Tight Bounds for Private Graph Data Release

Rasmus Pagh and Lukas Retschmeier

Jan 2025

1 Introduction

Privately releasing the set of edges of a *Minimum-Weight Spanning Trees*, *Shortest Path Trees* or a *Minimum-Weight Matching* under edge-weight differential privacy has been introduced by Sealfon [2016]. In this setting, a graph topology $G = (V, E)$ is publicly known and we want to protect the weights on each individual edge. It is natural to consider the ℓ_1 or the ℓ_∞ neighboring relationship, where neighboring datasets are bounded by how much the weights can change.

While recently, there has been much focus on privately releasing an MST, tight bounds on the other problems are less understood. While we know, that all of them can be solved by simply perturbating the input, tightness results are only known up to logarithmic factors.

Interestingly, for the MST problem, it has been shown that a simply input perturbation together with privately running *any* MST algorithms yields exactly the same utility as all previous techniques.

1.1 Previous Work

We will shortly survey the state of the problems individually.

Minimum Spanning Trees (MST). Tight bounds for pure DP and both the ℓ_1 and ℓ_∞ neighboring relationship are known, but for approximate DP we only have tight bounds for the ℓ_∞ relationship. This work solved the gap by providing tight bounds for this case as well.

Minimum-Weight Perfect Matchings (MWPM). We are unaware of any other work releasing the set of edges of a MWPM other then the original Sealfon [2016]. There has been some work on private matching like in **?** or more recently in , but those consider a different setting [Lukas: TODO. Describe this setting and check. the STOC paper is actually before Sealfon!]

Shortest Path Trees (SPT). Again, releasing the edges of a SPT has not been subject to study after introduced by Sealfon [2016] in 2016.

1.2 Releasing Minimium Spanning tree under ℓ_1 and (ϵ, δ)-DP

1.3 Our Contributions

We generalize a lower bound due to Sealfon and construct a hard, dense mst instance.

Short Preliminaries and Notation That is for all neighboring vectors $\mathbf{X} \sim_H \mathbf{X}' \in [r]^d$ if their hamming distance is at most 1. We write $w_{i,j}$ to denote the weight of w_e for an edge $e = \{i, j\}$. If not clear from the context, we subscript the neighborhood relationship and write $\sim_H$ for the hamming and $\sim_{\ell_1}$ for the ℓ_1 neighborhood.

We use the operator $\in_R \colon 2^A \to A$ and denote $X \in_R A$ is drawn uniformly at random from the set A.

[Rasmus: Maybe abstract to vectors in $[r]^d$ for integers r and d; I think we want Hamming neighborhood, i.e., differ in one entry; maybe mention the neighborhood relation in the theorem statement]

1.4 MSTs under ℓ_1 and approximate DP

We start by generalizing a proof technique that appeared in Sealfon [2016] and bound the probability that any (ϵ, δ)-DP mechanism B can leak some of the input bits.

[Lukas: Do we rather need a $B : [n]^d \to [n]^d$?]

[Lukas: Observation: Could we already use the ℓ_1 neighborhood here and just set the neighboring $X_i + -1$. Then we would get away with some ℓ_1 assumption.]

Theorem 1.1. *Assuming $n > 1$, if algorithm $B : [n]^d \to [n]$ is (ϵ, δ)-dp under the hamming neighborhood, then for $\mathbf{X} \in_R [n]^d$ drawn uniformly at random, we get* $\Pr[B(\mathbf{X}) = \mathbf{X_i}] \leq \frac{e^\epsilon}{n} + \delta$.

Proof. First fix some coordinate $i \in [d]$ and assume that $\mathbf{X} = (\mathbf{X}_{<i}, X_i, \mathbf{X}_{>i}) \in_R [n]^d$ is uniformly drawn. Then we can bound the probability that $B(\mathbf{X})$ outputs

some $\mathbf{X}_i$ by fixing entries that are not X_i: [Lukas: Fix -d]

$$Pr[B(\mathbf{X}) = \mathbf{X}_i] = \frac{1}{n^{d-1}} \sum_{\mathbf{X}_{<i} \in [n]^{i-1}} \sum_{\mathbf{X}_{>i} \in [n]^{d-i}} \left(\Pr_{X_i \in_R [n]} [B(\mathbf{X}_{<i}, X_i, \mathbf{X}_{>i}) = X_i] \right) \tag{1}$$

