Local Differential Privacy Meets Computational Social Choice - Resilience under Voter Deletion*

Liangde Tao[1] , **Lin Chen**[2†] , **Lei Xu**[3] and **Weidong Shi**[4]
[1]Zhejiang University
[2]Texas Tech University
[3]Kent State University
[4]University of Houston
{vast.tld,chenlin198662,xuleimath}@gmail.com, larryshi@ymail.com

Abstract

The resilience of a voting system has been a central topic in computational social choice. Many voting rules, like plurality, are shown to be vulnerable as the attacker can target specific voters to manipulate the result. What if a local differential privacy (LDP) mechanism is adopted such that the true preference of a voter is never revealed in pre-election polls? In this case, the attacker can only infer stochastic information about a voter's true preference, and this may cause the manipulation of the electoral result significantly harder. The goal of this paper is to provide a quantitative study on the effect of adopting LDP mechanisms on a voting system. We introduce the metric PoLDP (power of LDP) that quantitatively measures the difference between the attacker's manipulation cost under LDP mechanisms and that without LDP mechanisms. The larger PoLDP is, the more robustness LDP mechanisms can add to a voting system. We give a full characterization of PoLDP for the voting system with plurality rule and provide general guidance towards the application of LDP mechanisms.

1 Introduction

In this paper, we consider the attack on an abstract voting system. We compare the manipulation cost under two scenarios of the voting system: the classical voting system, and the voting system where a local differential privacy (LDP) mechanism is adopted. Our goal is to quantify the difference brought by LDP schemes to a voting system under electoral manipulation.

Extensive research has been conducted towards characterizing the resilience of various voting rules (see, e.g., Brandt et al. [2016] for a survey). In many of these research works, the attacker is assumed to know the true preference of voters through pre-election polls. Many voting rules, especially fundamental rules like plurality, are found to be vulnerable [Faliszewski *et al.*, 2009]. Can we enhance the resilience of a voting system by preventing the attacker from learning the preference of voters? Towards this, we observe that LDP mechanisms are a state-of-the-art approach to mitigating the leakage of sensitive information.

LDP mechanism serves as an important method in social science, particularly in voter turnout reports [Rosenfeld *et al.*, 2016]. However, most existing research works on LDP focus on the analysis of each user's privacy, not much is known from the aspect of the system's security (see, e.g., Bebensee [2019] as a survey). Does the introduction of LDP mechanisms make the system more resilient? The most relevant research in this direction is a very recent paper [Cheu *et al.*, 2021], where they showed that an attacker may manipulate a small percentage of users in an LDP mechanism to mislead the estimation of distribution parameters. However, their attack model is completely different from what we study in this paper.

We believe characterization of the resilience of a voting system under LDP mechanisms is a natural question that is worth investigating. The high-level description of the specific model we study in this paper is given as follows:

- *Election setting:* We consider the classical election where there are a set of candidates and a set of independent voters. We focus on the plurality rule (e.g., each voter votes for exactly one candidate). For each voter, we define the type[1] of his/her preference as the candidate he/she votes for. Candidate(s) receiving the highest score will be the co-winner(s).
- *LDP mechanism:* Every voter will locally and independently run a given LDP mechanism once to generate his/her reported preference which will be used for all pre-election polls. This is a common approach, see e.g., Google's RAPPOR [Erlingsson *et al.*, 2014]. Otherwise, the attacker may be able to figure out the voter's true preference via his/her multiple reported preferences [Bebensee, 2019]. In this paper, we focus on two fundamental LDP mechanisms: *randomized response* and *Laplace*.
- *Attack model:* We consider electoral control by deleting

*The full version of this paper is available at https://arxiv.org/abs/2205.00771

†Corresponding Author.

[1]For each voter, the type of his/her preference is private information, and will be referred to as the true type to distinguish from the reported type which is generated by the LDP mechanism.

voters. The *manipulation cost* is the minimal number of voters the attacker should delete to make the designated candidate win.

We emphasize that the adoption of LDP mechanisms will *not* change the electoral result since the mechanisms are only used in pre-election polls. Voters will still vote according to their true preference in the election, but will respond using the reported preference in any pre-election polls. Therefore when the attacker launches the voter deletion attack prior to the election, he/she only knows the reported preference of voters instead of their true preference.

Measuring the impact of LDP mechanisms. In the classic election setting, the attacker knows the true preference of all voters. Whereas, after adopting the LDP mechanism, the attacker only knows the reported preference.

Inspired by the concept of PoA (price of anarchy) in algorithmic game theory [Koutsoupias and Papadimitriou, 1999], we introduce the metric PoLDP (power of LDP) to quantitatively characterize the resilience brought by the LDP mechanism. Roughly, we can view PoLDP as follows:

$$\text{PoLDP} = \frac{\text{Minimal expected manipulation cost with LDP}}{\text{Minimal manipulation cost without LDP}}.$$

If PoLDP > 1, then it means the introduction of LDP mechanisms indeed increases the manipulation cost of the attacker, and thus enhances the resilience of the system; if PoLDP < 1, then it means the introduction of LDP mechanisms reduces the manipulation cost, and thus diminishes the resilience of the system. It should be noted that the value of PoLDP depends on the specific LDP mechanism as well as the value of the privacy parameter[2].

