Randomness in Cryptography February 7, 2013

Lecture 5: SV-robust Mechanisms and Bias-Control-Limited Source

Lecturer: Yevgeniy Dodis *Scribe: Abhishek Samanta*

We have seen that additive noise technique is not $(SV(\gamma), \varepsilon)$-DP. So, the question that we explore in todays class is whether differential privacy (DP) is achievable with SV sources. Interestingly, we give a differential private mechanism for approximate arbitrary "low sensitive" functions that works even with randomness coming from SV source, for any $\gamma < 1$. We conclude todays lecture with some key insights to a new imperfect random source, we call BCL source.

1 Last Class

Before delving into technical details, let us refresh our memory with some important metrics we defined and few important results we proved in last lecture.

DEFINITION 1 A mechanism M is $(\mathcal{R}, \varepsilon)$-DP for $\mathcal{Q}$, if $\forall\ x, x' \in \mathcal{X}$ s.t. $|x - x'| = 1$, $\forall q \in \mathcal{Q}$, and $\forall z$,

$$\frac{Pr_R[M(x', q; R) = z]}{Pr_R[M(x, q; R) = z]} \geq e^{-\varepsilon}$$

DEFINITION 2 A mechanism M is $(\mathcal{R}, \rho)$-accurate w.r.t $\mathcal{Q}$, if $\forall x \in \mathcal{X}, \forall q \in \mathcal{Q}$,

$$\mathbb{E}_{\mathcal{R}}[|M(x, q; \mathcal{R}) - q(x)|] \leq \rho$$

Theorem 1 ***(Theorem 2 lecture 4)*** *It is impossible to construct a "non-trivial" mechanism M with $(Weak_k(m), \varepsilon)$-differential-privacy and $(Weak_k(m), \rho)$-utility, if $k \leq m - \log(\varepsilon \cdot \rho) - o(1)$*

Remark 1 *γ-SV does not work for ρ-utiility, where $\rho > \frac{1}{\varepsilon}$, but imply the following restriction,*

$$\forall q, \forall x, x' \in \mathcal{X}, s.t. |x - x'| = 1$$
$$Pr_{r \leftarrow U_N}[M(x, q; r) \neq M(x', q; r)] \leq \frac{2 \cdot \varepsilon}{\gamma} = O(\varepsilon)$$

2 A new differential private mechanism

DEFINITION 3 A mechanism M has ε-consistent sampling (ε-CS) if $\forall x, x' \in \mathcal{X}$, s.t. $|x - x'| = 1$, $\forall q \in \mathcal{Q}$ and $\forall z$,

$$Pr_{r \leftarrow U_N}[M(x', q; r) = z | M(x, q; r) = z] \geq e^{-\varepsilon} \approx 1 - \varepsilon$$

Lemma 1 *If a mechanism M is ε-CS, then M is (U_N, ε)-DP*

Proof:

$$\frac{Pr_{r\leftarrow U_N}[M(x',q;r)=z]}{Pr_{r\leftarrow U_N}[M(x,q;r)=z]}$$
$$\geq \frac{Pr_{r\leftarrow U_N}[M(x,q;r)=z]\cdot Pr_{r\leftarrow U_N}[M(x',q;r)=z|M(x,q;r)=z]}{Pr_{r\leftarrow U_N}[M(x,q;r)=z]}$$
$$\geq e^{-\varepsilon}$$

□

Now, recall that we have defined additive noise mechanism as follows,

$$M_{lap}(x,q;r) = q(x) + Rlap_{0,1/\varepsilon}(r),\ with\ \rho = O(\frac{1}{\varepsilon}), \tag{1}$$

where $Rlap_{0,1/\varepsilon}(r)$ is number of flips of $(1-\varepsilon)$-biased coin until a head.
Additive-noise mechanism, Equation (1), fails to handle SV sources because such algorithms use disjoint sets of coins to produce "noisy answer" on two databases having different "real answers". We explain this with the example shown in Figure 1, with different values of $q(x)$. $\overline{M}_{lap}$, Equation (2), solves this problem by clustering together $1/\varepsilon$ number of intervals,

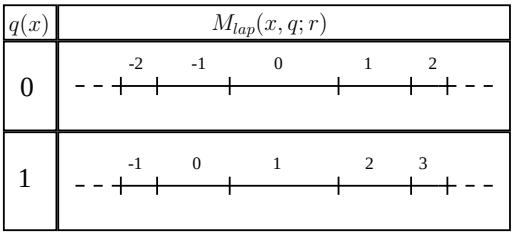

Figure 1: Example illustrating the fact that additive-noise mechanism fails to handle SV sources.

$$\overline{M}_{lap}(x,q;r) = \lceil q(x) + Rlap_{0,1/\varepsilon}(r) \rfloor_{1/\varepsilon}, \tag{2}$$

where $\lceil a \rfloor_b$ is rounding a to nearest multiple of b. As can be seen from the example in Figure 2, $(\frac{1}{\varepsilon}-1)$ of $\frac{1}{\varepsilon}$ intervals overlap. So, size of overlap is $\frac{1/\varepsilon - 1}{1/\varepsilon} = (1-\varepsilon)$ fraction of the size of a set.