$$\leq n^{1-d} \sum_{\mathbf{X}_{<i} \in [n]^{i-1}} \sum_{\mathbf{X}_{>i} \in [n]^{d-i}} \left(e^{\epsilon} \Pr_{X_i \in_R [n]} [B(\mathbf{X}_{<i}, 1, \mathbf{X}_{>i}) = X_i] + \delta \right] \tag{2}$$

$$\leq n^{1-d} \sum_{\mathbf{X}_{<i} \in [n]^{i-1}} \sum_{\mathbf{X}_{>i} \in [n]^{d-i}} \left(e^{\epsilon} \frac{1}{n} + \delta \right) \tag{3}$$

$$\leq \frac{e^{\epsilon}}{n} + \delta \tag{4}$$

In line 2, we use the fact that the full vector $\mathbf{X}$ is uniformly drawn, and we can just enumerate over all events. Recall that we use the hamming neighborhood relationship. Then finally in step 3, we use the privacy guarantees of mechanism B and flip to a neighboring dataset (wlog we set $X_i = 1$).

Small side note: we also get

$$\Pr[B(\mathbf{X}) \neq j] = 1 - \Pr[B(\mathbf{X}) = j] \leq \frac{e^{\epsilon}}{n} + \epsilon$$
$$\Leftrightarrow \Pr[B(\mathbf{X}) = j] \geq \frac{n \cdot (1 - \delta) - e^{\epsilon}}{n}$$

□

1.5 Reduction from MST

We are now ready to tighten the lower boind lower bound given by Sealfon in Sealfon [2016] by a factor of (...).

The idea is to encode a vector $\mathbf{X}$ into a dense graph G where privately releasing the MST exactly corresponds to privately releasing the vector. In a second step, we transfer the previously established bound from Theorem 2.4. More precisely, we are going to prove the following statement.

Theorem 1.2. *There exists a graph for which there exists no (ϵ, δ)-dp private mechanism $M_{V,E} : \mathcal{R}^E \to \{T | T \text{is a mst on} (V, E, \mathbf{W})\}$ under the ℓ_1 neighboring relationship with error xx for sufficiently $\delta < xx$.*

Proof. □

Graph Construction We start by describing how to encode some $\mathbf{X} \in [n]^d$ into a graph where privately releasing the vector corresponds exactly to privately releasing the MST of this graph.

Encoding. We first state a function $\mathcal{A}_{\text{ENCODE}} : [n]^d \to (\{V_l, V_r\}, E)$ returning a graph G with $(d + n)$ and $(n \cdot d + n - 1)$ edges E. We construct the graph in the following way.

1. We create a path of length n: $V_r = \{r_1, ..., r_n\}$ be vertices encoding the possible values $[n]$ and link r_i and r_{i+1} with edges weighted $w_{r_i, r_{i+1}} = 0$.

2. For each x_i, we create a vertex l_i and connect it to each r_i while

$$\forall i, j \text{ we set } w_{l_i, r_j} = \begin{cases} 0 \text{ if } x_i = j \\ R \text{ else} \end{cases}$$

   We denote these $V_l = \bigcup l_i$.

Decoding. Let's define $\mathcal{A}_{\text{DECODE}} : \{\text{spanning\_trees}(G)\} \to \{1, ..., n\}^n$ that takes an MST on this instance and converts it back to a vector in $[n]^d$. Let

$$x_i = \min_{j \in [n]} w_{l_i, r_j}$$

For any arbitrarily chosen mst algorithm $\mathcal{A}_{\text{MST}} : G \to \text{spanning\_trees(G)}$ and any vector $\mathbf{X}$, we have $\mathbf{X} = \mathcal{A}_{\text{DECODE}}\left(\mathcal{A}_{\text{MST}}\left(\mathcal{A}_{\text{ENCODE}}\right)\right)(\mathbf{X})$
[Lukas: Maybe a proof for that? At least some explanation]

Theorem 1.3. *There exists a graph for which there exists no (ϵ, δ)-dp private mechanism $M_{V,E} : \mathcal{R}^E \to \text{spanning\_trees}(G)$ under the ℓ_1 neighboring relationship with error xx for sufficiently large δ.*

Proof. □

Assuming two hamming neighboring vectors, we can bound the ℓ_1 neighboring relationship on the encoded graph:

Lemma 1.4. *Let $\mathbf{X} \sim \mathbf{X}' \in [n]^d$ be two neighboring vectors under the hamming neighboring relationship. Furthermore, let $M_{V,E}$ be an (ϵ, δ)-DP MST algorithm under the ℓ_1 neighboring relationship. Then*

$$X \longrightarrow M_{V,E}(\mathcal{A}_{ENCODE}(X))$$

is $(2R\epsilon, 2Re^{2R\epsilon}\delta)$-DP

Proof. Observe that for $\mathbf{X} \sim \mathbf{X}'$, we have

$$||\mathcal{A}_{\text{ENCODE}}(\mathbf{X}) - \mathcal{A}_{\text{ENCODE}}(\mathbf{X}')|| \leq 2R$$

. Notice that our graph is defined by the ℓ_1 neighboring relationship. Denote $z = n(d+1) - 1$. For two given $\mathbf{X} \sim_H \mathbf{X}'$ that differ in coordinate i (thus $x_i \neq x_i'$ only on coordinate i) we can give a chain of ℓ_1 neighboring graph weights $W^{(z)}$, such that

$$\mathbf{W}^{(0)} = \mathcal{A}_{\text{ENCODE}}(\mathbf{X}) \sim_1 \mathbf{W}^{(1)} \sim_1 \cdots \sim_1 \mathbf{W}^{(2R-1)} \sim_1 \mathcal{A}_{\text{ENCODE}}(\mathbf{X}') = \mathbf{W}^{(2R)}$$

For $i \in [z-1]$, we inductively increase (decrease) the weights for the two edges $e = \{l_i, r_{x_i}\}$ and $e' = \{l_i, r_{x'_i}\}$ that are different in the encoding. Let $\mathbf{W}_0 = \mathcal{A}_{\text{ENCODE}}(\mathbf{X})$, and let

$$\mathbf{W}^i = \begin{cases} \mathbf{W}^{(i-1)}, \text{ but } w_e^{(i)} = w_e^{(i-1)} - 1 \text{ if } i \leq R \\ \\ \mathbf{W}^{(i-1)}, \text{ but } w_{e'}^{(i)} = w_{e'}^{(i-1)} + 1 \text{ if } i > R \end{cases}$$

Because M is (ϵ, δ)-DP, we know for all $Z \subseteq E$:

$$\begin{aligned} \Pr[M_{V,E}(\mathcal{A}_{\text{ENCODE}}(\mathbf{X}) = Z] &\leq e^{\epsilon} \Pr[M_{V,E}(W^{(1)}) = Z] + \delta \\ &\leq e^{\epsilon} \left(e^{\epsilon} \Pr[M_{V,E}(W^{(2)}) = Z] + \delta \right) + \delta \\ &\leq \cdots \\ &\leq e^{2R\epsilon} \left(\Pr[M_{V,E}(\mathcal{A}_{\text{ENCODE}}(\mathbf{X}'))) = Z] \right) + 2R\delta e^{2R\epsilon} \end{aligned}$$

The last inequality follows because we can upper bound the sum of the deltas by $\delta \sum_{i=0}^{2R-1} e^{i\epsilon} \leq 2R\delta e^{2R}$.

Now, we are ready to derive a contradiction. □

[Lukas: TODO: Fill this.]

1.6 Lowering the Upper Bound of Shortest Paths

Lemma 1.5 (Tailbound on sum of laplacians Dwork and Roth [2014])**.** *Let $Y_1, \cdots Y_k \sim \texttt{Lap}(b_i)$ be independent variables and let $Y = \sum_i Y_i$ and $b_{\max} = \max_i b_i$. Let $\tau \geq \sqrt{\sum_i (b_i)^2}$, and $0 < \lambda < \frac{2\sqrt{2}\tau^2}{b_{\max}}$. Then*

$$\Pr[Y > \lambda] \leq \exp\left(-\frac{\lambda^2}{8\tau^2} \right)$$

Lemma 1.6. *For any graph G, the number of paths with k steps starting on a vertex v is at most n^k*

Proof. By a simple combinatorial argument on a complete graph K_n, we get:

$$\#\text{Distinct Paths} = \frac{(n-1)!}{(n-k)!} \leq n^k$$

□

Recall that Sealfon [2016]'s graph construction takes a P_i and adds two edges between each pair of vertices.