As randomness is involved in an LDP mechanism, the manipulation cost under the LDP mechanism is a random variable, and our definition uses its expected value. One may ask what if this random variable takes a value significantly different from the expectation? Indeed, we will show that since the manipulation cost can be expressed as a summation of independent binary variables, probabilistic analysis guarantees that the manipulation cost as a random variable lies around its expectation with an extremely high probability, and consequently our definition of PoLDP by using the expected value is without loss of generality.

Our contributions. The main contribution of this paper is to give the first quantitative analysis of the effect brought by local differential privacy in a voting system under electoral control of voter deletion. We study two major LDP mechanisms that are widely adopted, randomized response and Laplace, and show that they can generally enhance the resilience of a voting system. We quantify such an effect through a measure called PoLDP. The larger PoLDP is, the more resilience LDP adds to the system.

Since LDP mechanisms introduce uncertainty, the attacker may need to pay a different manipulation cost to make sure the designated candidate wins with a different probability. That is, the manipulation cost, and consequently PoLDP, is a function of the winning probability.

[2]See Definition 1 for the meaning of the privacy parameter ϵ.

For specific winning probability, we establish two integer linear programs to give a general upper and lower bound on the manipulation cost. Under some mild assumptions, we show that the upper and lower bound approach to the same value for a big range of the winning probability. That is the manipulation cost for a winning probability of 99.9% is almost the same as that for a winning probability of 0.1%.

Using probabilistic analysis, we give an efficient method to calculate the APoLDP (Asymptotic PoLDP) for voting systems satisfying certain conditions. Furthermore, we study the relationship between the value of APoLDP and the parameter ϵ of LDP mechanisms. For randomized response and Laplace mechanism, we give the closed-form expression of APoLDP for voting systems with only two candidates. For voting systems with multiple candidates, it becomes too complicated to obtain a simple mathematical formula. Instead, we analyze the maximal1 value of APoLDP, and the range of ϵ where APoLDP achieves the maximal value.

Interestingly, we observe that when the parameter ϵ of LDP mechanisms is below a certain threshold (we call it security threshold), APoLDP stays at its maximal value. Generally, LDP mechanisms with smaller ϵ can guarantee better privacy. But for manipulation via voter deletion, such kind of extra privacy provided by LDP mechanisms is redundant and can not add more resistance. The security threshold provides general guidance toward the application of LDP mechanisms.

Related works. Local differential privacy has been increasingly accepted as a state-of-the-art approach for statistical computations while protecting the privacy of each participant. It was first formalized in [Kasiviswanathan *et al.*, 2011]. Several important LDP mechanisms are studied and compared in the literature [Wang and Blocki, 2017; Bun *et al.*, 2019; Qin *et al.*, 2016], including the two major LDP mechanisms, randomized response and Laplace considered in this paper. LDP mechanisms have also been adopted by IT companies, e.g., RAPPOR by Google Chrome [Erlingsson *et al.*, 2014]. We refer the reader to a comprehensive survey [Bebensee, 2019].

The study of the computational complexity of electoral control was initiated by Bartholdi III et al. [1992], who mainly analyzed the voting rule of plurality and condorcet, where the attacker's goal is to make a designated candidate win. Later, Hemaspaandra et al. [2007] studied a closely related model where the attacker's goal is to make the original winner lose. Following their research work, extensive research has been conducted towards understanding the resilience of voting systems under different electoral control methods and voting rules [Hemaspaandra *et al.*, 2017; Magiera and Faliszewski, 2017; Maushagen and Rothe, 2016; Rey and Rothe, 2016]. While the research has characterized different voting rules as resilient or vulnerable, not much is known about protecting a vulnerable voting rule like plurality. Electoral control under partial information is also considered in the literature [Conitzer *et al.*, 2011; Dey *et al.*, 2018]. Very recently, Chen et al. [2018] and Yin et al. [2018] studied the protection of election through deploying defending resources.

2 Preliminary

In this section, we briefly introduce the concept of local differential privacy mechanism. We use the definition given by [Erlingsson *et al.*, 2014].

Definition 1. *ϵ-Local Differential Privacy: We say that an mechanism $\mathcal{R}$ satisfies ϵ-local differential privacy where $\epsilon > 0$ if and only if for any input v, v' and any $y \in range(\mathcal{R})$ it holds that*

$$\frac{\Pr[\mathcal{R}(v) = y]}{\Pr[\mathcal{R}(v') = y]} \leq e^{\epsilon},$$

where $range(\mathcal{R})$ denotes the set of all possible outputs of the mechanism $\mathcal{R}$.

We call $\mathcal{R}$ an ϵ-LDP mechanism if it satisfies the ϵ-local differential privacy. The privacy parameter ϵ characterizes the level of privacy provided by the LDP mechanism $\mathcal{R}$. For example, perfect privacy is ensured when $\epsilon = 0$. And, no privacy is guaranteed when $\epsilon = +\infty$.

As mentioned before, the type of true preference of each voter (we refer to it as true type) is private information. Voters will vote according to their true types in the election. In the meantime, we let every voter V_i independently and locally run an ϵ-LDP mechanism (which is the randomized response or Laplace in this paper) to generate a type of his/her reported preference (we refer to it as reported type) and use the reported type in all pre-election polls. Consequently, the reported type of each voter is public information.

