CIS 700 Differential Privacy in Game Theory and Mechanism Design February 14, 2014 ♡

Lecture 5

Lecturer: Aaron Roth *Scribe: Aaron Roth*

Private Equilibrium Computation Yields Truthful Mediators (Part 1)

1 Introduction

When we were being read fairy tails growing up, we first learned about games of "Complete information". So the story goes, in such games, all n players know one another's types, and use this information to somehow select and play one of the (possibly many) Nash equilibria of the game.

To recap, a game $\mathcal{G}$ (here we will use *costs* rather than *utilities*, but it doesn't matter) is defined by a set of n players, a set of actions A, a set of *types* $\mathcal{T}$, and a cost function $c : \mathcal{T} \times \mathcal{A}^n \to [0, 1]$. Given a choice of actions $a = (a_1, \ldots, a_n)$ for each player, we will sometimes write $c_i(a) = c(t_i, a)$ to denote agent i's cost.

A game is one of *complete information* if the types $t_1, \ldots, t_n$ are common knowledge. We can define a pure strategy Nash equilibrium of a game of complete information:

Definition 1 *Fixing a game of complete information defined by types $(t_1, \ldots, t_n)$, a set of actions $a = (a_1, \ldots, a_n)$ is an (η-approximate) pure strategy Nash equilibrium if for every player i and for every action a_i':*

$$c_i(a) \leq c_i(a_i', a_{-i}) + \eta$$

Nash equilibria are nice. They are of course stable. Additionally, in various games (e.g. congestion games with linear cost functions), the Nash equilibria also have nice welfare properties. For example, it is known that the worst pure strategy Nash equilibrium in linear congestion games has social welfare that is only a factor of at most 2.5 worse than the optimal social welfare [CK05, AAE05] (i.e. the *price of anarchy* is 2.5).

However, the idea that people will always play a pure strategy Nash equilibrium of a complete information game has a number of problems:

1. Pure strategy Nash equilibria are not guaranteed to exist. When they do exist, they are not guaranteed to be unique, and different players may prefer different equilibria. Thus, coordinating on an equilibrium (the *equilibrium selection problem*) is highly non-trivial.

2. In an n player game, it is often unreasonable to assume that players know each others types! Without knowing the types in the game, the Nash equilibria (of the complete information) game are not even defined.

The usual solution is to move to study games of *incomplete information*. There are two things we could do. First, we could move to a Bayesian setting, and assume everyone knows a common prior over types, as well as a function that all other players use to map realized type to action. We will discuss this later, but needless to say, this still assumes a great deal about the knowledge and rationality of the players. We could alternately hope that there exists an *ex-post* equilibrium of the game: a set of actions that form an equilibrium, no matter what player types turn out to be.

Definition 2 *Fix a collection of functions $s_i : T \to A$ mapping types to actions. $s = (s_1, \ldots, s_n)$ defines an (η-approximate) ex-post equilibrium if for every vector of realized player types $t = (t_1, \ldots, t_n)$ and for every player i and for every action a_i':*

$$c_i(s(t)) \leq c_i(a_i', s_{-i}(t_{-i})) + \eta$$

This is obviously a quite robust notion, requiring players to know nothing about each other's types. Moreover, it results in a Nash equilibrium of the complete information game, even though players don't know each others types! The difficulty is that in general, ex-post equilibria will exist only in extremely rare circumstances.

In this lecture (and subsequent ones), we will lay out and implement the following agenda: We wish to take a game, and augment it with an extremely weak *mediator*, which essentially has only the power to make suggestions. Agents will be able to:

1. Ignore the mediator and play an action directly in the original game, or
2. Report a type to the mediator, and receive a suggested action. They can then play any action in the original game, which need not be the suggested action.

If agents choose the second option, they are not obligated to report their true type – and the mediator cannot enforce the action, it can only make suggestions. What we want is to design a mediator such that the strategy that has everybody:

1. Report their *true* type to the mediator, and then
2. Follow the suggested action

is an asymptotic ex-post equilibrium in the (modified) game which is augmented by the mediator. Moreover, we want that the play that results should be a pure strategy Nash equilibrium of the original game. This solves the equilibrium selection problem, and results in players playing Nash equilibria of the complete information game (which as we mentioned often have nice welfare properties) even in more realistic settings of incomplete information.

