Generalization in Graph Neural Networks: Improved PAC-Bayesian Bounds on Graph Diffusion

Haotian Ju*, Dongyue Li*, Aneesh Sharma†, and Hongyang R. Zhang*

*Northeastern University, Boston
†Google, Mountain View

Abstract

Graph neural networks are widely used tools for graph prediction tasks. Motivated by their empirical performance, prior works have developed generalization bounds for graph neural networks, which scale with the graph structure in terms of the max degree. This paper shows generalization bounds that instead scale with the largest singular value of the graph neural network's diffusion matrix. These bounds are numerically much smaller than prior bounds for real-world graphs. A lower bound of the generalization gap is also constructed, which matches the bounds on the graph diffusion matrix. To achieve these results, we analyze a unified model that includes prior works' settings (i.e., convolutional and message-passing networks) and new settings (i.e., graph isomorphism networks). Our analysis examines the model's noise stability in PAC-Bayesian bounds using Hessians. Empirical results reveal that the Hessian-based bound closely matches observed generalization gaps of graph neural networks. Optimizing the noise stability properties of the model also leads to better generalization and test performance for several graph classification tasks.

1 Introduction

A central measure of success for a machine learning model is the ability to generalize well from the training set to the test set. For linear and shallow models, the generalization gap between their testing performance and training performance can be quantified by complexity notions such as the Vapnik–Chervonenkis dimension and Rademacher complexity. However, formally explaining the empirical generalization performance of deep models remains a challenging problem and an active research area [22]. There are by now many studies for fully-connected and convolutional neural networks that provide an explanation for their superior empirical performance [4, 39]. Our work seeks to formally understand generalization in graph neural networks (GNN) [41], which are commonly used for learning on graphs [20].

As an example of a concrete setting in which understanding generalization is crucial for performance, we consider the pretraining and fine-tuning of graph neural networks [24]. In the pretraining stage, a GNN is trained on a diverse range of graphs. Then, in the fine-tuning stage, the pretrained model is adapted to a specific prediction task. A practical challenge is that, on the one hand, the model requires enough parameters to fit the set of input graphs in the pretraining stage. On the other hand, the pretrained model could overfit and incur large generalization losses in the fine-tuning stage. Therefore, a better understanding of generalization could inform more robust fine-tuning of graph neural networks.

Published in Artificial Intelligence and Statistics (AISTATS), 2023. Email correspondence should be directed to ⟨ho.zhang@northeastern.edu⟩.

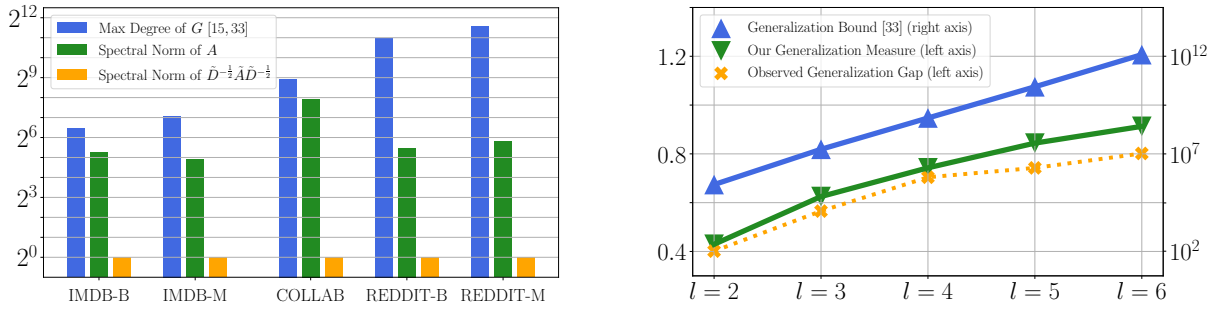


Figure 1: **Left**: Our spectral norm bounds for GNN are orders of magnitude smaller than maximum degree bounds on real-world graphs; see also Figure 2, which further measures weight norms in the comparison. **Right:** Our noise stability analysis yields a generalization measure that matches observed generalization gaps of graph neural networks.

Previous work has studied generalization in GNN by incorporating the graph structure into generalization bounds of feedforward neural networks [4, 38]. Verma and Zhang (2019) find that one-layer graph neural networks satisfy uniform stability properties [45], following the work of Hardt et al. [23]. Their generalization bound scales with the largest singular value (a.k.a. spectral norm) of the graph diffusion matrix of the model. However, their analysis only applies to a single layer and node prediction. Garg et al. (2020) analyze an l layer message passing neural net — with $l-1$ graph diffusion layers and 1 pooling layer — for graph prediction tasks [15]. Their generalization bound scales with d^{l-1}, where d is the maximum degree of the graph. Subsequently, Liao et al. (2021) develop a tighter generalization bound but still scales with d^{l-1} [33]. For both results, the max degree is used to upper bound the complexity of node aggregation in each diffusion step.

Our analysis approach measures the stability of a graph neural network against noise injections. Let f denote an l-layer GNN. It is known that the generalization gap of f will be small if f remains stable against noise injections; otherwise, the generalization gap of f will be large, using PAC-Bayesian bounds [35]. By quantifying noise stability via Lipschitzness of activation functions, one can get strong generalization bounds for feedforward nets that correlate with their observed generalization gaps [12, 2, 28, 31]. Thus, using a refined noise stability analysis in the setting of graph neural networks, we can obtain much tighter generalization bounds on the graph diffusion matrix.

Our Contributions. The goal of this work is to improve the theoretical understanding of generalization in GNN, and in that vein, we highlight two results below:

- First, we prove sharp generalization bounds for message passing neural networks [9, 16, 30], graph convolutional networks [32], and graph isomorphism networks [50]. Our bounds scale with the spectral norm of P_G^{l-1} for an l-layer network, in which P_G denotes a diffusion matrix on a graph G and varies between different models (see Theorem 3.1 for the full statement). We also show a matching lower bound whose generalization gap scales with the spectral norm of P_G^{l-1} in the instance (see Theorem 3.2).
- Second, our stability analysis of graph neural networks provides a practical tool for measuring generalization. Namely, we show that the noise stability of GNN can be measured by the trace of the (loss) Hessian matrix. The formal statement is given in Lemma 4.3, and our techniques, which include Taylor's expansion and uniform convergence, may be of independent interest. We note that the proof applies to twice-differentiable and Lipschitz-continuous activation functions (e.g., tanh and sigmoid).

Taken together, these two results provide a sharp understanding of generalization in terms of the graph structure for graph neural networks. We note that the numerical value of our bounds is much smaller than

prior results [15, 33], as is clear from Figure 1 (left). Further, the bounds provided by our noise stability analysis also match the empirically observed generalization gap as shown in Figure 1 (right).

Finally, motivated by the above analysis, we also present an algorithm that performs gradient updates on perturbed weight matrices of a GNN. The key insight is that minimizing the average loss of multiple perturbed models with independent noise injections is equivalent to regularizing f's Hessian in expectation. We conduct experiments on several graph classification tasks with Molecular graphs that show the benefit of this algorithm in the fine-tuning setting.

2 Related Work

Generalization Bounds: An article by Zhang et al. (2017) finds that deep nets have enough parameters to memorize real images with random labels, yet they still generalize well if trained with true labels. This article highlights the overparametrized nature of modern deep nets (see also a recent article by Arora [1]), motivating the need for complexity measures beyond classical notions. In the case of two-layer ReLU networks, Neyshabur et al. (2019) show that (path) norm bounds better capture the "effective number of parameters" than VC dimension—which is the number of parameters for piecewise linear activation mappings [5].

For multilayer networks, subsequent works have developed norm, and margin bounds, either via Rademacher complexities [4, 17, 34], or PAC-Bayesian bounds [2, 38, 31]. All of these bounds apply to the fine-tuning setting following the distance from the initialization perspective. Our analysis approach builds on the work of Arora et al. (2018) and Ju et al. (2022). The latter work connects perturbed losses and Hessians for feedforward neural networks, with one limitation Hessians do not show any explicit dependence on the data. This is a critical issue for GNN as we need to incorporate the graph structure in the generalization bound. Our result instead shows an explicit dependence on the graph and applies to message-passing layers that involve additional nonlinear mappings. We will compare our analysis approach and prior analysis in more detail when we present the proofs in Section 4 (see Remark 4.4).