Lemma 1.7 (Upper bound for path graph)**.** *The graph in the lower bound in Sealfon [2016] actually has utility $\mathcal{O}(n)$.*

Proof. Assume $X_1, \cdots, X_k \sim \texttt{Lap}(1/\epsilon)$ and $X = \sum X_i$ for any path P of length k Then:

$$\Pr[X > cn/\epsilon] \leq \exp\left(-\frac{c^2n^2/\epsilon^2}{8k/\epsilon^2}\right) = \exp\left(-\frac{c^2}{8} \cdot \frac{n^2}{k}\right).$$

By a union bound on all 2^n paths:

$$\Pr[\exists P : X_P \geq cn/\epsilon] \leq 2^n e^{-c'n} \leq e^n e^{-c'n} = e^{n(1-c')}$$

which is exponentially small for $c > 1$. □

[Lukas: TODO: I guess we have to reformulate these lemmas.]

Lemma 1.8. *This is not working! Needs some more thought. For all graphs G and $c \in (0, 1)$, adding noise drawn from $\texttt{Lap}(1/\epsilon)$ is ϵ-dp allows to compute paths between each pair of vertices with error less than $\mathcal{O}(n\sqrt{\log n}/\epsilon)$*

Proof. We observe that any graph has at most n^n distinct paths. WLOG assumes that we have a complete graph K_n. Let $c = 2\sqrt{2c' \log n}$ and hence, for each path of length k, we get

$$\begin{aligned}\Pr[\exists P : X_P > c'n \log n/\epsilon] &\leq n^k \exp\left(-\frac{c'n^2 \log n}{k}\right) \\ &= \exp\left(n \log n - \frac{c'n^2 \log n}{k}\right)\end{aligned}$$

The longest path can have up to $n - 1$ edges, then the term above is exponentially small in n, if $n \log n - c'n \log n < 1$. Hence, $c' > 1$ suffices, giving the desired result. □

Using the same technique, we get a similar result if the diameter of the graph is bounded by $\frac{n}{\sqrt{\log n}}$.

Lemma 1.9. *For all graphs G where the diameter is bounded by $k \leq \frac{n}{\sqrt{\log n}}$, adding noise drawn from $\texttt{Lap}(1/\epsilon)$ is ϵ-dp allows to compute paths between each pair of vertices with error less than $\mathcal{O}(n/\epsilon)$ and therefore matches the known lower bound by Sealfon Sealfon [2016].*

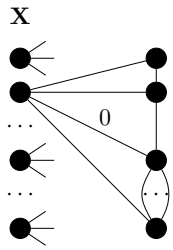
As previously, we have:

Proof.

$$\begin{aligned}\Pr[\exists P : X_P > cn/\epsilon] &\leq \exp(k \log n) \exp\left(-\frac{c^2n^2}{8k}\right) \\ &= \exp\left(k \log n - \frac{c^2n^2}{8k}\right)\end{aligned}$$

which is negative, exactly if $k^2 \log n < \frac{c^2 n^2}{8}$ and hence $k < \frac{cn}{2\sqrt{2 \log n}}$. Setting $c = 2\sqrt{2}$ finishes the proof. □

What about the gap of



Questions / Ideas

1. Graphs with bounded diameter?

2 Minimum Weight Perfect Matching

We are now turning our attention to the following problem: Given a publicly known graph $G = (V, E)$ with private weight vector $\mathbf{w} \in \mathbb{R}^E$, we want to release a set of edges $M \subseteq E$ where each vertex v is adjacent to exactly on edge in M and minimizes the weight of $w(M)$. We denote A as the mechanism that returns an approximately minimum weight perfect matching (MWPM) under edge-weight differential privacy, where we want to protect the individual weights. Denote $\mathcal{M}$ as the set of all possible perfect matching on G, and then we want for all $M \in \mathcal{M}(G)$ and neighboring (ℓ_1 and ℓ_∞ resp.):

$$\Pr[A(G, \mathbf{w}) = M] \leq e^\epsilon \Pr[A(G, \mathbf{w}') = M] + \delta \ .$$

We are interested in ensuring that the weight of the released matching is not too bad (in an additive sense). Therefore, we want to minimize the error [1] $w(A(G, \mathbf{w}) - w(M^*)$, the absolute difference to the real MWPM M^* with probability γ.

Recently, **?** have shown tight bounds for releasing an MST under pure DP. They used a packing argument to show that the errors of $\Theta(n \log n/\epsilon)$ and $\Theta(n^2 \log n)$ are tight for the ℓ_1 and ℓ_∞ neighboring relationship. Their main argument relies on the fact that one can greedily construct a large set of weight vectors on a very dense graph that are $\Theta(n)$ in hamming distance apart. We will now show that the same argument hold for Minimum-Weight Perfect Matching as well and prove the following theorem

[1] Note that the bound can be stated as *expected error* as well

Matchings : We define the *hamming metric* on matchings as $d_H(M_1, M_2) = |M_1 \setminus M_2| = |M_2 \setminus M_1|$.

