Exponential Separations In Local Differential Privacy

Matthew Joseph

Jieming Mao

Aaron Roth

Problem

Problem

Surgeon General Jerome Adams

Problem

Surgeon General Jerome Adams

“How many Americans have used a schedule-I drug?

Problem

Surgeon General Jerome Adams

“How many Americans have used a schedule-I drug?

People are probably reluctant to tell the federal government honestly…

Problem

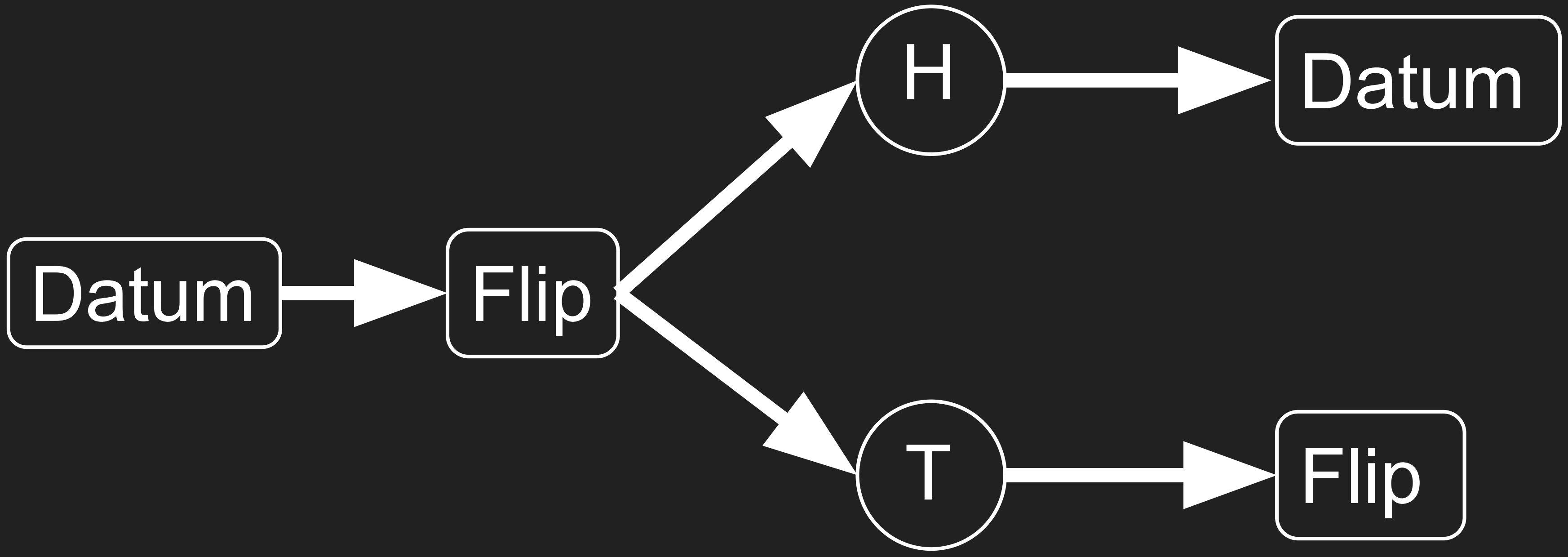
Surgeon General Jerome Adams

“How many Americans have used a schedule-I drug?

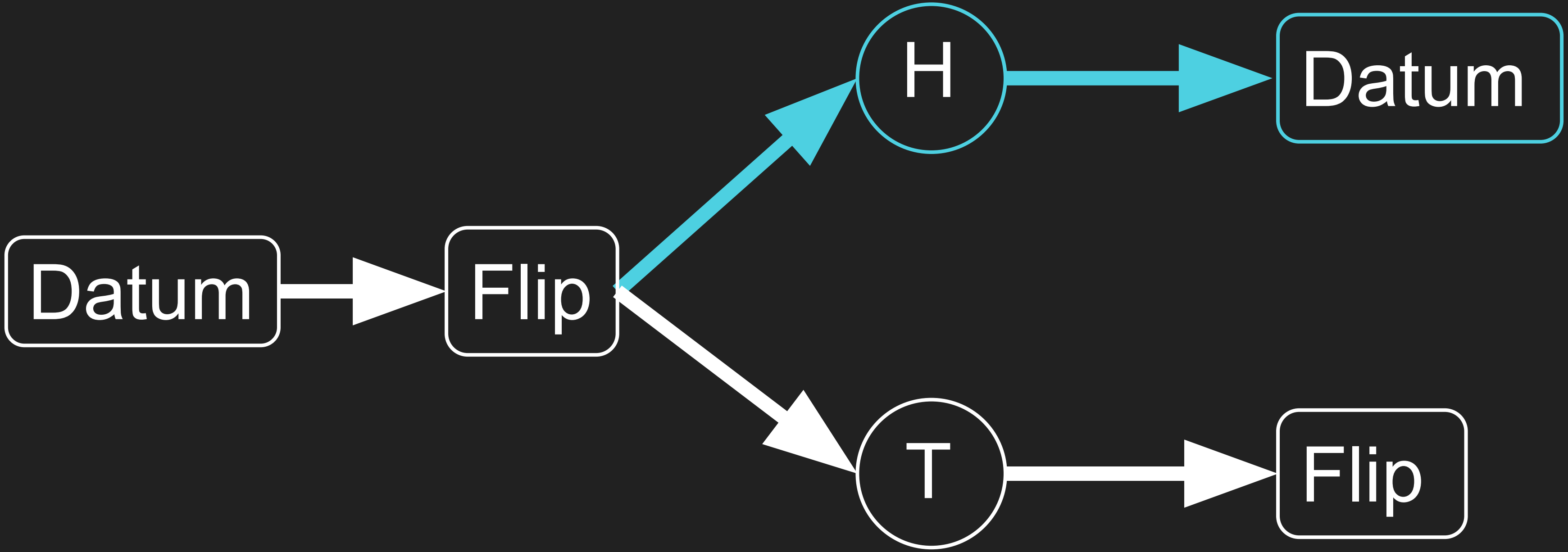
People are probably reluctant to tell the federal government honestly…

...so how can I get an accurate answer while guaranteeing plausible deniability for everyone?”

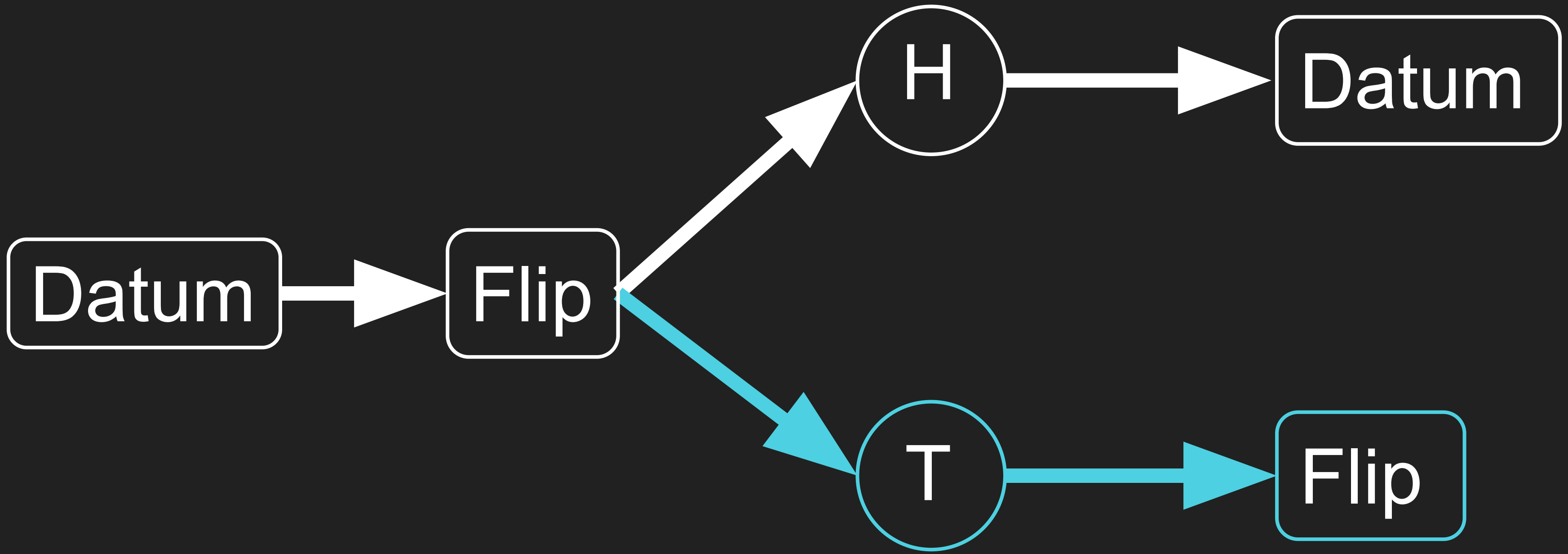
Solution: Randomized Response [W65]



