Node and Edge Differential Privacy for Graph Laplacian Spectra: Mechanisms and Scaling Laws

Calvin Hawkins, Bo Chen, Kasra Yazdani, and Matthew Hale, Member, IEEE

Abstract—This paper develops a framework for privatizing the spectrum of the Laplacian of an undirected graph using differential privacy. We consider two privacy formulations. The first obfuscates the presence of edges in the graph and the second obfuscates the presence of nodes. We compare these two privacy formulations and show that the privacy formulation that considers edges is better suited to most engineering applications. We use the bounded Laplace mechanism to provide $(\epsilon, \delta)$-differential privacy to the eigenvalues of a graph Laplacian, and we pay special attention to the algebraic connectivity, which is the Laplacian's second smallest eigenvalue. Analytical bounds are presented on the accuracy of the mechanisms and on certain graph properties computed with private spectra. A suite of numerical examples confirms the accuracy of private spectra in practice.

Index Terms—Network topology, network structure, network architecture and protocols, social network security and privacy.

I. INTRODUCTION

Graphs are used to model a wide range of interconnected systems, including multi-agent control systems [1], social networks [2], and others [3]. Various properties of these graphs have been used to analyze controllers and dynamical processes over them, such as reaching a consensus [4], the spread of a virus [5], robustness to connection failures [6], and others. Graphs in these applications may contain sensitive information, e.g., one’s close friendships in the case of a social network, and it is essential that these analyses do not inadvertently leak any such information.

Unfortunately, it is well-known that even aggregate, graph-level analyses can reveal sensitive information about individuals in a graph, such as the absence or presence of individual nodes in a graph [7] and the absence or presence of specific edges between them [8]. Similar privacy threats have been explored in the data science community, where graphs are used to represent datasets and the goal is to enable data analysis while formally protecting the data of individuals contained in those datasets.

Differential privacy is one well-studied tool for doing so. Differential privacy is a statistical notion of privacy that has several desirable properties: (i) it is robust to side information, in that learning additional information about data-producing entities does not weaken privacy by much [9], and (ii) it is immune to post-processing, in that arbitrary post-hoc computations on private data do not weaken privacy [10]. There exist numerous differential privacy implementations for graph properties specifically, including counts of certain subgraphs [8], the degree distribution of a graph [11], and various other frequent patterns in graphs [12].

The need for privacy for the aforementioned graph properties comes from the inferences that one can draw about a graph from these quantities, as detailed in [13], [14], [15]. Decades of research in algebraic graph theory have quantified connections between the Laplacian spectrum and a myriad of other graph properties; see [16] for a summary. Accordingly, the Laplacian spectrum, especially the algebraic connectivity $\lambda_2$, implicates the same ability to draw inferences as other graph properties and hence gives rise to the same types of privacy concerns. Nonetheless, it is still desirable to share private spectra to enable graph analyses.

One specific motivation for privacy of Laplacian spectra is interest in graphs as data sets in machine learning [17]. It is well known that training machine learning models on sensitive data can cause privacy breaches [18]. Laplacian spectra can specifically be used for clustering, embedding, and indexing large graphs [19], [20], [21], as well as understanding randomness and centrality of social networks [22], [23]. This paper presents a method to share a private Laplacian spectrum that enables all of these existing analyses while ensuring that the graph is kept private from the recipient of these private spectra.

We therefore protect the values the graph Laplacian spectrum using two notions of privacy: edge and node differential privacy [24]; to the best of our knowledge this is the first work to do so. Edge privacy obfuscates the absence and/or presence of a pre-specified number of edges, while node privacy obfuscates the absence or presence of a single node. In this paper we show that the differences in guarantees of these two notions of privacy result in drastic differences in the accuracy of the private values of the Laplacian spectrum. Specifically, in Section IV we show that the variance of noise required to obfuscate the presence of one node in a graph of size $n$ scales with $n^2$, which rapidly grows large. For this reason, Sections V and VI focus on edge privacy and obfuscating the connections in a network. We note that while differential privacy has been applied to protect various...
quantities in multi-agent systems [25], [26], [27], [28], privacy for properties of a multi-agent network itself has received less attention, and that is what we focus on.

In this paper we pay special attention to the algebraic connectivity. A graph’s algebraic connectivity (also called its Fiedler value [29]) is equal to the second-smallest eigenvalue of its Laplacian. This value plays a central role in the study of multi-agent systems because it sets the convergence rates of consensus algorithms [30], which appear directly or in modified form in formation control [31], connectivity control [32], and many distributed optimization algorithms [33].

Our implementation uses the recent bounded Laplace mechanism [34], which ensures that private scalars lie in a specified interval. The algebraic connectivity of a graph is bounded below by zero and above by the number of nodes in a graph, and we confine private outputs to this interval by applying the mechanism in [34] to the privatization of Laplacian spectra.

Contributions: We provide closed-form values for the sensitivity and other constants needed to define edge and node differential privacy mechanisms for the Laplacian spectrum, and this is the first contribution of this paper. The second contribution is showing the detrimental scaling of node privacy and the benefits of edge privacy. Our third contribution is the use of the private values of algebraic connectivity to analytically bound other graph properties, namely the diameter of graphs and the mean distance between their nodes. Our fourth contribution is providing guidelines on using these mechanisms by providing a series of examples to demonstrate how to use the mechanisms and the accuracy of information they provide.

Related work in [35] has developed a different method to privatize the spectrum and other properties of a graph’s adjacency matrix. We instead focus on privatizing the spectrum of a graph’s Laplacian, which commonly appears in the analysis of large data sets in machine learning and the analysis of social networks. We also derive simpler forms for the distribution of noise required, and we develop a privacy mechanism that does not require any post-processing, which is different from [35].

A preliminary version of this paper appeared in [36]. This paper extends the edge privacy mechanism for \( \lambda_2 \) to the rest of the Laplacian spectrum, develops the node privacy mechanism for \( \lambda_2 \), compares the scaling of the edge and node privacy mechanisms, and provides further applications and uses of the private Laplacian spectrum.

The rest of the paper is organized as follows. Section II provides background and problem statements. Section III develops the differential privacy mechanisms for the Laplacian spectrum. Next, Section IV compares the scaling of edge and node differential privacy and as a result we shift our attention to edge privacy exclusively. Then, we use the output of the edge mechanism to bound other graph properties in Section V. Section VI provides guidelines and examples and Section VII concludes.

Notation: We use \( \mathbb{R} \) and \( \mathbb{N} \) to denote the real and natural numbers, respectively. We use \( |S| \) to denote the cardinality of a finite set \( S \), and we use \( S_1 \triangle S_2 = (S_1 \setminus S_2) \cup (S_2 \setminus S_1) \) to denote the symmetric difference of two sets. For \( n \in \mathbb{N} \), we use \( G_n \) to denote the set of graphs on \( n \) nodes.
are adjacent if they differ by $A$ edges. We express this mathematically via

$$\text{Adj}_{e,A}(G, G') = \begin{cases} 1 & |E(G) \Delta E(G')| \leq A, \\ 0 & \text{otherwise}. \end{cases}$$

**Definition 1.B (Node Adjacency relation):** Fix $n \in \mathbb{N}$. Two graphs, $G, G' \in \mathcal{G}_n$, are adjacent if they differ by one node with the corresponding edges added or deleted. We express this mathematically via

$$\text{Adj}_n(G, G') = \begin{cases} 1 & |V(G) \Delta V(G')| \leq 1, \\ 0 & \text{otherwise}. \end{cases}$$

In Definition 1.A, the parameter $A$ is the number of edges whose absence or presence must be concealed by privacy. It can encode, for example, the concealment of the absence or presence of $A$ connections of a single user, the concealment of the absence or presence of $\frac{A}{n}$ connections of each of three distinct users, or the concealment of the absence or presence of a single connection belonging to each of $A$ distinct users.

On the other hand, Definition 1.B specifies that the absence or presence of a single node must be concealed by privacy. Its interpretation is that the addition or removal of a single user from a graph will not change its privatized spectrum by much.