Design Matrix. We now introduce the design matrix of an LDP mechanism. In our problem, the design matrix maps the true type to the reported type for each voter. Consider an arbitrary voter V_i. Let t_i be the true type, and r_i be the reported type generated by ϵ-LDP mechanisms. There are m candidates in the voting system. Consequently, there are m different types under plurality. Hence, we know that $t_i, r_i \in [1, m]$.

Consider the probability that V_i has a true type v but reports u, i.e., the event that its reported type is u conditioned on the event that its true type is v, or $\Pr[r_i = u|t_i = v]$. It is clear that since each voter independently and locally runs the LDP mechanism, $\Pr[r_i = u|t_i = v]$ is the same for all voters. Hence, we denote by $p_{uv} = \Pr[r_i = u|t_i = v]$, and let $P = (p_{uv})_{m\times m}$. P is called the design matrix and is only dependent on the LDP mechanism.

We consider two major LDP mechanisms, randomized response and Laplace. Given the privacy parameter ϵ, we denote by $P^{\epsilon\text{-ran}} = (p^{\epsilon\text{-ran}}_{uv})_{m\times m}$ and $P^{\epsilon\text{-lap}} = (p^{\epsilon\text{-lap}}_{uv})_{m\times m}$ the design matrices for randomized response and Laplace, respectively. According to Wang et al. [2016], the design matrices are given by:

$$p^{\epsilon\text{-ran}}_{uv} = \begin{cases} \theta + (1-\theta)/m & \text{if } u = v \\ (1-\theta)/m & \text{if } u \neq v \end{cases} \tag{1}$$

where $\theta = 1 - m/(m - 1 + e^{\epsilon})$, and

$$p^{\epsilon\text{-lap}}_{uv} = \begin{cases} F_{v,\frac{m-1}{\epsilon}}(\frac{3}{2}) & \text{if } u = 1 \\ 1 - F_{v,\frac{m-1}{\epsilon}}(m - \frac{1}{2}) & \text{if } u = m \\ F_{v,\frac{m-1}{\epsilon}}(u + \frac{1}{2}) - F_{v,\frac{m-1}{\epsilon}}(u - \frac{1}{2}) & \text{otherwise} \end{cases} \tag{2}$$

where $F_{\mu,b}(x) = \frac{1}{2} + \frac{1}{2} sgn(x-\mu)(1 - e^{(-\frac{|x-\mu|}{b})})$ denotes the cumulative distribution function of Laplace distribution with mean μ and the variance $2b^2$.

From Reported Type to True Type. Consider an arbitrary voter V_i. Again, let t_i be its true type, and r_i be its reported type. If we observe that V_i reports v as its type, what is the probability that its true type is u? More precisely, we consider $\Pr[t_i = u|r_i = v]$. As each voter independently and locally runs the LDP mechanism, $\Pr[t_i = u|r_i = v]$ is the same for all voters. Hence, we denote by $q_{uv} = \Pr[t_i = u|r_i = v]$, and let $Q = (q_{uv})_{m\times m}$.

The probabilities p_{uv} and q_{uv} are connected through Bayesian formulation. Let λ_j be the fraction of voters whose true type is j (i.e., there are $n\lambda_j$ voters whose true type is j). We assume λ_j's are fixed positive values and denote $\mathbf{p}_i = (p_{i1}, p_{i2}, \ldots, p_{im})$ as the i-th row of the design matrix P. Then, we have

$$q_{uv} = \frac{\Pr[r_i = v, t_i = u]}{\sum_{u'} \Pr[r_i = v|t_i = u'] \Pr[t_i = u']} = \frac{p_{vu}\lambda_u}{\boldsymbol{\lambda} \cdot \mathbf{p}_v}. \tag{3}$$

It is easy to see that p_{uv}'s and q_{uv}'s are independent of the number of voters n.

3 Formal Definition of PoLDP

The goal of this paper is to quantitatively analyze the effect of LDP mechanisms when applied to voting systems. Towards this, we consider two scenarios, the classical scenario without LDP and the new scenario with LDP, and compare the manipulation costs of the attacker under these two scenarios. Below we give details.

We denote $\mathcal{S}_{m,\boldsymbol{\lambda}}(n)$ as the voting system which contains n voters namely $\{V_1, \ldots, V_n\}$, and m candidates namely $\{C_1, \ldots, C_m\}$. The parameter $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)$ which denotes there are $n\lambda_i$ voters whose true type is i. The deletion cost for each voter is unit. Hence, there is no need to distinguish two voters who have the same true type.

There is an attacker who tries to make a designated candidate win via voter deletion. Without loss of generality, we assume that the designated candidate is C_1.

In the classical scenario without LDP mechanisms, the attacker knows the true type of voters[3] (e.g., the parameter $\boldsymbol{\lambda}$ of the voting system), and can thus strategically delete voters (e.g., decide the number of voters that need to be deleted for each kind of true type) to make the designated candidate C_1 be one of the co-winners. Hence, for the voting system $\mathcal{S} = \mathcal{S}_{m,\boldsymbol{\lambda}}(n)$, we define the minimal number of voters the attacker should delete as the *manipulation cost* of the attacker, and define it as $f(\mathcal{S})$.