2 Privately Computing Nash Equilibria Yields a Good Mediator

First, let us define formally what we mean by adding a mediator to a game.

Definition 3 *A* mediator *is given by an algorithm* $M : (\mathcal{T} \cup \{\perp\})^n \to (A \cup \{\perp\})^n$ *mapping reported types (or* $\perp$*, which denotes declining to report any type) to suggested actions (or* $\perp$*, which denotes not suggesting any action).*

Given a game $\mathcal{G}$ and a mediator M, we can define the game $\mathcal{G}$ augmented with mediator M, denoted as $\mathcal{G}_M$, as follows. The new game has action space:

$$A' = \{(t', f) : t' \in \mathcal{T} \cup \{\perp\}, f : (A \cup \{\perp\}) \to A\}$$

i.e. each player chooses both a type t_i to report (possibly $\perp$ if not participating), and a function f specifying how to map the advice received by the mediator to an action a to play in the underlying game $\mathcal{G}$ (Note that picking f equal to the identity function corresponds to following the advice of the mediator, but the player can pick anything else, including a constant function which corresponds to completely disregarding the advice. Note also that if a player reports $t' = \perp$ to the mediator, then it will return the advice $\perp$, and a player will then play the action $f(\perp) \in A$, which encodes the strategy he would have used in the original game). The new game has the following cost function:

$$c'(t_i, ((t'_1, f_1), \ldots, (t'_n, f_n))) = \mathrm{E}_{a \sim M(t')}[c(t_i, f(a))]$$

i.e. it is the expected cost in the original game, given the actions players play, as a function of the suggestions made by the mediator.

Let $\mathrm{id} : A \to A$ denote the identity function on actions. For a player with type t_i, write $g_i = (t_i, \mathrm{id})$ denote the strategy in $\mathcal{G}_M$ that corresponds to "good behavior" – i.e. truthfully reporting his type, and then following the suggested action of the mediator. We want to design a mediator M that makes $g_1, \ldots, g_n$ an asymptotic ex-post equilibrium for a large class of games $\mathcal{G}_M$.

The idea will be to have M *privately* compute a Nash equilibrium of the complete information game defined by the reported types t'. This should at first seem problematic: if M satisfies differential privacy, then in particular, the distribution on suggested actions given to a player i must be almost independent of his reported type! But such an action can only be a best response to his opponents' play if his cost function is (almost) independent of his type, which does not lead to a very interesting class of games.

To remedy this situation, we will rely on a (slight) relaxation of differential privacy known as *joint differential privacy*. The idea will be to allow the suggested action given to player i to depend in a sensitive way on his own reported type, while still maintaining that the joint distribution on actions given to players $j \neq i$ remain insensitive in the reported type of agent i.

Definition 4 (Joint Differential Privacy [KPRU14]) *Let $M : \mathcal{A}^n \to \mathcal{B}^n$. M satisfies ϵ-joint-differential privacy if for every $a \in \mathcal{A}^n$, for every i, for every $a'_i \in \mathcal{A}$, and for every $S \subseteq \mathcal{B}^{n-1}$:*

$$\Pr[M(a)_{-i} \in S] \leq \exp(\epsilon) \Pr[M(a'_i, a_{-i})_{-i} \in S]$$

We can now state the motivating theorem of the next couple of lectures: informally, that mediators $M(t')$ which are jointly differentially private and compute approximate Nash equilibria of a complete information game $\mathcal{G}$ defined by types t' make "good behavior" $g = (g_1, \ldots, g_n)$ an approximate ex-post equilibrium of $\mathcal{G}_M$.