In Section III-B, we show that a mechanism that obfuscates the absence or presence of only a single node (in the sense of Definition 1.B) requires impractically large variance of noise and produces highly inaccurate private outputs. The obfuscation of more nodes would require even more noise, and thus node differential privacy is not practical for protecting the spectra of many realistic networks. Therefore we do not consider obfuscating the absence or presence of arbitrary numbers of nodes, and we focus on edge differential privacy and the obfuscation of edges instead.

Next, we briefly review differential privacy; see [10] for a complete exposition. A privacy mechanism $\mathcal{M}$ for a function $f$ can be obtained by first computing the function $f$ on a given input $x$, and then adding noise to $f(x)$. The distribution of noise depends on the sensitivity of the function $f$ to changes in its input, described below. It is the role of a mechanism to approximate functions of sensitive data with private responses, and we next state this formally. The guarantees of privacy are defined with respect to the adjacency relation. Since we consider two notions of adjacency, we define two types of privacy: (i) edge differential privacy using the standard definition of differential privacy equipped with the edge adjacency relation, $\text{Adj}_{e,A}$ appearing in Definition 1.A, and (ii) node differential privacy using the standard definition of differential privacy equipped with $\text{Adj}_n$ in Definition 1.B.

**Definition 2.A (Edge differential privacy; [10]):** Let $\epsilon > 0$, $\delta \in [0, 1)$ be given, use $\text{Adj}_{e,A}$ from Definition 1.A, and fix a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Then a mechanism $\mathcal{M} : \Omega \times \mathcal{G}_n \rightarrow \mathbb{R}$ is $(\epsilon, \delta)$-differentially private if, for all adjacent graphs $G, G' \in \mathcal{G}_n$,

$$\mathbb{P} \left[ \mathcal{M}(G) \in S \right] \leq \exp(\epsilon) \cdot \mathbb{P} \left[ \mathcal{M}(G') \in S \right] + \delta$$

for all sets $S$ in the Borel $\sigma$-algebra over $\mathbb{R}$.

**Definition 2.B (Node differential privacy; [10]):** Let $\epsilon > 0$, $\delta \in [0, 1)$ be given, use $\text{Adj}_n$ from Definition 1.B, and fix a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Then a mechanism $\mathcal{M} : \Omega \times \mathcal{G}_n \rightarrow \mathbb{R}$ is $(\epsilon, \delta)$-differentially private if, for all adjacent graphs $G, G' \in \mathcal{G}_n$,

$$\mathbb{P} \left[ \mathcal{M}(G) \in S \right] \leq \exp(\epsilon) \cdot \mathbb{P} \left[ \mathcal{M}(G') \in S \right] + \delta$$

for all sets $S$ in the Borel $\sigma$-algebra over $\mathbb{R}$.

The value of $\epsilon$ controls the amount of information shared, and typical values range from $0.1$ to $\log 3$ [10]. The value of $\delta$ can be regarded as the probability that more information is shared than $\epsilon$ should allow, and typical values range from $0$ to $0.05$. Smaller values of both imply stronger privacy. Given $\epsilon$ and $\delta$, a privacy mechanism must enforce Definition 2.A or 2.B for all graphs adjacent in the sense of Definition 1.A or 1.B, respectively.

We next define the sensitivity of $\lambda_i$, which will be used later to calibrate the variance of privacy noise. With a slight abuse of notation, we treat $\lambda_i$ as a function $\lambda_i : \mathcal{G}_n \rightarrow \mathbb{R}$, and we will develop differential privacy mechanisms to approximate each $\lambda_i$. The sensitivity will depend on which adjacency relation is used, and this is made explicit in the following definitions.

**Definition 3.A (Edge Sensitivity):** The edge sensitivity of $\lambda_i$ is the greatest difference between its values on Laplacians of graphs that are adjacent with respect to $\text{Adj}_{e,A}$ in Definition 1.A. Formally, for a fixed $A$, the edge sensitivity of $\lambda_i$ is given as

$$\Delta \lambda_{i,e} = \max_{G,G' \in \mathcal{G}_n, \text{Adj}_{e,A}(G, G') = 1} |\lambda_i(L) - \lambda_i(L')|,$$

where $L$ and $L'$ are the Laplacians of $G$ and $G'$.

**Definition 3.B (Node Sensitivity):** The node sensitivity of $\lambda_i$ is the greatest difference between its values on Laplacians of graphs that are adjacent with respect to $\text{Adj}_n$ in Definition 1.B. Formally, the node sensitivity of $\lambda_i$ is given as

$$\Delta \lambda_{i,n} = \max_{G,G' \in \mathcal{G}_n, \text{Adj}_n(G, G') = 1} |\lambda_i(L) - \lambda_i(L')|,$$

where $L$ and $L'$ are the Laplacians of $G$ and $G'$.

Noise is added by a mechanism, which is a randomized map used to implement differential privacy. The Laplace mechanism is widely used, and it adds noise from a Laplace distribution to sensitive data (or functions thereof). The standard Laplace mechanism has support on all of $\mathbb{R}$. For graphs on $n$ nodes, $\lambda_i \in [0, n]$ for all $i$. To generate a private output, one can add Laplace noise and then project the result onto $[0, n]$ (which is differentially private because the projection is post-processing), though similar approaches have been shown to produce highly inaccurate private data [27]. Instead, we use the bounded Laplace mechanism in [34]. We state it in a form amenable to use with $\lambda_i$.

**Definition 4:** Let $b > 0$ and let $D = [0, n]$. Then the bounded Laplace mechanism $W_{\lambda_i} : \Omega \rightarrow D$, for each $\lambda_i \in D$, is given by its probability density function $f_{W_{\lambda_i}}$ as

$$f_{W_{\lambda_i}}(x) = \begin{cases} 0 & \text{if } x \notin D, \\ \frac{1}{C(\lambda_i, b)} \cdot \frac{1}{2b} e^{-\frac{|x - \lambda_i|}{b}} & \text{if } x \in D’, \end{cases}$$

where $C(\lambda_i, b) = \int_D \frac{1}{2b} e^{-\frac{|x - \lambda_i|}{b}} dx$. $\diamondsuit$
Remark 1: This mechanism can be implemented by any means of sampling from the p.d.f. \( f_{W_i}(x) \). For the numerical results presented in Section VI, we use inverse transform sampling. In short, we sample a uniform random variable and transform it using the inverse cumulative distribution function defined by the mechanism [39].

Throughout the rest of the paper, we use \( b_e \) to denote the parameter \( b \) in Definition 4 when used to implement edge privacy in Definition 2.A, and we use \( b_n \) to denote the parameter \( b \) when used to implement node privacy in Definition 2.B.

C. Problem Statements

We now give formal problem statements. The first two pertain to the development of privacy mechanisms.

Problem 1: Develop a mechanism to provide \((\epsilon, \delta)-edge differential privacy in the sense of Definition 2.A for the spectrum of the graph Laplacian \( L(G) \) of a graph \( G \).

Problem 2: Develop a mechanism to provide \((\epsilon, \delta)-node differential privacy in the sense of Definition 2.B for the algebraic connectivity of a graph \( G \).

We note that Problem 2 considers the algebraic connectivity specifically because that will be used to show the poor scaling of node privacy for the full Laplacian spectrum. Comparisons of the two mechanisms are the subject of the next problem.

Problem 3: Given a graph \( G \) on \( n \) nodes and two privacy mechanisms, \( M_n \) and \( M_e \), that provide \((\epsilon, \delta)-node privacy and \((\epsilon, \delta)-edge privacy for the spectrum of \( L(G) \), respectively, analyze how the variances of the two mechanisms scale with respect to the size of the network \( n \).

The final two problem statements pertain to the accuracy of graph properties when bounded using private spectra.

Problem 4: Given a private algebraic connectivity, develop bounds on the expectation of the graph diameter and mean distance between nodes in the graph.

Problem 5: Given private values of the Laplacian spectrum, provide examples to numerically quantify the accuracy of using these private values to estimate the trace of the Laplacian, Kemeny’s constant, and Cheeger’s inequality.