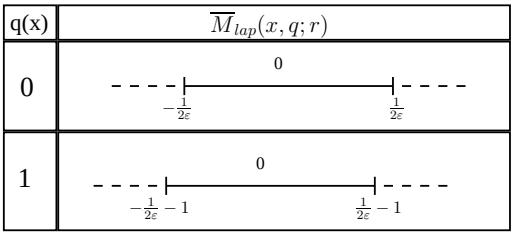

Figure 2: $\overline{M}_{lap}$ distribution for $q(x) = 0, 1$

Lemma 2 *$\overline{M}_{lap}$ is a ε-CS ((U_N, ε)-DP) and has utility $\rho' = O(\frac{1}{\varepsilon})$.*

Proof: Since, two consecutive sets have $(1-\varepsilon)$ fraction of overlap,

$$\frac{Pr_{r \leftarrow U_N}[\overline{M}_{lap}(x', q; r) = z]}{Pr_{r \leftarrow U_N}[\overline{M}_{lap}(x, q; r) = z]} \approx 1 - \varepsilon$$

Thus $\overline{M}_{lap}$ is ε-CS.
Now, as discussed in the previous lecture (Theorem 1 Lecture 4) M_{lap} is $(U_N, \mathrm{O}(1/\varepsilon))$-accurate. So, utility ρ' of $\overline{M}_{lap}$ holds the following condition,

$$\begin{aligned} \rho' &\leq O(1/\varepsilon) + \frac{1}{2\varepsilon} \\ &= O(1/\varepsilon) \end{aligned}$$

Thus, $\overline{M}_{lap}$ is $(U_N, O(1/\varepsilon))$-accurate. □

2.1 Differential privacy offered by ε-CS

Now, the question that we ask is if ε-CS is $(SV(\gamma), \mathrm{O}(\varepsilon))$-DP. But, unfortunately the answer is no. Let us illustrate this with the help of following example shown in Figure 3.
A $SV(\gamma)$ can be viewed as a tree where each branch (labeled 0 or 1) can be chosen with a probability $p \in [\frac{1}{2}(1-\gamma), \frac{1}{2}(1+\gamma)]$. Now, let us assume coins as follows,

$$\begin{aligned} \{r : \overline{M}_{lap}(x, q; r) = z\} &= S_0 \cup S_1 \\ \{r : \overline{M}_{lap}(x', q; r) = z\} &= S' \end{aligned} \tag{3}$$

Let us also assume that $|S_1| = \varepsilon \cdot |S_0|$.

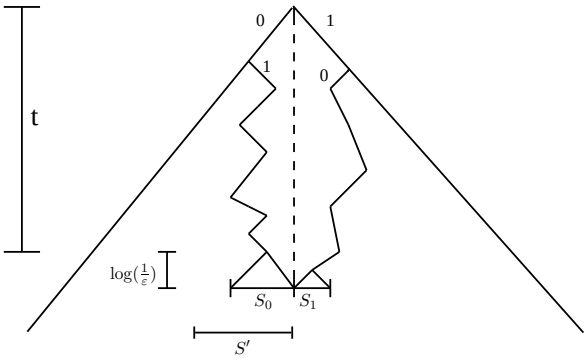

Figure 3: Example of how a $SV(\gamma)$ adversary can decrease the ratio $\frac{Pr_{r \leftarrow SV(\gamma)}[r \in S']}{Pr_{r \leftarrow SV(\gamma)}[r \in S_0 \cup S_1]}$

2.1.1 Goal of adversary

The goal of the adversary is decreasing the ratio $\frac{Pr_{r \leftarrow SV(\gamma)}[r \in S']}{Pr_{r \leftarrow SV(\gamma)}[r \in S]}$

2.1.2 Attack Strategy

- Pick $r \leftarrow \{0,1\}$
- If $r = 1$, then, bias towards S_1
- If $r = 0$, then, avoid $S_0 \cup S'$

So,

$$\begin{aligned}\frac{Pr_{r\leftarrow SV(\gamma)}[r \in S']}{Pr_{r\leftarrow SV(\gamma)}[r \in S_0 \cup S_1]} &= \frac{|S'|}{|S_0 \cup S_1|} \\ &\leq \frac{[\frac{1}{2}(1-\gamma)]^t}{[\frac{1}{2}(1-\gamma)]^t \cdot [\frac{1}{2}(1+\gamma)]^{t+\log(1/\varepsilon)}} \\ &= \frac{1}{1+\frac{[\frac{1}{2}(1+\gamma)]^{t+\log(1/\varepsilon)}}{[\frac{1}{2}(1-\gamma)]^t}} \\ &\leq \frac{1}{1+\varepsilon\cdot[\frac{1+\gamma}{1-\gamma}]^t}\end{aligned}$$

Now,

$$\frac{1}{1+\varepsilon\cdot[\frac{1+\gamma}{1-\gamma}]^t} \to 0 \text{ as } t >> \frac{1}{\gamma}\log(\frac{1}{\varepsilon})$$

2.2 SV-consistent Sampling

DEFINITION 4 A mechanism M is ($\overline{\varepsilon}$, c)-SV-consistent sampling (SVCS), if following two conditions are satified,

- M is ε-CS
- M is "c-nice"