Solution: Randomized Response [W65]

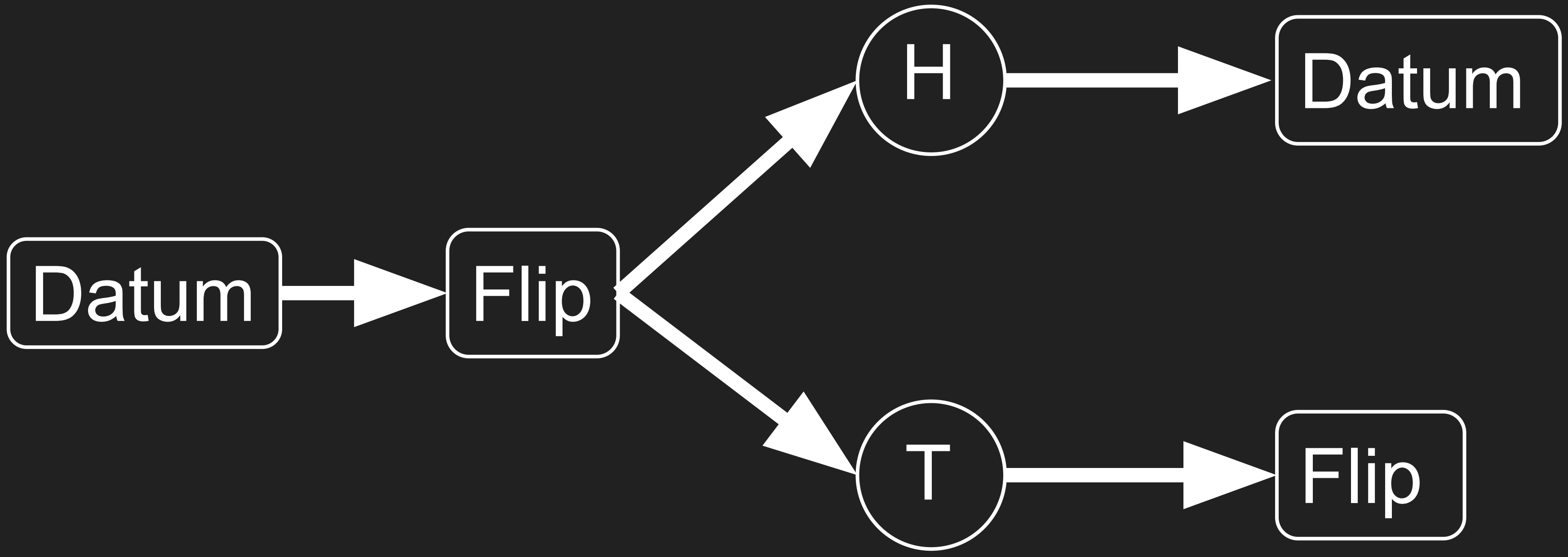


Solution: Randomized Response [W65]



Solution: Randomized Response [W65]

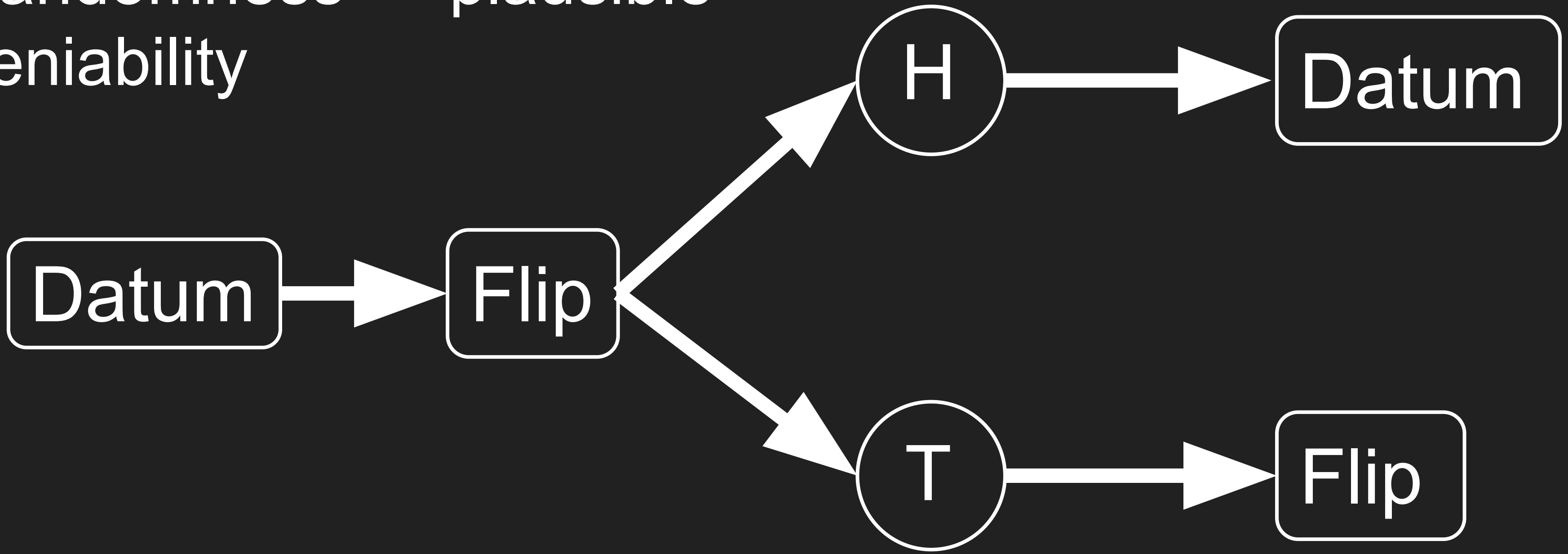
$O(\sqrt{\text{\# responses}})$ accuracy



Solution: Randomized Response [W65]

$O(\sqrt{\#\ \text{responses}})$ accuracy

Randomness → plausible deniability



“Sounds good! Let’s do that.”

Surgeon General
Jerome Adams

“Sounds good! Let’s do that.”

Surgeon General
Jerome Adams

“Or maybe we can do better if we ask many questions?

First ask person one q, then use the answer to ask person second q, and so on.”

Dep. Surgeon General
Erica Schwartz

“Sounds good! Let’s do that.”

Surgeon General
Jerome Adams

“Or maybe we can do better if we ask many questions?

First ask person one q, then use the answer to ask person second q, and so on.”

Dep. Surgeon General
Erica Schwartz

“Sounds cumbersome! We need **proof** that the extra effort is worth it first.”

This Talk

Prove **adaptive** questioning with **plausible deniability** is **worth it**.

This Talk

Prove **adaptive** questioning with **plausible deniability** is **worth it**.