Theorem 5 ([RR13], based on [KPRU14]) *Let $\mathcal{G}$ be any game with costs bounded in $[0, 1]$, and let M be a mechanism satisfying ϵ-joint differential privacy such that on any set of reported types t', M outputs an η-approximate pure strategy Nash equilibrium of the complete information game induced by t'. Then good behavior $g = (g_1, \ldots, g_n)$ is an η'-approximate ex-post equilibrium of $\mathcal{G}_M$ for:*

$$\eta' = 2\epsilon + \eta.$$

Proof Fix any vector of player types $t \in \mathcal{T}^n$: we will show that g is a pure strategy Nash equilibrium of $\mathcal{G}_M$ with realized types t. Consider any potential deviation $b_i = (t'_i, f_i)$: our goal is to show that no agent can subsantially decrease their cost by unilaterally deviating to b_i.

For a vector of actions $a_{-i} \in A^{n-1}$, we define the best response of player i:

$$\mathrm{BR}_i(a_{-i}) = \arg\min_{a'_i \in A} (c_i(a'_i, a_{-i}))$$

Now, we compute:

$$\begin{aligned} c'(t_i, g) &= \mathrm{E}_{a \sim M(t)}[c_i(a)] \\ &\leq \mathrm{E}_{a \sim M(t)}[c_i(\mathrm{BR}_i(a_{-i}), a_{-i})] + \eta \end{aligned}$$

where the inequality follows from the fact that M computes an η-approximate Nash equilibrium of the game defined by the reported types. Next we invoke the privacy condition:

$$\begin{aligned} c'(t_i, g) &\leq \exp(\epsilon)\mathrm{E}_{a \sim M(t'_i, t_{-i})}[c_i(\mathrm{BR}_i(a_{-i}), a_{-i})] + \eta \\ &\leq \mathrm{E}_{a \sim M(t'_i, t_{-i})}[c_i(\mathrm{BR}_i(a_{-i}), a_{-i})] + 2\epsilon + \eta \\ &\leq \mathrm{E}_{a \sim M(t'_i, t_{-i})}[c_i(f_i(a_i), a_{-i})] + 2\epsilon + \eta \\ &= c'(t_i, (b_i, g_{-i})) + 2\epsilon + \eta \end{aligned}$$

Here the first inequality follows from joint differential privacy, the second from the fact that (for $\epsilon \leq 1$) $\exp(\epsilon) \leq 1 + 2\epsilon$, and the third from the definition of best response. ■

Now that we have this theorem, it remains to actually design a jointly-differentially private algorithm that computes a Nash equilibrium in a large class of games. For this, we will need to learn a bit more differential privacy...

3 Some more differential privacy tools

So far, we have been skating by knowing nothing except the exponential mechanism. We will need to develop some more sophisticated tools now, however. Ok, not that sophisticated – all we will need is the ability to count (privately). It turns out the ability to count is enough to do all sorts of sophisticated things.

3.1 Numeric Valued Queries

First, we will give one that is even simpler – the Laplace mechanism! This is really just a special case of the exponential mechanism.

Definition 6 *The ℓ_1 norm of a vector $v \in \mathbb{R}^d$ is:*

$$||v||_1 = \sum_{i=1}^{d} |v_i|.$$

A function $f : \mathcal{T}^n \to \mathbb{R}^d$ has ℓ_1 sensitivity k if for every $t \in \mathcal{T}^n$, $t'_i \in \mathcal{T}$:

$$||f(t) - f(t'_i, t_{-i})||_1 \leq k$$

Let $\mathrm{GS}(f)$ denote the minimum k such that f has sensitivity k.

Definition 7 *The Laplace Distibution with scale parameter b, denoted $\mathrm{Lap}(b)$ is the distribution with pdf:*

$$p(x) = \frac{1}{2b} \exp\left(-\frac{|x|}{b}\right)$$

Some useful facts about $Z \sim \mathrm{Lap}(b)$:

$$\mathrm{E}[|Z|] = b \qquad \Pr[|Z| \geq t \cdot b] = \exp(-t)$$

Definition 8 ([DMNS06]) *The Laplace mechanism $M_L(t; f)$ computes $f(t) + Z$ where $Z \in \mathbb{R}^d$ such that each $Z_i \sim \mathrm{Lap}\left(\frac{\mathrm{GS}(f)}{\epsilon}\right)$.*