III. PRIVACY MECHANISMS

In this section, we solve Problems 1 and 2. Specifically, we develop two mechanisms to provide \((\epsilon, \delta)-differential privacy to eigenvalues of a graph Laplacian \( L \). In Section III-A, we use edge differential privacy to privatize each of the Laplacian eigenvalues, \( \lambda_i \) for \( i \in [n] \). Then in Section III-B, we use node differential privacy to privatize \( \lambda_2 \). In both subsections we first bound the sensitivity appearing in Definition 3 and then use these sensitivity bounds to develop the privacy mechanisms.

A. Edge Privacy

We now design a mechanism to implement \((\epsilon, \delta)-edge differential privacy. We first bound the sensitivity \( \Delta \lambda_{i,e} \) appearing in Definition 3.A.

Lemma 1 (Edge sensitivity bound): Fix an adjacency parameter \( A \in \mathbb{N} \). Then for the edge sensitivity \( \Delta \lambda_{i,e} \) in Definition 3.A, we have

\[
\Delta \lambda_{i,e} \leq 2A
\]

for \( i \in \{1, \ldots, n\} \).

Proof: See Appendix A. ■

Next, we establish an algebraic relation for \( b_e \), which lets the bounded Laplace mechanism satisfy the theoretical guarantees of \((\epsilon, \delta)-edge differential privacy in Definition 2.A.

Theorem 1: Let \( \epsilon > 0 \) and \( \delta \in (0,1) \) be given. Fix \( n \in \mathbb{N} \) and consider graphs in \( G_n \). Then for the bounded Laplace mechanism \( W_{A \Delta} \) in Definition 4, choosing \( b_e \) according to

\[
b_e \geq \frac{2A}{\epsilon - \log \left( \frac{2e^{-\Delta \lambda_{i,e}/b_e}}{1-e^{-\Delta \lambda_{i,e}/b_e}} \right) - \log(1-\delta)}
\]

satisfies \((\epsilon, \delta)-edge differentially privacy with respect to \( \text{Adj}_{e,A} \) as defined in Definition 2.A.

Proof: By [34, Theorem 3.5], the bounded Laplace mechanism provides \((\epsilon, \delta)-differential privacy if

\[
\lambda_i \in [0, n], \Delta C(b_e) \text{ is defined as}
\]

\[
\Delta C(b_e) := \frac{C(\Delta \lambda_{i,e}, b_e)}{C(0, b_e)},
\]

where \( C \) is from Definition 4. Next, we find

\[
C(\lambda_i, b_e) = \int_0^{\lambda_i} \frac{1}{2b_e} e^{-\frac{x-\lambda_i}{b_e}} dx = \frac{1}{2b_e} \int_0^{\lambda_i} e^{-\frac{x-\lambda_i}{b_e}} dx + \frac{1}{2b_e} \int_{\lambda_i}^{\lambda_i} e^{-\frac{x-\lambda_i}{b_e}} dx
\]

\[
= 1 - \frac{1}{2} \left( e^{-\frac{-\lambda_i}{b_e}} + e^{-\frac{-n-\lambda_i}{b_e}} \right).
\]

Using (2) to compute \( C(\Delta \lambda_{i,e}, b_e) \) and \( C(0, b_e) \) in (1) gives

\[
\Delta C(b_e) = \frac{1 - \frac{1}{2} \left( e^{-\frac{-\lambda_i}{b_e}} + e^{-\frac{-n-\lambda_i}{b_e}} \right)}{1 - \frac{1}{2} (1 + e^{-\frac{-\lambda_i}{b_e}})}.
\]

Using the sensitivity bound in Lemma 1, we put \( \Delta \lambda_{i,e} = 2A \), which completes the proof. ■

Theorem 1 solves Problem 1, and we now have an \((\epsilon, \delta)-differential privacy mechanism for the spectrum of a graph Laplacian \( L \). We now shift our attention to node privacy and the algebraic connectivity, \( \lambda_2 \).

B. Node Privacy

Here we develop an \((\epsilon, \delta)-node differential privacy mechanism for \( \lambda_2 \). We will use the same process as the last subsection: we first bound \( \Delta \lambda_{2,n} \) from Definition 3.B for a graph \( G \in G_n \), then use this sensitivity to find an algebraic relation for the bounded Laplace mechanism to satisfy Definition 2.B.

In Lemma 1, we were able to derive a common bound on the sensitivity of each eigenvalue of \( L \) when edge sensitivity is used. There is no common bound when edge sensitivity is used. In Section IV, we show that the node privacy scales poorly with the size of the network and will not be usable in most engineering
problems. Thus, in this section we focus on $\lambda_2$ rather than the entire spectrum, as this is sufficient to illustrate the poor scaling of node privacy in this context.

Definition 1.B considers adjacent graphs as graphs that have an additional or absent node from $G$. Thus, for a $G'$ satisfying $\text{Adj}_n(G, G') = 1$, it is possible that $G' \in \mathcal{G}^{n-1}$ or $G' \in \mathcal{G}^{n+1}$. Because of this, we require $n \geq 3$ and the two cases will be handled separately in our analysis. We have the following result.

**Lemma 2:** Fix $n \in \mathbb{N}$ and consider graphs in $\mathcal{G}_n$. Then the node sensitivity of $\lambda_2$ in Definition 3.B is bounded as

$$\Delta \lambda_{2,n} \leq n - 1.$$  

**Proof:** See Appendix B.

With this sensitivity bound, we now establish an algebraic relation for $b_n$, which lets the bounded Laplace mechanism satisfy the theoretical guarantees of $(\epsilon, \delta)$-node differential privacy in Definition 2.B.

**Theorem 2:** Let $\epsilon > 0$ and $\delta \in (0, 1)$ be given. Fix $n \in \mathbb{N}$ and consider graphs in $\mathcal{G}_n$. Then for the bounded Laplace mechanism $W_{\lambda_2}$ in Definition 4, choosing $b_n$ according to

$$b_n \geq \frac{n - 1}{\epsilon - \log \left(\frac{2 - \epsilon}{1 - \epsilon} \frac{n!}{n^{n-1}}\right) - \log(1 - \delta)}$$

satisfies $(\epsilon, \delta)$-node differential privacy with respect to $\text{Adj}_n$ from Definition 1.B.

**Proof:** See Appendix C.

Theorem 2 solves Problem 2 and gives an $(\epsilon, \delta)$-node differential privacy mechanism for the algebraic connectivity, $\lambda_2$, of the graph Laplacian $L$. The expressions given in Theorems 1 and 2 define $b$ implicitly since $b$ appears on both sides of these expressions. In particular, although each expression has a unique minimum value of $b$ that satisfies it, this minimum value does not have a closed form representation. However, in [34], the authors provided a method to solve for $b$ numerically using the bisection method, and we use this algorithm in the rest of the paper to numerically solve for the value of $b$ any time one is needed to implement a mechanism.

The lack of an analytical expression for the required $b$ prevents us from immediately comparing the amounts of noise required by the two notions of privacy. The next section derives necessary conditions for the variances of noise required for edge and node privacy, which will allow us to compare how the two notions of privacy scale with the size of the network $n$.

**IV. SCALING LAWS**

In this section we will compare the notions of edge and node differential privacy to solve Problem 3. More specifically, we will analyze how the required variance of each privacy notion scales with the size of the network $n$. Here, we focus on the algebraic connectivity $\lambda_2$ to draw accurate comparisons between edge and node privacy. However, the edge privacy results can immediately be applied to the rest of the Laplacian’s spectrum and the scaling trends found here persist for each value of the Laplacian spectrum. To compare the two mechanisms, we fix a graph $G \in \mathcal{G}_n$ and privacy parameters $\epsilon$ and $\delta$. Then we define an edge and node privacy mechanism to provide $(\epsilon, \delta)$-differential privacy with parameters $b_e$ and $b_n$, respectively. Then we will analyze and compare the required values of $b_e$ and $b_n$ given this $\epsilon$ and $\delta$.