≈

Construct problem where we can prove **fully interactive locally differentially private** protocols get **much better sample complexity** than **sequentially interactive** ones.

Outline

1. Preliminaries

2. Tool: LDP ≈ Noisy Communication

3. Application: Exponential Separation

Outline

1. **Preliminaries**

2. Tool: LDP ≈ Noisy Communication

3. Application: Exponential Separation

Local Differential Privacy (LDP) [DMNS06]

Each user has their own private datum

Protocol $\mathcal{A}$ learns about the data through public communication with users

Users send messages through *randomizers* $\boldsymbol{R}$

Randomness ensures privacy

LDP in Math

Definition: Protocol $\boldsymbol{A}$ is (ε,δ)-locally differentially private (LDP) if the transcript of communications it generates is an (ε,δ)-DP function of the user data.

For neighboring distributed databases X and X',

$$\mathrm{P}[T(X) \text{ in } Y] \leq e^{\varepsilon}\mathrm{P}[\mathrm{T}(X') \text{ in } Y] + \delta$$

LDP: Pros and Cons

Pros:

✓ Data never leaves user device, only DP outputs

✓ Don’t have to store any private data

LDP: Pros and Cons

Pros:

✓ Data never leaves user device, only DP outputs

✓ Don’t have to store any private data

Cons:

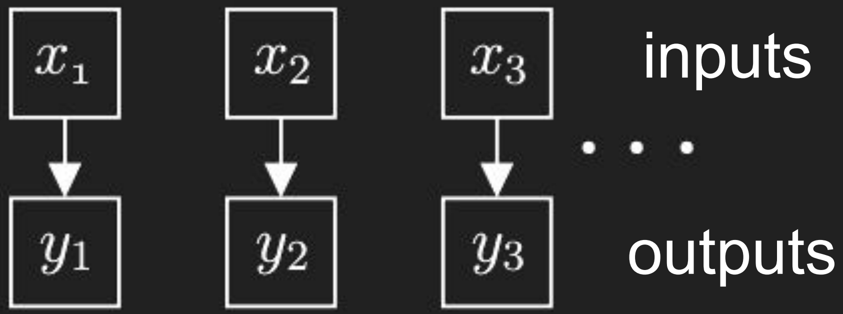
✗ More noise → worse utility

✗ Don’t get to store any private data

Types of LDP Interactivity

Definition: Protocol A is *noninteractive* if all users speak once, simultaneously and independently.

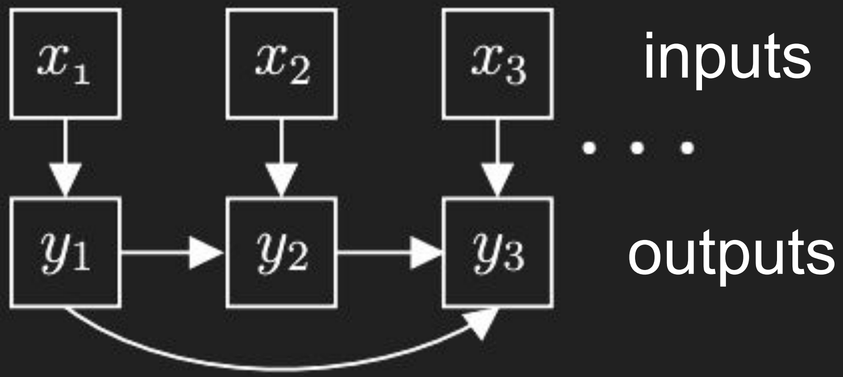
Make all randomizer assignments beforehand.



Types of LDP Interactivity

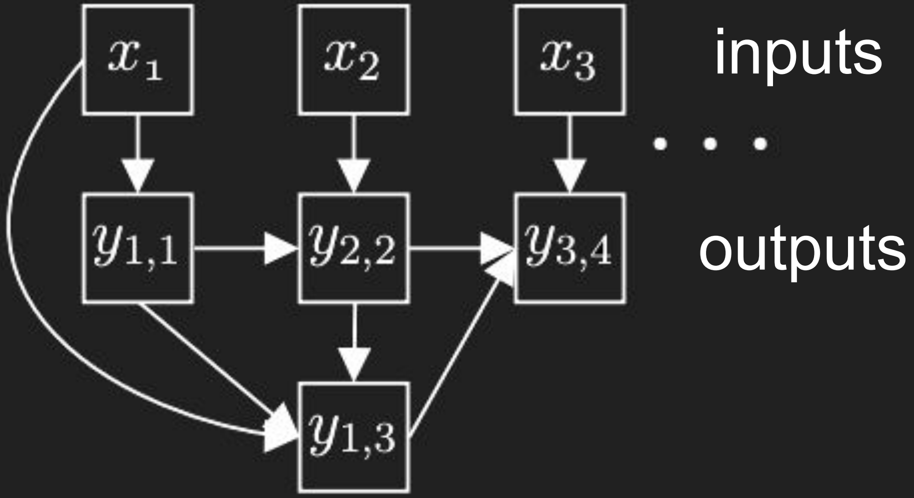
Definition: Protocol A is *sequentially interactive* [DJW13] if all users speak once (possibly in multiple rounds).

Make randomizer assignments adaptively.



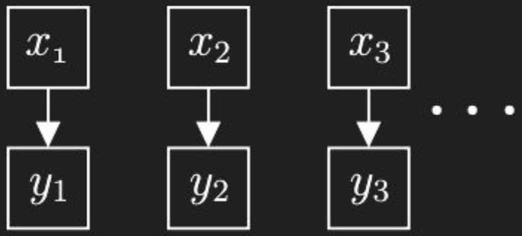
Types of LDP Interactivity

Definition: Protocol $\mathcal{A}$ is *fully interactive* if users may interact arbitrarily (possibly speak multiple times, in multiple rounds).