♢

DEFINITION 5 A mechanism M is c-nice is $\forall x \in \mathcal{X}$ and $q \in \mathcal{Q}$

$$\frac{|suffix(x)|}{|S \cup S'|} \leq c,$$

where,

$$S = \{r, \overline{M}_{lap}(x, q; r) = z\}$$
$$S' = \{r, \overline{M}_{lap}(x', q; r) = z\}$$

♢

DEFINITION 6 A node x of SV-tree is least common ancestor of S and S' if x is the node with least height satisfying the following condition,

$$S \cup S' \subset suffix(x)$$

♢

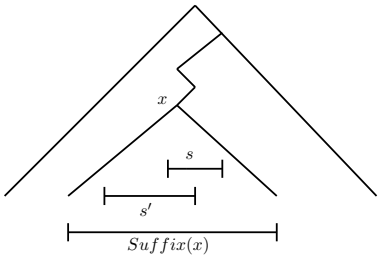

Figure 4: Example SV-tree to illustrate least common ancestor and suffix function. x is called the least common ancestor of S and S'

Theorem 2 *A mechanism M is ($\overline{\varepsilon}$, c)-SVCS $\Rightarrow$ M is ($SV(\gamma)$, ε)-DP, where* $\varepsilon = 2(8\overline{\varepsilon})^{(1-\log(1+\gamma))} \cdot [\frac{1-\gamma}{1+\gamma}]^{\log(8c)}$ *and* $c = O(1)$.

Proof: Let us consider the highest node w, s.t. $suffix(w) \subset S'$, and v, s.t. v is the least common ancestor $S \setminus S'$, Figure 5. Let us partition $S \setminus S'$ in I_0 and I_1, s.t. $|I_0| = |I_1| = \frac{|S \setminus S'|}{2}$ Let v_0 and v_1 be the least common ancestors of I_0 and I_1, respectively. Without loss of generality, let us also consider that $|v_0| \leq |v_1|$. Now,

$$\begin{aligned}
\frac{Pr_{r \leftarrow SV(\gamma)}[r \in S]}{Pr_{r \leftarrow SV(\gamma)}[r \in S']} &\leq 1 + \frac{Pr_{r \leftarrow SV(\gamma)}[r \in S]}{Pr_{r \leftarrow SV(\gamma)}[r \in S']} \\
&= 1 + \frac{Pr_{r \leftarrow SV(\gamma)}[r \in S \setminus S' | r \in suffix(u)]}{Pr_{r \leftarrow SV(\gamma)}[r \in S' | r \in suffix(u)]} \\
&\leq 1 + \frac{Pr_{r \leftarrow SV(\gamma)}[r \in suffix(v_0) \cup r \in suffix(v_1) | r \in suffix(u)]}{Pr_{r \leftarrow SV(\gamma)}[r \in suffix(w) | r \in suffix(u)]} \\
&= 1 + \frac{[\frac{(1+\gamma)}{2}]^{|v_0|-|u|} + [\frac{(1+\gamma)}{2}]^{|v_1|-|u|}}{[\frac{(1-\gamma)}{2}]^{|w|-|u|}} \\
&\leq 1 + 2 \cdot \frac{[\frac{(1+\gamma)}{2}]^{|v_0|-|u|}}{[\frac{(1-\gamma)}{2}]^{|w|-|u|}} \\
&= 1 + 2 \cdot \Big[\frac{(1+\gamma)}{2}\Big]^{|v_0|-|w|} \cdot \Big[\frac{(1+\gamma)}{(1-\gamma)}\Big]^{|w|-|u|} \\
&\leq 1 + 2 \cdot (8\overline{\varepsilon})^{1-\log(1+\gamma)} \cdot \Big[\frac{(1+\gamma)}{(1-\gamma)}\Big]^{\log(8c)}, \quad \text{[By, Lemma 4]} \qquad (4)
\end{aligned}$$

Since, M is ($\overline{\varepsilon}$, c)-SVCS,

$$\frac{Pr_{r \leftarrow SV(\gamma)}[r \in S]}{Pr_{r \leftarrow SV(\gamma)}[r \in S']} \leq 1 + \varepsilon \qquad (5)$$

Comparing right hand side of Equation (4) and Equation (5),

$$\varepsilon = 2 \cdot (8\overline{\varepsilon})^{1-\log(1+\gamma)} \cdot \Big[\frac{(1+\gamma)}{(1-\gamma)}\Big]^{\log(8c)}$$

□

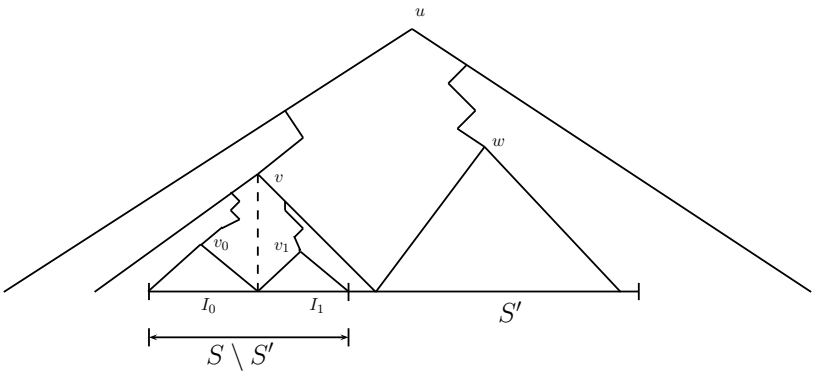

Figure 5: Construction of ($SV(\gamma)$, ε)-DP

Corollary 3 $\forall \gamma < 1$, $\varepsilon \to 0$ *and* $\overline{\varepsilon} \to 0$.