**A. Comparison of Mechanisms**

Recall that the requirements for the bounded Laplace mechanism to achieve $(\epsilon, \delta)$-differential privacy appearing in Theorems 1 and 2 are defined implicitly in $b_e$ and $b_n$, and the minimal values must be found numerically. To compare the two notions of privacy we find different conditions for $(\epsilon, \delta)$—edge and node differential privacy, which give an analytical expression for the growth of $b$. For node privacy, Corollary 1 presents a weaker, necessary condition for differential privacy. For edge privacy, Corollary 2 presents a stronger, sufficient condition for differential privacy. Corollary 1 will show that the required parameter for the bounded Laplace mechanism to achieve $(\epsilon, \delta)$—node differential privacy is strictly larger than the parameter required for the standard, unbounded Laplace mechanism from [10] to achieve the same level of privacy. This recovers a general-purpose result of the same kind presented in [34, Theorem 3.5].

We now introduce the two corollaries that quantify the scaling behavior of node and edge privacy. Corollary 1 provides a necessary condition for achieving node privacy. By examining a necessary condition, we gain insights into the graph properties that contribute to the scaling of any node privacy mechanism.

In contrast, Corollary 2 offers a separate sufficient condition for achieving edge privacy. By examining a sufficient condition, we show that edge privacy can be achieved with no dependence on $n$.

**Corollary 1:** Fix a graph $G \in \mathcal{G}_n$, $\epsilon > 0$, and $\delta \in (0, 1)$. Let $W_{\lambda_2}^n$ be a bounded Laplace mechanism with parameter $b_n$. Then

$$b_n > \frac{n - 1}{\epsilon - \log(1 - \delta)}$$

is a necessary condition for $W_{\lambda_2}^n$ to provide $(\epsilon, \delta)$—node differential privacy.

**Proof:** See Appendix D.

**Corollary 2:** Fix a graph $G \in \mathcal{G}_n$, $\epsilon > 0$, and $\delta \in (0, 1)$. Let $W_{\lambda_2}^n$ be a bounded Laplace mechanism with parameter $b_e$. Then

$$b_e > \frac{2A}{\epsilon - \log(2) - \log(1 - \delta)}$$

is a sufficient condition for $W_{\lambda_2}^n$ to provide $(\epsilon, \delta)$—edge differential privacy.

**Proof:** See Appendix E.

**Remark 2:** In Corollary 1, the necessary condition on $b_n$ for $(\epsilon, \delta)$—node differential privacy scales linearly with $n$. A standard Laplace distribution with parameter $b$ has variance $2b^2$. This means that as the size of the network $n$ grows, the variance required for $(\epsilon, \delta)$—node differential privacy grows quadratically in $n$. Simultaneously, in Corollary 2, the sufficient condition for $b_e$ has no dependence on the size of the network. Thus, $b_e$ and the variance of privacy noise needed for edge differential privacy can remain constant as a network grows. That is, the variance of
noise for node privacy is $O(n^2)$ while the variance of noise for edge privacy is $O(1)$.

The aforementioned scaling laws are further illustrated by numerical results shown in Fig. 1. These results show that the minimal $b_n$ required for node privacy grows quickly, while $b_c$ remains constant.

An appealing feature of differential privacy is that it provides a means to share private information that can still be useful. However, in many applications, variance that is $O(n^2)$ will render private information useless. Node differential privacy requires variance that is $O(n^2)$ and hence we do not expect node differentially private spectra to be useful. Thus, we focus on edge privacy for the rest of the paper.

### B. Accuracy of Edge Privacy

The edge privacy mechanism has the following accuracy.

**Theorem 3:** For a fixed $G \in \mathcal{G}_n$, $\epsilon > 0$, $\delta \in (0, 1)$, and $A \in \mathbb{N}$, the accuracy of a private eigenvalue $\tilde{\lambda}_i$ generated using the bounded Laplace mechanism with parameter $b_c$ is given by

$$E \left[ \tilde{\lambda}_i - \lambda_i \right] = \frac{1}{2C(\lambda_i, b_c)} \left( 2 \lambda_i b_c e^{-\frac{\lambda_i}{b_c}} - \left( n + b_c \right) e^{-\frac{n-\lambda_i}{b_c}} \right) - \lambda_i.$$

**Proof:** See Appendix F.

**Remark 3:** The error bound in Theorem 3 can be used by data curators to calibrate the strength of privacy. Specifically, when choosing $\epsilon$ and $\delta$, a data curator releasing private spectra can use Theorem 3 as a measure of the quality those private spectra in order to balance the strength of privacy with the accuracy of private data. To interpret Theorem 3, note that (i) $E[\tilde{\lambda}_i - \lambda_i] < 0$ when $\lambda_i < \frac{n}{2}$, i.e., the expected error is negative when $\lambda_i < \frac{n}{2}$, (ii) $E[\tilde{\lambda}_i - \lambda_i] > 0$ when $\lambda_i > \frac{n}{2}$, i.e., the expected error is positive when $\lambda_i > \frac{n}{2}$, and (iii) $E[\tilde{\lambda}_i - \lambda_i] = 0$ when $\lambda_i = \frac{n}{2}$, i.e., the expected error is zero when $\lambda_i = \frac{n}{2}$ exactly. Overall, this shows that the mechanism adds a bias to the private output unless the value being privatized is in the center of the interval $[0, n]$. Example 2 and Fig. 5 of Section VI demonstrate this bias and its relation to the privacy level with numerical experiments.

**Theorem 3** provides an analytical expression for the accuracy of the edge privacy mechanism. Since differential privacy is immune to post-processing, we can use private spectra to estimate other graph properties without harming privacy guarantees. The rest of the paper focuses on estimating other graph properties using the edge privacy mechanism. Specifically, in Section V we develop statistical bounds on other graph properties given a private $\lambda_2$, and in Section VI we provide a series of examples that demonstrate the accuracy of the edge privacy mechanism and illustrate how these private values of the Laplacian spectrum can be used to estimate other graph properties.

### V. Bounding Other Graph Properties

In this section we solve Problem 4. There exist numerous inequalities relating $\lambda_2$ to other quantitative graph properties [16], [38], and one can therefore expect that the private $\lambda_2$ will be used to estimate other quantitative characteristics of graphs.

For example, a network analyst may only have access to the private value $\tilde{\lambda}_2$ and wish to use it in some standard graph analyses. To illustrate the utility of doing so, in this section we bound the graph diameter $d$ and mean distance $\rho$ (defined in Section V-A below) in terms of the private value $\tilde{\lambda}_2$. Specifically, we post-process the private value $\tilde{\lambda}_2$ to estimate $d$ and $\rho$ by using it in bounds on $d$ and $\rho$ from the existing literature. We denote these estimates by $\hat{d}$ and $\hat{\rho}$.

It is important to note that differential privacy is immune to post-processing [10]. Therefore, if $\tilde{\lambda}_2$ is $(\epsilon, \delta)$-differentially private, then the estimates $\hat{d}$ and $\hat{\rho}$ are also $(\epsilon, \delta)$-differentially private if they are functions of $\tilde{\lambda}_2$. Hence, there is no need to develop new mechanisms for the private estimation of $d$ and $\rho$.

Instead, the mechanisms we develop can be applied to $\lambda_2$ to generate $\tilde{\lambda}_2$ and then the value of $\tilde{\lambda}_2$ can be used to bound $d$ and $\rho$ in a privacy-preserving way.

**A. Analytical Bounds**

Both $d$ and $\rho$ measure graph size and provide insight into how easily information can be transferred across a network [40]. Formally, they are defined as follows.

**Definition 5:** Fix $n \in \mathbb{N}$ and a graph $G = (V, E) \in \mathcal{G}_n$. Let $d_{ij}$ be the length of the shortest path from node $i \in V$ to node $j \in V$. The graph diameter is defined as the length of the longest shortest path, namely

$$d = \max_{i,j \in V} d_{ij}.$$  

The mean distance is defined to be the average of the shortest paths, namely

$$\rho = \frac{1}{n(n-1)} \sum_{i,j \in V} d_{ij}.$$
We will next estimate each one in terms of the private \( \lambda_2 \) and bound the error induced in these estimates by privacy. These bounds represent the types of calculations one can do with \( \lambda_2 \), and similar bounds can be easily derived, e.g., on minimal/maximal degree, edge connectivity, etc., because their bounds are proportional to \( \lambda_2 \) [29].