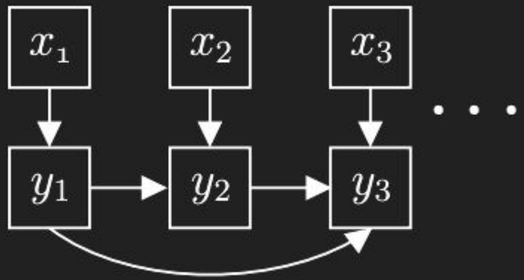


Types of LDP Interactivity

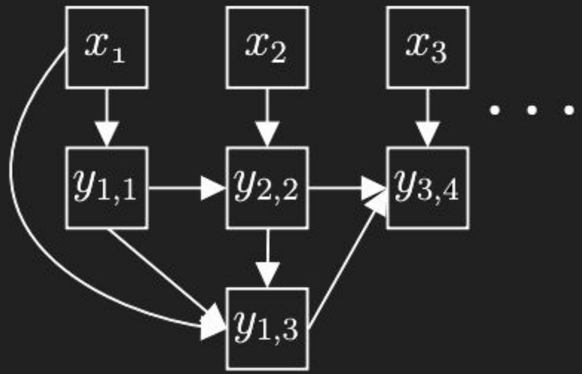
Noninteractive



Sequentially Interactive

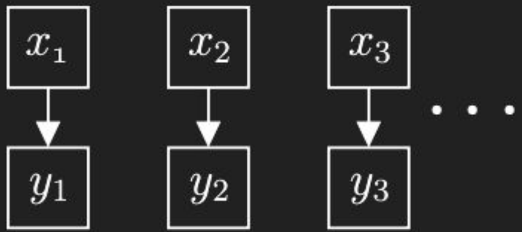


Fully Interactive



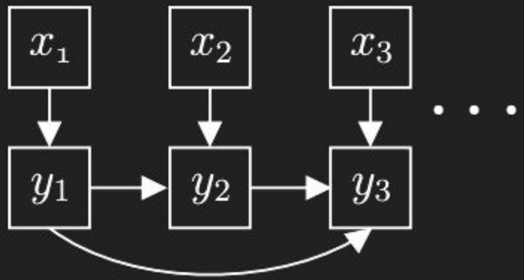
Types of LDP Interactivity

Noninteractive



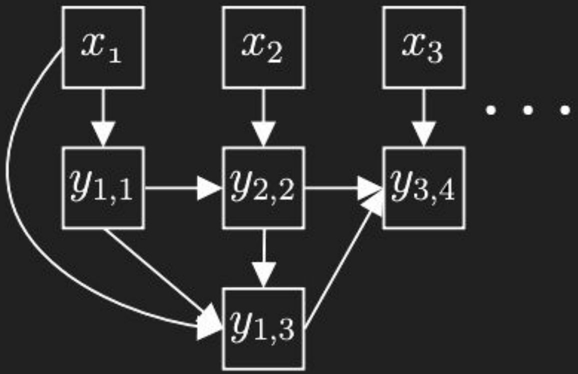
rounds = 1

Sequentially Interactive



rounds ≤ # users

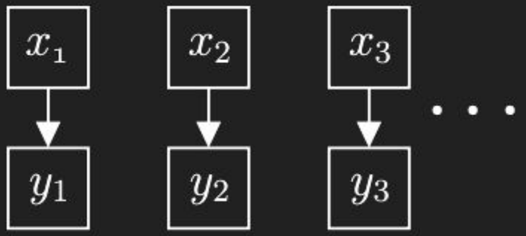
Fully Interactive



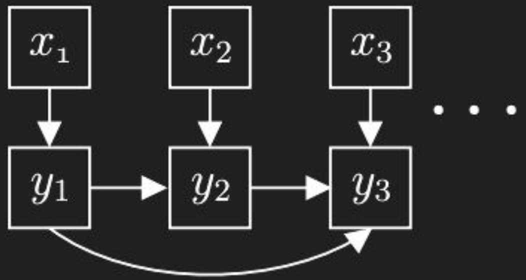
rounds = ???

Types of LDP Interactivity

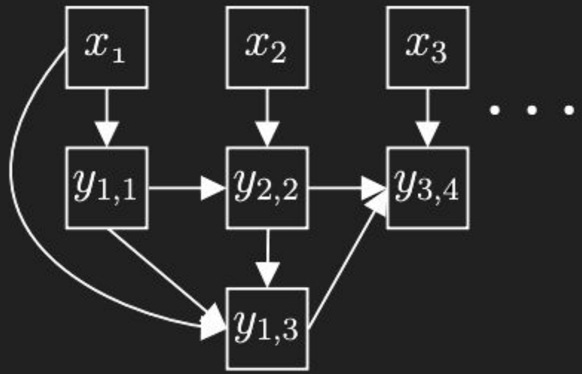
Noninteractive



Sequentially Interactive



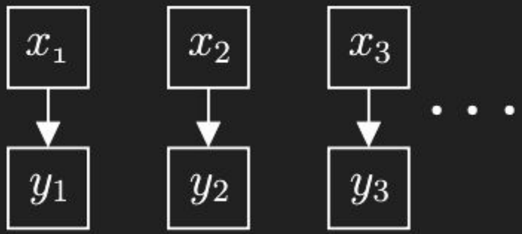
Fully Interactive



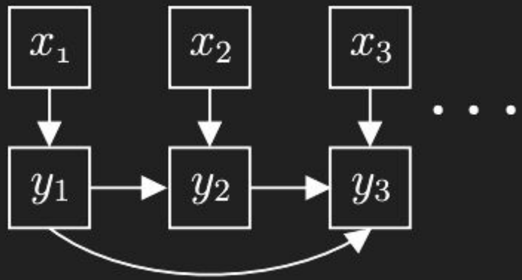
[KLNRS08]
[DF19]

Types of LDP Interactivity

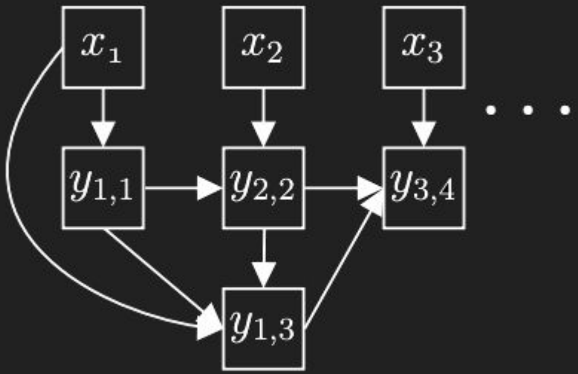
Noninteractive



Sequentially Interactive



Fully Interactive



[KLNRS08]
[DF18]

This Work

Local Differential Privacy

Outline

1. Preliminaries

2. **Tool: LDP ≈ Noisy Communication**

3. Application: Exponential Separation

Tool: LDP ≈ Noisy Communication

General connection between two-party communication complexity (**CC**) and multi-party locally private sample complexity (**SC**).

Tool: LDP ≈ Noisy Communication

General connection between two-party communication complexity (**CC**) and multi-party locally private sample complexity (**SC**).

Two-party problem: Alice has input **a**, Bob has input **b**, want to compute some function of **a** and **b**.

Tool: LDP ≈ Noisy Communication

General connection between two-party communication complexity (**CC**) and multi-party locally private sample complexity (**SC**).

Two-party problem: Alice has input **a**, Bob has input **b**, want to compute some function of **a** and **b**.

Multi-party problem: each user randomly gets **a** or **b**, want to compute some function of **a** and **b**.

Tool: LDP ≈ Noisy Communication

Theorem: Given two-party problem P_2 and multi-party analogue P_m, for $\varepsilon = O(1)$, $SC^{\varepsilon,S}(P_m) = \Theta(CC(P_2)/\varepsilon^2)$.

Tool: LDP ≈ Noisy Communication

Theorem: Given two-party problem P_2 and multi-party analogue P_m, for $\varepsilon = O(1)$, $\boxed{SC^{\varepsilon,S}(P_m)} = \Theta(CC(P_2)/\varepsilon^2)$.

Tool: LDP ≈ Noisy Communication

Theorem: Given two-party problem P_2 and multi-party analogue P_m, for $\varepsilon = O(1)$, $SC^{\varepsilon,S}(P_m) = \Theta(\boxed{CC(P_2)}/\varepsilon^2)$.

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

$CC(P_2)$ $\longleftrightarrow$ ε'-noisy $CC(P_2)$ $\longleftrightarrow$ 1-bit $SC^{\varepsilon,S}(P_m)$ $\longleftrightarrow$ $SC^{\varepsilon,S}(P_m)$

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

1. Connect **CC** of noiseless and **CC** of **ε'**-noisy two-party communication problems [BR14, BM15]

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

1. Connect **CC** of noiseless and **CC** of **ε'**-noisy two-party communication problems [BR14, BM15]
2. Connect **CC** of **ε'**-noisy two-party and **SC** of 1-bit-per-person sequentially interactive **ε**-locally private multi-party communication problems

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\rightarrow$

* $P_m^{\varepsilon,S,1} \rightarrow P_2^{\varepsilon',\text{noisy}}$: Alice and Bob randomly partition users in $P_m^{\varepsilon,S,1}$ between them

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\rightarrow$

* $P_m^{\varepsilon,S,1} \rightarrow P_2^{\varepsilon',\mathrm{noisy}}$: Alice and Bob randomly partition users in $P_m^{\varepsilon,S,1}$ between them

* for each bit in $P_m^{\varepsilon,S,1}$ Alice or Bob sends bit with probabilities calibrated to ε' and current randomizer

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\rightarrow$

* $P_m^{\varepsilon,S,1} \rightarrow P_2^{\varepsilon',\mathrm{noisy}}$: Alice and Bob randomly partition users in $P_m^{\varepsilon,S,1}$ between them
* for each bit in $P_m^{\varepsilon,S,1}$ Alice or Bob sends bit with probabilities calibrated to ε' and current randomizer
* # bits for $P_2^{\varepsilon',\mathrm{noisy}}$ = # users for $P_m^{\varepsilon,S,1}$