Lemma 4 *If M has $(\overline{\varepsilon}, c)$-SVCS then for all neighboring databases $D_1, D_2 \in \mathcal{D}$, which define u, v_0, v_1, w as in Figure 5, we have,*

$$|v_0| - |w| \geq \log(\frac{1}{8 \cdot \overline{\varepsilon}})$$
$$|w| - |u| \leq \log(8c)$$

Proof: By definition of $(\overline{\varepsilon}, c)$-SVCS, we have,

$$\frac{S \setminus S'}{S'} \leq \overline{\varepsilon}$$
$$\frac{|suffix(u, n)|}{|S|} \leq c$$

So,

$$\begin{aligned} \frac{|suffix(v_0, n)|}{2} + \frac{|suffix(v_1, n)|}{2} &\leq |I_0| + |I_1| \\ &= |S \setminus S'| \\ &\leq \overline{\varepsilon} \cdot |S'| \\ &\leq 4 \cdot \overline{\varepsilon} \cdot |suffix(w, n)| \end{aligned}$$

Therefore,

$$n - |v_0| \leq \log(8\overline{\varepsilon}) + n - |w|$$
$$\Rightarrow |v_0| - |w| \geq \log(\frac{1}{8\overline{\varepsilon}})$$

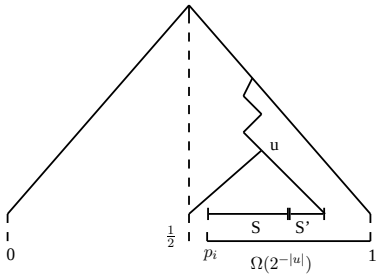

Figure 6: SVCS implementation of $\overline{M}_{lap}$

Again, we have,

$$\begin{aligned} |suffix(u,n)| &\leq c \cdot |S| \\ &\leq c \cdot (|S \setminus S'| + |S|) \\ &\leq (1+\overline{\varepsilon}) \cdot c \cdot |S'| \\ &\leq 2 \cdot c \cdot |S'| \\ &\leq 8 \cdot c \cdot |suffix(w,n)| \end{aligned}$$

Thus,

$$\begin{aligned} n - |u| &\leq \log(8 \cdot c) + n - |w| \\ \Rightarrow |w| - |u| &\leq \log(8 \cdot c) \end{aligned}$$

□

Theorem 3 *$\forall\overline{\varepsilon}$, $\exists(\overline{\varepsilon}, O(1))$-SVCS implementation of $\overline{M}_{lap}$.*

Proof:

It is to be noted that, we have already proved that $\overline{M}_{lap}$ is ε-cs, Lemma 2. So, by definition of SVCS, we have to prove that $\overline{M}_{lap}$ is "c-nice". To prove this, we use arithmatic coding as follows,

Construction 1

- *Place interval (I_r) corresponding to $Pr(R = r)$ on $[0, 1]$.*
- *Map real number $X \in [0, 1]$ to unique r, s.t. $x \in I_r$.*
- *Sample $X = 0.X_1X_2X_2...$ by sampling random bits $X_1, X_2, X_3, ...$, until it identifies an interval, uniquely.*

Let, u be the least common ancestor of $S \cup S'$, Figure 6. By construction, $|suffix(u)| = 2^{-|u|}$. Moreover, arithmatic coding and our use of Laplacian distribution ensure that smaller intervals are farther from center than bigger ones. So, $|S \cup S'| = \Omega(2^{-|u|})$. Therefore,

$$\frac{|suffix(u)|}{|S \cup S'|} = O(1)$$

□

Note 1 $(1+\frac{1}{\varepsilon})$ *consecutive intervals starting from* p_i *cover a constant fraction of the range* $[p_i, 1]$*, Figure 6. So, for any consecutive interval* S, S'*,*

$$1 - p_i = \Omega(2^{-|u|})$$
$$\Rightarrow p_i \approx 1 - 2^{-|u|}$$

Project 1 *What about other DP-mechanisms(viz. exponential mechanism)? What are the effects of CS and SVCS on them?*

Project 2 *Is* ε*-CS enough if allow* (ε, δ)*-DP, with negligible* δ*.*

Project 3 *This project consists of following two parts,*

- *Generalize DP to* $Weak_m(k)$ *sources if* $k \geq m - \log(\varepsilon \cdot \rho)$*.*
- *Define more realisitic source between* $Weak_m(k)$ *and* $SV(\gamma)$*.*

3 New Imperfect Random Source with Applications to Coin-Flipping

We are going to look into a new imperfect random source which realistically generalizes SV-source [4] and the bit fixing (LLS) source [3]. This new source is called *Bias Control Limited (BCL)* sources.
A BCL source is characterized by a "noise" oarameter $\gamma \in [0, \frac{1}{2}]$ and a "number of error" parameter $b \geq 0$. It is also convenient to fix the number of bits, N, emitted by the source. Hence, we define BCL source as follows,
DEFINITION 7 A $\underline{(\gamma, b, N) - BCL}$ source generates N bits x_i, where $i \in [1, N]$. The value of x_i depends on x_j, where $j \in [1, i-1]$ in one of the following two ways,

- x_i is determined by x_j. But, this happens for at most b bits. This process of determining a bit is called intervention.
- $Pr[x_i = 1 | x_1, x_2, ..., x_{i-1}] \in [\frac{(1-\gamma)}{2}, \frac{(1+\gamma)}{2}]$.