Consider the scenario where an LDP mechanism $\mathcal{R}$ is adopted. We also assume the attacker knows the parameter $\boldsymbol{\lambda}$ of the voting system[4]. But, the attacker needs to delete voters according to reported types instead of true types. Adopting the LDP mechanism introduces two kinds of uncertainties

[3] This is a common assumption in computational social choice, see, e.g., Brandt et al. [2016].

[4] For the rationality of this assumption, please refer to Section "Discussion".

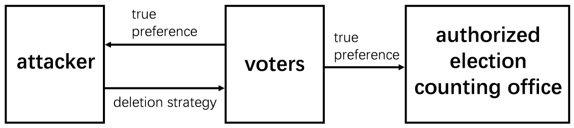

Figure 1: Illustration of Manipulation without LDP

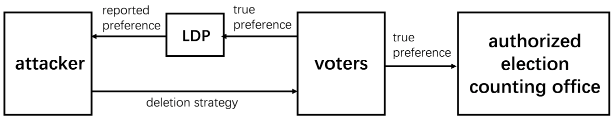

Figure 2: Illustration of Manipulation with LDP

when considering the manipulation problem, as we discuss below.

• *Realizations:* Recall that there are $n\lambda_i$ voters whose true type is i. We know that it is fixed for the voting system $\mathcal{S}_{m,\boldsymbol{\lambda}}(n)$. However, the number of voters whose reported type is i, may be different[5]. Let $n\tau_i$ be the number of voters whose reported type is i where $1 \leq i \leq m$, and let $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_m)$. Then, we know that

$$\boldsymbol{\tau} \in \mathcal{P} := \{\mathbf{z} = (z_1, \cdots, z_m) \in [0,1]^m : \sum_{i=1}^{m} z_i = 1\}.$$

For any $\mathbf{z} \in \mathcal{P}$, $\boldsymbol{\tau} = \mathbf{z}$ is called a *realization*, and will occur with the probability $\Pr[\boldsymbol{\tau} = \mathbf{z}]$ that depends on the LDP mechanism $\mathcal{R}$.

• *Winning Probability:* The attacker observes a realization $\boldsymbol{\tau}$ and decides the number of voters that need to be deleted for each reported type. More precisely, a feasible solution to the attacker under a fixed realization $\boldsymbol{\tau}$ can be represented as a vector $\mathbf{x} = (x_1, x_2, \cdots, x_m)$ where $x_i \leq n\tau_i$ is the number of deleted voters whose reported type is i. As voters of the same reported type are indistinguishable from the attacker, we assume the attacker will randomly delete x_i out of $n\tau_i$ voters of reported type i. Among these x_i voters, the number of voters whose true type is i' is a stochastic value for each i'. Consequently, the attacker may or may not succeed in making C_1 win. In other words, any feasible solution $\mathbf{x}$ to the attacker is associated with a *winning probability* which indicates the probability that C_1 becomes a co-winner if the attacker deletes arbitrary x_i voters whose reported type is i, given the observation $\boldsymbol{\tau}$.

Above all, for the voting system $\mathcal{S}_{m,\boldsymbol{\lambda}}(n)$, we define the manipulation cost of the attacker under the LDP mechanism as follows:

Definition 2. *The manipulation cost of a voting system $\mathcal{S} = \mathcal{S}_{m,\boldsymbol{\lambda}}(n)$ under an LDP mechanism $\mathcal{R}$, given a winning probability ξ and a realization $\boldsymbol{\tau}$, is the minimal number of voters that need to be deleted to make the designated candidate one co-winner with a probability at least ξ and is denoted by $f(\mathcal{S}, \mathcal{R}, \xi : \boldsymbol{\tau})$.*

As the mechanism $\mathcal{R}$ defines a distribution of $\boldsymbol{\tau}$ over $\mathcal{P}$, we can define the expectation of $f(\mathcal{S}, \mathcal{R}, \xi : \boldsymbol{\tau})$ over all the realizations as below.

Definition 3. *The expected manipulation cost of a voting system $\mathcal{S} = \mathcal{S}_{m,\boldsymbol{\lambda}}(n)$ under an LDP mechanism $\mathcal{R}$, given a winning probability ξ, is the expected minimal number of voters that need to be deleted to make the designated candidate one co-winner with a probability at least ξ over all the realizations, and is denoted by $f(\mathcal{S}, \mathcal{R}, \xi) = \mathbb{E}_{\boldsymbol{\tau}}[f(\mathcal{S}, \mathcal{R}, \xi : \boldsymbol{\tau})]$.*

Compare the manipulation cost of the attacker in the above two scenarios (with or without LDP), the introduction of LDP may cause the cost to increase or decrease, and we measure such an increase or decrease through PoLDP, interpreted as the power of LDP. More precisely,

Definition 4. *The PoLDP for a voting system $\mathcal{S} = \mathcal{S}_{m,\boldsymbol{\lambda}}(n)$ under an LDP mechanism $\mathcal{R}$ is defined as*

$$\textit{PoLDP}(\mathcal{S}, \mathcal{R}, \xi) = f(\mathcal{S}, \mathcal{R}, \xi)/f(\mathcal{S}).$$

We are interested in the value of PoLDP when the number of voters in the voting system is sufficiently large. Hence, we define the APoLDP (Asymptotic PoLDP).