We will use the indicator function to denote the weights of our hard instance:

Definition 2.1. $\mathbb{1}_M(e) = \begin{cases} 0 & \text{if } e \in M \\ 1 & \text{otherwise} \end{cases}$.

And extend this notation to $\mathbb{1}_M(M') = \sum_{e \in M'} \mathbb{1}_M(e)$. Observe that $\mathbb{1}_M(M) = 0$ and $||\mathbb{1}_{M_1}(M_2) - \mathbb{1}_{M_2}(M_1)||_1 \leq 2d_H(M_1, M_2)$.

[Lukas: TODO: Reflect $d \in \Theta(n)$. Check constants.]

Theorem 2.2 (Lower Bound Pure DP)**.** *Fix the topology of some graph G. Given a set of matchings $M \subseteq \mathcal{M}(G)$, let $d \in \Theta(n)$ be such that $d_H(M_1, M_2) > d$ for all $M_1, M_2 \in S$. let $\mathcal{A}$ be ϵ-differentially private. with respect to $\sim_1$. Then there exists weights $\mathbf{w} \in \mathbb{R}^E$, such that*

$$\Pr\left[w(M) \leq w(M^*) + \log n/(c \cdot \epsilon) - 1/4\right] \leq 1/\sqrt{|S|}$$

where M^* denotes the real minimal-weight perfect matching.

We will prove that such an exponentially sized set S with $|S| \in 2^{\Theta(n \log n}$ exists later in xx.

We are using the packing argument due to xxx:

Lemma 2.3. *Let $\mathcal{W} \subseteq \mathcal{X}$ be a collection of datasets all at distance at most r from some fixed dataset $w_0 \in \mathcal{X}$, and let $\{\mathcal{L}_w\}_{w \in \mathcal{W}}$ be a collection fo disjoint subsets of the output space $\mathcal{M}$. If there is a ϵ-differentially private mechanism $\mathcal{A} : \mathcal{X} \to \mathcal{W}$ such that $\Pr[\mathcal{A}(w) \in \mathcal{L}_w] \geq p$ for every $w \in \mathcal{W}$, then*

$$p \leq r^{r\epsilon}/|\mathcal{W}|$$

1. Can we invoke the Packing Argument for pure DP to show tight (worst-case) $\Theta(n \log n)$? Can we also show $\Theta(n^2 \log n)$ for ℓ_∞? YES!
2. Greedy Algorithm https://www.ee.columbia.edu/ jelena/lec21.pdf should also give a $\tilde{\mathcal{O}}(n^{3/2})$ approximation for ℓ_∞ / approximate DP. Is it tight?

References

Adam Sealfon. Shortest Paths and Distances with Differential Privacy, April 2016. URL http://arxiv.org/abs/1511.04631. arXiv:1511.04631 [cs].

Cynthia Dwork and Aaron Roth. *The Algorithmic Foundations of Differential Privacy*, volume 9. 2014. URL http://dx.doi.org/10.1561/0400000042.

2.1 old stuff

Theorem 2.4. *Assuming $n > 1$, if algorithm $B : [n]^{[n]} \to [n]$ is (ϵ, δ)-dp and we uniformly sample a random input $X \leftarrow U([n])^n$, then we have for all indices i:*

$$\Pr[B(X) \neq X_i] \geq \frac{(n-1)(1-n^2\delta)}{e^\epsilon + n}$$

(((Maybe more natural to upper bound the probability of equality?)))

Proof. We consider a family of algorithms $B_{i,j}$ that returns $B(Y)$ where Y is equal to X except that coordinate i is replaced by j; if B is differentially private with respect to X then so is $B_{i,j}$. First, note that for $B_{i,j}$:

$$\begin{aligned}
\Pr[B(X) = X_i] &= \frac{1}{n}\sum_{j=1}^{n} \Pr[B_{i,j}(\mathbf{X}) = j] \\
FIXME. \Pr[B(X) = X_i] &= \frac{1}{n}\sum_{j=1}^{n} \Pr[B_{i,j}(\mathbf{X}) = j] \\
&\leq \frac{1}{n}\sum_{j=1}^{n} \frac{1}{n-1}\sum_{j' \neq j} (e^\epsilon \Pr[B_{i,j'}(\mathbf{X}) = j] + \delta) \\
&\leq \frac{1}{n}\sum_{j=1}^{n} \left(\frac{1}{n-1}\sum_{j' \neq j} (e^\epsilon \Pr[B_{i,j'}(\mathbf{X}) = j]) + \delta \right) \\
FIXME &\leq \frac{e^\epsilon}{n^2 - n} \Pr[B(\mathbf{X}) \neq X_i] + n^2\delta
\end{aligned}$$