Graph Representation Learning: Most contemporary studies of learning on graphs consider either node-level or graph-level prediction tasks. Our result applies to graph prediction while permitting an extension to node prediction: see Remark 4.2 in Section 4. Most graph neural networks follow an information diffusion mechanism on graphs [41]. Early work takes inspiration from ConvNets and designs local convolution on graphs, e.g., spectral networks [6], GCN [32], and GraphSAGE [21] (among others). Subsequent works have designed new architectures with graphs attention [44] and isomorphism testing [50]. Gilmer et al. (2017) synthesize several models into a framework called message-passing neural networks. Besides, one could also imbue graph structure in the pooling layer (e.g., differentiable pooling and hierarchical pooling [55, 57]). It is conceivable that one can incorporate the model complexity of these approaches into our analysis. Recent work applies pretraining to large-scale graph datasets for learning graph representations [24]. Despite being an effective transfer learning approach, few works have examined the generalization of graph neural nets in the fine-tuning step.

Besides learning on graphs, GNNs are also used for combinatorial optimization [43] and causal reasoning [51]. There is another approach for graph prediction using kernels [46, 11]. For references, see review articles [20, 7, 13, 49].

Generalization in GNN: It is known that the VC dimension of GNN scales with the number of nodes in the graph [42]. By contrast, norm-based bounds can only be measured from data. Recent work explores generalization by formalizing the role of the algorithm, and the alignment between networks and tasks [52]. Besides, there are works about size generalization, which refer to performance degradation when models extrapolate to graphs of different sizes from the input [43, 54]. It is conceivable that the new tools we have developed may be useful for studying extrapolation.

Expressivity of GNN: The expressivity of GNN for graph classification can be related to graph isomorphism tests. It is known that GNNs are equivalent to one-dimensional Weisfeiler-Lehman testing of graph isomorphism [37, 50]. This implies limitations of GNN for expressing tasks such as counting cycles [40, 8, 3]. The expressiveness view seems orthogonal to generalization, which instead concerns the sample efficiency of learning. For further discussions and references, see an excellent survey by Jegelka [27].

3 Sharp Generalization Bounds for Graph Neural Networks

We first introduce the problem setup for analyzing graph neural networks. Then, we state our generalization bounds for graph neural networks and compare them with the prior art. Lastly, we construct an example to argue that our bounds are tight.

3.1 Problem setup

Consider a graph prediction task. Suppose we have N examples in the training set; each example is an independent sample from a distribution denoted as $\mathcal{D}$. In each example, we have an undirected graph denoted as $G = (V, E)$, which describes the connection between n entities, represented by nodes in V. For example, a node could represent a molecule, and an edge between two nodes indicates a bond between two molecules. Each node also has a list of d features. Denote all node features as an n by d matrix X. For graph-level prediction tasks (We will describe a few standard datasets later in Section 5.2), the goal is to predict a graph label y for every example.

Message passing neural networks (MPNN). We study a model based on several prior works for graph-level prediction tasks [9, 16, 15, 33]. Let l be the number of layers: the first $l-1$ layers are diffusion layers, and the last layer is a pooling layer. Let d_t denote the width of each layer for t from 1 up to l. There are several nonlinear mappings in layer t, denoted as ϕ_t, ρ_t, and ψ_t; further, they are all centered at zero. There is a weight matrix $W^{(t)}$ (of size d_{t-1} by d_t) for transforming neighboring features and another matrix $U^{(t)}$ (of size d by d_t) for transforming the anchor node feature.

For the first $l-1$ layers, the model recursively computes the node embedding from the input features $H^{(0)} = X$ as:

$$H^{(t)} = \phi_t\Big(XU^{(t)} + \rho_t\big(P_G\psi_t(H^{(t-1)})\big)W^{(t)}\Big). \tag{1}$$

For the last layer l, we aggregate the embedding of all nodes: let $\mathbf{1}_n$ be a vector with n values of one, the final output is:

$$H^{(l)} = \frac{1}{n}\mathbf{1}_n^\top H^{(l-1)}W^{(l)}. \tag{2}$$

Our model takes many existing graph diffusion designs into consideration. In step (1), the graph diffusion matrix P_G could be the adjacency matrix A of the graph. Or, P_G could be the normalized adjacency matrix—$D^{-1}A$, with D being the degree-diagonal matrix—for averaging the neighboring embedding. Adding an identity matrix in P_G is the same as adding self-loops in G. To get the GCN, one can set $U^{(t)}$ as zero, ρ_t and ψ_t as identity mappings.

Notations. For any matrix X, let $\|X\|$ denote the largest singular value (a.k.a. spectral norm) of X. Let $\|X\|_F$ denote the Frobenius norm of X. We use the notation $f(N) \lesssim g(N)$ to indicate that there exists a fixed constant c that does not grow with N such that $f(N) \le c \cdot g(N)$ for large enough values of N. Let $\boldsymbol{W}$ and $\boldsymbol{U}$ denote the union of the W and U matrices in a model f, respectively.

3.2 Main results

Given a message-passing neural network denoted as f, what can we say about its generalization performance? Let $f(X, G)$ denote the output of f, given input with graph G, node feature matrix X, and label y. The loss of f for this input example is denoted as $\ell(f(X, G), y)$. Let $\hat{L}(f)$ denote the empirical loss of f over the training set. Let $L(f)$ denote the expected loss of f over a random example of distribution $\mathcal{D}$. We are interested in the generalization gap of f, i.e., $L(f) - \hat{L}(f)$. How would the graph structure of G affect the generalization gap of graph neural networks?

To motivate our result, we examine the effect of incorporating graph diffusion in a one-layer linear neural network. That is, we consider $f(X, G)$ to be $\frac{1}{n}\mathbf{1}_n^\top P_G X W^{(1)}$, which does not involve any nonlinear mapping for simplicity of our discussion. In this case, by standard spectral norm inequalities for matrices, the Euclidean norm of f (which is a vector) satisfies:

$$\begin{aligned} \|f(X, G)\| &= \left\| \frac{1}{n}\mathbf{1}_n^\top P_G X W^{(1)} \right\| \\ &\leq \left\| \frac{1}{n}\mathbf{1}_n^\top \right\| \cdot \left\| P_G \right\| \cdot \|X\| \cdot \left\| W^{(1)} \right\| \end{aligned} \tag{3}$$

Thus, provided that the loss function $\ell(\cdot, y)$ is Lipschitz-continuous, standard arguments imply that the generalization gap of f scales with the spectral norm of P_G (divided by $\sqrt{N}$) [36]. Let us compare this statement with a fully-connected neural net that averages the node features, i.e., the graph diffusion matrix P_G is the identity matrix. The spectral norm of P_G becomes one. Together, we conclude that the graph structure affects the generalization bound of a single layer GNN by adding the spectral norm of P_G.

Our main result is that incorporating the spectral norm of the *graph diffusion matrix* P_G^{l-1} is sufficient for any l layer MPNN (recall its definition from equation (1), which involves $l-1$ graph diffusion layers). Let f be an l-layer network whose weights $\boldsymbol{W}, \boldsymbol{U}$ are defined within a hypothesis set $\mathcal{H}$: For every layer i from 1 up to l, we have that

$$\begin{aligned} \left\| W^{(i)} \right\| \leq s_i, \quad \left\| W^{(i)} \right\|_F \leq s_i r_i, \\ \left\| U^{(i)} \right\| \leq s_i, \quad \left\| U^{(i)} \right\|_F \leq s_i r_i, \end{aligned} \tag{4}$$

where $s_1, s_2, \ldots, s_l$ and $r_1, r_2, \ldots, r_l$ are bounds on the spectral norm and stable rank and are all greater than or equal to one, without loss of generality. We now present the full statement.

Theorem 3.1. *Suppose the nonlinear activation in $\{\phi_t, \rho_t, \psi_t : \forall\, t\}$ and the loss $\ell(\cdot, y)$ are twice-differentiable, Lipschitz-continuous, and their first-order and second-order derivatives are all Lipschitz-continuous.*

With probability at least $1 - \delta$ over the randomness of N independent samples from $\mathcal{D}$, for any $\delta > 0$, and any $\epsilon > 0$ close to zero, any model f with weight matrices in the set $\mathcal{H}$ satisfies:

$$L(f) \leq (1+\epsilon)\hat{L}(f) + \mathrm{O}\left(\frac{\log(\delta^{-1})}{N^{3/4}}\right) + \sum_{i=1}^{l} \sqrt{\frac{CBd_i\left(\max_{(X,G,y)\sim\mathcal{D}} \|X\|^2 \|P_G\|^{2(l-1)}\right)\left(r_i^2 \prod_{j=1}^{l} s_j^2\right)}{N}}, \tag{5}$$

where B is an upper bound on the value of the loss function ℓ across the entire data distribution, and C is a fixed Lipchitz constant depending on the activation's and the loss function's Lipschitz-continuity (see equation (43), Appendix A.2.4).