Definition 5. *The APoLDP for a voting system $\mathcal{S} = \mathcal{S}_{m,\boldsymbol{\lambda}}(n)$ under an LDP mechanism $\mathcal{R}$ is defined as*

$$\textit{APoLDP}(\mathcal{S}, \mathcal{R}, \xi) = \lim_{n\to\infty} f(\mathcal{S}, \mathcal{R}, \xi)/f(\mathcal{S}).$$

4 Characterizing APoLDP

The goal of this section is to provide an efficient method to calculate APoLDP and study the relationship between APoLDP and the parameter ϵ of LDP mechanisms.

We prove that under certain conditions, the value of APoLDP is robust to the winning probability ξ, and can be calculated via a linear program. Specifically, we have the following theorem:

Theorem 1. *If the voting system $\mathcal{S} = \mathcal{S}_{m,\boldsymbol{\lambda}}(n)$ under an LDP mechanism $\mathcal{R}$ satisfies that for any $1 \leq j \leq m$ either $q_{ij} - q_{1j} > 0$ for all $2 \leq i \leq m$, or $q_{ij} - q_{1j} < 0$ for all $2 \leq i \leq m$, then for any $\xi \in (0, 1)$ it holds that*

$$\textit{APoLDP}(\mathcal{S}, \mathcal{R}, \xi) = \frac{OPT_{LP(\hat{\boldsymbol{\tau}})}}{\sum_{j=2}^{m} \max\{0, n\lambda_j - n\lambda_1\}},$$

where $\hat{\tau}_i = \mathbb{E}[\tau_i] = \sum_{j=1}^{m} \lambda_j p_{ij}$.

See the full version of this paper for the complete proof of Theorem 1. Here, we only give the roadmap of the proof.

- First, we use two integer linear programs $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ and $\underline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ to give the upper and lower bound of $f(\mathcal{S}, \mathcal{R}, \xi : \boldsymbol{\tau})$ respectively. We establish $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ as

[5]To understand the effect of an LDP mechanism to a fixed voting system on average, we consider the independent application of the same LDP mechanism multiple times, and each time we may observe a different $\boldsymbol{\tau}$.

follows:

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{m} x_i \\
s.t. \quad & \hat{\Gamma}_i = n\lambda_i - \sum_{j=1}^{m} q_{ij} x_j && \forall\, 1 \le i \le m \quad \text{(4a)} \\
& \hat{\Gamma}_1 \ge \hat{\Gamma}_i + \delta \sum_{j=1}^{m} (q_{1j} + q_{ij}) x_j && \forall\, 2 \le i \le m \quad \text{(4b)} \\
& x_i \le n\tau_i && \forall\, 1 \le i \le m \quad \text{(4c)} \\
& x_i \in \mathbb{N}
\end{aligned}
$$

Here, the decision variable x_j denotes the number of voters with reported type j that need to be deleted. Recall that q_{jk} is the probability that the true type of a voter is j conditioned on its reported type is k. Consequently, Eq (4a) implies that $\hat{\Gamma}_j$ is the expected number of voters of true type j that are left after voter deletion, Eq (4b) ensures that the designated candidate C_1 receives the highest votes in expectation, with a margin of $\delta \sum_{j=1}^{m}(q_{1j} + q_{ij})x_j$. The margin will guarantee that C_1 can win with a sufficiently large probability despite that we only consider the expected scores of candidates. More precisely, we have the following lemma.

Lemma 1. *Let $\mathbf{x}^* = (x_1^*, \ldots, x_m^*)$ denote the optimal solution to $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$. If the attacker deletes arbitrary x_i^* voters whose reported type is i, then the winning probability of the designated candidate is at least $1 - me^{-\frac{c\delta^2}{3}n}$, where $c = q_{\min}(\lambda_{\max} - \lambda_1)$,$q_{\min} = \min_{i,j} q_{ij}$, $\lambda_{\max} = \max_j \lambda_j$.*

Replacing constraint Eq (4b) of $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ with $\hat{\Gamma}_1 \ge \hat{\Gamma}_i - \delta \sum_{j=1}^{m}(q_{1j} + q_{ij})x_j$, we obtain another integer linear program and denote it as $\underline{\text{ILP}}(\boldsymbol{\tau}, \delta)$. Then, we can have the following lemma.

Lemma 2. *Let $\mathbf{x}^* = (x_1^*, \ldots, x_m^*)$ denote the optimal solution to $\underline{\text{ILP}}(\boldsymbol{\tau}, \delta)$. If the attacker deletes y_i voters whose reported type is i such that $\sum_j y_j < \sum_j x_j^* := OPT_{\underline{ILP}(\boldsymbol{\tau},\delta)}$, then the winning probability of the designated candidate is less than $2e^{-\frac{c\delta^2}{3}n}$ where $c = q_{\min}(\lambda_{\max} - \lambda_1)$, $q_{\min} = \min_{i,j} q_{ij}$, $\lambda_{\max} = \max_j \lambda_j$.*

- Second, we show that if certain condition holds, then the value of $f(\mathcal{S}, \mathcal{R}, \xi : \boldsymbol{\tau})$ is robust to the winning probability ξ. Specifically, we prove that for any $\epsilon \in (0, 1)$, the value of $f(\mathcal{S}, \mathcal{R}, 1 - \epsilon : \boldsymbol{\tau})$ can be calculated via linear program LP($\boldsymbol{\tau}$) when the number of voters is sufficiently large.