$\diamondsuit$

Note 2

- *If* $b = 0$*,* (γ, b, N)*-BCL source behaves as a* (γ, N)*-SV source*
- *If* $\gamma = 0$*,* (γ, b, N)*-BCL source behaves as a* (b, N)*-LLS source*

Project 4 *Do DP with BCL source for reasonably high* b*.*

Question 2 *For which value of* $b = b(N)$ *can there be a good extractor?*

Now, we quantitatively measure the "goodness" of BCL source for the problem of bit-extraction.

DEFINITION 8 Let $\mathcal{A}$ be some (γ, b, N)-BCL source, and $e : \{0,1\}^N \to \{0,1\}$ be a 1-bit-extractor. Let us define,

- $\underline{q(\gamma, b, n, e, \mathcal{A})}$ be the bias of the coin $e(x)$, where $x = x_1...x_N$ was produced by $\mathcal{A}$.
- $\underline{q(\gamma, b, N, e)} = max_{\mathcal{A}} q(\gamma, b, N, e, \mathcal{A})$, for all (γ, b, N)-BCL sources.
- $\underline{q(\gamma, b, N)} = min_e q(\gamma, b, N, e)$, for all $e : \{0,1\}^N \to \{0,1\}$.

$\diamondsuit$

Thus, $q(\gamma, b, N)$ is the smallest bias of a coin that can be extracted from any (γ, b, N)-BCL source.

Theorem 4 *([4]) $q(\gamma, 0, N) = \gamma$. Thus, it is possible to extract an almost perfect bit iff $\gamma = o(1)$, and a slightly random bit iff $\gamma = 1 - \Omega(1)$.*

Theorem 5 *([3]) For any b, majority is the best bit extraction function for the LLS source. In particular, $q(0, c_1 \cdot \sqrt{N}, N) = o(1)$, while $q(0, c_2 \cdot \sqrt{N}, N) = 1 - o(1)$, for some constants $c_1 < c_2$.*

Note 3 *A random function $f : \{0,1\}^N \to \{0,1\}$ is a bad bit extractor for a LLS source even for $b = \omega(1)$. Although, with high probability the first $(N - b)$ bits do not fix f, $\mathcal{A}$ can use last b interventions to fix f. Another bad function (even for $b = 1$) is any parity function: it can be fixed by fixing the last emitted bit. On the other hand, majority is the best extraction funtion and can tolerate $b = \Theta(\sqrt{N})$.*

Theorem 6 *If $b \cdot \gamma = \Theta(1)$, it is impossible to extract a slightly random bit from a (γ, b, N)-BCL source, irrespective of the value of N. More precisely,*

$$q(\gamma, b, N) \geq 1 - \frac{2}{(1+\gamma)^b} = 1 - 2^{1-\Theta(b \cdot \gamma)}$$

Lemma 5 *If $b \cdot \gamma = O(1)$, $b = O(\sqrt{N})$, and $\gamma = o(1)$, it is possible to extraxt an almost random bit from a (γ, b, N)-BCL source: $q(\gamma, b, N) = o(1)$. In particular, such extraction can be done by applying the majority function to any $min(N, O(1/\gamma^2))$ bits of the source.*

p-sparse

Given an extractor $e : \{0,1\}^N \leftarrow \{0,1\}$, we can associate an event ξ such that "ξ happend $\Leftrightarrow e(x) = 1$". So, the natural probability of ξ is the probability that ξ happend for an ideal source, which in our case, emits N perfect unbiased bits. More precisely,

$$p = Pr_{r \leftarrow U_N}[e(r) = 1] = Pr_{r \leftarrow U_N}[\xi]$$

We then say that ξ is p-sparse.

DEFINITION 9 $\underline{F_\gamma(p, N, b)} = max_\xi min_{\mathcal{A}} Pr[e(x) = 0]$, taken over all p-sparse ξ, and all (γ, b, N)-BCL source. $\diamondsuit$

Observation 7

$$q(\gamma, b, N) \geq 1 - F_\gamma(\frac{1}{2}, b, N)$$

Theorem 8

$$F_\gamma(p, N, b) \leq \frac{1}{p \cdot (1+\gamma)^b} = 2^{\log(1/p) - \Theta(b \cdot \gamma)}$$

In particular , if $b = \omega(\frac{1}{\gamma} \cdot \log(1/p))$, $\mathcal{A}$ *can force any p-sparse* ξ *with probability* $1 - o(1)$.