Theorem 9 ([DMNS06]) *For every f, $M_L(t; f)$ is ϵ-differentially private.*

Proof Consider any output of the Laplace mechanism $x \in \mathbb{R}^d$. Since the Laplace mechanism produces a continuous random variable, we will understand $\Pr(M_L(D; f) = x)$ to refer to the probability density function for the Laplace mechanism. We can compute:

$$\begin{aligned}
\frac{\Pr[M_L(t; f) = x]}{\Pr[M_L(t'_i, t_{-i}; f) = x]} &= \prod_{i=1}^{d} \frac{\Pr[M_L(t; f)_i = x_i]}{\Pr[M_L(t'_i, t_{-i}; f)_i = x_i]} \\
&= \prod_{i=1}^{d} \frac{\Pr[Z_i = x_i - f(t)_i]}{\Pr[Z_i = x_i - f(t'_i, t_{-i})_i]} \\
&= \prod_{i=1}^{d} \frac{\exp\left(-\frac{\epsilon|x_i - f(t)_i|}{\mathrm{GS}(f)}\right)}{\exp\left(-\frac{\epsilon|x_i - f(t'_i, t_{-i})_i|}{\mathrm{GS}(f)}\right)} \\
&= \prod_{i=1}^{d} \exp\left(\frac{\epsilon\left(|x_i - f(t'_i, t_{-i})_i| - |x_i - f(t)_i|\right)}{\mathrm{GS}(f)}\right)
\end{aligned}$$

$$
\begin{aligned}
&\leq \prod_{i=1}^{d} \exp\left(\frac{\epsilon\left(|f(t)_i - f(t'_i, t_{-i})_i|\right)}{\mathrm{GS}(f)}\right) \\
&= \exp\left(\frac{\epsilon\left(\sum_{i=1}^{d} |f(t)_i - f(t'_i, t_{-i})_i|\right)}{\mathrm{GS}(f)}\right) \\
&\leq \exp(\epsilon)
\end{aligned}
$$

■

So now we know how to answer numeric valued queries privately.

The following basic facts (which we have encountered before) will also be useful:

Fact 10 *Suppose $M_1 : \mathcal{T}^n \to \mathcal{O}$ is ϵ_1-differentially private. Let $f : \mathcal{O} \to \mathcal{O}'$ be any randomized mapping. Then $f(M_1(t)) : \mathcal{T}^n \to \mathcal{O}'$ is ϵ_1 differentially private. Moreover, suppose $M_2 : \mathcal{T}^n \to \mathcal{O}'$ is ϵ_2-differentially private. Then $M : \mathcal{T}^n \to \mathcal{O} \times \mathcal{O}'$ defined by $M(t) = (M_1(t), M_2(t))$ is $\epsilon_1 + \epsilon_2$ differentially private. This is true even if M_2 can be chosen after observing the realized value of $M_1(t)$.*

The first property says in English "Applying arbitrary post-processing to a differentially private algorithm does not harm its privacy guarantee", and the second says "Private algorithms can be combined, and are still private, where their privacy parameters add up."

3.2 Counting on a stream

We want to use the Laplace mechanism to allow us maintain a *running count* of a stream $\sigma \in X^L$ where $X \in [-1, 1]$: i.e. each element σ_i of the stream is a real number between -1 and 1. You should think of the stream as really being a function $\sigma(t)$ of some set of reported agent types. What we want is to have an algorithm that "sees" the stream one number at a time, and maintains a running count of the numbers seen so far (i.e. produces an output at every time step), while guaranteeing that the whole computation is differentially private.

Definition 11 *A continual observation mechanism on a stream $\sigma(t) \in X^L$ is a randomized mapping $M(\sigma(t)) : \mathcal{N} \to \mathbf{R}$ such that $M(\sigma(t))(j)$ is independent of $\sigma(t)_i$ for all $i > j$.*

Lets give a name to the quantity we want to estimate – each of the prefix sums of the stream:

$$c_\sigma(j) = \sum_{i=1}^{j} \sigma_i$$

We can now define what we mean by a mechanism which estimates the running count on a stream accurately:

Definition 12 *A continual observation mechanism $M(\sigma)$ is $(\alpha(j), \beta)$ accurate on a stream σ if except with probability at most β we have for all $j \leq L$:*

$$|c_\sigma(j) - M(\sigma)(j)| \leq \alpha(j)$$

Let us warm up by considering a couple of simple mechanisms that we might try to solve this problem. Suppose that the stream $\sigma(t)$ has sensitivity k. To analyze these,use a concentration theorem for sums of Laplace random variables:

Theorem 13 *Suppose $Y_1, \ldots, Y_j \sim Lap(k/\epsilon)$. Let $Y = \sum_{i=1}^{j} Y_i$. Then:*

$$\Pr[|Y| \geq t \cdot k] \leq \exp\left(\frac{-t^2\epsilon^2}{6j}\right)$$

In particular:

$$\Pr\left[|Y| \geq \frac{k \cdot \sqrt{6j \log \frac{1}{\beta}}}{\epsilon}\right] \leq \beta$$

Algorithm1(ϵ)
  Let $\epsilon' \leftarrow \frac{\epsilon}{k \cdot L}$
  Let $c_0 \leftarrow 0$
  for $j = 1$ to L **do**
    Let $c_j \leftarrow c_{j-1} + \sigma_j$, $\nu_j \leftarrow \text{Lap}(1/\epsilon')$
    Output $c_j + \nu_j$
  end for

Ok, this algorithm sucks. We are adding just enough noise to guarantee ϵ-differential privacy: each entry σ_i appears in $\leq L$ counts $c_1, \ldots, c_L$. Therefore, each count is at most k sensitive, and the vector of all L counts is $k \cdot L$ sensitive. W add noise $\text{Lap}(k \cdot L/\epsilon)$ to each count, which is enough to guarantee differential privacy. The count c_j is never larger than L, but we are adding noise $\text{Lap}(k \cdot L/\epsilon)$ at every step! We don't get non-trivial error.

Here is another algorithm that sucks less:

Algorithm2(ϵ)
  Let $c_0 \leftarrow 0$
  for $j = 1$ to L **do**
    Let $\nu_j \leftarrow \text{Lap}(k/\epsilon)$, $\hat{\sigma}_j \leftarrow \sigma_j + \nu_j$, $c_j \leftarrow c_{j-1} + \hat{\sigma}_j$,
    Output c_j
  end for

This algorithm is a little better. Note that it still preserves ϵ-differential privacy: the algorithm can be viewed as directly computing $\sigma(t)$ using the Laplace mechanism (adding noise k/ϵ) to each coordinate, and then simply computing the partial sums of the already perturbed stream, which is "postprocessing" does not harm the privacy guarantee. Moreover, we have for all j:

$$c_j = \sum_{i=1}^{j} \sigma_i + \sum_{i=1}^{j} \nu_i$$

So the error of this mechanism at each step j is simply $E_j = |\sum_{i=1}^{j} \nu_i|$. By our concentration theorem above, except with probability β, we have for all j:

$$|E_j| \leq O\left(\frac{k \cdot \sqrt{j} \log \frac{j}{\beta}}{\epsilon}\right)$$

This is already non-trivial error! Can we do better? Lets examine the sources of error in the previous two algorithms. Both of these algorithms work by computing partial sums:

Definition 14 *A P-sum is a partial sum of consecutive items. Write:*

$$\Sigma[i, j] = \sum_{\ell=i}^{j} \sigma_\ell$$

We can think of both of the algorithms that we have seen as simply releasing a collection of p-sums. Algorithm 1 releases L noisy p-sums $\hat{\Sigma}[1,j]$ for each $j \leq L$. Algorithm 2 releases L noisy p-sums $\hat{\Sigma}[i,i]$ for $i \leq L$ and computes $M(\sigma)(j) = \sum_{i=1}^{j} \hat{\Sigma}[i,i]$.