Replacing constraint Eq (4b) of $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ with $\hat{\Gamma}_1 \ge \hat{\Gamma}_i$, we introduce an "intermediate" integer linear program ILP($\boldsymbol{\tau}$) between $\underline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ and $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$. Denote LP($\boldsymbol{\tau}$) as the linear relaxation of ILP($\boldsymbol{\tau}$). We show that under certain conditions, if the number of voters is sufficiently large, then the upper bound and lower bound given by $\underline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ and $\overline{\text{ILP}}(\boldsymbol{\tau}, \delta)$ approaches the same value which is exactly $\text{OPT}_{\text{LP}(\boldsymbol{\tau})}$.

- Finally, we show a efficient way to calculate $f(\mathcal{S}, \mathcal{R}, \xi)$. Recall the definition, we need to compute the optimal objective value of linear program LP($\boldsymbol{\tau}$) for each fixed realization $\boldsymbol{\tau}$, and then taking the expectation over $\boldsymbol{\tau}$. We prove that if certain condition holds, we can first compute the expectation of $\boldsymbol{\tau}$ (denoted as $\hat{\boldsymbol{\tau}} = \mathbb{E}[\boldsymbol{\tau}]$), and then compute the optimal objective value of integer program LP($\hat{\boldsymbol{\tau}}$).

With Theorem 1, we are able to study the relationship between APoLDP and the privacy parameter ϵ for two classes of LDP mechanisms, randomized response and Laplace.

For voting systems with only two candidates, we prove that for any $\xi \in (0, 1)$ APoLDP has the following closed-form expression which is derived by solving LP($\hat{\boldsymbol{\tau}}$).

$$
\text{APoLDP}(\mathcal{S}_2, \epsilon\text{-ran}, \xi) = \begin{cases} 1/\phi & \text{if } \epsilon \le \ln \frac{1+\phi}{1-\phi} \\ \frac{e^\epsilon + 1 + (e^\epsilon - 1)\phi}{e^\epsilon - 1 + (e^\epsilon + 1)\phi} & \text{otherwise} \end{cases}
$$

$$
\text{APoLDP}(\mathcal{S}_2, \epsilon\text{-lap}, \xi) = \begin{cases} 1/\phi & \text{if } \epsilon \le 2\ln \frac{1}{1-\phi} \\ \frac{e^{\epsilon/2} + (e^{\epsilon/2} - 1)\phi}{e^{\epsilon/2} - 1 + e^{\epsilon/2}\phi} & \text{otherwise} \end{cases}
$$

Here $\phi = \lambda_2 - \lambda_1$, "$\epsilon$-ran" stands for randomized response mechanism which satisfies ϵ-LDP and "ϵ-lap" stands for Laplace mechanism which satisfies ϵ-LDP.

For voting systems with multiple candidates, it is difficult to give the closed-form expression of APoLDP. Instead, we can have the following result.

Theorem 2. *If the voting system $\mathcal{S}$ under an randomized response mechanism $\mathcal{R}$ satisfies ϵ-LDP and $\lambda_i > \lambda_1$ for all $2 \le i \le m$, then for any $\xi \in (0, 1)$, APoLDP achieves the maximal value $\frac{1}{1 - \sum_{i=2}^{m}(\lambda_i - \lambda_1)}$ when $\frac{\theta\lambda_1}{\lambda_{\max} - \lambda_1} \le \frac{1-\theta}{m}$ where $\lambda_{\max} = \max_i \lambda_i$, i.e., when $\epsilon \le \ln \frac{\lambda_{\max}}{\lambda_1}$.*

The proof of Theorem 2 relies on showing two statements on LP($\hat{\boldsymbol{\tau}}$): (i) if $\epsilon \le \ln \frac{\lambda_{\max}}{\lambda_1}$, then LP($\hat{\boldsymbol{\tau}}$) only admits one feasible solution where $x_k = n\mathbb{E}[\tau_k]$, meaning that deleting all the voters is the only solution to make the designated candidate C_1 win; (ii) if $\epsilon > \ln \frac{\lambda_{\max}}{\lambda_1}$, then there exists a feasible solution to LP($\hat{\boldsymbol{\tau}}$) whose objective value is strictly smaller than n. Unfortunately, the proof heavily utilizes the special structure of the design matrix for randomized response, and cannot be carried over to Laplace.

5 Numeric Experiments

In this section, we demonstrate PoLDP through numeric experiments[6]. In our experiments, the number of voters is set to be $n = 10^8$. The number of candidates is set to be $m = 2; 5$. In this experiment, we generate the true type of each voter using two methods.

- The first method guarantees the difference in score between the designated candidate and the winner equals $m\phi$ where the true type of each voter is randomly generated from $[1, m]$.

[6]All experiments are performed on a computer with an Intel i7-6700 processor and 32GB memory. The code is available at https://github.com/polyapp/poldp

- The second method randomly generates the true type of each voter according to the real-world data—Sushi Data [Kamishima, 2003] which is a commonly used data set for generating preferences [Azari *et al.*, 2012].

For ease of notation, we denote by $\mathcal{T}_m^{\phi}$ the voting system generated by the first method, and $\mathcal{T}_m$ the voting system generated by the second method. For each kind of voting system, we generate 2000 instances.

