# Differentially Private Multi-party Computation

**Peter Kairouz**[1] **Sewoong Oh**[2] **Pramod Viswanath**[1]
[1]Department of Electrical & Computer Engineering
[2]Department of Industrial & Enterprise Systems Engineering
University of Illinois Urbana-Champaign
Urbana, IL 61801, USA
{kairouz2,swoh,pramodv}@illinois.edu

## Abstract

We study the problem of multi-party computation under approximate $(\varepsilon, \delta)$ differential privacy. We assume an interactive setting with $k$ parties, each possessing a private bit. Each party wants to compute a function defined on all the parties' bits. Differential privacy ensures that there remains uncertainty in any party's bit even when given the transcript of interactions and all the other parties' bits. This paper is a follow up to our work in [9], where we studied multi-party computation under $(\varepsilon, 0)$ differential privacy. We generalize the results in [9] and prove that a simple non-interactive randomized response mechanism is optimal. Our optimality result holds for all privacy levels (all values of $\varepsilon$ and $\delta$), heterogenous privacy levels across parties, all types of functions to be computed, all types of cost metrics, and both average and worst-case (over the inputs) measures of accuracy.

## 1 Introduction

Multi-party computation (MPC) is a general framework where multiple parties exchange information over a broadcast channel towards the goal of computing a function over their inputs while keeping those inputs private [12, 2, 6, 3]. In this paper, we study the problem of multi-party computation under differential privacy [1, 5, 10, 7]. Each party possesses a single bit of information; the information bits are statistically independent. Each party is interested in computing a function, which could differ from party to party, and there could be a central observer (observing the entire transcript of the interactive communication protocol) that is interested in computing a separate function. The interactive communication is achieved via a broadcast channel that all parties and central observer can hear. It is useful to distinguish between two types of communication protocols: *interactive* and *non-interactive*. We say that a communication protocol is non-interactive if a message broadcasted by one party does not depend on the messages broadcasted by any other party. In contrast, interactive protocols allow the messages at any stage of the communication to depend on all the previous messages that were communicated over the broadcast channel.

**Our contributions.** Our main result is the exact optimality of a simple non-interactive protocol in terms of maximizing accuracy for any given privacy levels: each party randomizes (sufficiently) its own bit and broadcasts the noisy version. Each party and the central observer then separately compute their respective decision functions to maximize the appropriate notion of their accuracy measure. The optimality is general: it holds for all types of functions, heterogeneous privacy conditions on the parties, all types of cost metrics, and both average and worst-case (over the inputs) measures of accuracy. Finally, the optimality result is *simultaneous*, in terms of maximizing accuracy at each of the parties and the central observer. Each party only needs to know its own desired level of privacy, its own function to be computed, and its measure of accuracy. Optimal data release and optimal decision making are naturally separated.

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

**Related work.** Private MPC was first addressed in [5]. The study of accuracy-privacy tradeoffs in the MPC context was first initiated by [1], which studies a paradigm where differential privacy and secure function evaluation (SFE) co-exist. Specific functions, such as the SUM function, were studied under this setting, but no exact optimality results were provided. In the context of two parties, privacy-accuracy tradeoffs have been studied in [10, 7] where a single function is computed by a "third-party" observing the transcript of an interactive protocol. [7] showed that every non-trivial privacy setting incurs loss on any non-trivial boolean function. Further, focusing on the specific scenario where each one of the two parties has a single bit of information, [7] characterized the exact accuracy-privacy tradeoff for AND and XOR functions; the corresponding optimal protocol turns out to be non-interactive. However, this result was derived under some assumptions: only two parties are involved, the central observer is the only entity that computes a function, the function has to be either XOR or AND, symmetric privacy conditions are used for both parties, and accuracy is measured only as worst-case over the four possible inputs. Further, their analysis technique does not generalize to the case when there are more than two parties.

The proof of our result critically relies on an operational interpretation of differential privacy in [9]. Precisely, we show that a simple non-interactive randomized response protocol dominates all $(\varepsilon, \delta)$-differentially private multi-party protocols. This powerful technique bypasses the previous results on the same setting, where weaker results were proved using more sophisticated proof techniques. Specifically, our work generalizes the results in [9], which only addressed $(\varepsilon, 0)$-differential privacy.

