacy

A crucial property of differential privacy is its behavior under composition. If we run multiple distinct differentially private algorithms on the same dataset, the resulting composed algorithm is also differentially private, with some degradation in the privacy parameters (ε, δ). This property is especially important and useful since in practice we rarely want to release only a single statistic about a dataset. Releasing many statistics may require running multiple differentially private algorithms on the same database. Composition is also a very useful tool in algorithm design. In many cases, new differentially private algorithms are created by combining several simpler algorithms. The composition theorems help us analyze the privacy properties of algorithms designed in this way.

Formally, let $\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_{k-1}$ be differentially private mechanisms, we define the composition of these mechanisms by independently executing them. Specifically, we define

$$\mathcal{M} = \mathrm{Comp}(\mathcal{M}_0, \mathcal{M}_1, \ldots, \mathcal{M}_{k-1})$$

as

$$\mathcal{M}(x) = (\mathcal{M}_0(x), \ldots, \mathcal{M}_{k-1}(x))$$

where each $\mathcal{M}_i$ is run with independent coin tosses. For example, this is how we might obtain a mechanism answering a k-tuple of queries.

A handful of composition theorems already exist in the literature. The Basic Composition Theorem [7] says that the privacy

[1] This work was previously appeared in ICML Theory and Practice of Differential Privacy Workshop (TPDP-2021).

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

degrades at most linearly with the number of mechanisms executed. However, if we are willing to tolerate an increase in the δ term, we obtain the Advanced Composition Theorem [11] where the privacy parameter ε only needs to degrade proportionally to the square root of number of mechanisms executed. Despite giving an asymptotically correct upper bound for the global privacy parameter, the Advanced Composition Theorem is not exact. Kairouz, Oh, and Viswanath [17] shows how to compute the *optimal* bound for composing k mechanisms where all of them are (ε, δ)-differentially private. Murtagh and Vadhan [20] further extends the optimal composition for the more general case where the privacy parameters may differ for each algorithm in the composition.

1.3 Interactive Differential Privacy

The standard treatment of differential privacy, as captured by Definition 1, refers to a *noninteractive* algorithm $\mathcal{M}$ that takes a dataset x as input and produces a statistical release $\mathcal{M}(x)$, or a batch by taking $\mathcal{M} = \mathrm{Comp}(\mathcal{M}_0, \ldots, \mathcal{M}_{k-1})$. However, in many of the motivating applications of differential privacy, we don't want to perform all of our releases in one shot, but rather allow analysts to make adaptive queries to a dataset. Thus, we should view the mechanism $\mathcal{M}$ as a party in a two-party protocol, interacting with a (possibly adversarial) analyst.

To formalize the concept of interactive DP, we recall one of the standard formalizations of an *interactive protocol* between two parties A and B. We do this by viewing each party as a function, taking its private input x, all messages it has received $(m_0, m_1, \ldots,)$, and the party's random coins r, to the party's next message to be sent out.

Definition 2 (Interactive protocols). *An* interactive protocol (A, B) *is any pair of functions. The interaction between A with input x_A and B with input x_B is the following random process (denoted $(A(x_A), B(x_B))$):*

(1) Uniformly choose random coins r_A and r_B (binary strings) for A and B, respectively.

(2) Repeat the following for $i = 0, 1, \ldots$:
  - *(a) If i is even, let $m_i = A(x_A, m_1, m_3, \ldots, m_{i-1}; r_A)$.*
  - *(b) If i is odd, let $m_i = B(x_B, m_0, m_2, \ldots, m_{i-1}; r_B)$.*
  - *(c) If $m_{i-1} = \mathtt{halt}$, then exit loop.*

We further define the *view* of a party in an interactive protocol to capture everything the party "sees" during the execution:

Definition 3 (View of a party in an interactive protocol). *Let (A, B) be an interactive protocol. Let r_A and r_B be the random coins for A and B, respectively. A's view of $(A(x_A; r_A), B(x_B; r_B))$ is the tuple $\mathtt{View}_A\langle A(x_A; r_A), B(x_B; r_B)\rangle = (r_A, x_A, m_0, m_1, \ldots)$ consisting of all the messages received by A in the execution of the protocol together with the private input x_A and random coins r_A. If we drop the random coins r_A and/or r_B, $\mathtt{View}_A\langle A(x_A), B(x_B)\rangle$ becomes a random variable. B's view of $(A(x_A), B(x_B))$ is defined symmetrically.*

In our case, A is the adversary and B is the mechanism whose input is usually a database x. Since A does not have an input in our case, we will denote the interactive protocol as $(A, B(x))$ for the ease of notation. Since we will only be interested in A's view and A does not have an input, we will drop the subscript and write A's view as $\mathtt{View}\langle A, B(x)\rangle$.

Now we are ready to define the interactive differential privacy as a type of interactive protocol between an adversary (without any computational limitations) and an interactive mechanism of special properties.

Definition 4 (Interactive Differential Privacy). *A randomized algorithm $\mathcal{M}$ is an (ε, δ)-differentially private interactive mechanism if for every pair of adjacent datasets $x, x' \in \mathrm{MultiSets}(\mathcal{X})$, for every adversary algorithm $\mathcal{A}$ we have: we have*

$$\forall T \subseteq \mathrm{Range}\left(\mathtt{View}\langle \mathcal{A}, \mathcal{M}(\cdot)\rangle\right),$$
$$\Pr\left[\mathtt{View}\langle \mathcal{A}, \mathcal{M}(x)\rangle \in T\right] \leq e^{\varepsilon} \Pr\left[\mathtt{View}\langle \mathcal{A}, \mathcal{M}(x')\rangle \in T\right] + \delta \tag{2}$$

where the randomness is over the coin flips of both the algorithm $\mathcal{M}$ and the adversary $\mathcal{A}$.

In addition to being the "right" modelling for many applications of differential privacy, interactive differential privacy also captures the full power of fundamental DP mechanisms such as the Sparse Vector Technique [8, 21] and Private Multiplicative Weights [16], which are in turn useful in the design of other DP algorithms (which can use these mechanisms as subroutines and issue adaptive queries to them). Interactive DP was also chosen as the basic abstraction in the programming framework for the new open-source software project OpenDP [13], which was our motivation for this research.

Despite being such a natural and useful notion, interactive DP has not been systematically studied before. It has been implicitly studied in the context of distributed forms of DP, starting with [1], where the sensitive dataset is split amongst several parties, who execute a multiparty protocol to estimate a joint function of their data, while each party ensures that their portion of the dataset has the protections of DP against the other parties. Indeed, in an m-party protocol, requiring DP against malicious coalitions of size $m - 1$ is equivalent to requiring that each party's strategy is an interactive DP mechanism in the sense of Definition 4. An extreme case of this is the *local model* of DP, where each party holds a single data item in $\mathcal{X}$ representing data about themselves [18]. There has been extensive research about the power of interactivity in local DP; see [5] and the references therein. In contrast to these distributed models, in Definition 4 we are concerned with the *centralized DP* scenario where only one party ($\mathcal{M}$) holds sensitive data, and how an adversarial data analyst ($\mathcal{A}$) may exploit adaptive queries to extract information about the data subjects.

Some of the aforementioned composition theorems for noninteractive DP, such as in [11, 20], are framed in terms of an adaptive "composition game" where an adversary can adaptively select the mechanisms $\mathcal{M}_0, \ldots, \mathcal{M}_{k-1}$, and thus the resulting composition $\mathrm{Comp}(\mathcal{M}_0, \ldots, \mathcal{M}_{k-1})$ can be viewed as an interactive mechanism, but the results are not framed in terms of a general definition of Interactive DP. In particular, the mechanisms $\mathcal{M}_i$ being composed are restricted to be noninteractive in the statements and proofs of these theorems.