The parameter is set to be $\xi = 0.999$, $\delta = 0.001$, $\epsilon = 0.001$. We observe that randomized response and Laplace mechanism on our randomly generated voting systems hold that:

$$\text{avg}[\frac{|\text{OPT}_{\overline{\text{LP}}((1-\epsilon)\hat{\boldsymbol{\tau}},\delta)} - \text{OPT}_{\text{LP}(\hat{\boldsymbol{\tau}})}|}{\text{OPT}_{\text{LP}(\hat{\boldsymbol{\tau}})}}] \leq 5\%,$$

and

$$\text{avg}[\frac{|\text{OPT}_{\underline{\text{LP}}((1+\epsilon)\hat{\boldsymbol{\tau}},\delta)} - \text{OPT}_{\text{LP}(\hat{\boldsymbol{\tau}})}|}{\text{OPT}_{\text{LP}(\hat{\boldsymbol{\tau}})}}] \leq 5\%.$$

This result suggests if we compare two solutions, one guarantees that the designated candidate wins by at least 99.9%, and the other guarantees that the designated candidate wins by at most 0.1%, then the two manipulation costs differ by at most 10%. In other words, on the voting systems we created PoLDP$(\mathcal{S}, \mathcal{R}, 0.1\%)$ and PoLDP$(\mathcal{S}, \mathcal{R}, 99.9\%)$ differ by at most 10% in average. Based on this, we simply calculate the manipulation cost with LDP mechanisms using $\text{OPT}_{\text{LP}(\hat{\boldsymbol{\tau}})}$.

For voting systems with multiple candidates, we perform experiments on voting systems $\mathcal{T}_5$ and $\mathcal{T}_5^{\phi}$ for $\phi = 0.1; 0.2$. For $\epsilon = 0, 0.01, \ldots, 0.49, 5$, we compare the efficiency of randomized response and Laplace mechanism from two aspects: the average and the standard deviation of PoLDP (solid lines represent the average, and dotted shadows represent the standard deviation); the percentage of instances which has PoLDP larger than 99% of its maximal value (in short, the maximal percentage). We summarize the comparison in Figure 3-4.

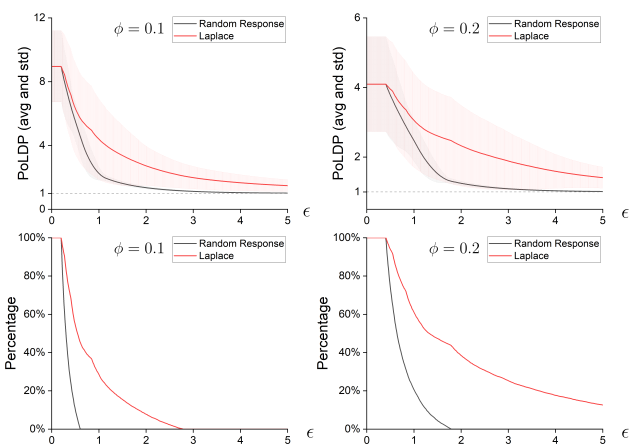

Figure 3: PoLDP and the maximal percentage of randomized response and Laplace mechanism on voting systems $\mathcal{T}_5^{\phi}$.

We observe that the Laplace mechanism outperforms randomized response by always giving a larger PoLDP on all the voting systems we test. Moreover, the randomized mechanism is more sensitive to the selection of ϵ on voting systems $\mathcal{T}_5^{\phi}$. We can see from Figure 3, the average of PoLDP for randomized response decreases quicker than Laplace in the region $\epsilon \in [0, 2]$. Interestingly, we also observe that the maximal percentage coincides with the value of PoLDP (e.g., the maximal percentage equals 100% where PoLDP stays at its maximal value) on voting systems $\mathcal{T}_5^{\phi}$.

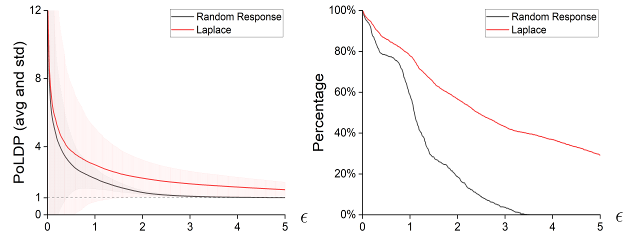

Figure 4: PoLDP and the maximal percentage of randomized response and Laplace mechanism on voting systems $\mathcal{T}_5$.

More experimental results, including the analysis, are given in the full version of this paper.

6 Discussion

In real-world applications, although the exact value of $\boldsymbol{\lambda}$ is unknown, we can use a two-round implementation of LDP in voting to estimate $\boldsymbol{\lambda}$: In the first round, a random subset $\mathcal{A}$ of voters are selected to report their types via the LDP mechanism; In the second round, all voters except those in $\mathcal{A}$, report their types via the LDP mechanism. We use the reported types given by voters in $\mathcal{A}$, to get an estimation of $\boldsymbol{\lambda}$. It is generally sufficient to obtain a good statistical estimation through a reasonably small sample [Kenny, 1986]. In particular, $|\mathcal{A}|$ can be $o(n)$ for sufficiently large n, and therefore it will only marginally affect the value of PoLDP.