We defer a proof sketch of the above result until Section 4. As a remark, prior works by Garg et al. [15] and Liao et al. [33] consider an MPNN with $W^{(t)}$ and $U^{(t)}$ being the same for t from 1 up to l, motivated by practical designs [16, 30]. Thus, their analysis needs to be conducted separately for GCN and MPNN with weight tying. Instead, our result allows $W^{(t)}$ and $U^{(t)}$ to be arbitrarily different across different layers. This unifies GCN and MPNN without weight tying in a unified framework, and we will present their analysis via a unified approach. Further discussion about our proof techniques will be presented in Section 4.

Table 1: How does the generalization gap scale with graph properties in a graph neural network? We compare our results with prior results in the following table: We let A denote the adjacency matrix, D be the degree-diagonal matrix of A, and l be the depth of the GNN model. Prior results scale with the graph's maximum degree, denoted as d. Our result instead scales with the spectral norm of the graph diffusion matrix. Furthermore, our analysis applies to graph isomorphism networks [50] and GraphSAGE with mean aggregation [21], both of which are the first from our work.

Graph Dependence	GCN	MPNN	GIN	GraphSAGE-Mean
Garg et al. (2020) [15]	d^{l-1}	d^{l-1}	-	-
Liao et al. (2021) [33]	$d^{(l-1)/2}$	d^{l-1}	-	-
Ours (cf. Theorems 3.1 and 4.5)	1	$\|A\|^{l-1}$	$(l-1)^{-1}\sum_{i=1}^{l-1}\|A\|^i$	$\left\|D^{-1}A\right\|^{l-1}$

3.3 Comparison with prior art

In Table 1, we compare our result with prior results. We first illustrate the effects of graph properties on the generalization bounds. Then we will also show a numerical comparison to incorporate the other components of the bounds.

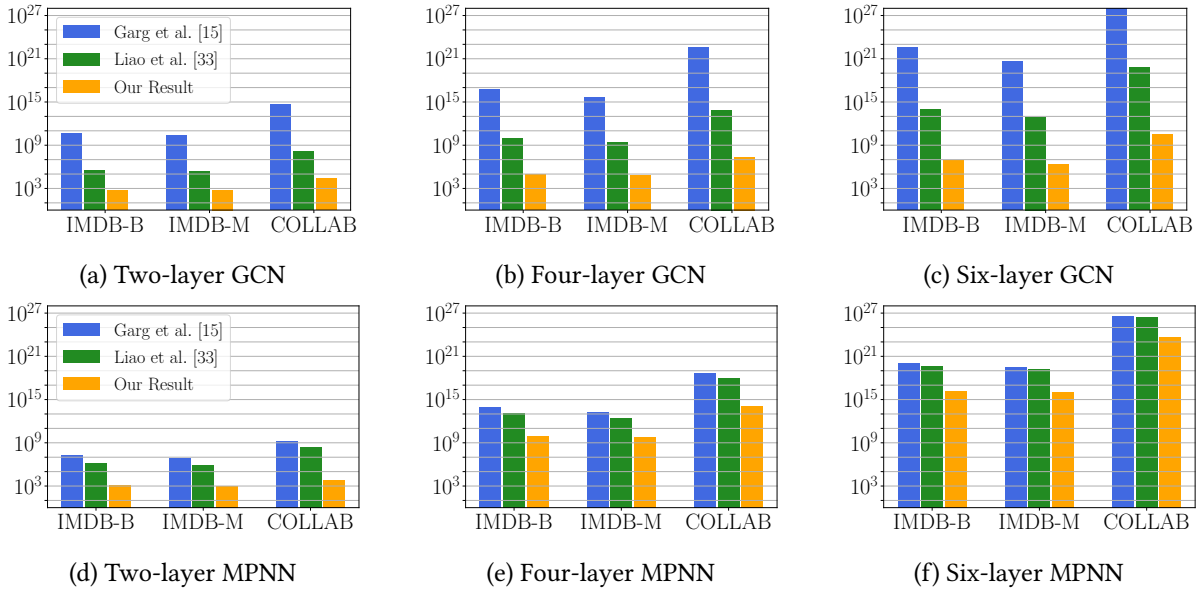
- Suppose P_G is the adjacency matrix of G. Then, one can show that for any undirected graph G, the spectral norm of P_G is less than the maximum degree d (cf. Fact A.1, Appendix A for a proof). This explains why our result is strictly less than prior results for MPNN in Table 1.
- Suppose P_G is the normalized and symmetric adjacency matrix of G: $P_G = \tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}$, where $\tilde{A}$ is $A + \mathrm{Id}$ and $\tilde{D}$ is the degree-diagonal matrix of $\tilde{A}$. Then, the spectral norm of P_G is at most one (cf. Fact A.1, Appendix A for a proof). This fact explains why the graph dependence of our result for GCN is 1 in Table 1. Thus, we can see that this provides an exponential improvement compared to the prior results.

Thus, for the above diffusion matrices, we conclude that the spectral norm of P_G is strictly smaller than the maximum degree of graph G (across all graphs in the distribution $\mathcal{D}$).

Numerical Comparison. Next, we conduct an empirical analysis to compare our results and prior results numerically. Following the setting of prior works, we use two types of models that share their weight matrices across different layers, including GCN [32] and the MPNN specified in Liao et al. [33]. For both models, we evaluate the generalization bounds by varying the network depth l between 2, 4, and 6.

We consider graph prediction tasks on three collaboration networks, including IMDB-B, IMDB-M, and COLLAB [53]. IMDB-B includes a collection of movie collaboration graphs. In each graph, a node represents an actor or an actress, and an edge denotes a collaboration in the same movie. The task is to classify each graph into the genre of the movie as Action or Romance. The IMDB-M is a multi-class extension with the movie graph label Comedy, Romance, or Sci-Fi. COLLAB includes a list of ego-networks of scientific researchers. Each graph includes a researcher and her collaborators as nodes. An edge in the graph indicates a collaboration between two researchers. The task is to classify each ego-network into the field of the researcher, including High Energy, Condensed Matter, and Astro Physics.

We report the numerical comparison in Figure 2. We report the averaged result over three random seeds. Our results are consistently smaller than previous results. As explained in Table 1, the improvement comes from the spectral norm bounds on graphs compared with the max degree bounds.



(a) Two-layer GCN (b) Four-layer GCN (c) Six-layer GCN

(d) Two-layer MPNN (e) Four-layer MPNN (f) Six-layer MPNN

Figure 2: Comparing our result and prior results [15, 33] on three graph classification tasks. **Upper**: The experiments are conducted on GCNs. **Lower**: The experiments are conducted on MPNNs following the setup of Liao et al. [33].

3.4 A matching lower bound

Next, we show an instance that exhibits the same dependence on the graph diffusion matrix as our upper bound. In our example:

- The graph G is the complete graph with self-loops inserted in each node. Thus, the adjacency matrix of G is precisely a square matrix with all ones. We will set P_G as the adjacency matrix of G.
- In the first $l-1$ graph diffusion layers, the activation functions ϕ, ρ, ψ are all linear functions. Further, we fix all the parameters of $\boldsymbol{U}$ as zero.
- The loss function ℓ is the logistic loss.

Then, we demonstrate a data distribution such that there always exists some weight matrices within $\mathcal{H}$ whose generalization gap must increase in proportion to the spectral norm of P_G^{l-1} and the product of the spectral norm of every layer $s_1, s_2, \ldots, s_l$.

Theorem 3.2. *Let N_0 be a sufficiently large value. For any norms $s_1, s_2, \ldots, s_n$, there exists a data distribution $\mathcal{D}$ on which with probability at least 0.1 over the randomness of N independent samples from $\mathcal{D}$, for any $N \geq N_0$, the generalization gap of f is larger than:*

$$\left|L(f) - \hat{L}(f)\right| \gtrsim \sqrt{\frac{\left(\max_{(X,G,y)\sim\mathcal{D}} \left\|P_G\right\|^{2(l-1)}\right)\left(\prod_{i=1}^{l} s_i^2\right)}{N}}. \tag{6}$$

Notice that the lower bound in (6) exhibits the same scaling in terms of G—$\left\|P_G\right\|^{l-1}$—as our upper bound from equation (5). Therefore, we conclude that our spectral norm bound is tight for multilayer MPNN in terms of G. The proof of the lower bound can be found in Appendix A.3.