1.4 Our Contributions.

In this paper, we initiate a study of the composition of interactive DP mechanisms. Like in the context of cryptographic protocols, there are several different forms of composition we can consider. The simplest is *sequential composition*, where all of the queries to $\mathcal{M}_{i-1}$ must be completed before any queries are issued to $\mathcal{M}_i$. It is straightforward to extend the proofs of the noninteractive DP composition theorems to handle sequential composition of interactive DP mechanisms; in particular the Optimal Composition Theorem extends to this case. (Details omitted.)

Thus, we turn to *concurrent composition*, where an adversary can arbitrarily interleave its queries to the k mechanisms. Although the mechanisms use independent randomness, the adversary may create correlations between the executions by coordinating its actions; in particular, its queries in one execution may also depend on messages it received in other executions. Concurrent composability is important for the deployment of interactive DP in practice, as one or more organizations may set up multiple DP query systems on datasets that refer to some of the same individuals, and we would not want the privacy of those individuals to be violated by an adversary that can concurrently access those systems. Concurrent composability may also be useful in the design of DP algorithms; for example, one might design a DP machine learning algorithm that uses adaptive and interleaved queries to two instantiations of an interactive DP mechanism like the Sparse Vector Technique [8, 21].

Although the concurrent composition for the case of differential privacy has not been explored before, it has been studied extensively for many primitives in cryptography, and it is often much more subtle than the sequential composition (See the surveys [4, 14]). For example, standard zero-knowledge protocols are no longer zero-knowledge when a single prover is involved in multiple, simultaneous zero-knowledge proofs with one or multiple verifiers [12, 15].

Our findings is summarized as follows:

- We prove a composition theorem for interactive mechanisms that satisfy approximate differential privacy, where the bound for δ is weaker than the basic composition theorem for noninteractive differential privacy.
- We prove that when the interactive mechanisms being composed are *pure* differentially private, their concurrent composition achieves privacy parameters (with respect to pure or approximate differential privacy) that match the (optimal) composition theorem for noninteractive differential privacy.
- Based on computer experiments, we conjecture that the Optimal Composition Theorems can be extended to the concurrent composition of approximate DP mechanisms. We leave closing the gap as a direction for future research, along with understanding concurrent composition for other variants of differential privacy.

2 DEFINITIONS AND BASIC PROPERTIES

The formal definition of the concurrent composition of interactive protocols is provided here.

Definition 5 (Concurrent Composition of Interactive Protocols). *Let $\mathcal{M}_0, \ldots, \mathcal{M}_{k-1}$ be interactive mechanisms. We say*

$$\mathcal{M} = \mathrm{ConComp}(\mathcal{M}_0, \ldots, \mathcal{M}_{k-1})$$

is the concurrent composition *of mechanisms $\mathcal{M}_0, \ldots, \mathcal{M}_{k-1}$ if $\mathcal{M}$ runs as follows:*

1. *Random coin tosses for $\mathcal{M}$ consist of $r = (r_0, \ldots, r_{k-1})$ where r_j are random coin tosses for $\mathcal{M}_j$.*
2. *Inputs for $\mathcal{M}$ consists of $x = (x_0, \ldots, x_{k-1})$ where x_j is private input for $\mathcal{M}_j$.*
3. *$\mathcal{M}(x, m_0, \ldots, m_{i-1}; r)$ is defined as follows:*
   1. *(a) Parse m_{i-1} as (q, j) where q is a query and $j \in [k]$. If m_{i-1} cannot be parsed correctly, output* `halt`.
   2. *(b) Extract history $(m_0^j, \ldots, m_{t-1}^j)$ from $(m_0, \ldots, m_{i-1})$ where m_i^j are all of the queries to mechanism $\mathcal{M}_j$.*
   3. *(c) Output $\mathcal{M}_j(x_j, m_0^j, \ldots, m_{t-1}^j; r_j)$.*

We are mainly interested in the case where all mechanisms operate on the same dataset, i.e., the private input for each $\mathcal{M}_i$ are all the same.