## 2 Problem Statement

Consider the setting where there are $k$ parties, each with its own private binary data $x_i \in \{0,1\}$ generated independently. The independence assumption here is necessary because without it each party can learn something about others, which violates differential privacy, even without revealing any information. Differential privacy implicitly imposes independence in a multi-party setting. The goal of each party $i \in [k]$ is to compute an arbitrary function $f_i : \{0,1\}^k \to \mathcal{Y}$ of interest by interactively broadcasting messages. There might be a central observer who listens to all the messages being broadcasted, and wants to compute another arbitrary function $f_0 : \{0,1\} \to \mathcal{Y}$. The $k$ parties are honest in the sense that once they agree on what protocol to follow, every party follows the rules. At the same time, they can be curious, and each party needs to ensure that other parties cannot learn its bit with sufficient confidence. This is done by imposing local differential privacy constraints. This setting is similar to the one studied in [4, 8] in the sense that there are multiple privacy barriers, each one separating an individual party from the rest of the world. However, the main difference is that we consider multi-party computation, where there are multiple functions to be computed, and each node might possess a different function to be computed.

Let $x = [x_1, \ldots, x_k] \in \{0,1\}^k$ denote the vector of $k$ bits, and $x_{-i} = [x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_k] \in \{0,1\}^{k-1}$ is the vector of bits except for the $i^{th}$ bit. The parties agree on an interactive protocol $P$ to achieve the goal of multi-party computation. A 'transcript' $\tau$ is the output of $P$, and is it contains the the sequence of messages exchanged between the parties. Let the probability that a transcript $\tau$ is broadcasted (via a series of interactive communications) when the data is $x$ be denoted by $P_{x,\tau} = \mathbb{P}(\tau \,|\, x)$ for $x \in \{0,1\}^k$ and for $\tau \in \mathcal{T}$. Then, a protocol can be represented as a matrix denoting the probability distribution over a set of transcripts $\mathcal{T}$ conditioned on $x$: $P = [P_{x,\tau}] \in [0,1]^{2^k \times |\mathcal{T}|}$.

In the end, each party makes a decision on what the value of function $f_i$ is, based on its own bit $x_i$ and the transcript $\tau$ that was broadcasted. A decision rule is a mapping from a transcript $\tau \in \mathcal{T}$ and private bit $x_i \in \{0,1\}$ to a decision $y \in \mathcal{Y}$ represented by a function $\hat{f}_i(\tau, x_i)$. We allow randomized decision rules, in which case $\hat{f}_i(\tau, x_i)$ can be a random variable. For the central observer, a decision rule is a function of just the transcript, denoted by a function $\hat{f}_0(\tau)$.

We consider two notions of accuracy: the average accuracy and the worst-case accuracy. For the $i^{th}$ party, consider an accuracy measure $w_i : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ (or equivalently a negative cost function) such that $w_i(f_i(x), \hat{f}_i(\tau, x_i))$ measures the accuracy when the function to be computed is $f_i(x)$ and the

approximation is $\hat{f}_i(\tau, x_i)$. Then the average accuracy for this $i^{th}$ party is defined as

$$\mathrm{ACC}_{\mathrm{ave}}(P, w_i, f_i, \hat{f}_i) \equiv \frac{1}{2^k} \sum_{x \in \{0,1\}^k} \mathbb{E}_{\hat{f}_i, P_{x,\tau}}[w_i(f_i(x), \hat{f}_i(\tau, x_i))] \ , \tag{1}$$

where the expectation is taken over the random transcript $\tau$ and any randomness in the decision function $\hat{f}_i$. For example, if the accuracy measure is an indicator such that $w_i(y, y') = \mathbb{I}_{(y=y')}$, then $\mathrm{ACC}_{\mathrm{ave}}$ measures the average probability of getting the correct function output. For a given protocol $P$, it takes $(2^k \, |\mathcal{T}|)$ operations to compute the optimal decision rule:

$$f^*_{i,\mathrm{ave}}(\tau, x_i) = \arg\max_{y \in \mathcal{Y}} \sum_{x_{-i} \in \{0,1\}^{k-1}} P_{x,\tau} \, w_i(f_i(x), y) \ , \tag{2}$$

for each $i \in [k]$. The computational cost of $(2^k \, |\mathcal{T}|)$ for computing the optimal decision rule is *unavoidable in general*, since that is the inherent complexity of the problem: describing the distribution of the transcript requires the same cost. We will show that the optimal protocol requires a set of transcripts of size $|\mathcal{T}| = 2^k$, and the computational complexity of the decision rule for a general function is $2^{2k}$. However, for a fixed protocol, this decision rule needs to be computed only once before any message is transmitted. Further, it is also possible to find a closed form solution for the decision rule when $f$ has a simple structure. One example is the XOR function where the optimal decision rule is as simple as evaluating the XOR of all the received bits, which requires $O(k)$ operations. When there are multiple maximizers $y$, we can choose either one of them arbitrarily, and it follows that there is no gain in randomizing the decision rule for average accuracy.