4 Proof Techniques and Extensions

Our analysis for dealing with the graph structure seems fundamentally different from the existing analysis. In the margin analysis of Liao et al. [33], the authors also need to incorporate the graph structure in the perturbation error. For bounding the perturbation error, the authors use a triangle inequality that results in a $(1,\infty)$ norm of the matrix P_G (see Lemma 3.1 of Liao et al. [33] for GCN). We note that this norm can be larger than the spectral norm by a factor of $\sqrt{n}$, where n is the number of nodes in G: in the case of a star graph, this norm for the graph diffusion matrix of GCN is $\sqrt{n}$. By comparison, the spectral norm of the same matrix is less than one (see Fact A.1, Appendix A).

How can we tighten the perturbation error analysis and the dependence on P_G in the generalization bounds then? Our proof involves two parts:

- **Part I:** By expanding the perturbed loss of a GNN, we prove a bound on the generalization gap using the trace of the Hessian matrix associated with the loss.
- **Part II:** Then, we explicitly bound the trace of the Hessian matrix with the spectral norm of the graph using the Lipschitzness of the activation functions.

Part I: *Measuring noise stability using the Hessian.* We first state an implicit generalization bound that measures the trace of the Hessian matrix. Let $\mathbf{H}^{(i)}$ denote the Hessian matrix of the loss $\ell(f(X,G),y)$ with respect to layer i's parameters, for each i from 1 up to l. Particularly, $\mathbf{H}^{(i)}$ is a square matrix whose dimension depends on the number of variables within layer i. Let $\mathbf{H}$ denote the Hessian matrix of the loss $\ell(f(X,G),y)$ over all parameters of f.

Lemma 4.1. *In the setting of Theorem 3.1, with probability at least $1-\delta$ over the randomness of the N training examples, for any $\delta > 0$ and ϵ close to 0, we get:*

$$L(f) \le (1+\epsilon)\hat{L}(f) + \mathrm{O}\left(\frac{\log(\delta^{-1})}{N^{3/4}}\right) + (1+\epsilon)\sum_{i=1}^{l}\sqrt{\frac{B \max\limits_{(X,G,y)\sim\mathcal{D}} \mathrm{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X,G),y)]\right] s_i^2 r_i^2}{N}}. \tag{7}$$

Proof Sketch. At a high level, the above result follows from Taylor's expansion of the perturbed loss. Suppose each parameter of f is perturbed by an independent noise drawn from a Gaussian distribution with mean zero and variance σ^2. Let $\tilde{\ell}(f(X,G),y)$ be the perturbed loss value of an input example X, G with label y. Let $\mathcal{E}$ denote the noise injections organized in a vector. Using Taylor's expansion of the perturbed loss $\tilde{\ell}$, we get:

$$\tilde{\ell}(f(X,G),y) - \ell(f(X,G),y) = \mathcal{E}^\top \nabla \ell(f(X,G),y) + \frac{1}{2}\mathcal{E}^\top \mathbf{H}\big[\ell(f(X,G),y)\big]\mathcal{E} + \mathrm{O}(\sigma^3). \tag{8}$$

Notice that the expectation of the first-order expansion term above is equal to zero. The expectation of the second-order expansion term becomes σ^2 times the trace of the loss Hessian. To derive equation (7), we use a PAC-Bayes bound of McAllester [35, Theorem 2]. There are two parts to this PAC-Bayes bound:

- The expectation of the noise perturbation in equation (8), taken over the injected noise $\mathcal{E}$;
- The KL divergence between the prior and the posterior, which is at most $s_i^2 r_i^2$ for layer i, for i from 1 up to l, within the hypothesis set $\mathcal{H}$.

Thus, one can balance the two parts by adjusting the noise variance at each layer—this leads to the layerwise Hessian decomposition in equation (7).

A critical step is showing the uniform convergence of the Hessian matrix. We achieve this based on the Lipschitz-continuity of the first and twice derivatives of the nonlinear activation mappings. With these

conditions, we prove the uniform convergence with a standard ϵ-cover argument. The complete proof can be found in Appendix A.2.1.

Remark 4.2. Our argument in Lemma 4.1 applies to graph-level prediction tasks, which assume an unknown distribution of graphs. A natural question is whether the analysis applies to node-level prediction tasks, which are often treated as semi-supervised learning problems. The issue with directly applying our analysis to semi-supervised learning is that the size of a graph is only finite. Instead, a natural extension would be to think about our graph as a random sample from some population, and then argue about generalization in expectation of the random sample. It is conceivable that one can prove a similar spectral norm bound for node prediction in this extension. This would be an interesting question for future work.

Part II: *Spectral norm bounds of the Hessian.* Next, we explicitly analyze the trace of the Hessian at each layer. We bound the trace of the Hessian with the spectral norm of the weight matrices and the graph, based on the Lipschitzness conditions from Theorem 3.1. Notice that the last layer is a linear pooling layer, which can be deduced from layer $l-1$. Hence, we consider the first $l-1$ layers below.

Lemma 4.3. *In the setting of Theorem 3.1, the trace of the Hessian $\mathbf{H}^{(i)}$ over $W^{(i)}$ and $U^{(i)}$ satisfies the following, for any $i = 1, 2, \cdots, l-1$,*

$$\left|\mathrm{Tr}\left[\mathbf{H}^{(i)}\left[\ell\big(f(X,G),y\big)\right]\right]\right| \lesssim s_l^2\left(\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(l-1)}}{\partial\big(W_{p,q}^{(i)}\big)^2}\right\|_F + \sum_{p=1}^{d_0}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(l-1)}}{\partial\big(U_{p,q}^{(i)}\big)^2}\right\|_F + \left\|\frac{\partial H^{(l-1)}}{\partial W^{(i)}}\right\|_F^2 + \left\|\frac{\partial H^{(l-1)}}{\partial U^{(i)}}\right\|_F^2\right) \tag{9}$$

$$\lesssim \|X\|^2\left\|P_G\right\|^{2(l-1)}\prod_{j=1:\,j\neq i}^{l} s_j^2. \tag{10}$$

Proof Sketch. Equation (9) uses the chain rule to expand out the trace of the Hessian and then applies the Lipschitzness of the loss function. Based on this result, equation (10) then bounds the first and second derivatives of $H^{(l-1)}$. This step is achieved via an induction of $\partial H^{(j)}$ and $\partial^2 H^{(j)}$ over $W^{(i)}$ and $U^{(i)}$, for $j = 1, \ldots, l-1$ and $i = 1, \ldots, j$. The induction relies on the feedforward architecture and the Lipschitzness of the first and second derivatives. We leave out a few details such as the constants in equations (9) and (10) that can be found in Appendix A.2.2 and A.2.3. Combining both parts together, we get equation (1).

Remark 4.4. We compare our analysis approach with the approach of Liao et al. [33]. While our analysis also follows the PAC-Bayesian approach, we additionally explore Lipschitz properties of first and twice derivatives of the activation functions (e.g., tanh and sigmoid). This allows us to measure the perturbation loss with Hessians, which captures graph properties much more accurately than the margin analysis of Liao et al. [33]. It would be interesting to understand if one could still achieve spectral norm bounds on graphs under weaker conditions (e.g., with the tool of Wei and Ma [47]). This is left for future work.

4.1 Extensions

Graph isomorphism networks. This architecture concatenates every layer's embedding together for more expressiveness [50]. A classification layer is used after the layers. Let $V^{(i)}$ denote a d_i by k matrix (recall k is the output dimension). Denote the set of these matrices by $\mathcal{V}$. We average the loss of all of the classification layers. Let $\hat{L}_{GIN}(f)$ denote the average loss of f over N independent samples of $\mathcal{D}$. Let $L_{GIN}(f)$ denote the expected loss of f over a random sample of $\mathcal{D}$. See also equation (44) in Appendix A.4 for their precise definitions.

Next, we state a generalization guarantee for graph isomorphism networks. Let f be any l-layer MPNN with weights defined in a hypothesis space $\mathcal{H}$: the parameters of f reside within the constraints from equation (4); further, for every i from 1 up to l, the spectral norm of $V^{(i)}$ is less than s_l. Building on Lemma

4.3, we show a bound that scales with the spectral norm generalization of the averaged graph diffusion matrices. Let P_{GIN} denote the average of $l-1$ matrices: $P_G, P_G^2, \ldots, P_G^{l-1}$. We state the result below.

