A Simple Differentially Private Algorithm for Global Minimum Cut

George Z. Li*
University of Maryland

Abstract
In this note, we present a simple differentially private algorithm for the global minimum cut problem using only one call to the exponential mechanism. This problem was first studied by Gupta et al. [2010], and they gave a differentially private algorithm with near-optimal utility guarantees. We improve upon their work in many aspects: our algorithm is simpler, more natural, and more efficient than the one given in Gupta et al. [2010], and furthermore provides slightly better privacy and utility guarantees.

1 Introduction
Given an undirected graph $G = (V, E)$, the global minimum cut problem asks us to find a cut $(S, V - S)$ minimizing the size of the cut $E(S, V - S) = |\{(i, j) \in E : i \in S, j \in V - S\}|$. We consider this problem under the edge-differential privacy model, where the cut we output can’t depend too much on the existence of any edge in the graph. Formally, edge-differential privacy is defined as follows:

Definition 1 (Dwork and Roth, 2014). Let $G$ be the set of undirected graphs and call two graphs $G_1 = (V_1, E_1), G_2 = (V_2, E_2) \in G$ neighbors if their vertex sets are the same and their edge sets differ by exactly one edge. A mechanism $M : G \rightarrow \mathcal{Y}$ is said to be $(\epsilon, \delta)$-edge differentially private if for every pair of neighboring graphs $G_1, G_2 \in G$ and any set of outputs $S \subseteq \mathcal{Y}$, we have

$$\Pr[M(G_1) \in S] \leq \exp(\epsilon) \cdot \Pr[M(G_2) \in S] + \delta.$$  

If $\delta = 0$, we say $M$ is $\epsilon$-edge differentially private.

Without differential privacy, this problem has been studied extensively, culminating recently in a deterministic almost linear time algorithm for the problem [Kawarabayashi and Thorup, 2019]. Under differential privacy, this problem was studied in the seminal work of Gupta et al. [2010] along with many other fundamental problems in combinatorial optimization. Combining the exponential mechanism with some ideas from Karger [1993], they gave an $(\epsilon, \delta)$-edge differentially private algorithm (for $\delta = O(1/n^2)$) which incurs an additive error of $3 \log n \over \epsilon$ over the optimal minimum cut. Furthermore, they showed that this result is tight up to constant factors: any $\epsilon$-edge differentially private algorithm for minimum cut must (information theoretically) incur an $\Omega(\log n / \epsilon)$ additive error.

In this work, we give a simpler algorithm which only incurs an additive error of $3 \log n \over \epsilon$ over the optimal solution while satisfying $\epsilon$-edge differentially privacy, a stronger privacy guarantee than $(\epsilon, \delta)$-edge differential privacy. In addition to the improved privacy and accuracy guarantees, our algorithm also benefits from other practical advantages over Gupta et al. [2010]’s algorithm. Our algorithm uses very little randomness, requiring only one call to the exponential mechanism, and has run-time equivalent to $n - 1$ runs of the fastest minimum $s - t$ cut algorithm. In contrast, Gupta et al. [2010]’s algorithm requires 2 calls to the exponential mechanism and $O(n^2)$ calls to Karger’s (randomized) algorithm for global minimum cut. Finally, our proof is also simpler: while their utility proof uses a deep result by Karger on the number of near-optimal cuts, ours is entirely self contained.

*Email: gzli929@gmail.com
2 The Exponential Mechanism

In this section, we state the exponential mechanism and its privacy and utility guarantees; we refer the readers to the original paper for the proofs of the claims [McSherry and Talwar, 2007]. The exponential mechanism serves as the backbone of essentially all of differentially private combinatorial optimization [Gupta et al., 2010; Mitrovic et al., 2017; Esencayi et al., 2019; Nguyen and Vullikanti, 2021], and will also be the primary component of our algorithm.