We show that to prove an interactive DP mechanism is (ε, δ)-differentially private, it suffices to consider all deterministic adversaries.

Lemma 1. *An interactive mechanism $\mathcal{M}$ is (ε, δ)-differentially private if and only if for every pair of adjacent datasets x, x', for every* deterministic *adversary algorithm $\mathcal{A}$, for every possible output set $T \subseteq \mathrm{Range}\,(\mathtt{View}\langle\mathcal{A}, \mathcal{M}(\cdot)\rangle)$ we have*

$$\Pr\left[\mathtt{View}\langle\mathcal{A}, \mathcal{M}(x)\rangle \in T\right] \leq e^{\varepsilon} \Pr\left[\mathtt{View}\langle\mathcal{A}, \mathcal{M}(x')\rangle \in T\right] + \delta \quad (3)$$

More properties of concurrent compositions are be provided in the full version of the paper.

3 BASIC COMPOSITION THEOREM FOR CONCURRENT COMPOSITION

Our first result is roughly an analogue of the Basic Composition Theorem.

Theorem 2. *If interactive mechanisms $\mathcal{M}_0, \ldots, \mathcal{M}_{k-1}$ are each (ε, δ)-differentially private, then their concurrent composition* $\mathrm{ConComp}(\mathcal{M}_0, \ldots, \mathcal{M}_{k-1})$ *is $\left(k \cdot \varepsilon, \frac{e^{k\varepsilon}-1}{e^{\varepsilon}-1} \cdot \delta\right)$-differentially private.*

More generally, if interactive mechanism $\mathcal{M}_i$ is $(\varepsilon_i, \delta_i)$-differentially private for $i = 0, \ldots, k-1$, then $\mathrm{ConComp}(\mathcal{M}_0, \ldots, \mathcal{M}_{k-1})$ *is $(\varepsilon_g, \delta_g)$-differentially private where $\varepsilon_g = \sum_{i=0}^{k-1} \varepsilon_i$, and $\delta_g = \sum_{i=0}^{k-1} e^{\sum_{j=0}^{i-1} \varepsilon_j} \cdot \delta_i$.*

Just like in the Basic Composition Theorem for noninteractive DP, the privacy-loss parameters ε_i just sum up. However, the bound on δ_g is worse by a factor of at most e^{ε_g}. In the typical setting where we want to enforce a global privacy loss of $\varepsilon_g = O(1)$, this is only a constant-factor loss compared to the Basic Composition Theorem, but that constant can be important in practice. Note that expression for δ_g depends on the ordering of the k mechanisms $\mathcal{M}_0, \ldots, \mathcal{M}_{k-1}$, so one can optimize it further by taking a permutation of the mechanisms that minimizes δ_g.

The proof idea is that in an interactive protocol where the adversary is concurrently interacting with multiple mechanisms, its interaction with one particular mechanism could be viewed as the combination of the adversary and the remaining mechanisms interacting with that mechanism, and the differential privacy guarantee still holds for the "combined adversary". We can then construct a standard hybrid argument. Specifically, we compare the distributions of $H_0 = \mathtt{View}\langle\mathcal{A}, \mathrm{ConComp}(\mathcal{M}_0(x), \mathcal{M}_1(x), \ldots, \mathcal{M}_{k-1}(x))\rangle$ and $H_k = \mathtt{View}\langle\mathcal{A}, \mathrm{ConComp}(\mathcal{M}_0(x'), \mathcal{M}_1(x'), \ldots, \mathcal{M}_{k-1}(x'))\rangle$ on adjacent datasets x, x' by changing x to x' for one mechanism at a time, so that H_{i-1} and H_i differ only on the input to $\mathcal{M}_{i-1}$.