Corollary 4.5. *Suppose the nonlinear activation mappings and the loss function satisfy the conditions stated in Theorem 3.1. With probability at least $1-\delta$ for any $\delta \geq 0$, and any ϵ close to zero, any f in $\mathcal{H}$ satisfies:*

$$L_{GIN}(f) \leq (1+\epsilon)\hat{L}_{GIN}(f) + \mathrm{O}\left(\frac{\log(\delta^{-1})}{N^{3/4}}\right) + \sum_{i=1}^{l} \sqrt{\frac{CBd_i\left(\max_{(X,G,y)\sim\mathcal{D}} \|X\|^2 \left\|P_{GIN}\right\|^2\right)\left(r_i^2 \prod_{j=1}^{l} s_j^2\right)}{N}}, \tag{11}$$

where B is an upper bound of the loss ℓ and C is a fixed constant that only depends on the Lipschitz-continuity of the activation mappings and the loss.

The proof can be found in Appendix A.4. In particular, we apply the trace norm bound over the model output of every layer. The classification layer, which only uses a linear transformation, can also be incorporated.

Fine-tuned Graph Neural Networks. We note that all of our bounds can be applied to the fine-tuning setting, in which a graph neural network is initialized with pretrained weights and then fine-tuned on the target task. The results can be extended to this setting by setting the norm bounds within equation (4) as the distance between the pretrained and fine-tuned model.

5 Optimizing Noise Stability Properties for Fine-tuning GNN

Our PAC-Bayesian analysis shows that maintaining a small perturbed loss ensures better generalization. Motivated by this observation, we present an algorithm to minimize the perturbed loss of a model.

Let f denote a model and $\tilde{\ell}(f)$ be the perturbed loss of f, with noise injected inside f's weight matrices. Recall from step (8) that $\tilde{\ell}(f)$ is equal to $\ell(f)$ plus several expansion terms. In particular, minimizing the expectation of $\tilde{\ell}(f)$ is equivalent to minimizing $\hat{L}(f)$ plus the trace of the Hessian matrix. To estimate this expectation, we sample several noise perturbations independently. Because Taylor's expansion of $\tilde{\ell}(f)$ also involves the gradient, we cancel this out by computing the perturbed loss with the negated perturbation. Algorithm 1 describes the complete procedure.

We evaluate the above algorithm for fine-tuning pretrained GNNs. Empirical results reveal that this algorithm achieves better test performance compared with existing regularization methods for five graph classification tasks.

5.1 Experimental setup

We focus on graph classification tasks, including five datasets from the MoleculeNet benchmark [48]. In each dataset, the goal is to predict whether a molecule has a certain chemical property given its graph representation. We use pretrained GINs from Hu et al. [24] and fine-tune the model on each downstream task. Following their experimental setup, we use the scaffold split for the dataset, and the model architecture is fixed for all five datasets. Each model has 5 layers; each layer has 300 hidden units and uses average pooling in the readout layer. We set the parameters such as the learning rate and the number of epochs following their setup.

We compare our algorithm with previous regularization methods that serve as benchmark approaches for improving generalization. This includes early stopping, weight decay, dropout, weight averaging [26], and distance-based regularization [18]. For implementing our algorithm, we set the number of perturbations as 10 and choose the noise standard deviation σ with a grid search in $\{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$.

Algorithm 1 Noise stability optimization for fine-tuning graph neural networks

Input: A training dataset $\{(X_i, G_i, y_i)\}_{i=1}^N$ with node feature X_i, graph G_i, and graph-level label y_i, for $i = 1, \ldots, N$.
Require: Number of perturbations m, noise variance σ^2, learning rate η, and number of epochs T.
Output: A trained model $f^{(T)}$.

1: At $t = 0$, initialize the parameters of $f^{(0)}$ with pretrained GNN weight matrices.
2: **for** $1 \leq t \leq T$ **do**
3: **for** $1 \leq i \leq m$ **do**
4: Draw a random perturbation $\mathcal{E}_i$ for each parameter of $f^{(t-1)}$ from a Gaussian distribution with mean zero and variance σ^2.
5: Let $\tilde{L}_i(f^{(t-1)})$ be the training loss of the model $f^{(t-1)}$ with perturbation $\mathcal{E}_i$.
6: Let $\tilde{L}_i'(f^{(t-1)})$ be the training loss of the model $f^{(t-1)}$ with perturbation $-\mathcal{E}_i$.
7: **end for**
8: Use stochastic gradient descent to update $f^{(t)}$ as $f^{(t-1)} - \frac{\eta}{2m} \sum_{i=1}^m \left(\nabla \tilde{L}_i\left(f^{(t-1)}\right) + \nabla \tilde{L}_i'\left(f^{(t-1)}\right)\right)$.
9: **end for**

5.2 Experimental results

Table 2 reports the test ROC-AUC performance averaged over multiple binary prediction tasks in each dataset. Comparing the average ranks of methods across datasets, our algorithm outperforms baselines on all five molecular property prediction datasets. The results support our theoretical analysis that the noise stability property of GNN is a strong measure of empirical generalization performance. Next, we provide details insights from applying our algorithm.

First, we hypothesize that our algorithm is particularly effective when the empirical generalization gap is large. To test the hypothesis, we vary the size of the training set in the BACE dataset; we compare the performance of our algorithm with early stopping until epoch 100. We plot the generalization gap between the training and test losses during training, shown in Figure 3a-3b. As the trend shows, our algorithm consistently reduces the generalization gap, particularly when the training set size N is 600.

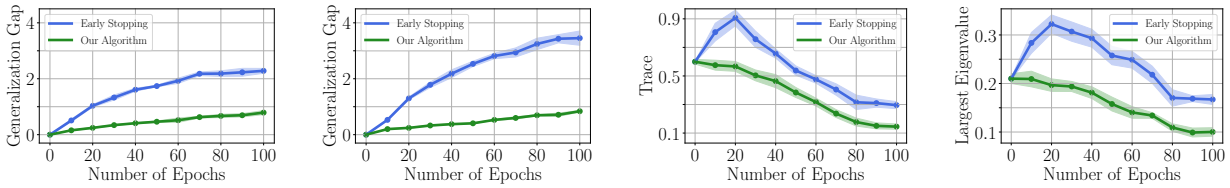
Second, we hypothesize that our algorithm helps reduce the trace of the Hessian matrix (associated with the loss). We validate this by plotting the trace of the Hessian as the number of epochs progresses during training, again using the BACE dataset as an example. Specifically, we average the trace over the training dataset. Figure 3c shows the averaged trace values during the fine-tuning process. The results confirm that noise stability optimization reduces the trace of the Hessian matrix (more significantly than early stopping). We note that noise stability optimization also reduces the largest eigenvalue of the Hessian matrix, along with reducing the trace. This can be seen in Figure 3d.

Lastly, we study the number of perturbations used in our algorithm. While more perturbations would lead to a better estimation of the noisy stability, we observe that using 10 perturbations is sufficient for getting the most gain. We also validate that using negated perturbations consistently performs better than not using them across five datasets.

Remark. We note that noise stability optimization is closely related to sharpness-aware minimization (SAM) [14]. Noise stability optimization differs in two aspects compared with SAM. First, SAM requires solving constrained minimax optimization, which may not even be differentiable [10]. Our objective remains the same after perturbation. Second, SAM reduces the largest eigenvalue of the Hessian matrix, which can be seen from Taylor's expansion of $\tilde{\ell}(f)$. We reduce the trace of the Hessian matrix, which includes reducing the largest eigenvalue as part of the trace. There is another related work that regularizes noise stability in NLP [25]. Their approach adds noise perturbation in the input and regularizes the loss change in the output. Our approach directly adds the perturbation in the weight matrices.

Table 2: Test ROC-AUC (%) score for five molecular property prediction datasets with different regularization methods. The reported results are averaged over five random seeds.

Dataset	SIDER	ClinTox	BACE	BBBP	Tox21	
# Molecule Graphs	1,427	1,478	1,513	2,039	7,831	Avg. Rank
# Binary Prediction Tasks	27	2	1	1	12	
Early Stopping	61.06±1.48	68.25±2.63	82.86±0.95	67.80±1.05	77.52±0.23	5.8
Weight Decay	61.30±0.21	67.43±2.88	83.72±0.99	67.98±2.41	78.23±0.35	5.0
Dropout	63.90±0.90	73.70±2.80	84.50±0.70	68.07±1.30	78.30±0.30	3.6
Weight Averaging	63.67±0.34	78.78±1.49	83.93±0.36	70.26±0.24	77.59±0.11	3.4
Distance-based Reg.	64.36±0.48	76.68±1.19	84.65±0.48	70.37±0.44	78.62±0.24	2.2
Ours (Alg. 1)	**65.13±0.18**	**80.18±0.82**	**85.07±0.43**	**71.22±0.36**	**79.31±0.24**	**1.0**



(a) $N = 1200$ (b) $N = 600$ (c) Trace (d) Largest Eigenvalue

Figure 3: **Left**: Our algorithm is particularly effective at reducing the generalization gap for small training dataset sizes N. **Right**: Both the trace and the largest eigenvalue of the Hessian matrix (associated with the loss) decreased during training.