Informally, the setup is as follows: you are given a set of \( n \) objects and and a utility value for each object. You wish to choose one of the objects to maximize your utility. However, the utility function depends on a private dataset, such as the edge set \( E \), so we cannot output the object which maximizes the utility every time. Instead, we hope to output an object which approximately maximizes the utility while preserving differential privacy.

The exponential mechanism, introduced in [McSherry and Talwar, 2007], shows that outputting an object with utility only an additive \( O(\log n) \) away from the optimal utility is possible (up to some normalization).

We will first define the sensitivity of the utility function, and then state the exponential mechanism and its privacy and utility guarantees. For consistency, all of our definitions will be written in terms of graphs and edge-differential privacy.

Definition 2. Let \( R \) be a set of objects and let \( u : R \times G \rightarrow \mathbb{R} \) be a utility function. We define the sensitivity of utility function as \( \Delta_u = \max_{G_1 \sim G_2} \max_{r \in R} |u(r, G_1) - u(r, G_2)| \), where \( G_1 \sim G_2 \) are neighboring graphs.

Definition 3. Let \( R \) be a (finite) set of objects and let \( u : R \times G \rightarrow \mathbb{R} \) be a utility function with sensitivity \( \Delta_u \). The exponential mechanism outputs object \( r \in R \) with probability proportional to \( \exp(\epsilon \cdot u(r, G)/\Delta_u) \).

Theorem 4. Suppose the utility function \( u \) is monotone (i.e., adding an edge to the graph doesn’t decrease the utility). Then the exponential mechanism, \( M_E \), satisfies the following:

- **Privacy:** \( M_E \) is \( \epsilon \)-edge differentially private.
- **Utility:** \( \Pr[u(M_E(G, u, R), G) \leq OPT - \frac{\Delta_u}{\epsilon} (\log n + r)] \leq \exp(-r) \).

3 Differentially Private Min-Cut

To give our differentially private algorithm for global minimum cut, we first need to define a related problem called minimum \( s-t \) cut. We are given an undirected graph \( G = (V, E) \) along with a source node \( s \) and a sink node \( t \). The minimum \( s-t \) cut problem asks us to find a cut \( (S, V - S) \) satisfying \( s \in S \) and \( t \in V - S \) which minimizes the size of the cut \( E(S, V - S) \). It is well know that this problem can be solved deterministically in polynomial time [Ford and Fulkerson, 1956].

Using the algorithm for computing the minimum \( s-t \) cut as a subroutine, we can give an algorithm for global minimum cut. Fix an arbitrary source node \( s \) and for each \( t \in V - \{s\} \), obtain the minimum \( s-t \) cut. Choosing the smallest of these \( n - 1 \) cuts will then give us the global minimum cut. We will use the exponential mechanism to adapt this algorithm to satisfy differential privacy.

Like before, fix \( s \in V \) and for each \( t \in V - \{s\} \), find the minimum \( s-t \) and denote it by \( (S_t, V - S_t) \). Since the size of the minimum \( s-t \) cut depends on the private edge set \( E \), we cannot directly choose the smallest cut. Instead, we define the utility of a cut as the size of the cut \( u(S_t) = E(S_t, V - S_t) \) for each \( t \in V - \{s\} \) and output a cut of approximately minimum size via the exponential mechanism.

To analyze our algorithm, we need to understand the sensitivity of the utility function. For a given a cut \( (S, V - S) \), it is obvious that the size of the cut changes by at most 1 between neighboring graphs \( G_1, G_2 \). However, our utility is the size of the minimum \( s-t \) cut between a fixed vertex \( s \) and an input vertex \( t \). Though adding/removing an edge can change the minimum \( s-t \) cut completely, we observe that the size of the minimum \( s-t \) cut changes by at most 1 (see Lemma 4). With this in hand, our results will follow directly by the guarantees of the exponential mechanism in Theorem 4.
Lemma 5. The sensitivity of our utility function $u$ is $\Delta_u = 1$.