Suppose an algorithm releases a collection of p-sums such that a single element in the stream can appear in at most c of the p-sums. Then the sensitivity of the output is at most $c \cdot k$, and to preserve privacy, each p-sum must be perturbed with noise $\text{Lap}(c \cdot k/\epsilon)$. Suppose further that each answer $M(\sigma)(j)$ is the sum of ℓ of these noisy p-sums. Then the error term is at most:

$$|E_j| = |M(\sigma)(j) - c_\sigma(j)| = |\sum_{i=1}^{\ell} \text{Lap}\left(\frac{c \cdot k}{\epsilon}\right)| \leq O\left(c \cdot k \frac{\sqrt{\ell} \log \frac{j}{\beta}}{\epsilon}\right)$$

except with probability β.

Indeed, Algorithm 1 had $c = L$ and $\ell = 1$, and Algorithm 2 had $c = 1$ and $\ell = j$. So, to develop an algorithm with lower error, we can simply try and develop a way of releasing a count by combining partial sums that has a better tradeoff between c and ℓ. This is what the binary mechanism does:

BinaryCount(ϵ, L)

Let $\epsilon' \leftarrow \frac{\epsilon}{k \cdot \log L}$.
for $j = 1$ to L **do**
  Express j in binary: $j = \sum_{i=1}^{\log L} b_i(j) \cdot 2^i$
  Let $i \leftarrow \min_\ell : b_\ell(j) \neq 0$ be the least non-zero bit of j.
  Let $a_i \leftarrow \sum_{\ell < i} a_\ell + \sigma_j$ $(a_i = \Sigma[j - 2^i + 1, j])$
  For $\ell < i$ Let $a_\ell \leftarrow 0$, $\hat{a}_\ell \leftarrow 0$.
  Let $\hat{a}_i \leftarrow a_i + \text{Lap}(1/\epsilon')$.
  Output $M(\sigma)(j) = \sum_{i:b_i(j)=1} \hat{a}_i$
end for

To analyze this algorithm in the p-sum framework, first note that by design, every output is the sum of at most $\ell = \{i : b_i(j) = 1\} \leq \log L$ p-sums. Moreover, each σ_j is a member of at most a single p-sum of length 2^i for each i, and so is a member of at most $c = \log L$ p-sums. Hence, the algorithm preserves differential privacy, and moveover, except with probability β, we have at each time step the error is at most:

$$|E_j| = O\left(\frac{k \cdot \log(L) \cdot \sqrt{\log L} \cdot \log(\frac{j}{\beta})}{\epsilon}\right) = O\left(\frac{k \cdot \log\left(\frac{L}{\beta}\right)^{5/2}}{\epsilon}\right)$$

Ok! So now we know how to count.

Next class, we will see how to use this new-fangled ability to privately compute Nash equilibria in "large" congestion games. We will then go on and show how to privately compute correlated equilibria in arbitrary large games, and examine the game-theoretic implications of this.

Bibliographic Information The method of privately maintaining a counter over a stream of numbers was developed independently by [CSS10] and [DNPR10]. We follow the presentation of Chan et al. [CSS10]. The goal of implementing a mediator in a game by privately computing an equilibrium is due to [KPRU14], who show how to privately compute correlated equilibria. The implication that privately computing a *Nash* equilibrium gives a mediator that has ex-post truthfulness properties is due to [RR13], who show how to privately compute Nash equilibria in congestion games.

References

[AAE05] Baruch Awerbuch, Yossi Azar, and Amir Epstein. The price of routing unsplittable flow. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 57–66. ACM, 2005.

[CK05] George Christodoulou and Elias Koutsoupias. The price of anarchy of finite congestion games. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 67–73. ACM, 2005.

[CSS10] TH Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. In *Automata, Languages and Programming*, pages 405–417. Springer, 2010.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC '06*, pages 265–284, 2006.

[DNPR10] Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N Rothblum. Differential privacy under continual observation. In *Proceedings of the 42nd ACM symposium on Theory of computing*, pages 715–724. ACM, 2010.

[KPRU14] Michael Kearns, Mallesh M Pai, Aaron Roth, and Jonathan Ullman. Mechanism design in large games: Incentives and privacy. *Proceedings of the annual 5th Innovations in Theoretical Computer Science (ITCS) conference*, 2014.

[RR13] Ryan Rogers and Aaron Roth. Asymptotically truthful equilibrium selection in large congestion games. *arXiv preprint arXiv:1311.2625*, 2013.