6 Conclusion

This work develops generalization bounds for graph neural networks that have a sharp dependence on the graph diffusion matrix. The results are achieved within a unified setting that significantly extends prior works. In particular, we answer an open question mentioned in Liao et al. [33]: a refined PAC-Bayesian analysis can improve the generalization bounds for message-passing neural networks. These bounds are obtained by analyzing the trace of the Hessian matrix with the Lipschitzness of the activation functions. Empirical findings suggest that the Hessian-based bound matches observed gaps on real-world graphs. Thus, our work also provides a practical tool to measure generalization in graph neural networks. The algorithmic results with noise stability optimization further demonstrate the practical implication of our findings.

Our work opens up many interesting questions for future work. Could the new tools we have developed be used to study generalization in graph attention networks [44]? Could Hessians be used for measuring out-of-distribution generalization gaps of graph neural networks?

Acknowledgement

Thanks to Renjie Liao, Haoyu He, Jan-Willem van de Meent, and the anonymous referees for providing constructive feedback on our work. Thanks to Yang Yuan for the helpful discussions. HJ and DL acknowledge the financial support from the startup fund of Khoury College of Computer Sciences, Northeastern University.

References

[1] Sanjeev Arora (2021). "Technical perspective: Why don't today's deep nets overfit to their training data?" In: *Communications of the ACM*.

[2] Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang (2018). "Stronger generalization bounds for deep nets via a compression approach". In: *ICML*.

[3] Waiss Azizian and Marc Lelarge (2021). "Expressive power of invariant and equivariant graph neural networks". In: *ICLR*.

[4] Peter Bartlett, Dylan J Foster, and Matus Telgarsky (2017). "Spectrally-normalized margin bounds for neural networks". In: *NeurIPS*.

[5] Peter Bartlett, Nick Harvey, Christopher Liaw, and Abbas Mehrabian (2019). "Nearly-tight VC-dimension and pseudodimension bounds for piecewise linear neural networks". In: *JMLR*.

[6] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun (2014). "Spectral networks and locally connected networks on graphs". In: *ICLR*.

[7] Ines Chami, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy (2020). "Machine learning on graphs: A model and comprehensive taxonomy". In: *arXiv preprint arXiv:2005.03675*, p. 1.

[8] Zhengdao Chen, Lei Chen, Soledad Villar, and Joan Bruna (2020). "Can graph neural networks count substructures?" In: *NeurIPS*.

[9] Hanjun Dai, Bo Dai, and Le Song (2016). "Discriminative embeddings of latent variable models for structured data". In: *ICML*.

[10] Constantinos Daskalakis, Stratis Skoulakis, and Manolis Zampetakis (2021). "The complexity of constrained min-max optimization". In: *STOC*.

[11] Simon S Du, Kangcheng Hou, Russ R Salakhutdinov, Barnabas Poczos, Ruosong Wang, and Keyulu Xu (2019). "Graph neural tangent kernel: Fusing graph neural networks with graph kernels". In: *NeurIPS*.

[12] Gintare Karolina Dziugaite and Daniel M Roy (2017). "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data". In: *UAI*.

[13] Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli (2020). "A fair comparison of graph neural networks for graph classification". In: *ICLR*.

[14] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur (2021). "Sharpness-aware minimization for efficiently improving generalization". In: *ICLR*.

[15] Vikas Garg, Stefanie Jegelka, and Tommi Jaakkola (2020). "Generalization and representational limits of graph neural networks". In: *ICML*.

[16] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl (2017). "Neural message passing for quantum chemistry". In: *ICML*.

[17] Noah Golowich, Alexander Rakhlin, and Ohad Shamir (2018). "Size-independent sample complexity of neural networks". In: *COLT*.

[18] Henry Gouk, Timothy M Hospedales, and Massimiliano Pontil (2021). "Distance-Based Regularisation of Deep Networks for Fine-Tuning". In: *ICLR*.

[19] Benjamin Guedj (2019). "A Primer on PAC-Bayesian Learning". In: *Proceedings of the French Mathematical Society*.

[20] Will Hamilton, Rex Ying, and Jure Leskovec (2017a). "Representation learning on graphs: Methods and applications". In: *arXiv preprint arXiv:1709.05584*.

[21] Will Hamilton, Zhitao Ying, and Jure Leskovec (2017b). "Inductive representation learning on large graphs". In: *NeurIPS*.

[22] Moritz Hardt and Benjamin Recht (2021). "Patterns, predictions, and actions: A story about machine learning". In: *arXiv preprint arXiv:2102.05242*.

[23] Moritz Hardt, Benjamin Recht, and Yoram Singer (2016). "Train faster, generalize better: Stability of stochastic gradient descent". In: *ICML*.

[24] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec (2020). "Strategies for pre-training graph neural networks". In: *ICLR*.

[25] Hang Hua, Xingjian Li, Dejing Dou, Cheng-Zhong Xu, and Jiebo Luo (2021). "Noise stability regularization for improving BERT fine-tuning". In: *ACL*.

[26] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson (2018). "Averaging weights leads to wider optima and better generalization". In: *UAI*.

[27] Stefanie Jegelka (2022). "Theory of Graph Neural Networks: Representation and Learning". In: *arXiv preprint arXiv:2204.07697*.

[28] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio (2020). "Fantastic generalization measures and where to find them". In: *ICLR*.

[29] Chi Jin, Praneeth Netrapalli, Rong Ge, Sham M Kakade, and Michael I Jordan (2019). "A short note on concentration inequalities for random vectors with subgaussian norm". In: *arXiv preprint arXiv:1902.03736*.

[30] Wengong Jin, Regina Barzilay, and Tommi Jaakkola (2018). "Junction tree variational autoencoder for molecular graph generation". In: *ICML*.

[31] Haotian Ju, Dongyue Li, and Hongyang R Zhang (2022). "Robust Fine-Tuning of Deep Neural Networks with Hessian-based Generalization Guarantees". In: *ICML*.

[32] Thomas N Kipf and Max Welling (2017). "Semi-supervised classification with graph convolutional networks". In: *ICLR*.

[33] Renjie Liao, Raquel Urtasun, and Richard Zemel (2021). "A PAC-Bayesian Approach to Generalization Bounds for Graph Neural Networks". In: *ICLR*.

[34] Philip M Long and Hanie Sedghi (2020). "Generalization bounds for deep convolutional neural networks". In: *ICLR*.

[35] David McAllester (2013). "A PAC-Bayesian tutorial with a dropout bound". In: *arXiv preprint arXiv:1307.2118*.

[36] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar (2018). *Foundations of machine learning*. MIT press.

[37] Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe (2019). "Weisfeiler and leman go neural: Higher-order graph neural networks". In: *AAAI*.

[38] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro (2018). "A pac-bayesian approach to spectrally-normalized margin bounds for neural networks". In: *ICLR*.

[39] Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro (2019). "Towards understanding the role of over-parametrization in generalization of neural networks". In: *ICLR*.

[40] Ryoma Sato, Makoto Yamada, and Hisashi Kashima (2019). "Approximation ratios of graph neural networks for combinatorial problems". In: *NeurIPS*.

[41] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini (2008). "The graph neural network model". In: *IEEE transactions on neural networks*.

[42] Franco Scarselli, Ah Chung Tsoi, and Markus Hagenbuchner (2018). "The vapnik–chervonenkis dimension of graph and recursive neural networks". In: *Neural Networks*.

[43] Daniel Selsam, Matthew Lamm, Benedikt Bünz, Percy Liang, Leonardo de Moura, and David L Dill (2019). "Learning a SAT solver from single-bit supervision". In: *ICLR*.

[44] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio (2018). "Graph attention networks". In: *ICLR*.

[45] Saurabh Verma and Zhi-Li Zhang (2019). "Stability and generalization of graph convolutional neural networks". In: *KDD*.

[46] S. Vishwanathan, Nicol Schraudolph, Risi Kondor, and Karsten Borgwardt (2010). "Graph kernels". In: *JMLR*.