Proof. Let $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ be arbitrary neighboring graphs with $E_1 = E_2 \cup \{e\}$. Fix a source node $s$ and let the terminal node $t \in V - \{s\}$ be arbitrary. Let $(S_1, V - S_1)$ and $(S_2, V - S_2)$ be the minimum $s$-$t$ cuts in graphs $G_1$ and $G_2$, respectively. We claim $E_1(S_1, V - S_1) \leq E_2(S_2, V - S_2) \leq E_1(S_1, V - S_1) + 1$; since $t$ and $G_1, G_2$ are arbitrary, this directly implies our desired result.

We will first prove $E_1(S_1, V - S_1) \leq E_2(S_2, V - S_2)$. Since $(S_2, V - S_2)$ also forms a cut in $G_1$, we have $E_1(S_1, V - S_1) \leq E_1(S_2, V - S_2)$ by the optimality of $(S_1, V - S_1)$. But since $E_1 \subseteq E_2$, we also have $E_1(S_2, V - S_2) \leq E_2(S_2, V - S_2)$. Combining the results give us the first part of the desired inequality. Note that this implies that our utility function is monotone, which we will need later.

Next, let us prove $E_2(S_2, V - S_2) \leq E_1(S_1, V - S_1) + 1$. Since $(S_1, V - S_1)$ also forms a cut in $G_2$, we have $E_2(S_2, V - S_2) \leq E_2(S_1, V - S_1)$ by the optimality of $(S_2, V - S_2)$. But since $E_2$ only contains one more edge than $E_1$, we also have $E_2(S_1, V - S_1) \leq E_1(S_1, V - S_1) + 1$. Combining the results give us second part of the desired inequality, and completes the proof.\]

Theorem 6. Let ALG denote the algorithm described above. We have:

- ALG is $\epsilon$-edge differentially private.
- With probability $1 - \frac{1}{n^2}$, ALG outputs a cut $(S, V - S)$ with $E(S, V - S) \geq \text{OPT} - \frac{3\log n}{r}$.

Proof. Since we already proved that $\Delta_u = 1$ and that the utility is monotone in Lemma 5, the guarantees of the exponential mechanism in Theorem 4 apply. Our privacy guarantee follows directly and our utility guarantee follows by setting $r = 2\log n$ in the utility guarantee of Theorem 4.\]

References

C. Dwork and A. Roth. The Algorithmic Foundations of Differential Privacy. Found. Trends Theor. Comput. Sci., 9(3-4):211–407, 2014.

Y. Esencayi, M. Gaboardi, S. Li, and D. Wang. Facility Location Problem in Differential Privacy Model Revisited. In Advances in Neural Information Processing Systems 32, pages 8489–8498, 2019.

L. R. Ford and D. R. Fulkerson. Maximal Flow Through a Network. Canadian Journal of Mathematics, 8:399–404, 1956. doi: 10.4153/CJM-1956-045-5.

A. Gupta, K. Ligett, F. McSherry, A. Roth, and K. Talwar. Differentially Private Combinatorial Optimization. In M. Charikar, editor, Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, January 17-19, 2010, pages 1106–1125. SIAM, 2010.

D. R. Karger. Global Min-cuts in RNC, and Other Ramifications of a Simple Min-Cut Algorithm. In Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms, pages 21–30. ACM/SIAM, 1993.

K. Kawarabayashi and M. Thorup. Deterministic Edge Connectivity in Near-Linear Time. Journal of the ACM, 66(1):4:1–4:50, 2019.

F. McSherry and K. Talwar. Mechanism Design via Differential Privacy. In 48th Annual IEEE Symposium on Foundations of Computer Science, pages 94–103. IEEE Computer Society, 2007.

M. Mitrovic, M. Bun, A. Krause, and A. Karbasi. Differentially Private Submodular Maximization: Data Summarization in Disguise. In Proceedings of the 34th International Conference on Machine Learning, volume 70, pages 2478–2487, 2017.

D. Nguyen and A. Vullikanti. Differentially Private Densest Subgraph Detection. In Proceedings of the 38th International Conference on Machine Learning, volume 139, pages 8140–8151, 2021.