We first recall bounds from the literature.

**Lemma 3 (Diameter and Mean Distance Bounds [41]):** For an undirected, unweighted graph \( G \) on \( n \) nodes, define

\[
\bar{d}(\lambda_2, \alpha) = \left( 2 \sqrt{\frac{\lambda_n}{\lambda_2}} \left( \frac{\alpha^2 - 1}{4\alpha} + 2 \right) \right) \left( \log_\alpha \frac{n}{2} \right)
\]

\[
\bar{\rho}(\lambda_2, \alpha) = \left( \sqrt{\frac{\lambda_n}{\lambda_2}} \left( \frac{\alpha^2 - 1}{4\alpha} + 1 \right) \right) \left( \frac{n}{n-1} \right) \left( \frac{1}{2} + \log_\alpha \frac{n}{2} \right).
\]

Then for any fixed \( \lambda_2 > 0 \) and any \( \alpha > 1 \), the diameter \( d \) and mean distance \( \rho \) of the graph \( G \) are bounded via

\[
d(\lambda_2) = \frac{4}{n\lambda_2} \leq d \leq \bar{d}(\lambda_2, \alpha)
\]

\[
\rho(\lambda_2) = \frac{2}{(n-1)\lambda_2} + \frac{n-2}{2(n-1)} \leq \rho \leq \bar{\rho}(\lambda_2, \alpha).
\]

The least upper bounds can be derived by finding values of \( \alpha_d \) and \( \alpha_p \) which minimize \( \bar{d}(\lambda_2, \alpha) \) and \( \bar{\rho}(\lambda_2, \alpha) \), respectively.

A list of \( \alpha_d \) and \( \alpha_p \) values can be found in Table I in [41]. To quantify the impacts of using the private \( \lambda_2 \) in these bounds, we next bound the expectations of the private forms of \( d \) and \( \rho \).

These bounds use the upper incomplete gamma function \( \Gamma(\cdot, \cdot) \) and the imaginary error function \( \text{erfi}(\cdot) \), defined as

\[
\Gamma(s, x) = \int_x^{\infty} t^{s-1} e^{-t} dt
\]

and \( \text{erfi}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{t^2} dt \).

We generate private estimates of \( d \) and \( \rho \) using

\[
\hat{d} = \left( 2 \sqrt{\frac{n}{\lambda_2}} \left( \frac{\alpha^2 - 1}{4\alpha} + 2 \right) \right) \left( \log_\alpha \frac{n}{2} \right)
\]

\[
\hat{\rho} = \left( \sqrt{\frac{n}{\lambda_2}} \left( \frac{\alpha^2 - 1}{4\alpha} + 1 \right) \right) \left( \frac{1}{2} + \log_\alpha \frac{n}{2} \right),
\]

i.e., we simply plug \( \tilde{\lambda}_2 \) into the upper bound for each term.

Using the private \( \lambda_2 \), expectation bounds are as follows.

**Theorem 4 (Expectation bounds for \( d \) and \( \rho \): Solution to Problem 4):** For any \( \lambda_2 > 0 \), denote its private value by \( \lambda_2 \). Let \( \hat{d} \) and \( \hat{\rho} \) denote the estimates of the diameter and mean distance, respectively, when computed with \( \lambda_2 \). Then the expectations \( E[\hat{d}] \) and \( E[\hat{\rho}] \), obey

\[
\frac{4}{n E[\lambda_2]} \leq E[\hat{d}] \leq E[\tilde{\rho}(\lambda_2, \alpha_d)]
\]

and

\[
\frac{2}{n - 2} \leq \frac{n - 2}{2(n - 1)} \leq E[\hat{\rho}] \leq E[\tilde{\rho}(\lambda_2, \alpha_p)].
\]

where

\[
E[\tilde{\rho}(\lambda_2, \alpha_d)] = \left[ 2 \sqrt{\frac{n}{\lambda_2}} \left( \frac{\alpha_d^2 - 1}{4\alpha_d} \right) \right] \left( \log_\alpha \frac{n}{2} \right)
\]

\[
E[\tilde{\rho}(\lambda_2, \alpha_p)] = \left[ \sqrt{\frac{n}{\lambda_2}} \left( \frac{\alpha_p^2 - 1}{4\alpha_p} \right) \right] \left( \log_\alpha \frac{n}{2} \right)
\]

We can compute the expectation terms with \( \lambda_2 \) via

\[
E\left[ \frac{1}{\sqrt[\lambda_2]} \right] = \frac{1}{C(\lambda_2, b_c)} \left( \sqrt{\frac{\alpha_d}{b_c}} e^{-\frac{\alpha_d}{b_c}} \left( \text{erfi} \left( \frac{\lambda_2}{b_c} \right) \right) \right)
\]

\[
+ \sqrt{\frac{\alpha_p}{b_c}} \left( \Gamma \left( \frac{n}{2} \frac{\lambda_2}{b_c} \right) - \Gamma \left( \frac{\lambda_2}{b_c} \right) \right)
\]

\[
E[\tilde{\lambda}_2] = \frac{1}{2C(\lambda_2, b_c)} \left( 2\lambda_2 + b_c e^{-\frac{\lambda_2}{b_c}} - b_c e^{-\frac{n-\lambda_2}{b_c}} - n e^{-\frac{n-\lambda_2}{b_c}} \right),
\]

where \( C \) is from Definition 4.

**Proof:** See Appendix G.

**Remark 4:** Figs. 2 and 3 demonstrate that the bounds presented in Theorem 4 do not lose much accuracy as the strength of privacy increases. For example, in Fig. 2 when \( \lambda_2 = 10 \), we can use \( \epsilon = 1 \) which corresponds to strong privacy and incur only around 5% error in the upper bound, along with negligible error in the lower bound. Thus, the key takeaway from Theorem 4 is that bounds on diameter and mean distance with privacy are about as accurate as using those same bounds without privacy.

**Remark 5:** A larger \( \epsilon \) gives weaker privacy, and it results in a smaller value of \( b_c \) and a distribution of privacy noise that is more

| Quantity | D | n | Average % error | Variance of Error |
|----------|---|---|-----------------|------------------|
| \( \lambda_2(G) \) | 5 | 535 | 8.48% | 0.35 |
| \( \bar{d}(L(G)) \) | 5 | 535 | 1.62% | 7.47 x 10^{-3} |
| \( K(P) \) | 5 | 535 | 7.56% | 1.48 x 10^{-4} |
| \( \rho(\lambda_2) \) | 2.5 | 14 | 9.01% | 0.27 |

Values were computed using \( M = 10^6 \) private spectrum values. There are no columns for \( A \) and \( \delta \) since they are fixed at \( A = 1 \) and \( \delta = 0.05 \) for all simulations.
We can compute the eigenvalue $\lambda_i$ and the minimum $b$ required for $(\epsilon, \delta)$-differential privacy, either edge or node, then add noise with the bounded Laplace mechanism to get the private eigenvalue $\tilde{\lambda}_i$. More care must be taken when answering queries of multiple eigenvalues or the entire spectrum. Specifically, since we only consider connected graphs we will always have $\lambda_1 = 0$ and thus there is no need to privatize it. Furthermore, for $\lambda_i$ with $i \in \{2, \ldots, n\}$ we can define $n - 1$ independent mechanisms that provide $(\epsilon, \delta)$-differential privacy to each $\lambda_i$. In general, since we have $n - 1$ queries that are each individually $(\epsilon, \delta)$-differentially private, the privacy level for querying the entire spectrum is $((n - 1)\epsilon, (n - 1)\delta)$—differentially private due to the Composition Theorem [10, Theorem 3.16]. After privatizing the spectrum, the set $\{\tilde{\lambda}_i\}_{i=1}^n$ is no longer guaranteed to have the ordering $\lambda_1 \leq \cdots \leq \lambda_n$. In applications where the sorting of the private values is critical we can sort the private values prior to sharing them. Sorting does not harm privacy because it is post-processing on privatized data, but it will change the statistics of each $\tilde{\lambda}_i$.