Proof: The statement is true for $\gamma = 0$ or $b = 0$, since $F_\gamma(\cdot, \cdot, \cdot) \leq 1 \leq 1/p$, so assume $\gamma > 0$ and $b \geq 1$. Define $g(p, b) = \frac{1}{p(1+\gamma)^b}$. We need to show that $F_\gamma(p, N, b) \leq g(p, b)$ for any $N \geq 1$, $1 \leq b \leq N$ and $0 \leq p \leq 1$. We prove this by induction on N.

Base Case For $N = 1$, $F_\gamma(0, 1, b) = 1 < \infty = g(0, b)$, and $F_\gamma(p, 1, b) = 0 \leq g(p, b)$ for $p > 0$ (here we used $b \geq 1$).

Induction step Assume now the claim is true for $(N-1)$ and we want to show it for N.

Take any p-sparse $\mathcal{E}$ given by a function e. Let $e_0 : \{0,1\}^{N-1} \to \{0,1\}$ be the restriction of e when $x_1 = 0$. Similarly for e_1. This defines a p_0-sparse event $\mathcal{E}_0$ and a p_1-sparse event $\mathcal{E}_1$ satisfying $\frac{1}{2}(p_0 + p_1) = p$. Without loss of generality assume $p_0 \geq p \geq p_1$. Given such $\mathcal{E}$, our particular adversary $\mathcal{A}$ will consider two options and pick the best (using his unbounded computational resources): either he will use an intervention (he can do it since we assumed $b \geq 1$) and fix $x_1 = 0$, reducing the question to that of analyzing the p_0-sparse event $\mathcal{E}_0$ on $(N-1)$ variables and also reducing b by 1, or he will use rule (B) making the 0-probability of x_1 equal to $\frac{1}{2} \cdot (1+\gamma)$ and leaving the same b. By the definition of function $F_\gamma(p, N, b)$, we know that in the first case the failure probability of $\mathcal{A}$ will be at most $F_\gamma(p_0, N-1, b-1)$, and in the second case it will be at most $(\frac{1}{2} \cdot (1-\gamma)) F_\gamma(p_1, N-1, b) + (\frac{1}{2} \cdot (1+\gamma)) F_\gamma(p_0, N-1, b)$. Since, the choice of $p_0 \geq p_1$ (i.e., how $\mathcal{E}$ splits into $\mathcal{E}_0$ and $\mathcal{E}_1$) such that $p_0 + p_1 = 2p$ is outside of our control, we will take the maximum over all such choices and obtain the following recurrence.

$$F_\gamma(p, N, b) \leq \max_{\substack{p_0 \geq p_1 \\ p_0 + p_1 = 2p}} \min[F_\gamma(p_0, N-1, b-1), \left(\frac{1}{2} \cdot (1-\gamma)\right) \cdot F_\gamma(p_1, N-1, b) + \left(\frac{1}{2} \cdot (1+\gamma)\right) \cdot F_\gamma(p_0, N-1, b)]$$

Let $p_0 = p(1+\beta)$ and $p_1 = p(1-\beta)$, where $0 \leq \beta \leq 1$ (since $p_0 \geq p \geq p_1$). Using our inductive assumption.

$$F_\gamma(p, N, b) \leq \max_{0 \leq \beta \leq 1} \min[g(p(1+\beta), b-1), \left(\frac{1}{2} \cdot (1-\gamma)\right) \cdot g(p(1+\beta), b) + \left(\frac{1}{2} \cdot (1+\gamma)\right) \cdot g(p(1+\beta), b)]$$

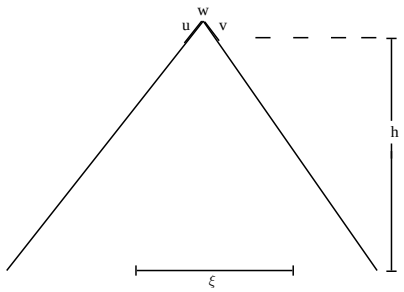

Figure 7: Definition of u, v, ξ in binary tree representation of SV-source

Recalling the definition of g, it thus suffices to show that

$$\max_{0\leq\beta\leq 1} \min\left(\frac{1}{p(1+\beta)(1+\gamma)^{b-1}}, \frac{\frac{1}{2}\cdot(1-\gamma)}{p(1-\beta)(1+\gamma)^{b}} + \frac{\frac{1}{2}\cdot(1+\gamma)}{p(1+\beta)(1+\gamma)^{b}}\right) \leq \frac{1}{p(1+\gamma)^{b}}$$

$$\iff \qquad \max_{0\leq\beta\leq 1} \min\left(\frac{1+\gamma}{1+\beta}, \frac{\frac{1}{2}\cdot(1-\gamma)}{1-\beta} + \frac{\frac{1}{2}\cdot(1+\gamma)}{1+\beta}\right) \leq 1 \tag{6}$$

We see that the expressions under the minimum are equal when $\beta = \gamma$. We consider two cases.