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\rightarrow$

* $P_m^{\varepsilon,S,1} \rightarrow P_2^{\varepsilon',\text{noisy}}$: Alice and Bob randomly partition users in $P_m^{\varepsilon,S,1}$ between them
* for each bit in $P_m^{\varepsilon,S,1}$ Alice or Bob sends bit with probabilities calibrated to ε' and current randomizer
* # bits for $P_2^{\varepsilon',\text{noisy}}$ = # users for $P_m^{\varepsilon,S,1}$

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\rightarrow$

uses $SC^{\varepsilon,S}$ = # randomizer calls

may fail for $SC^{\varepsilon,F}$

* $P_m^{\varepsilon,S,1} \rightarrow P_2^{\varepsilon',noisy}$: Alice and Bob randomly partition users in $P_m^{\varepsilon,S,1}$ between them

* for each bit in $P_m^{\varepsilon,S,1}$ Alice or Bob sends bit with probabilities calibrated to ε' and current randomizer

* # bits for $P_2^{\varepsilon',noisy}$ = # users for $P_m^{\varepsilon,S,1}$

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\leftarrow$

* $P_m^{\varepsilon,S,1} \leftarrow P_2^{\varepsilon',\mathrm{noisy}}$: for each bit in $P_2^{\varepsilon',\mathrm{noisy}}$, draw a new user

Tool: LDP ≈ Noisy Communication

Proof Sketch

2.: $\leftarrow$

* $P_m^{\varepsilon,S,1} \leftarrow P_2^{\varepsilon',noisy}$: for each bit in $P_2^{\varepsilon',noisy}$, draw a new user

* user sends bit through ε-RR if correct of Alice and Bob otherwise uniform random

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

1. Connect **CC** of noiseless and **CC** of **ε'**-noisy two-party communication problems [BR14, BM15]
2. Connect **CC** of **ε'**-noisy two-party and **SC** of 1-bit-per-person sequentially interactive **ε**-locally private multi-party communication problems

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

1. Connect **CC** of noiseless and **CC** of **ε'**-noisy two-party communication problems [BR14, BM15]
2. Connect **CC** of **ε'**-noisy two-party and **SC** of 1-bit-per-person sequentially interactive **ε**-locally private multi-party communication problems
3. Connect **SC** of 1-bit-per-person and **SC** of generic sequentially interactive **ε**-locally private multi-party communication problems [BS15]

Tool: LDP ≈ Noisy Communication

Proof Sketch

Three parts:

$CC(P_2)$ $\longleftrightarrow$ ε'-noisy $CC(P_2)$ $\longleftrightarrow$ 1-bit $SC^{\varepsilon,S}(P_m)$ $\longleftrightarrow$ $SC^{\varepsilon,S}(P_m)$

Tool: LDP ≈ Noisy Communication

Theorem: Given two-party problem P_2 and multi-party analogue P_m, for $\varepsilon = O(1)$, $SC^{\varepsilon,S}(P_m) = \Theta(CC(P_2)/\varepsilon^2)$.

Tool: LDP ≈ Noisy Communication

Theorem: Given two-party problem P_2 and multi-party analogue P_m, for $\varepsilon = O(1)$, $SC^{\varepsilon,S}(P_m) = \Theta(CC(P_2)/\varepsilon^2)$.

Can now get multi-party $SC^{\varepsilon,S}$ lower bounds straight from two-party CC lower bounds.

Outline

1. Prelims
2. Tool: LDP ≈ Noisy Communication
3. **Application: Exponential Separation**

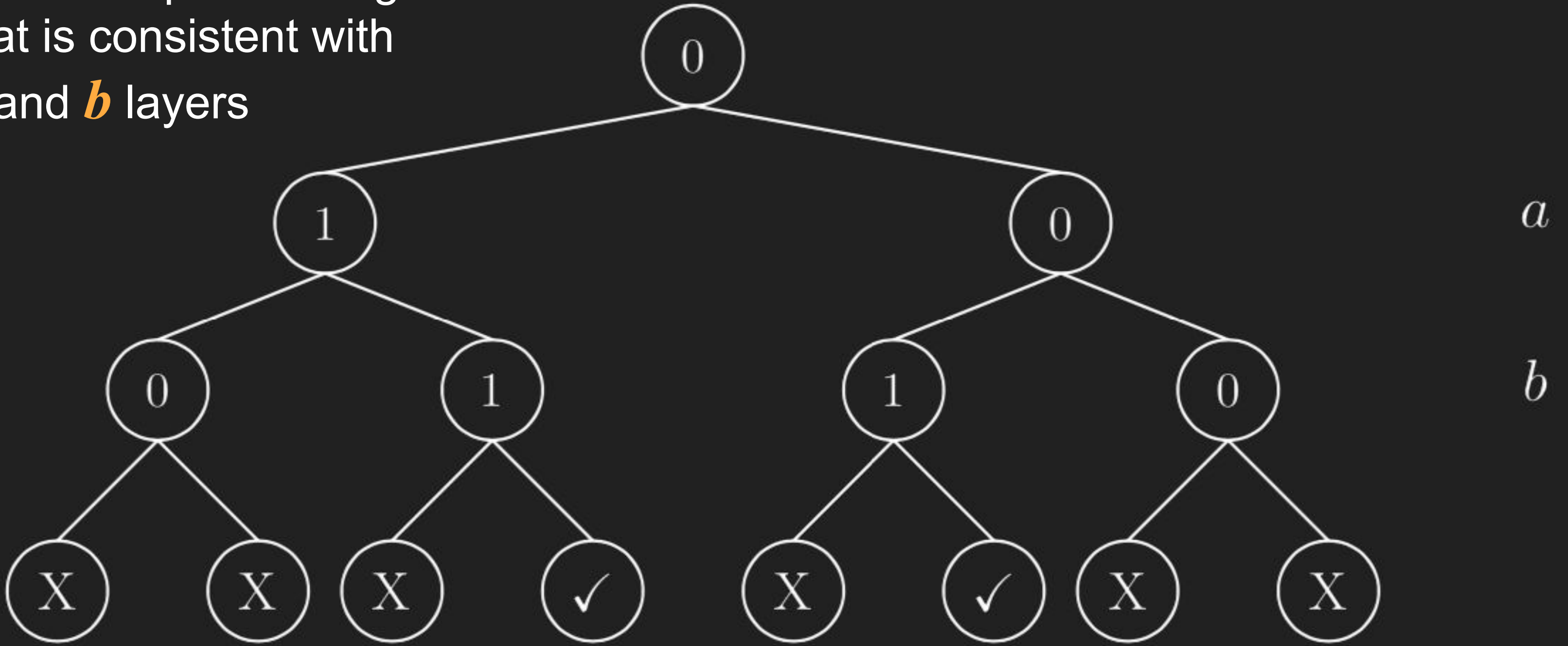
Application: Exponential Separation

Useful because two-party **CC** lower bounds are well-studied.

Lemma [GKR16]: Solving the two-party *hidden layers* problem requires $\mathbf{CC = \Omega(2^k)}$.