For the remainder of this section we provide a series of examples illustrating the accuracy and utility of the edge privacy mechanism developed in Theorem 1. In each of the examples, we calculate a metric to quantify accuracy of the private information, and Table I gives statistical summaries of these quantities. In Examples 1-4 we fix the sensitive graph $G$ to be a Facebook user’s local social network which is available at [42]. This data set was originally presented in [17] and the graph we use here corresponds to the ego agent with ID 3437. This graph is un-directed and has $n = 535$ nodes, $|E| = 10,160$ edges, and an algebraic connectivity of $\lambda_2(G) \approx 1$.

**Example 1 (Accuracy):** Fix $G \in \mathcal{G}_{535}$ to be the graph from [42] with ego ID 3437. Fix $\epsilon = 5, \delta = 0.05$, and $A = 1$. We generated $M = 10^4$ private $\tilde{\lambda}_i$’s for each $i \in \{2, 200, 300, 400\}$ using an edge privacy mechanism $W_{i,b}^\epsilon$ with parameter $b_i$. Solving for the minimum $b_i$ required for $(5, 0.05)$—differential privacy gives $b_i = 0.458$. To quantify the accuracy of the private spectrum for a fixed $\epsilon$ and $\delta$ we analyze $\tilde{\lambda}_i - \lambda_i$ for $i \in \{2, 200, 300, 400\}$. A histogram of the accuracy for the $M = 10^4$ queries is shown in Fig. 4. For each of the eigenvalues, the error in the private information is heavily concentrated near 0. This trend persists for the rest of the $n = 535$ eigenvalues that are not shown here, as well as for larger networks with larger values of $n$. This shows that edge privacy provides accurate spectrum values for large networks, even with strong privacy.

In Fig. 4, it appears that there is a slight bias in the private spectrum values because the plots are not perfectly symmetric. This bias is made precise by Theorem 3, and it is a function of the underlying graph $G$ through its eigenvalues and a function of the privacy parameters $\epsilon$ and $\delta$ through $b_i$. This bias appears as a result of adding bounded noise. Specifically, the density we use to generate $\tilde{\lambda}_i$ has a peak at the true value $\lambda_i$, but is only supported on the interval $[0, n]$, which means that the expected value will not be $\lambda_i$ unless $\lambda_i = \frac{n}{2}$. Nonetheless, Fig. 4 shows that this bias is small even when using strong privacy.

**Example 2 (The Effect of $\epsilon$):** Fix $G \in \mathcal{G}_{535}$ to be the graph from [42] with ego ID 3437. Fix $\delta = 0.05$ and $A = 1$. Let $\epsilon$
presents the values of $\lambda_i$ for $i \in \{2, 200, 300, 400\}$. These results illustrate that edge privacy is able to achieve high accuracy, even under strong privacy.

For each $\epsilon$, $M = 10^5$ private values were generated to empirically compute the values of $E[\hat{\lambda}_i - \lambda_i]$ and $\text{Var}[\hat{\lambda}_i - \lambda_i]$. These results vary and take on values $\epsilon \in [0.1, 10]$. Then for each $\epsilon$, generate $M = 10^4$ private $\lambda_i$’s for $i \in \{2, 206, 535\}$ using an edge privacy mechanism $W^\epsilon_{\lambda_i}$ with parameter $b_e$. For a given $\epsilon$ and eigenvalue $\lambda_i$, we quantify the quality of the private information with the empirical values of $E[\hat{\lambda}_i - \lambda_i]$ and $\text{Var}[\hat{\lambda}_i - \lambda_i]$ taken over the $M = 10^4$ private values. Fig. 5 presents the values of $E[\hat{\lambda}_i - \lambda_i]$ and $\text{Var}[\hat{\lambda}_i - \lambda_i]$ for $\epsilon \in [0.1, 5]$. Recall that a larger $\epsilon$ implies weaker privacy.

In Fig. 5, as $\epsilon$ grows and privacy is weakened, both $E[\hat{\lambda}_i - \lambda_i]$ and $\text{Var}[\hat{\lambda}_i - \lambda_i]$ converge to 0 relatively quickly. This trend is consistent across the entire spectrum of the graph Laplacian. This shows that even with relatively strong privacy, for example $\epsilon = 3$, the private spectra we share are highly accurate. Here we also see that under strong privacy, given by small $\epsilon$, we are sharing values of $\lambda_2$ that are much larger than the true value, and we are sharing much smaller values of $\lambda_{535}$. This occurs because of adding bounded noise and because $\lambda_2$ and $\lambda_{535}$ are near the boundaries of the allowable output range $[0, n]$. This example also illustrates the loss of accuracy as privacy is strengthened.

**Example 3 (Trace of the Laplacian):** Fix $G \in \mathcal{G}_{535}$ to be the graph from [42] with ego ID 3437. Fix $\epsilon = 1$, $\delta = 0.05$, and $\lambda = 1$. Recall that the trace of a matrix $R \in \mathbb{R}^{n \times n}$ is given by the sum of its eigenvalues, i.e., $\text{Tr}(R) = \sum_{i=1}^{n} \lambda_i(R)$. Applying this to the graph Laplacian, we have $\text{Tr}(L) = \sum_{i=1}^{n} \lambda_i(L)$. The trace of the graph Laplacian can, for example, be used to compute the average degree of the network as $d_{avg} = \frac{1}{n} \text{Tr}(L)$. Suppose that we do not have access to $G$ or $L(G)$ and we only have the private spectrum values $\{\tilde{\lambda}_i\}_{i=1}^{n}$. Then we can use these eigenvalues to estimate the trace of $L$, as $\tilde{\text{Tr}}(L) = \sum_{i=1}^{n} \tilde{\lambda}_i(L)$. To analyze the accuracy of this estimate, $M = 10^4$ sets of private spectra were generated and used to estimate the trace. In Fig. 6 we give a histogram of values of $\tilde{\text{Tr}}(L) - \text{Tr}(L)$ for these trace estimates.

We can see in Fig. 6 that edge privacy generally provides accurate estimates of the trace, with the majority of private trace estimates falling within ±5% of the true trace value.

However, there is a bias in the distribution of private trace estimates, and we tend to overestimate the trace. To quantify this overestimate, we analyze $E[\tilde{\text{Tr}}(L) - \text{Tr}(L)]$. Plugging in $\tilde{\text{Tr}}(L) = \sum_{i=1}^{n} \tilde{\lambda}_i(L)$ and simplifying gives

$$E\left[\tilde{\text{Tr}}(L) - \text{Tr}(L)\right] = \sum_{i=1}^{n} E\left[\tilde{\lambda}_i(L)\right] - \lambda_i(L).$$

Then applying Theorem 3 gives

$$E\left[\tilde{\text{Tr}}(L) - \text{Tr}(L)\right] = \sum_{i=1}^{n} \frac{1}{2C(\lambda_i, b)} \left(2\lambda_i + be^{-\lambda_i} - (n + b)e^{-\frac{n+\lambda_i}{b}}\right) - \lambda_i(L),$$

where $C$ is from Definition 4. In Example 1, there was a small bias in the values of $\lambda_i$ due to using bounded noise to achieve differential privacy. Here the bias for the trace is larger because we are summing each $\lambda_i$ and the bias is amplified due to summing.
We use the private $\tilde{\phi}$ from [4] and use them to privately estimate $\lambda_i$. This is more connected than it really is, though these estimates are often fairly accurate.

Example 4 (Kemeny’s Constant): In network control, network level discrete-time consensus dynamics are governed by the matrix $P = I - \gamma L(G)$, where $\gamma$ is a step-size which must obey $\gamma \leq \frac{1}{\max_d \lambda_i}$ in order to achieve consensus [43, Theorem 2].

When $G$ is a connected, undirected graph, $P$ can be interpreted as the transition matrix of a symmetric Markov chain. The Kemeny constant of a Markov chain is the expected time it takes to transition from a state $i$ in a Markov chain to another state sampled from its stationary distribution and can be used to compute the error in consensus protocols subject to noise [44].