[47] Colin Wei and Tengyu Ma (2020). "Data-dependent sample complexity of deep neural networks via lipschitz augmentation". In: *NeurIPS*.

[48] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande (2018). "MoleculeNet: a benchmark for molecular machine learning". In: *Chemical science*.

[49] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Yu Philip (2020). "A comprehensive survey on graph neural networks". In: *IEEE transactions on neural networks and learning systems*.

[50] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka (2019). "How powerful are graph neural networks?" In: *ICLR*.

[51] Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka (2020). "What can neural networks reason about?" In: *ICLR*.

[52] Keyulu Xu, Mozhi Zhang, Jingling Li, Simon S Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka (2021). "How neural networks extrapolate: From feedforward to graph neural networks". In: *ICLR*.

[53] Pinar Yanardag and S. Vishwanathan (2015). "Deep graph kernels". In: *KDD*.

[54] Gilad Yehudai, Ethan Fetaya, Eli Meirom, Gal Chechik, and Haggai Maron (2021). "From local structures to size generalization in graph neural networks". In: *ICML*.

[55] Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec (2018). "Hierarchical graph representation learning with differentiable pooling". In: *NeurIPS*.

[56] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals (2017). "Understanding deep learning requires rethinking generalization". In: *ICLR*.

[57] Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen (2018). "An end-to-end deep learning architecture for graph classification". In: *AAAI*.

A Proofs

This section provides the complete proofs for all of our results in Section 3. First we state several notations and facts that will be needed in the proofs. Then we provide the proof of the Hessian-based generalization bound for MPNN, which is stated in Lemma 4.1. After that, in Appendix A.2, we provide the proof of Theorem 3.1, a key step of which is the proof of Lemma 4.3. Next in Appendix A.3, we state the proof of the lower bound. Lastly in Appendix A.4, we will provide the proof for the case of graph isomorphism networks.

First we state several facts about graphs and provide a short proof of them.

Fact A.1. *Let $G = (V, E)$ be an undirected graph. Let d_G be the maximum degree of G.*

a) Let A be the adjacency matrix of G. Then, the adjacency matrix satisfies: $\sqrt{d_G} \le \|A\| \le d_G$.

b) The symmetric and degree-normalized adjacency matrix satisfies: $\left\|D^{-1/2}AD^{-1/2}\right\| \le 1$.

Proof. Based on the definition of the spectral norm, we get

$$\|A\| = \max_{\|x\|=1} x^\top A x = \max_{\|x\|=1} \sum_{(i,j)\in E} x_i x_j \le \max_{\|x\|=1} \sum_{(i,j)\in E} \frac{1}{2}(x_i^2 + x_j^2) \le d_G \sum_{i\in V} x_i^2 = d_G.$$

Assume that node i has the maximum degree d_G. Denote edges set $E_i = \{(i, i_k)\}_{k=1}^{d_G} \subseteq E$. Let $x_i = \frac{1}{\sqrt{2}}$, $x_{i_k} = \frac{1}{\sqrt{2d_G}}$ for all $k = 1, \ldots, d_G$. The rest entries of x are equal to zero. Thus, x is a normalized vector. Next, we have

$$\|A\| = \max_{\|x\|=1} \sum_{(i,j)\in E} x_i x_j \ge \max_{\|x\|=1} 2 \sum_{(i,j)\in E_i} x_i x_j = 2d_G \cdot \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2d_G}} = \sqrt{d_G}.$$

An example in which $\|P_G\|$ gets close to $\sqrt{d_G}$ is the star graph. An example in which $\|P_G\|$ gets close to d_G is the complete graph.

Next we focus on case b). From the definition of the spectral norm, we know

$$\left\|D^{-1/2} A D^{-1/2}\right\| = \max_{\|x\|=1} x^\top \left(D^{-1/2} A D^{-1/2}\right) x = \max_{\|x\|=1} \sum_{(i,j)\in E} \frac{x_i x_j}{\sqrt{d_i d_j}} \le \max_{\|x\|=1} \sum_{(i,j)\in E} \frac{x_i^2}{2d_i} + \frac{x_j^2}{2d_j} = \sum_{i\in V} x_i^2 = 1.$$

During the middle of the above step, we have used the Cauchy-Schwartz inequality. The proof of this result is now completed. □

Notations: For two matrices X and Y that are both of dimension d_1 by d_2, the Hadamard product of X and Y, denoted as $X \odot Y$, is equal to the entrywise product of X and Y.

A.1 Proof of our PAC-Bayesian bound (Lemma 4.1)

To be precise, we will restate the conditions required in Theorem 3.1 separately below. The conditions are exactly the same as stated in Section 3.

Assumption A.2. *Assume that all the activation functions $\phi_i(\cdot), \rho_i(\cdot), \psi_i(\cdot)$ for any $1 \le i \le l-1$ and the loss function $\ell(x, y)$ over x are twice-differentiable and κ_0-Lipschitz. Their first-order derivatives are κ_1-Lipschitz and their second-order derivatives are κ_2-Lipschitz.*

Based on the above assumption, we provide the precise statement for the Taylor's expansion, used in equation (8).

Proposition A.3. *In the setting of Theorem 3.1, suppose each parameter in layer i is perturbed by an independent noise drawn from $\mathcal{N}(0, \sigma_i^2)$. Let $\tilde{\ell}(f(X, G), y)$ be the perturbed loss function with noise perturbation injection vector $\mathcal{E}$ on all parameters $\boldsymbol{W}$ and $\boldsymbol{U}$. There exist some fixed value C_1 that do not grow with N and $1/\delta$ such that*

$$\left|\tilde{\ell}(f(X, G), y) - \ell(f(X, G), y) - \frac{1}{2}\sum_{i=1}^{l} \sigma_i^2 \operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X, G), y)]\right]\right| \le C_1 \sum_{i=1}^{l} \sigma_i^3.$$

Proof. By the Taylor's expansion, the following identity holds

$$\tilde{\ell}(f(X, G), y) - \ell(f(X, G), y) = \mathop{\mathbb{E}}_{\mathcal{E}}\left[\mathcal{E}^\top \nabla \ell(f) + \frac{1}{2}\mathcal{E}^\top \mathbf{H}[\ell(f)]\mathcal{E} + R(\ell(f), \mathcal{E})\right].$$

where $R(\ell(f), \mathcal{E})$ be the rest of the first order and the second order terms. Since each entry in $\mathcal{E}$ follows the normal distribution, we have $\mathbb{E}_{\mathcal{E}}\left[\mathcal{E}^\top \nabla \ell(f)\right] = 0$. The Hessian term turns to

$$\mathcal{E}^\top \mathbf{H}[\ell(f)]\mathcal{E} = \sum_{i=1}^{l} \sigma_i^2 \operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X,G),y)]\right].$$

Since the readout layer is linear, by Proposition A.4, there exists a fixed constant $\bar{C}$ that does not grow with N and δ^{-1} such that $|R(\ell(f), \mathcal{E})| \le \bar{C}\|\mathcal{E}\|^3$. Based on Jin et al. [29, Lemma 2], for any x drawn from a normal distribution $\mathcal{N}(0, \sigma^2)$, we have $\mathbb{E}\left[x^3\right] \le 6\sigma^3$. Hence, we get $\mathbb{E}\left[R(\ell(f), \mathcal{E})\right] \le C_1 \sum_{i=1}^{l} \sigma_i^3$, where $C_1 = \mathrm{O}(h^2\bar{C})$ is a fixed constant. Thus, we have finished the proof. □

Next, we state a Lipschitz upper bound of the network output at each layer. This will be needed in the ϵ-covering argument later in the proof of Theorem 3.1. To simplify the notation, we will abbreviate explicit constants that do not grow with N and $1/\delta$ in the notation $\lesssim$; more specifically, we use $A(n) \lesssim B(n)$ to indicate that there exists a function c that does not depend on N and $1/\delta$ such that $A(n) \le c \cdot B(n)$ for large enough values of n.