7 Conclusion

We provide the first systematic study on how LDP mechanisms may improve the resilience of a voting system under electoral control by deleting voters. The metric PoLDP introduced in this paper gives general guidance towards the choice of the privacy parameter ϵ in LDP mechanisms.

Acknowledgments

Liangde Tao is partly supported by "New Generation of AI 2030" Major Project (2018AAA0100902). Lin Chen is partly supported by NSF Grant 2004096.

References

[Azari *et al.*, 2012] Hossein Azari, David Parks, and Lirong Xia. Random utility theory for social choice. *Advances in Neural Information Processing Systems*, 25:126–134, 2012.

[Bartholdi III *et al.*, 1992] John J Bartholdi III, Craig A Tovey, and Michael A Trick. How hard is it to control an election? *Mathematical and Computer Modelling*, 16(8-9):27–40, 1992.

[Bebensee, 2019] Björn Bebensee. Local differential privacy: a tutorial. *arXiv preprint arXiv:1907.11908*, 2019.

[Brandt *et al.*, 2016] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.

[Bun *et al.*, 2019] Mark Bun, Jelani Nelson, and Uri Stemmer. Heavy hitters and the structure of local privacy. *ACM Transactions on Algorithms*, 15(4):1–40, 2019.

[Chen *et al.*, 2018] Lin Chen, Lei Xu, Shouhuai Xu, Zhimin Gao, Nolan Shah, Yang Lu, and Weidong Shi. Protecting election from bribery: New approach and computational complexity characterization. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*, pages 1894–1896, 2018.

[Cheu *et al.*, 2021] Albert Cheu, Adam Smith, and Jonathan Ullman. Manipulation attacks in local differential privacy. In *Proceedings of the 42nd IEEE Symposium on Security and Privacy*, pages 883–900, 2021.

[Conitzer *et al.*, 2011] Vincent Conitzer, Toby Walsh, and Lirong Xia. Dominating manipulations in voting with partial information. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, 2011.

[Dey *et al.*, 2018] Palash Dey, Neeldhara Misra, and Yadati Narahari. Complexity of manipulation with partial information in voting. *Theoretical Computer Science*, 726:78–99, 2018.

[Erlingsson *et al.*, 2014] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 21st ACM SIGSAC Conference on Computer and Communications Security*, pages 1054–1067, 2014.

[Faliszewski *et al.*, 2009] Piotr Faliszewski, Edith Hemaspaandra, and Lane A Hemaspaandra. How hard is bribery in elections? *Journal of artificial Intelligence Research*, 35:485–532, 2009.

[Hemaspaandra *et al.*, 2007] Edith Hemaspaandra, Lane A Hemaspaandra, and Jörg Rothe. Anyone but him: The complexity of precluding an alternative. *Artificial Intelligence*, 171(5-6):255–285, 2007.

[Hemaspaandra *et al.*, 2017] Edith Hemaspaandra, Lane A Hemaspaandra, and Jörg Rothe. The complexity of controlling candidate-sequential elections. *Theoretical Computer Science*, 678:14–21, 2017.

[Kamishima, 2003] Toshihiro Kamishima. Nantonac collaborative filtering: recommendation based on order responses. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data mining*, pages 583–588, 2003.

[Kasiviswanathan *et al.*, 2011] Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

[Kenny, 1986] David A Kenny. *Statistics for the social and behavioral sciences*. Little, Brown, 1986.

[Koutsoupias and Papadimitriou, 1999] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science*, pages 404–413, 1999.

[Magiera and Faliszewski, 2017] Krzysztof Magiera and Piotr Faliszewski. How hard is control in single-crossing elections? *Autonomous Agents and Multi-Agent Systems*, 31(3):606–627, 2017.

[Maushagen and Rothe, 2016] Cynthia Maushagen and Jörg Rothe. Complexity of control by partitioning veto and maximin elections and of control by adding candidates to plurality elections. In *Proceedings of the 22nd European Conference on Artificial Intelligence*, pages 277–285, 2016.

[Qin *et al.*, 2016] Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. Heavy hitter estimation over set-valued data with local differential privacy. In *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security*, pages 192–203, 2016.

[Rey and Rothe, 2016] Anja Rey and Jörg Rothe. Structural control in weighted voting games. In *Proceedings of the 15th International Conference on Autonomous Agents & Multiagent Systems*, pages 1371–1372, 2016.

[Rosenfeld *et al.*, 2016] Bryn Rosenfeld, Kosuke Imai, and Jacob N Shapiro. An empirical validation study of popular survey methodologies for sensitive questions. *American Journal of Political Science*, 60(3):783–802, 2016.

[Wang and Blocki, 2017] Tianhao Wang and Jeremiah Blocki. Locally differentially private protocols for frequency estimation. In *Proceedings of the 26th USENIX Security Symposium*, pages 729–745, 2017.

[Wang *et al.*, 2016] Yue Wang, Xintao Wu, and Donghui Hu. Using randomized response for differential privacy preserving data collection. In *EDBT/ICDT Workshops*, volume 1558, pages 0090–6778, 2016.

[Yin *et al.*, 2018] Yue Yin, Yevgeniy Vorobeychik, Bo An, and Noam Hazon. Optimal defense against election control by deleting voter groups. *Artificial Intelligence*, 259:32–51, 2018.