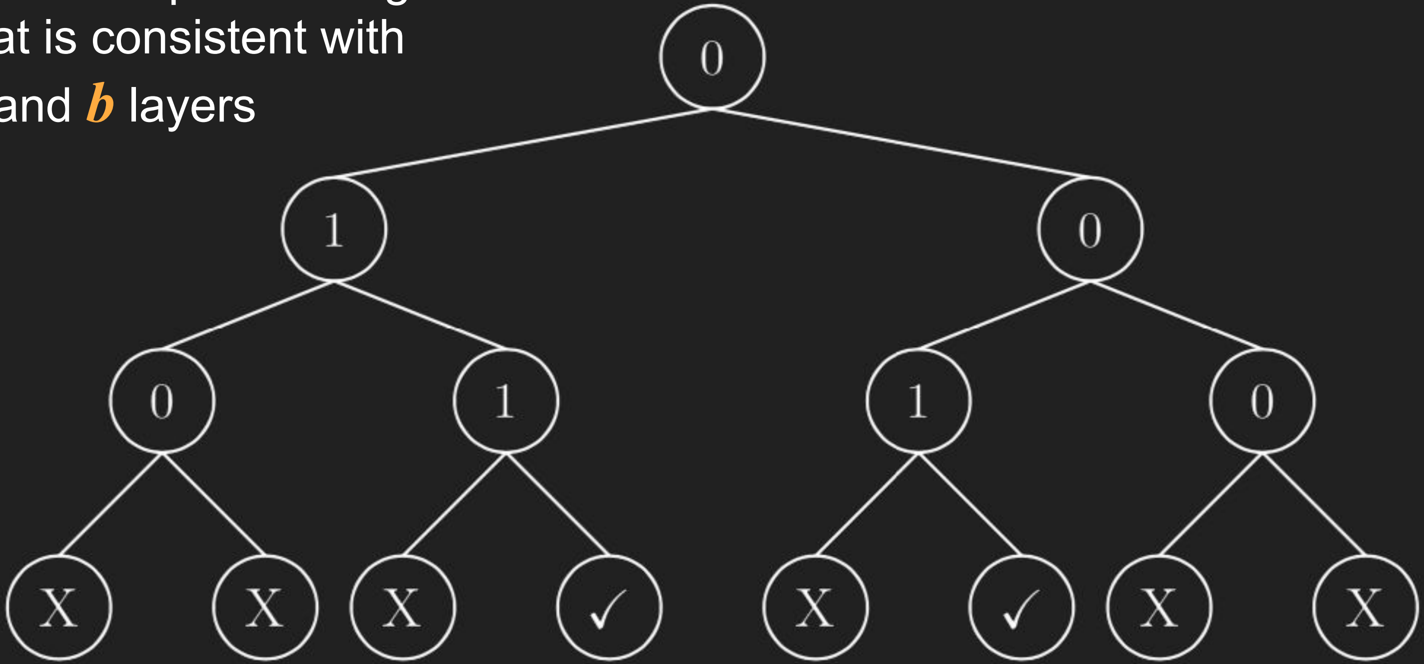
Application: Exponential Separation

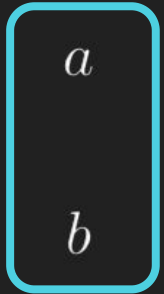
Goal: find path through tree that is consistent with a and b layers



Application: Exponential Separation

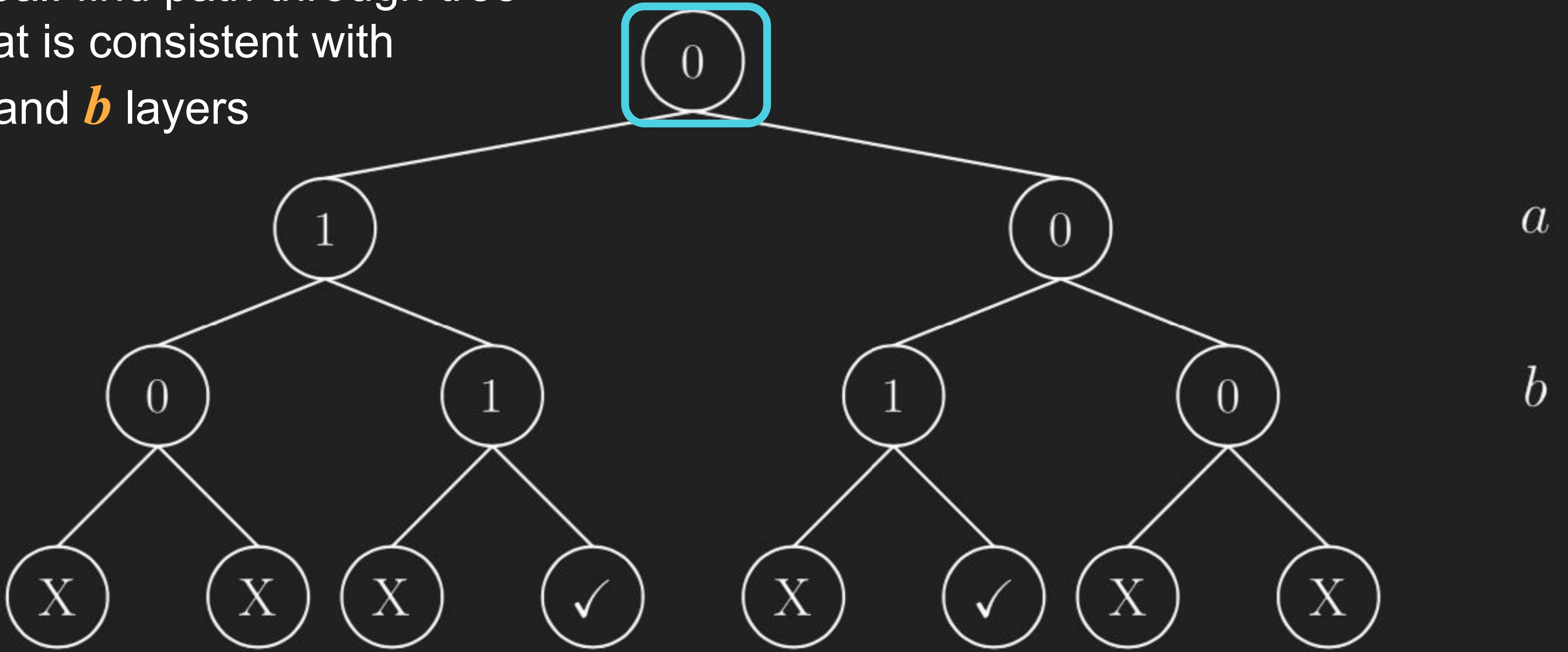
Goal: find path through tree that is consistent with a and b layers