To relate H_{i-1} and H_i we consider an adversary strategy $\mathcal{A}_i$ that emulates $\mathcal{A}$'s interaction with $\mathcal{M}_{i-1}$, while internally simulating all of the other $\mathcal{M}_j$'s. Applying a "triangle inequality" to the distance notion given in Requirement (2) yields the result. We note that the proof is very similar to the proof of the group privacy property of (noninteractive) differential privacy, where (ε, δ)-DP for datasets that differ on one record implies $\left(k \cdot \varepsilon, \frac{e^{k\varepsilon}-1}{e^{\varepsilon}-1} \cdot \delta\right)$ for datasets that differ on k records.

4 ADVANCED COMPOSITION THEOREM FOR PURE CONCURRENT COMPOSITION

Next we show that the Advanced and Optimal Composition Theorems for noninteractive DP can be extend to interactive DP, provided that the mechanisms $\mathcal{M}_i$ being composed satisfy pure DP (i.e. $\delta_i = 0$). Note that the final composed mechanism $\mathrm{ConComp}(\mathcal{M}_0, \ldots, \mathcal{M}_{k-1})$ can be approximate DP, by taking $\delta_g = \delta' > 0$, and thereby allowing for a privacy loss ε_g that grows linearly in $\sqrt{k}$ rather than k.

We do this by extending the main proof technique of [17, 20] to interactive DP mechanisms. Specifically, we reduce the analysis of interactive $(\varepsilon, 0)$-DP mechanisms to that of analyzing the following simple "randomized response" mechanism:

Definition 6 ([17]). *For $\varepsilon > 0, \delta \in [0, 1]$, define a randomized noninteractive algorithm $\mathrm{RR}_{(\varepsilon,\delta)} : \{0, 1\} \to \{0, 1, \texttt{'Iam0'}, \texttt{'Iam1'}\}$ as follows:*

$$\begin{array}{ll}
\Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(0) = \texttt{'Iam0'}\right] = \delta & \Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(1) = \texttt{'Iam0'}\right] = 0 \\
\Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(0) = 0\right] = \frac{e^{\varepsilon}(1-\delta)}{1+e^{\varepsilon}} & \Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(1) = 0\right] = \frac{1-\delta}{1+e^{\varepsilon}} \\
\Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(0) = 1\right] = \frac{1-\delta}{1+e^{\varepsilon}} & \Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(1) = 1\right] = \frac{e^{\varepsilon}(1-\delta)}{1+e^{\varepsilon}} \\
\Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(0) = \texttt{'Iam1'}\right] = 0 & \Pr\left[\mathrm{RR}_{(\varepsilon,\delta)}(1) = \texttt{'Iam1'}\right] = \delta
\end{array}$$

Note that $\mathrm{RR}_{(\varepsilon,\delta)}$ is a noninteractive (ε, δ)-DP mechanism. For simplicity, when $\delta = 0$, we denote the mechanism as $\mathrm{RR}_{\varepsilon}$. We show that every interactive $(\varepsilon, 0)$-DP mechanism can be, in some sense, simulated from $\mathrm{RR}_{\varepsilon}$:

Theorem 3. *Suppose that $\mathcal{M}$ is an interactive $(\varepsilon, 0)$-differentially private mechanism. Then for every pair of adjacent datasets x_0, x_1 there exists an interactive mechanism T s.t. for every adversary $\mathcal{A}$ and every $b \in \{0, 1\}$ we have*

$$\texttt{View}(\mathcal{A}, \mathcal{M}(x_b)) \equiv \texttt{View}(\mathcal{A}, T(\mathrm{RR}_{\varepsilon}(b)))$$

Here T is an interactive mechanism that depends on $\mathcal{M}$ as well as a fixed pair of adjacent datasets x_0 and x_1. It receives a single bit as an output of $\mathrm{RR}_{\varepsilon}(b)$, and then interacts with the adversary $\mathcal{A}$ just like $\mathcal{M}$ would. Kairouz, Oh, and Viswanath [17] proved Theorem 3 result for the case that $\mathcal{M}$ and T are noninteractive. The interactive case is more involved because we need a single T that works for all adversary strategies $\mathcal{A}$. (If we allow T to depend on the adversary strategy $\mathcal{A}$, then the result would readily follow from that of [17], but this would not suffice for our application to concurrent composition.)