Similarly, the worst-case accuracy is defined as

$$\mathrm{ACC}_{\mathrm{wc}}(P, w_i, f_i, \hat{f}_i) \equiv \min_{x \in \{0,1\}^k} \mathbb{E}_{\hat{f}_i, P_{x,\tau}}[w_i(f_i(x), \hat{f}_i(\tau, x_i))] \ . \tag{3}$$

For worst-case accuracy, given a protocol $P$, the optimal decision rule of the $i^{th}$ party with a bit $x_i$ can be computed by solving the following convex program:

$$\begin{aligned} Q^{(x_i)} = & \\ \arg\max_{Q \, \in \, \mathbb{R}^{|\mathcal{T}| \times |\mathcal{Y}|}} & \quad \min_{x_{-i} \in \{0,1\}^{k-1}} \sum_{\tau \in \mathcal{T}} \sum_{y \in \mathcal{Y}} P_{x,\tau} \, w_i(f_i(x), y) Q_{\tau,y} \\ \text{subject to} & \quad \sum_{y \in \mathcal{Y}} Q_{\tau,y} = 1 \ , \ \forall \tau \in \mathcal{T} \text{ and } Q \geq 0 \end{aligned} \tag{4}$$

The optimal (random) decision rule $f^*_{i,\mathrm{wc}}(\tau, x_i)$ is to output $y$ given transcript $\tau$ according to $\mathbb{P}(y|\tau, x_i) = Q^{(x_i)}_{\tau,y}$. This can be formulated as a linear program with $|\mathcal{T}| \times |\mathcal{Y}|$ variables and $2^k + |\mathcal{T}|$ constraints. Again, it is possible to find a closed form solution for the decision rule when $f$ has a simple structure: for the XOR function, the optimal decision rule is again evaluating the XOR of all the received bits requiring $O(k)$ operations.

For a central observer, the accuracy measures are defined similarly, and the optimal decision rule is now

$$f^*_{0,\mathrm{ave}}(\tau) \quad = \quad \arg\max_{y \in \mathcal{Y}} \sum_{x \in \{0,1\}^k} P_{x,\tau} \, w_0(f_0(x), y) \ , \tag{5}$$

and for worst-case accuracy the optimal (random) decision rule $f^*_{0,\mathrm{wc}}(\tau)$ is to output $y$ given transcript $\tau$ according to $\mathbb{P}(y|\tau) = Q^{(0)}_{\tau,y}$.

$$\begin{aligned} Q^{(0)} = & \\ \arg\max_{Q \, \in \, \mathbb{R}^{|\mathcal{T}| \times |\mathcal{Y}|}} & \quad \min_{x \in \{0,1\}^k} \sum_{\tau \in \mathcal{T}} \sum_{y \in \mathcal{Y}} P_{x,\tau} \, w_0(f_0(x), y) Q_{\tau,y} \\ \text{subject to} & \quad \sum_{y \in \mathcal{Y}} Q_{\tau,y} = 1 \ , \ \forall \tau \in \mathcal{T} \text{ and } Q \geq 0 \end{aligned} \tag{6}$$

where $w_0 : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is the measure of accuracy for the central observer.