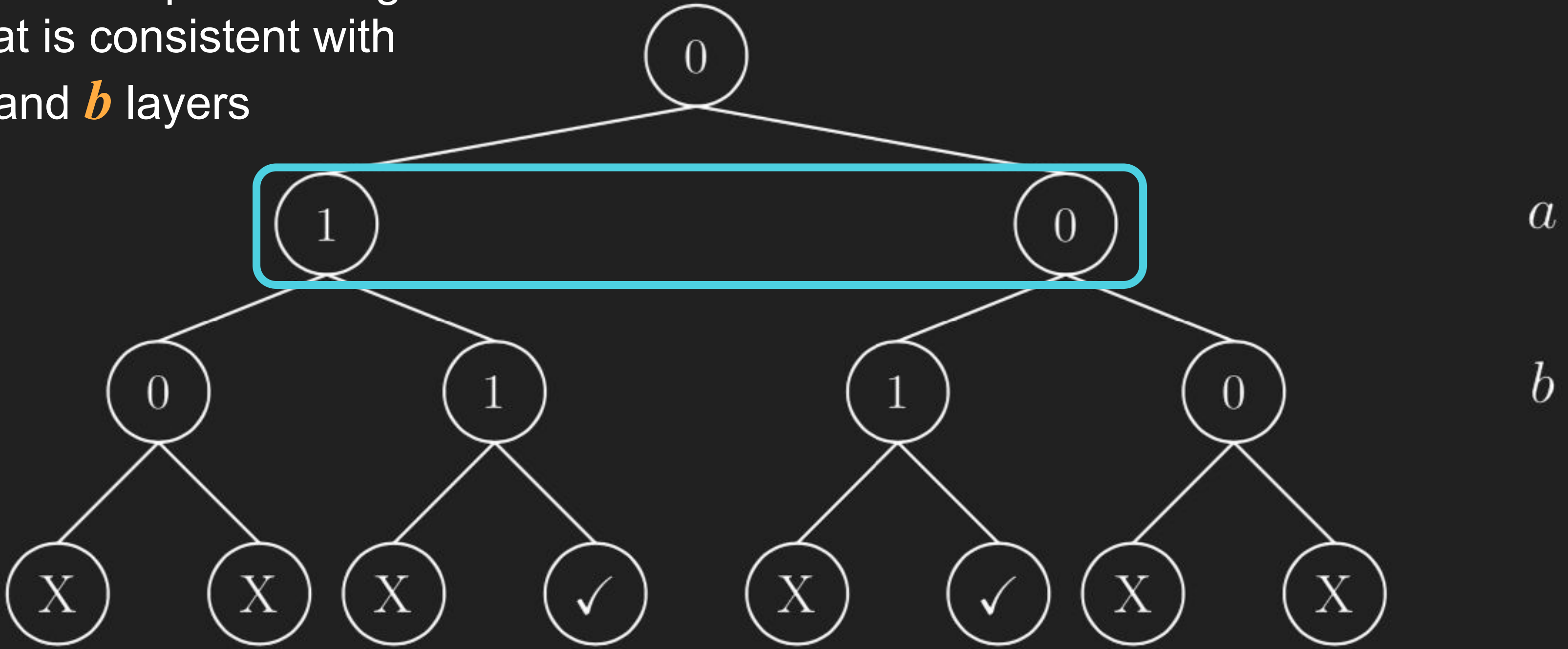
Application: Exponential Separation

Goal: find path through tree that is consistent with a and b layers



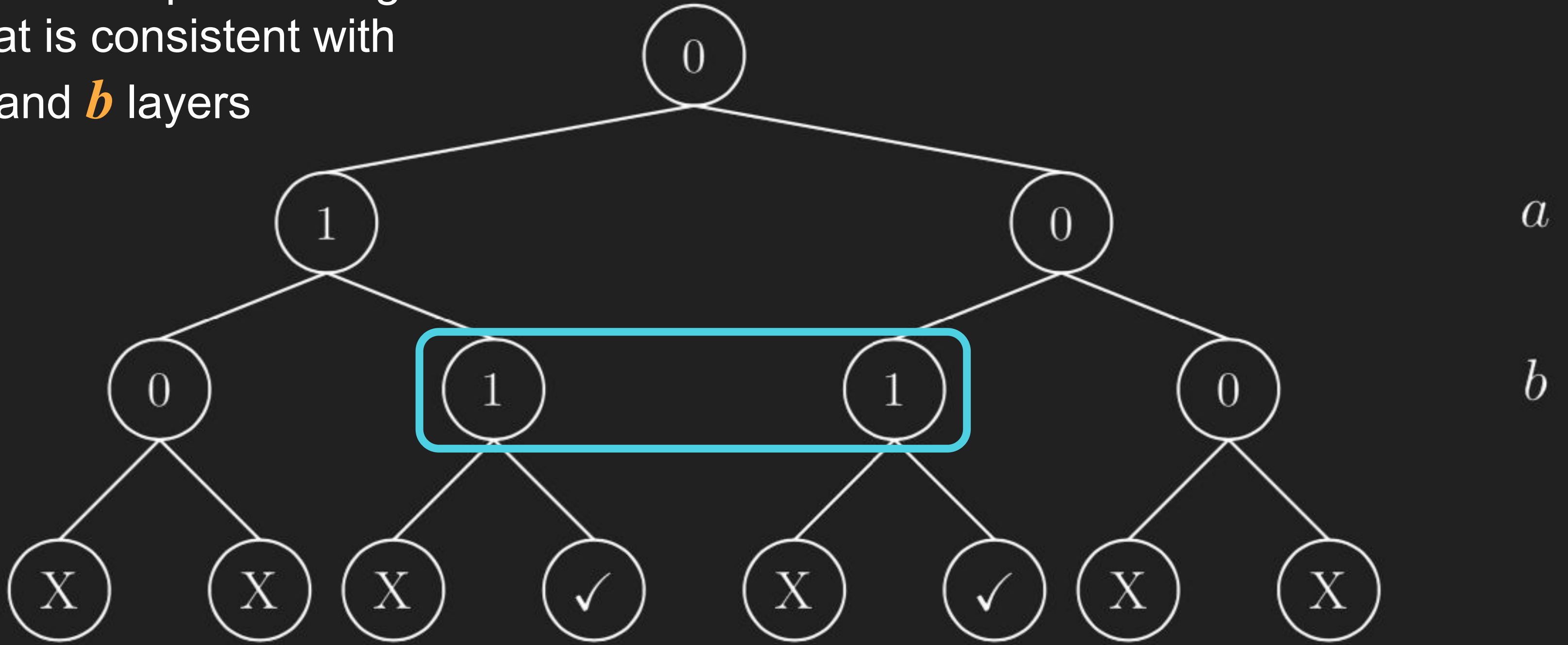
Application: Exponential Separation

Goal: find path through tree that is consistent with a and b layers



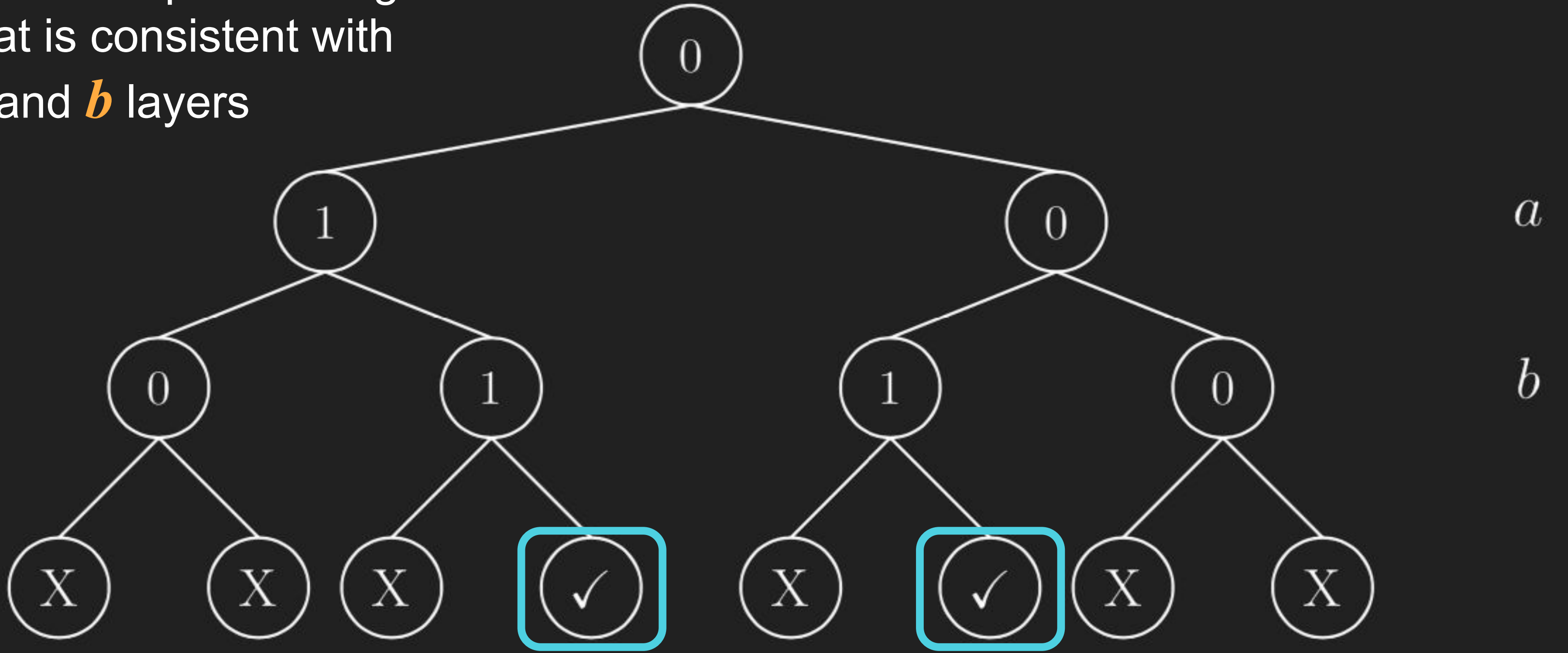
Application: Exponential Separation

Goal: find path through tree that is consistent with a and b layers



Application: Exponential Separation

Goal: find path through tree that is consistent with ***a*** and ***b*** layers

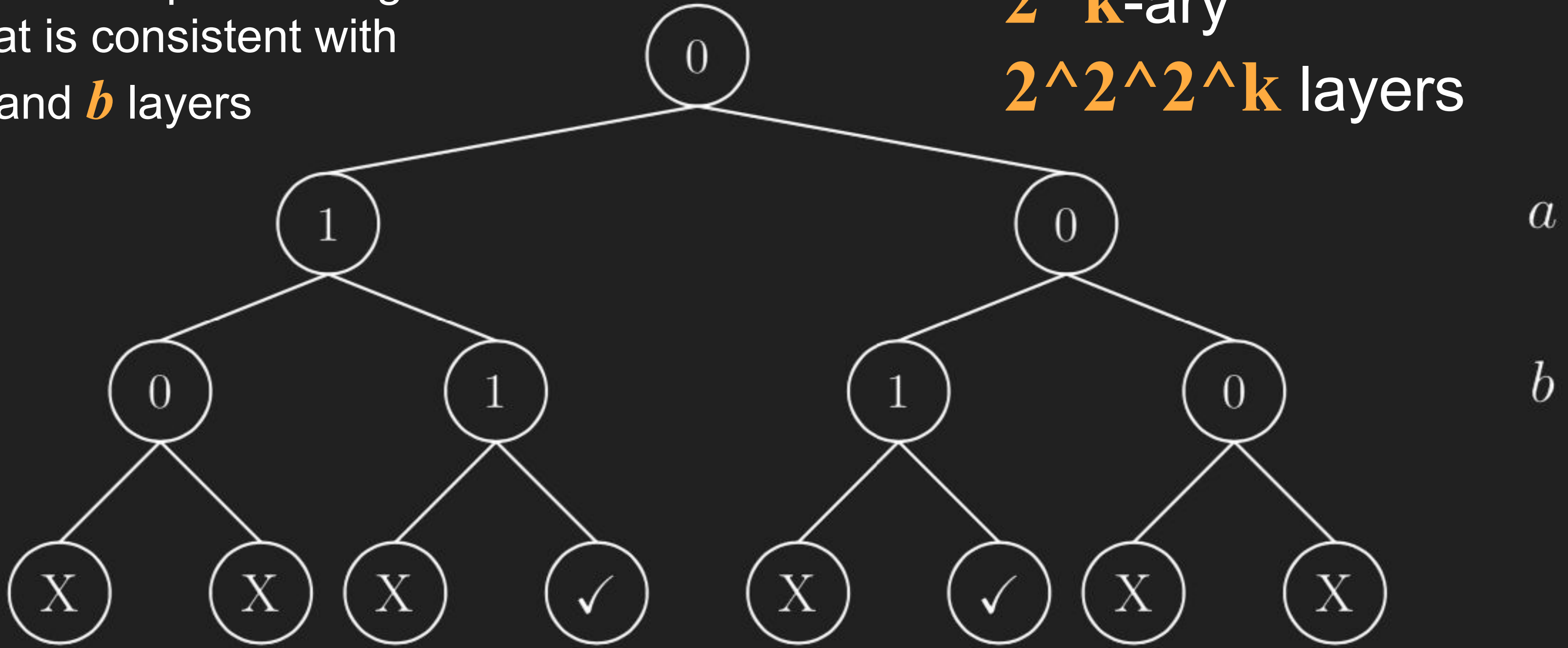


Application: Exponential Separation

Goal: find path through tree that is consistent with ***a*** and ***b*** layers

2^k-ary

2^2^2^k layers



Application: Exponential Separation

Useful because two-party **CC** lower bounds are well-studied.

Lemma [GKR16]: Solving the two-party *hidden layers* problem requires $\mathbf{CC} = \Omega(2^k)$.

Corollary: For P_m = multi-party hidden layers problem, $\mathbf{SC}^{\varepsilon,S}(P_m) = \Omega(2^k/\varepsilon^2)$.

Application: Exponential Separation

Useful because two-party **CC** lower bounds are well-studied.

Lemma [GKR16]: Solving the two-party *hidden layers* problem requires $\mathbf{CC} = \Omega(2^k)$.

Corollary: For P_m = multi-party hidden layers problem, $\mathbf{SC}^{\varepsilon,S}(P_m) = \Omega(2^k/\varepsilon^2)$. But $\mathbf{SC}^{\varepsilon,F}(P_m) = O(k/\varepsilon^2)$.

Application: Exponential Separation

Useful because two-party $\mathbf{CC}$ lower bounds are well-studied.

Lemma [GKR16]: Solving the two-party *hidden layers* problem requires $\mathbf{CC} = \Omega(2^k)$.

Corollary: For P_m = multi-party hidden layers problem, $SC^{\varepsilon,S}(P_m) = \Omega(2^k/\varepsilon^2)$. But $SC^{\varepsilon,F}(P_m) = O(k/\varepsilon^2)$.

(At each node, for all 2^k possible next nodes, ask all users if correct. Can handle 2^k by union bound on RR accuracy.)

Surgeon General
Jerome Adams

“The extra effort is worth it if we’re trying to solve the hidden layers problem!