Given the Theorem 3, to analyze

$$\mathrm{ConComp}(\mathcal{M}_0(x_b), \ldots, \mathcal{M}_{k-1}(x_b))$$

on $b = 0$ vs. $b = 1$, it suffices to analyze

$$\mathrm{ConComp}(T_0(\mathrm{RR}_{\varepsilon_0}(b)), \ldots, T_{k-1}(\mathrm{RR}_{\varepsilon_{k-1}}(b)))$$

. An adversary's view interacting with the latter concurrent composition can be simulated entirely from the output of $\mathrm{Comp}(\mathrm{RR}_{\varepsilon_0}(b), \ldots, \mathrm{RR}_{\varepsilon_{k-1}}(b))$, which is the composition of entirely noninteractive mechanisms. Thus, we conclude:

Corollary 4. *The Advanced and Optimal Composition Theorems extend to the concurrent composition of $(\varepsilon_i, \delta_i)$-interactive DP mechanisms $\mathcal{M}_i$ provided that $\delta_0 = \delta_1 = \cdots = \delta_{k-1} = 0$.*

5 FUTURE WORK

We leave the question of whether or not the Advanced and/or Optimal Composition Theorems extend to the concurrent composition of approximate DP mechanisms (with $\delta_i > 0$) for future work. The Optimal Composition Theorem for noninteractive approximate DP [20] is also proven by showing that any noninteractive (ε, δ)-DP mechanism can be simulated by an approximate-DP generalization of randomized response, $\mathrm{RR}_{(\varepsilon,\delta)}$, analogously to Theorem 3.

In the full version of the paper, we present empirical evidence for our conjecture that the Optimal Composition Theorems can be extended to the concurrent composition of approximate DP mechanisms. Specifically, we experimentally evaluate the conjecture for 3-message interactive mechanisms with 1-bit messages, and we test whether instantiations of this 2-round interactive mechanism that are (ε, δ)-DP can be simulated as the interactive post-processing of randomized response $\mathrm{RR}_{(\varepsilon,\delta)}$. In all of our trials, we find a feasible interactive post-processing algorithm. The implementation details of the experiment can be found in the full version of the paper. Based on the above findings, we conjecture that the concurrent composition of interactive DP mechanisms may still have the same bound as the composition for noninteractive DP mechanisms. Besides, we might be able to prove it through a similar construction of interactive post-processing mechanisms as we did in Theorem 3. This means that every interactive DP mechanisms can be reduced to noninteractive randomized response. We leave the resolution of these conjectures for future work.

Another interesting question for future work is analyzing concurrent composition for variants of differential privacy, such as Concentrated DP [2, 3, 10], Rényi DP [19], and Gaussian DP [6]. Some of these notions require bounds on Rényi divergences, e.g. that

$$D_{\alpha}(\texttt{View}\langle\mathcal{A}, \mathcal{M}(x)\rangle || \texttt{View}\langle\mathcal{A}, \mathcal{M}(x')\rangle) \leq \rho,$$

for adjacent datasets x, x' and certain pairs (α, ρ). Here sequential composition can be argued using a chain rule for Rényi divergence:

$$D_{\alpha}((Y, Z)||(Y', Z')) \leq D_{\alpha}(Y||Y') + \sup_{y} D_{\alpha}(Z|_{Y=y}||Z'|_{Y'=y}). \quad (4)$$