The Kemeny constant of the Markov chain with transition matrix $P = I - \gamma L(G)$ can be computed as $K(P) = \sum_{i=2}^{\infty} \frac{1}{1 - \gamma_i(P)}$ [45]. Note that $\lambda_i(P) = 1 - \gamma \lambda_i(L)$ and thus $K(P) = \frac{1}{\gamma} \sum_{i=2} \frac{1}{\lambda_i(L)}$. Given private spectrum values we can estimate the Kemeny constant as $\tilde{K}(P) = \frac{1}{\tilde{\gamma}} \sum_{i=2}^{\infty} \tilde{\lambda}^{-1}(i)$. We fix $\tilde{\gamma} = \frac{1}{\gamma}$, and with this step-size the graph $G \in G_{535}$ from [42] has $\tilde{K}(P) = 32,985.57$. Note that $\tilde{K}(P)$ is prone to numerical instabilities as it consists of summing $n - 1 = 534$ random variables that have a positive probability of being close to 0. Thus, to avoid numerical instability we constrain the domain of the mechanism to $[0.2, n]$ rather than $[0, n]$.

We now fix $\epsilon = 5$, $\delta = 0.05$, and $A = 1$. We generate $M = 10^4$ private spectra for $G \in G_{535}$ from [42] and these values are used to compute $K(P)$. To quantify the accuracy of the estimates of the Kemeny constant, we analyze the relative error $\frac{\tilde{K}(P) - K(P)}{K(P)}$ whose values for the $M = 10^4$ private spectra are presented in Fig. 7. Here, we can see that we overestimate the Kemeny constant, but the average error for these queries is only 7.56%. This shows that sharing the private spectrum can share relatively accurate information about the Kemeny constant and thus about discrete-time consensus dynamics while providing edge differential privacy.

Example 5 (Cheeger’s Inequality): In this example we discuss how private Laplacian spectra can be used to estimate the isoperimetric number, $\phi(G)$, of a graph $G$. The isoperimetric number, or the Cheeger constant, is a measure of how connected a graph is or more specifically how easy it is to disconnect a graph [46].

In general, the isoperimetric number is NP-hard to compute and Cheeger’s inequality gives an easily computable upper bound on the isoperimetric number via $\phi(G) \leq \phi_{ab} := \sqrt{\lambda_2(\max_i d_i - \lambda_2)}$ [46, Theorem 4.2].

In this example, we estimate $\phi(G)$ using Cheeger’s inequality for cases in which we do not have access to $G$ and only have its private Laplacian spectrum. To estimate $\lambda_2$, we use the private value $\tilde{\lambda}_2$. For $\max_i d_i$, we estimate this with $d(G) = \frac{1}{n} \sum_{i=1}^{n} \tilde{\lambda}_i$. Then plugging these estimates into Cheeger’s inequality, we have the estimate $\phi(G) = \sqrt{\tilde{\lambda}_2(2d(G) - \tilde{\lambda}_2)}$.

Since the isoperimetric number is not feasible to compute for large networks, we cannot run simulations on the graph $G \in G_{535}$ from [42] to demonstrate the accuracy of our estimates. Thus, we fix $G$ to be the cycle or ring graph on $n$ nodes, $C_n$, which has a known Cheeger’s constant of $\phi(C_n) = \frac{\pi}{n}$ [47]. For this example, we fix $n = 14$. The graph $G = C_{14}$ is shown in Fig. 8 and $\phi(C_{14}) = \frac{\pi}{7}$. To analyze the accuracy of using Cheeger’s inequality with private spectra, we generate $M = 10^4$ private Laplacian spectra $\{\tilde{\lambda}_i\}_{i=1}^{n}$ and use them to privately estimate $\phi(G)$.

Before discussing the accuracy of our estimates we will discuss the accuracy of the Cheeger’s inequality itself. For $C_{14}$, we have $\phi(C_{14}) = \frac{\pi}{7} = 0.2857$ and plugging in $\tilde{\lambda}_2$ and $\max_i d_i = 2$ into Cheeger’s inequality gives $\phi_{ab} = 0.8678$. This is more
than 3 times the true value. Thus, to distinguish between errors inherent to Cheeger’s inequality itself and errors due to privacy, we will compare our estimate to the upper bound from Cheeger’s inequality, $\phi_{ub}$.

In Fig. 9, we show the accuracy of the resulting estimates given by $\phi(G) \pm 3\phi_{ub}$ for $10^{4}$ queries satisfying (2.5, 0.05)–differential privacy. Here, we typically over estimate Cheeger’s constant. This means that we are estimating that the graph is more connected than it truly is. Comparing to the non-private Cheeger’s inequality upper bound given by $\phi_{ub}$, the use of private spectra in computations results in a slightly looser bound on average. However the estimates are relatively accurate with an average normalized error of 9.01%, with a variance of only 0.27. Overall, this example shows that using private spectrum information to estimate the isoperimetric number is relatively accurate and does not have much more error than when true spectrum values are used. $\triangle$

VII. CONCLUSION

This paper presented two differential privacy mechanisms for edge and node privacy of the spectra of graph Laplacians of unweighted, undirected graphs. Bounded noise was used to provide private values that are still accurate, and the private values of Laplacian spectrum were shown to give accurate estimates of the diameter and mean distance of a graph, the trace of the Laplacian, the Kemeny constant, and Cheeger’s inequality. Future work includes the development of new privacy mechanisms for other algebraic graph properties.