Consider the following simple protocol known as the *randomized response*, which is a term first coined by [11] and commonly used in many private communications including the multi-party setting [10]. We will show in Section 3 that this is the optimal protocol that simultaneously maximizes the accuracy for all the parties. Each party broadcasts a randomized version of its bit denoted by $\tilde{x}_i$ such that

$$\tilde{x}_i = \begin{cases} 0 & \text{if } x_i = 0 \text{ with probability } \delta_i \ , \\ 1 & \text{if } x_i = 0 \text{ with probability } \dfrac{(1-\delta_i)e^{\varepsilon_i}}{1+e^{\varepsilon_i}} \ , \\ 2 & \text{if } x_i = 0 \text{ with probability } \dfrac{(1-\delta_i)}{1+e^{\varepsilon_i}} \ , \\ 3 & \text{if } x_i = 0 \text{ with probability } 0 \ , \end{cases}$$

$$\tilde{x}_i = \begin{cases} 0 & \text{if } x_i = 1 \text{ with probability } 0 \ , \\ 1 & \text{if } x_i = 1 \text{ with probability } \dfrac{(1-\delta_i)}{1+e^{\varepsilon_i}} \ , \\ 2 & \text{if } x_i = 1 \text{ with probability } \dfrac{(1-\delta_i)e^{\varepsilon_i}}{1+e^{\varepsilon_i}} \ , \\ 3 & \text{if } x_i = 1 \text{ with probability } \delta_i \ . \end{cases} \tag{7}$$

The proof of optimality of this randomized response depends on an operational definition of differential privacy, and we refer to [9].

## 3 Main Result

We show, perhaps surprisingly, that the simple randomized response presented in (7) is the unique optimal protocol in a very general sense.

**Theorem 3.1** *Let the optimal decision rule be defined as in* (2) *for the average accuracy and* (5) *for the worst-case accuracy. Then, for any privacy levels* $(\varepsilon_i, \delta_i)$*, any function* $f_i : \{0,1\}^k \to \mathcal{Y}$*, and any accuracy measure* $w_i : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ *for* $i \in [k]$*, together with the optimal decision rule, the randomized response achieves the maximum accuracy for the* $i^{th}$ *party among all differentially private interactive and non-interactive protocols. For the central observer, the randomized response with the optimal decision rule defined in* (5) *and* (7) *achieves the maximum accuracy among all* $\{(\varepsilon_i, \delta_i)\}$*-differentially private interactive protocols and all decision rules for any arbitrary function* $f_0$ *and any measure of accuracy* $w_0$*.*

This is a strong optimality result. Every party and the central observer can simultaneously achieve the optimal accuracy, using a universal randomized response. Each party only needs to know its own desired level of privacy, its own function to be computed, and its measure of accuracy. Optimal data release and optimal decision making are naturally separated. It is not immediate at all that such a simple non-interactive randomized response mechanism would achieve the maximum accuracy. The proof critically harnesses the data processing inequalities and is provided in [9].

## 4 Conclusion

In this paper, we studied the problem of differentially private multi-party computation. We showed that a simple non-interactive randomized response is optimal for all privacy levels (all values of $\varepsilon$ and $\delta$), heterogenous privacy levels across parties, all types of functions to be computed, all types of cost metrics, and both average and worst-case (over the inputs) measures of accuracy. Though our results are general, they only handle settings where each party possesses a single bit. In the more general scenario where parities can have multiple bits, interaction might be critical to achieving the optimal privacy-utility tradeoffs.

## References


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

[2] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 1–10. ACM, 1988.

[3] David Chaum, Claude Crépeau, and Ivan Damgard. Multiparty unconditionally secure protocols. In *Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 11–19. ACM, 1988.

[4] J. C. Duchi, M. I. Jordan, and M. J. Wainwright. Local privacy and statistical minimax rates. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 429–438. IEEE, 2013.

[5] 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*, pages 486–503. Springer, 2006.

[6] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, pages 218–229, New York, NY, USA, 1987. ACM.

[7] Vipul Goyal, Ilya Mironov, Omkant Pandey, and Amit Sahai. Accuracy-privacy tradeoffs for two-party differentially private protocols. In *Advances in Cryptology–CRYPTO 2013*, pages 298–315. Springer, 2013.

[8] P. Kairouz, S. Oh, and P. Viswanath. Extremal mechanisms for local differential privacy. In *Advances in Neural Information Processing Systems 27*, pages 2879–2887, 2014.

[9] P. Kairouz, S. Oh, and P. Viswanath. Secure multi-party differential privacy. In *Advances in Neural Information Processing Systems*, 2015.

[10] Andrew McGregor, Ilya Mironov, Toniann Pitassi, Omer Reingold, Kunal Talwar, and Salil Vadhan. The limits of two-party differential privacy. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 81–90. IEEE, 2010.

[11] S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, 1965.

[12] Andrew C Yao. Protocols for secure computations. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 160–164. IEEE, 1982.