Dep. Surgeon General
Erica Schwartz

Surgeon General
Jerome Adams

“The extra effort is worth it if we’re trying to solve the hidden layers problem!

And we can prove it using a general connection between local differential privacy and communication complexity.”

Dep. Surgeon General
Erica Schwartz

Surgeon General
Jerome Adams

"The extra effort is worth it if we're trying to solve the hidden layers problem!

And we can prove it using a general connection between local differential privacy and communication complexity."

"Great! We'll keep that in mind if we ever need to solve the hidden layers problem."

Dep. Surgeon General
Erica Schwartz

Open Questions

- How large can the gap between SI and FI be?

Open Questions

- How large can the gap between SI and FI be?

- Separation for “natural” problem?

Open Questions

- How large can the gap between SI and FI be?
- Separation for “natural” problem?
- How powerful are FI protocols?

Open Questions

- How large can the gap between SI and FI be?

- Separation for “natural” problem?

- How powerful are FI protocols?

arxiv.org/abs/1907.00813

References

1. [BM15] “Simulating Noisy Channel Interaction”. Braverman, Mao. ITCS.
2. [BR14] “Toward Coding for Maximum Errors in Interactive Communication”. Braverman, Rao.Transactions on Information Theory.
3. [BS15] “Local, Private, Efficient Protocols for Succinct Histograms”. Bassily, Smith. STOC.
4. [DF19] “Learning without Interaction Requires Separation”. Daniely and Feldman. NeurIPS.
5. [DMNS06] “Calibrating Noise to Sensitivity in Private Data Analysis”. Dwork, Mcsherry, Nissim, Smith. TCC.
6. [GKR16] “Exponential Separation of Communication and External Information”. Ganor, Kol, Raz. STOC.
7. [JMNR19] “The Role of Interactivity in Local Differential Privacy”. Joseph, Mao, Neel, Roth. FOCS.
8. [KLNRS08] “What Can We Learn Privately?”. Kasiviswanathan, Lee, Nissim, Raskhodnikova, Smith. STOC.
9. [W65] "Randomised Response: A Survey Technique for Eliminating Evasive Answer Bias". Warner. JASA.