- Case 1. Assume $\beta \geq \gamma$. Then the minimum above is $\frac{1+\gamma}{1+\beta}$ and it suffices to show that $\frac{1+\gamma}{1+\beta} \leq 1$, which is equivalent to our assumption on β.
- Case 2. Assume $\beta \leq \gamma$. Then the minimum above equals to $\frac{\frac{1}{2}\cdot(1-\gamma)}{1-\beta} + \frac{\frac{1}{2}\cdot(1+\gamma)}{1+\beta}$ and it suffices to show that $\frac{\frac{1}{2}\cdot(1-\gamma)}{1-\beta} + \frac{\frac{1}{2}\cdot(1+\gamma)}{1+\beta} \leq 1$. But this is again equivalent to our assumption on β.

□

Theorem 9 *For $SV(\gamma, N)$, for all p-sparse ξ, $\exists$ an adversary $\mathcal{A}$ s.t. $\mathcal{A}$ increases $Pr(\xi)$ from p to atleast $p^{1-\log(1+\gamma)}$.*

Proof: Recall that a (γ, N)-SV source can be represented as a finite binary tree and let ξ be the subset of its leaves having probability p under unbiased coin. We are going to prove Theorem 9 by induction on height of the binary tree representation of (γ, N)-SV source.

Base case: The base case is a tree consisting of a single node labeled ξ. In this case $p = 1$. So, the induction hypothesis holds trivially.

Induction step: Let us assume that the induction hypothesis holds for a tree of height h. Let the probability of reaching ξ through left and right child of the root(w) be u and v, respectively, Figure 3. Since, p is the probability of hitting ξ if a child is chosen at uniformly at random, $p = \frac{(u+v)}{2}$. Let us also assume that without loss of generality $u \geq v$. Now, let us consider two cases as follows,

case 1 ($v = 0$):

If $v = 0$, $p = \frac{u}{2}$. So, by induction hypothesis probability of hitting ξ is,

$$\begin{aligned}(\frac{1}{2} \cdot (1+\gamma)) \cdot u^{1-\log(1+\gamma)} &= (\frac{1}{2} \cdot (1+\gamma)) \cdot (\frac{1}{2} \cdot (1+\gamma))^{-\log(u)} \\ &= (\frac{1}{2} \cdot (1+\gamma))^{-\log(u/2)} \\ &= p^{1-\log(1+\gamma)}\end{aligned}$$

case 2 ($v > 0$):

To prove the induction step, we have to verify that,

$$(\frac{1}{2} \cdot (1+\gamma)) \cdot u^{1-\log(1+\gamma)} + (\frac{1}{2} \cdot (1-\gamma)) \cdot v^{1-\log(1+\gamma)} \geq \left(\frac{(u+v)}{2}\right)^{1-\log(1+\gamma)}$$

Let us consider that $u = \alpha \cdot v$, where $\alpha \geq 1$. Then the above equation is equivalent to,

$$(\frac{1}{2} \cdot (1+\gamma)) \cdot \alpha^{1-\log(1+\gamma)} + (\frac{1}{2} \cdot (1-\gamma)) \geq \left(\frac{(1+\alpha)}{2}\right)^{1-\log(1+\gamma)}$$

Let the LHS of Equation (7) be $l(\alpha)$ and RHS be $r(\alpha)$. Now, let $l'(\alpha)$ and $r'(\alpha)$ be the first order derivative of $l(\alpha)$ and $r(\alpha)$, respectively. Now,

$$\begin{aligned}l'(\alpha) &= \frac{(1+\gamma)) \cdot (1-\log(1+\gamma))}{2 \cdot \alpha^{\log(1+\gamma)}} \\ r'(\alpha) &= \frac{(1+\gamma)) \cdot (1-\log(1+\gamma))}{2 \cdot (\alpha+1)^{\log(1+\gamma)}}\end{aligned}$$

The exponent $\log(1+\gamma) > 0$ for $\gamma > 0$. So, $l'(\alpha) > r'(\alpha)$. Again since, $l(1) = 1 = r(1)$, $l(\alpha) > r(\alpha)$. □

Project 5 *Translate to stronger impossibility of traditional privacy.*

$$\textit{Hint: } \xi = [f(r) \neq g(r)] = 1 - \frac{1}{T}$$

Project 6 *Develop MACs with BCL sources for rate* $< \frac{1}{2}$.

$$\textit{Hint: } 2^{-K} = [\frac{1}{2} \cdot (1+\gamma)]^{N-b}$$

Project 7 *Develop clear understanding of bounded budget source. The total budget* $B = b + (n-b) \cdot \gamma$

Project 8 *Break* $O(\sqrt{n})$ *barrier, (or, show optimality) for adaptively secured coin flipping.*

3.1 BCL source and collective coin flipping

The setting In this model n computationally unbounded processors are trying to generate a random bit in a setting where only a *s*ingle broadcast channel is available for communication. We assume that some of the players (at most b out of n) can be faulty or malicious, and in fact is controlled by a central adversary $\mathcal{A}$ (which is called b-bounded). In each round of the protocol every player can broadcast a message to the other players. A crucial complication is that the network is asynchronous within a round. For example, players cannot flip a coin by broadcasting a random bit and taking their exclusive OR: the last player to talk can completely control the output. Again taking the worst case scenario, we assume that in each round first $\mathcal{A}$ receives all the messages broadcast by the honest players, and only then decides which messages to send on behalf of the bad players. The output of the protocol is some pre-agreed deterministic function of the messages exchanged over the broadcast channel.