REFERENCES

[1] W. Ren, R. W. Beard, and E. M. Atkins, “A survey of consensus problems in multi-agent coordination,” in Proc. Amer. Control Conf., 2005, pp. 1859–1864.
[2] J. Scott, “Social network analysis,” Sociology, vol. 22, no. 1, pp. 109–127, 1988.
[3] M. D. Shirley and S. P. Rushton, “The impacts of network topology on disease spread,” Ecological Complexity, vol. 2, no. 3, pp. 287–299, 2005.
[4] Y. Zheng, L. Wang, and Y. Zhu, “Consensus of heterogeneous multi-agent systems,” IET Control Theory Appl., vol. 5, no. 16, pp. 1881–1888, 2011.
[5] P. Van Mieghem, J. Omic, and R. Kooij, “Virus spread in networks,” IEEE/ACM Trans. Netw., vol. 17, no. 1, pp. 1–14, Feb. 2009.
[6] S. Freitas, D. Yang, S. Kumar, H. Tong, and D. H. Chau, “Evaluating graph vulnerability and robustness using turing,” in Proc. 30th ACM Int. Conf. Inf. Knowl. Manage., 2021, pp. 4945–4950.
[7] S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith, “Analyzing graphs with node differential privacy,” in Proc. 10th Theory Cryptology: Conf. Theory Cryptology, 2013, pp. 457–476.
[8] V. Karwa, S. Raskhodnikova, A. Smith, and G. Yaroslavtsev, “Private analysis of graph structure,” ACM Trans. Database Syst., vol. 39, no. 3, pp. 1–33, 2014.
[9] V. Karwa, S. Raskhodnikova, and A. Smith, “On the ‘semantics’ of differential privacy: A Bayesian formulation,” J. Privacy Confidentiality, vol. 6, no. 1, pp. 1–16, 2014.
[10] C. Dwork and A. Roth, “The algorithmic foundations of differential privacy,” Found. Trends Theor. Comput. Sci., vol. 9, no. 3, pp. 211–407, 2013.
[11] W.-Y. Day, N. Li, and M. Lyu, “Publishing graph degree distribution with node differential privacy,” in Proc. Int. Conf. Manage. Data, 2016, pp. 123–138.
[12] E. Shen and T. Yu, “Mining frequent graph patterns with differential privacy,” in Proc. 19th ACM Int. Conf. Knowl. Discov. Data Mining, 2013, pp. 545–555.
[13] X. Ding, X. Zhang, Z. Bao, and H. Jin, “Preserving-privacy preserving counting in large graphs,” in Proc. 27th ACM Int. Conf. Inf. Knowl. Manage., 2018, pp. 1283–1292.
[14] M. Hay, C. Li, G. Miklau, and D. Jensen, “Accurate estimation of the degree distribution of private networks,” in Proc. IEEE 9th Int. Conf. Data Mining, 2009, pp. 169–178.
[15] C. Task and C. Clifton, “A guide to differential privacy theory in social network analysis,” in Proc. IEEE Int. Conf. Adv. Soc. Netw. Anal. Mining, 2012, pp. 411–417.
[16] N. M. M. de Abreu, “Old and new results on algebraic connectivity of graphs,” Linear Algebra Appl., vol. 423, no. 1, pp. 53–73, 2007.
[17] J. Leskovec and J. Mcauley, “Learning to discover social circles in ego networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 539–547.
[18] B. Liu, M. Ding, S. Shaham, W. Rahayu, F. Farokhi, and Z. Lin, “When machine learning meets privacy: A survey and outlook,” ACM Comput. Surv., vol. 54, no. 2, pp. 1–36, 2021.
[19] U. V. Luxburg, “A tutorial on spectral clustering,” Statist. Comput., vol. 17, pp. 395–416, 2007.
[20] M. Belkin and P. Niyogi, “Laplacian eigenmaps and spectral techniques for embedding and clustering,” in Proc. Adv. Neural Inf. Process. Syst., 2001, pp. 585–591.
[21] M. F. Demirci, R. H. van Leuken, and R. C. Veltkamp, “Indexing through Laplacian spectra,” Comput. Vis. Image Understanding, vol. 110, no. 3, pp. 312–325, 2008.
[22] X. Ying, L. Wu, and X. Wu, “A spectrum-based framework for quantifying randomness of social networks,” IEEE Trans. Knowl. Data Eng., vol. 23, no. 12, pp. 1842–1856, Dec. 2011.
[23] S. D. Pauls and D. Remondini, “Measures of centrality based on the spectrum of the Laplacian,” Phys. Rev. E, vol. 85, no. 6, 2012, Art. no. 066127.
[24] V. Karwa, S. Raskhodnikova, A. Smith, and G. Yaroslavtsev, “Private analysis of graph structure,” Proc. VLDB Endowment, vol. 4, no. 11, pp. 1146–1157, 2011.
[25] C. Hawkins and M. Hale, “Differentially private formation control,” in Proc. IEEE 59th Conf. Decis. Control, 2020, pp. 6260–6265.
[26] P. Gohari, M. Hale, and U. Topcu, “Privacy-preserving policy synthesis in Markov decision processes,” in Proc. IEEE 59th Conf. Decis. Control, 2020, pp. 6266–6271.
[27] P. Gohari, B. Wu, C. Hawkins, M. Hale, and U. Topcu, “Differential privacy on the unit simplex via the dirichlet mechanism,” IEEE Trans. Inf. Forensics Secur., vol. 16, pp. 2326–2340, 2021.
[28] J. Cortés, G. E. Dullerud, S. Han, J. Le Ny, S. Mitra, and G. J. Pappas, “Differential privacy in control and network systems,” in Proc. IEEE 55th Conf. Decis. Control, 2016, pp. 4252–4272.
[29] M. Fiedler, “Algebraic connectivity of graphs,” Czechoslovak Math. J., vol. 23, no. 2, pp. 298–305, 1973.
[30] R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” IEEE Trans. Autom. Control, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.
[31] W. Ren and E. Atkins, “Distributed multi-vehicle coordinated control via local information exchange,” Int. J. Robust Nonlinear Control, vol. 17, pp. 1002–1033, 2007.
[32] M. C. De Gennaro and A. Jadbabaie, “Decentralized control of connectivity for multi-agent systems,” in Proc. IEEE 45th Conf. Decis. Control, 2006, pp. 3628–3633.
[33] A. Nedich, A. Olshervsky, and W. Shi, Decentralized Consensus Optimization and Resource Allocation, Cham, Switzerland: Springer, 2018, pp. 247–287.
[34] N. Holohan, S. Antonatos, S. Braghin, and P. Mac Aonghusa, “The bounded Laplace mechanism in differential privacy,” J. Privacy Confidentiality, vol. 10, no. 1, pp. 1–15, 2020.
[35] Y. Wang, X. Wu, and L. Lu, “Differential privacy preserving spectral graph analysis,” in Proc. 17th Pacific-Asia Conf. Knowl. Discov. Data Mining, 2013, pp. 329–340.
[36] B. Chen, C. Hawkins, K. Yazdani, and M. Hale, “Edge differential privacy for algebraic connectivity of graphs,” in Proc. IEEE 60th Conf. Decis. Control, 2021, pp. 2764–2769.
[37] M. Fiedler, “A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory,” Czechoslovak Math. J., vol. 25, no. 4, pp. 619–633, 1975.
[38] M. Mesbah and M. Egerstedt, Graph Theoretic Methods in Multiagent Networks, Princeton, NJ, USA: Princeton Univ. Press, 2010.
[39] L. Devroye, “Sample-based non-uniform random variate generation,” in Proc. 18th Conf. Winter Simul., 1986, pp. 260–265.
M. J. Paldino, W. Zhang, Z. D. Chu, and F. Golriz, “Metrics of brain network architecture capture the impact of disease in children with epilepsy,” *NeuroImage: Clin.*, vol. 13, pp. 201–208, 2017.

B. Mohar, “Eigenvalues, diameter, and mean distance in graphs,” *Graph. Combinatorics*, vol. 7, no. 1, pp. 53–64, 1991.

J. Leskovec and A. Krevl, “SNAP datasets: Stanford large network dataset collection,” Jun. 2014. [Online]. Available: http://snap.stanford.edu/data

R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

A. Jadbabaie and A. Olshevsky, “Scaling laws for consensus protocols subject to noise,” *IEEE Trans. Autom. Control*, vol. 64, no. 4, pp. 1389–1402, Apr. 2019.

M. Levene and G. Loizou, “Kemeny’s constant and the random surfer,” *Amer. Math. Monthly*, vol. 109, no. 8, pp. 741–745, 2002.

B. Mohar, “Isoperimetric numbers of graphs,” *J. Combinatorial Theory, Ser. B*, vol. 47, no. 3, pp. 274–291, 1989.

C. Godsil and G. F. Royle, *Algebraic Graph Theory*. Berlin, Germany: Springer Science & Business Media, 2001.

**Calvin Hawkins** received the bachelor’s degree (summa cum laude) in mechanical engineering from Wayne State University, Detroit, MI, USA, in 2019. He is currently working toward the Ph.D. degree with the University of Florida, Gainesville, FL, USA. His research interests include the area of privacy in control, with an emphasis on bringing differential privacy to new classes of data typically used by control systems and quantifying the effects of privacy upon feedback. He was the recipient of the Graduate School Preeminence Award.

**Bo Chen** received the Bachelor of Engineering degree in aerospace engineering from Northwestern Polytechnical University, China, in 2017, and the Master of Science degree in aerospace engineering from the University of Florida, Gainesville, FL, USA, in 2019. He is currently working toward the Ph.D. degree in mechanical engineering under the guidance of Dr. Hale. His research interests include differential privacy and its applications in Markov decision processes and reinforcement learning.

**Kasra Yazdani** received the B.S. degree in mechanical engineering from Shiraz University, Shiraz, Iran, in 2015, and the M.S. degrees in mechanical engineering and computer science and the Ph.D. degree in mechanical engineering from the University of Florida, Gainesville, FL, USA, in 2019 and 2022, respectively. He is a Senior Machine Learning Research Engineer with Samsung Semiconductor Inc., San Jose, CA, USA.

**Matthew Hale** (Member, IEEE) received the B.S.E. degree in electrical engineering from the University of Pennsylvania, Philadelphia, PA, USA, in 2012, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2015 and 2017, respectively. He is currently an Assistant Professor of mechanical and aerospace engineering with the University of Florida, Gainesville, FL, USA. His research interests include multi-agent systems, mobile robotics, privacy in control, and distributed optimization. He was the recipient of the NSF CAREER Award, ONR YIP, and AFOSR YIP.