Furthermore, because $\Pr[B(\mathbf{X}) = X_i] = 1 - \Pr[B(\mathbf{X}) \neq X_i]$, we get

$$\begin{aligned}
1 - n^2\delta &\leq \left(\frac{e^\epsilon}{n(n-1)} + 1 \right) \Pr[B(\mathbf{X} \neq X_i)] &(5)\\
&\leq \frac{e^\epsilon + n}{(n-1)} \Pr[B(\mathbf{X} \neq X_i)] &(6)
\end{aligned}$$

and hence

$$Pr[B(\mathbf{X} \neq X_i)] \geq \frac{(1-n^2\delta)(n-1)}{e^\epsilon + n}$$

□

A second proof with a slightly different result:
TODO rephrase with r AND rephrase

Proof.

$$
\begin{aligned}
1 \geq \Pr[B(\mathbf{X}) \neq \mathbf{X}_i] &= \sum_{j \in [n]} \Pr[\mathbf{X}_i = j] \cdot \Pr[B(\mathbf{X}_i) \neq j | \mathbf{X}_i = j] \\
&= \frac{1}{n} \sum_{j \in [n]} \Pr[B_{i,j}(\mathbf{X}) \neq j] \\
&= \frac{1}{n} \sum_{j \in [n]} \Pr[B_{i,j}(\mathbf{X}) \in [n] \backslash \{j\}] \\
&= \frac{1}{n} \sum_{j \in [n]} \left(\frac{1}{n-1} \sum_{j' \neq j} \Pr[B_{i,j}(\mathbf{X}) \in [n] \backslash \{j\}] \right) \\
&\geq \frac{1}{n} \sum_{j \in [n]} \left(\frac{e^{-\epsilon}}{n-1} \sum_{j' \neq j} (\Pr[B_{i,j'}(\mathbf{X}) \in [n] \backslash \{j\}] - \delta) \right) \\
&= -e^{-\epsilon}\delta + \frac{1}{n} \sum_{j' \in [n]} \left(\frac{e^{-\epsilon}}{n-1} \sum_{j \neq j'} \Pr[B_{i,j'}(\mathbf{X}) \in [n] \backslash \{j\}] \right) \\
&= e^{-\epsilon} \left(-\delta + \sum_{j \in [n]} \sum_{j' \neq j} \frac{1}{n} \Pr[B_{i,j'}(\mathbf{X}) = j']) \right) \\
&= e^{-\epsilon} \left(-\delta + \sum_{j \in [n]} \sum_{j' \neq j} \Pr[\mathbf{X}_i = j'] \cdot \Pr[B(\mathbf{X}) = j' | \mathbf{X}_i = j']) \right) \\
& e^{-\epsilon} \left(-\delta + (n-1) \cdot \sum_{j' \in [n]} \Pr[B(\mathbf{X}) = j' \wedge \mathbf{X}_i = j'] \right) \\
& e^{-\epsilon} \left(-\delta + (n-1) \cdot \Pr[B(\mathbf{X}) = \mathbf{X}_i] \right)
\end{aligned}
$$

Hence, we can upper bound

$$\Pr[B(\mathbf{X} = \mathbf{X}_i)] \leq \frac{e^{\epsilon} + \delta n}{n-1}$$

□

Some notes for me. Or later.

1. First equality: Condition a single index on all distinct outcomes.
2. Second: $\Pr[\mathbf{X_i} = j] = 1/n$ because of uniformly drawing $\mathbf{X}$ and conditioning can be seen as "'replacing"'
3. (3) Holds because it is just the complementary event on all different $j' \neq j$
4. (4) Because B is dp

5. last steps: flipping sum and reverting steps

1. Why is it necessary to sum over all j'. Privacy should already be given for a single change. (((I think this is just a way of getting an expression that equals the probability of not outputting X_i, but maybe there is a more direct way of arguing.)))

2. I have the feeling that the bound also holds for shortest path trees because in our construction, the MST is also the shortest path tree - essentially regardless which vertex we start with.