The Goal The objective of collective coin-flipping (parameterized by the number of players, n) is for the players to agree on a "random" bit, even in the presence of an adversary. Of course, the adversary $\mathcal{A}$ will introduce some bias into the coin. We let $\Delta_\Pi(b)$ be the largest bias achieved by a b-bounded adversary against protocol Π. Then, a coin-flipping protocol Π is said to be (weakly) $b(n)$-resilient if Π produces a slightly random coin: $\Delta_\Pi(b(n)) \leq \frac{1}{2} - \Omega(1)$, where the constant is independent of n.

Coin-Flipping with adaptive adversaries Let us assume that, we are given with a protocol Π which is known be "very good" against static adversaries. The question that we ask is if it is possible to transform it in a "black-box" way so as to obtain a "somewhat-good" adaptively secure protocol Φ.

DEFINITION 10 Let N be any integer and $f : \{0,1\}^N \leftarrow \{0,1\}$ be any function. We let $\Phi(N, f, \Pi)$ be the protocol where players sequentially run N times the protocol Π, obtain coins $x_1, x_2, ..., x_N$ and outputs $f(x_1, x_2, ..., x_N)$ as the resulting coin. The class $\{\Phi(N, f, \Pi) | N \geq 1, f : \{0,1\}^N \leftarrow \{0,1\}\}$ is called the class of black box transformation of Π. ♢

Let us assume that given a fixed set B of faulty players, Π produces at most a $\Delta_\Pi(B)$-biased coin for any static adversary who corrupts at the beginning, and let $\Delta_\Pi(b) = max_{|B|=b}\Delta_\Pi(B)$ be the best bias that a b-bounded static adversary can achieve. Let us denote by Π_i the i^{th} run of Π, and by x_i the resulting coin. As before, $\mathcal{A}$ is called b-bounded if he corrupts at most b players overall. However, now we assume that $\mathcal{A}$ (the adversary for $\Phi(N, f, \Pi)$) has the following capabilities:

- (A) If $\mathcal{A}$ decides to corrupt at least one new player during the execution of any value.
- (B) If at the beginning of $\mathcal{A}$, the set of corrupted players is B and $\mathcal{A}$ decides not to corrupt new players during Π. The resulting coin x_i is at most $\Delta_\Pi(B)$-biased, but $\mathcal{A}$ can set the probability of $x_i = 0$ anywhere in the interval $[\frac{1}{2} \cdot (1 - \Delta_\Pi(B)), \frac{1}{2} \cdot (1 + \Delta_\Pi(B))]$.

Theorem 10 *For any family of coin-flipping protocols Π, there is no black-box transformation resulting in an adaptively $\omega(\sqrt{n})$-resilient family of protocols $\Phi(N, f, \Pi)$.*

Proof: Let us assume that, $\Phi(N, f, \Pi)$ is adaptively $2b(n)$-resilient. We construct the following $2b(n)$-bounded adversary for Φ satisfying properties (A) and (B). Let $b = b(n)$ and $\gamma = \Delta_\Pi(b)$ and let B be the set of players of cardinality achieving $\Delta_\Pi(B) = \Delta_\Pi(b) = \gamma$. Before Π_1 starts, $\mathcal{A}$ corrupts all the players in B. Therefore, from now on in each of the N invocations of Π, $\mathcal{A}$ can set the 0-probability of x_i anywhere in at least the interval $[\frac{1}{2} \cdot (1-\gamma), \frac{1}{2} \cdot (1+\gamma)]$. As $\mathcal{A}$ will later corrupt more players, this interval can only expand, but our particular $\mathcal{A}$ will not use it. If $\mathcal{A}$ decides to follow rule (A), he will corrupt a single player and set the corresponding bit x_i to the value he wants. Therefore, since Φ claims to be $2b$-resilient, $\mathcal{A}$ an use rule (A) exactly b times. Hence, now we exactly reduced the possible behavior of $\mathcal{A}$ to an arbitrary (γ, b, N)-BCL source.

From the upper-bound of Ben-Or and Linial [5], we know that $\Delta_\Pi(b) \geq \Omega(b/n)$. Thus, $b\gamma = \Omega(b^2/n)$. By Theorem 6, it is impossible to extract a slightly random bit whenever $b^2/n = \omega(1)$, i.e. $b = \omega(\sqrt{n})$. □

References

[1] Yevgeniy Dodis, Adriana Lpez-Alt, Ilya Mironov and Salil P. Vadhan. Differential Privacy with Imperfect Randomness. In *CRYPTO 2012: 497-516*

[2] Yevgeniy Dodis: New Imperfect Random Source with Applications to Coin-Flipping. In *ICALP 2001*

[3] D. Lichtenstein, N. Linial, M. Saks. Some Extremal Problems Arising from Discrete Control Processes. In *Combinatorica, 9:269287, 1989.*

[4] M. Santha, U. Vazirani. Generating Quasi-Random Sequences from Semi-Random Sources. In *Journal of Computer and System Sciences, 33(1):7587, 1986.*

[5] M. Ben-Or, N. Linial. Collective Coin-Flipping. In *Slivio Micali, editor, Randomness and Computation, pp. 91-115, Academic Press, New York, 1990.*