Taking Y to be the view of the analyst interacting with $\mathcal{M}_0(x)$, Z to be the view of the analyst in a subsequent interaction with $\mathcal{M}_1(x)$, and Y' and Z' to be analogously defined with respect to an adjacent dataset x', we obtain the usual composition bound of $\rho_0 + \rho_1$ on the overall Rényi divergence of order α, where ρ_0 and ρ_1 are the privacy-loss parameters of the individual mechanisms. However, this argument fails for concurrent DP, since we can no longer assert the privacy properties of $\mathcal{M}_1$ conditioned on any possible value y of the adversary's view of the interaction with $\mathcal{M}_0$. Unfortunately, the Chain Rule (4) does not hold if we replace the supremum with an expectation, so a new proof strategy is needed (if the composition theorem remains true).

REFERENCES

[1] Amos Beimel, Kobbi Nissim, and Eran Omri. 2008. Distributed private data analysis: Simultaneously solving how and what. In *Annual International Cryptology Conference*. Springer, 451–468.

[2] Mark Bun, Cynthia Dwork, Guy N Rothblum, and Thomas Steinke. 2018. Composable and versatile privacy via truncated cdp. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. 74–86.

[3] Mark Bun and Thomas Steinke. 2016. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*. Springer, 635–658.

[4] Ran Canetti, Uri Feige, Oded Goldreich, and Moni Naor. 1996. Adaptively secure multi-party computation. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 639–648.

[5] Lijie Chen, Badih Ghazi, Ravi Kumar, and Pasin Manurangsi. 2020. On Distributed Differential Privacy and Counting Distinct Elements. *arXiv preprint arXiv:2009.09604* (2020).

[6] Jinshuo Dong, Aaron Roth, and Weijie J Su. 2019. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383* (2019).

[7] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*. Springer, 265–284.

[8] Cynthia Dwork, Moni Naor, Omer Reingold, Guy N Rothblum, and Salil Vadhan. 2009. On the complexity of differentially private data release: efficient algorithms and hardness results. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*. 381–390.

[9] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3-4 (2014), 211–407.

[10] Cynthia Dwork and Guy N Rothblum. 2016. Concentrated differential privacy. *arXiv preprint arXiv:1603.01887* (2016).

[11] Cynthia Dwork, Guy N Rothblum, and Salil Vadhan. 2010. Boosting and differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE, 51–60.

[12] Uriel Feige and Adi Shamir. 1989. Zero knowledge proofs of knowledge in two rounds. In *Conference on the Theory and Application of Cryptology*. Springer, 526–544.

[13] Marco Gaboardi, Michael Hay, and Salil Vadhan. 2020. A Programming Framework for OpenDP. (2020).

[14] Oded Goldreich. 2019. *Providing sound foundations for cryptography: on the work of Shafi Goldwasser and Silvio Micali*. Morgan & Claypool.

[15] Oded Goldreich and Hugo Krawczyk. 1996. On the composition of zero-knowledge proof systems. *SIAM J. Comput.* 25, 1 (1996), 169–192.

[16] Moritz Hardt and Guy N Rothblum. 2010. A multiplicative weights mechanism for privacy-preserving data analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE, 61–70.

[17] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2015. The composition theorem for differential privacy. In *International conference on machine learning*. PMLR, 1376–1385.

[18] Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2011. What can we learn privately? *SIAM J. Comput.* 40, 3 (2011), 793–826.

[19] Ilya Mironov. 2017. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 263–275.

[20] Jack Murtagh and Salil Vadhan. 2016. The complexity of computing the optimal composition of differential privacy. In *Theory of Computing*. Theory of Computing, 157–175.

[21] Aaron Roth and Tim Roughgarden. 2010. Interactive privacy via the median mechanism. In *Proceedings of the forty-second ACM symposium on Theory of computing*. 765–774.

[22] Salil Vadhan. 2017. The complexity of differential privacy. In *Tutorials on the Foundations of Cryptography*. Springer, 347–450.