Proposition A.4. *In the setting of Theorem 3.1, for any $j = 1, \ldots, l-1$, the change in the Hessian of output of the j layer network $H^{(j)}$ with respect to W_i and U_i under perturbation on W and U can be bounded as follows:*

$$\left\|\mathbf{H}_{\boldsymbol{W}}^{(i)}[\tilde{H}^{(j)}] - \mathbf{H}_{\boldsymbol{W}}^{(i)}[H^{(j)}]\right\|_F \lesssim \sum_{t=1}^{j}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \tag{12}$$

$$\left\|\mathbf{H}_{U}^{(i)}[\tilde{H}^{(j)}] - \mathbf{H}_{U}^{(i)}[H^{(j)}]\right\|_F \lesssim \sum_{t=1}^{j}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \tag{13}$$

Above, the notation $\mathbf{H}_{\boldsymbol{W}}^{(i)}[\tilde{H}^{(j)}]$ is the perturbation of the Hessian matrix of $H^{(j)}$ by $\Delta\boldsymbol{W}$ and $\Delta\boldsymbol{U}$, specific to the variables of $\boldsymbol{W}$; likewise, $\mathbf{H}_{U}^{(i)}[\tilde{H}^{(j)}]$ is the perturbation of the Hessian matrix specific to the variables of $\boldsymbol{U}$.

The proof of Proposition A.4 will be deferred until Appendix A.1.1. Based on Propositions A.3 and A.4, now we are ready to present the proof of Lemma 4.1.

Proof of Lemma 4.1. First, we separate the gap of $L(f)$ and $\frac{1}{\beta}\hat{L}(f)$ into three parts:

$$\begin{aligned} L(f) - \frac{1}{\beta}\hat{L}(f) = {} & \underbrace{\mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\ell(f(X,G),y)\right] - \mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\tilde{\ell}(f(X,G),y)\right]}_{E_1} + \mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\tilde{\ell}(f(X,G),y)\right] \\ & - \frac{1}{\beta}\Big(\frac{1}{N}\sum_{i=1}^{N}\tilde{\ell}(f(X_i,G_i),y_i)\Big) + \underbrace{\frac{1}{\beta}\Big(\frac{1}{N}\sum_{i=1}^{N}\tilde{\ell}(f(X_i,G_i),y_i)\Big) - \frac{1}{\beta}\Big(\frac{1}{N}\sum_{i=1}^{N}\ell(f(X_i,G_i),y_i)\Big)}_{E_2}. \end{aligned}$$

for any $\beta \in (0,1)$. Above, $\tilde{\ell}(f(X,G),y)$ is the perturbed loss from $\ell(f(X,G),y)$ with noise injections $\mathcal{E}$ added to all the parameters in $\boldsymbol{W}$ and $\boldsymbol{U}$. By the Taylor's expansion from Proposition A.3, we can bound the

difference between $\tilde{\ell}(f(X,G),y)$ and $\ell(f(X,G)$ with the trace of the Hessian. Therefore

$$
\begin{aligned}
L(f) - \frac{1}{\beta}\hat{L}(f) \leq & - \mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\frac{1}{2}\sum_{i=1}^{l}\sigma_i^2 \operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X,G),y)]\right]\right] + \sum_{i=1}^{l} C_1\sigma_i^3 \qquad \text{(by Prop. A.3 for } E_1\text{)} \\
& + \left(\mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\tilde{\ell}(f(X,G),y)\right] - \frac{1}{\beta}\Big(\frac{1}{N}\sum_{i=1}^{N}\tilde{\ell}(f(X_i,G_i),y_i)\Big)\right) \\
& + \frac{1}{2\beta}\sum_{i=1}^{l}\sigma_i^2\Big(\frac{1}{N}\sum_{j=1}^{N}\operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X_j,G_j),y_j)]\right]\Big) + \frac{1}{\beta}\sum_{i=1}^{l} C_1\sigma_i^3. \qquad \text{(by Prop. A.3 for } E_2\text{)}
\end{aligned}
$$

By rearranging the above equation, we get the following:

$$
\begin{aligned}
L(f) - \frac{1}{\beta}\hat{L}(f) \leq & \frac{1}{2}\sum_{i=1}^{l}\sigma_i^2\underbrace{\left(\frac{1}{N}\sum_{j=1}^{N}\operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X_j,G_j),y_j)]\right] - \mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X,G),y)]\right]\right]\right)}_{E_3} \\
& + \frac{1}{2}\Big(\frac{1}{\beta}-1\Big)\underbrace{\sum_{i=1}^{l}\frac{\sigma_i^2}{N}\sum_{j=1}^{N}\operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X_j,G_j),y_j)]\right]}_{E_4} \\
& + \Big(1+\frac{1}{\beta}\Big)C_1\sum_{i=1}^{l}\sigma_i^3 + \underbrace{\mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\tilde{\ell}(f(X,G),y)\right] - \frac{1}{\beta N}\sum_{i=1}^{N}\tilde{\ell}(f(X_i,G_i),y_i)}_{E_5}.
\end{aligned}
$$

Based on Proposition A.4, the Hessian operator $\mathbf{H}^{(i)}$ is Lipschitz-continuous for some parameter that does not depend on N and $1/\delta$, for any $i = 1, 2\ldots, l$. Therefore, from Ju et al. [31, Lemma 2.4], there exist some fixed values C_2, C_3 that do not grow with N and $1/\delta$, such that with probability at least $1-\delta$ over the randomness of the training set. Therefore, the matrix inside the trace of E_3 satisfies

$$
\left\|\frac{1}{N}\sum_{j=1}^{N}\mathbf{H}^{(i)}[\ell(f(X_j,G_j),y_j)] - \mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\mathbf{H}^{(i)}[\ell(f(X,G),y)]\right]\right\|_F \leq \frac{C_2\sqrt{\log(C_3N/\delta)}}{\sqrt{N}}, \tag{14}
$$

for any $i = 1, \ldots, l$. Thus, by the Cauchy-Schwartz inequality, E_3 is less than $\sqrt{2h^2}$ times the RHS of equation (14). Suppose the loss function $\ell(f(X,G),y)$ lies in a bounded range $[0,B]$ given any $(X,G,y) \sim \mathcal{D}$. By the PAC-Bayes bound of McAllester [35, Theorem 2] (see also Guedj [19]), we choose $\boldsymbol{U}$ as a prior distribution and $\boldsymbol{W}+\boldsymbol{U}$ as a posterior distribution. For any $\beta \in (0,1)$ and $\delta \in [0,1)$, with probability at least $1-\delta$, E_5 satisfies:

$$
\begin{aligned}
\mathop{\mathbb{E}}_{(X,G,y)\sim\mathcal{D}}\left[\tilde{\ell}(f(X,G),y)\right] - \frac{1}{\beta N}\sum_{i=1}^{N}\tilde{\ell}(f(X_i,G_i),y_i) \leq & \frac{B}{2\beta(1-\beta)N}\Big(\sum_{i=1}^{l}\frac{\left\|W^{(i)}\right\|_F^2 + \left\|U^{(i)}\right\|_F^2}{2\sigma_i^2} + \log\frac{1}{\delta}\Big) \\
\leq & \frac{B}{2\beta(1-\beta)N}\Big(\sum_{i=1}^{l}\frac{s_i^2 r_i^2}{\sigma_i^2} + \log\frac{1}{\delta}\Big).
\end{aligned} \tag{15}
$$

The above is because $\boldsymbol{W}$ and $\boldsymbol{U}$ are both inside the hypothesis set $\mathcal{H}$. For any $i = 1, \ldots, l$, let

$$
\alpha_i = \max_{(X,G,y)\sim\mathcal{D}} \operatorname{Tr}\left[\mathbf{H}^{(i)}[\ell(f(X,G),y)]\right].
$$

Lastly, we use $\sigma_i^2\alpha_i$ above to upper bound E_4. Combined with equations (14) and (15), with probability at least $1-2\delta$, we get

$$L(f) - \frac{1}{\beta}\hat{L}(f) \le \frac{C_2\sqrt{2h^2\log(C_3N/\delta)}}{\sqrt{N}}\sum_{i=1}^{l}\sigma_i^2 + \Big(1+\frac{1}{\beta}\Big)C_1\sum_{i=1}^{l}\sigma_i^3 + \frac{1}{2}\Big(\frac{1}{\beta}-1\Big)\sum_{i=1}^{l}\alpha_i\sigma_i^2 + \frac{B}{2\beta(1-\beta)N}\left(\sum_{i=1}^{l}\frac{s_i^2r_i^2}{\sigma_i^2} + \log\frac{1}{\delta}\right).$$

Next, we will select σ_i to minimize the last line above. One can verify that this is achieved when

$$\sigma_i^2 = \frac{s_ir_i}{1-\beta}\sqrt{\frac{B}{\alpha_iN}}, \text{ for every } i = 1,2,\ldots,l.$$

With this setting of the noise variance, the gap between $L(f)$ and $\hat{L}(f)/\beta$ becomes:

$$L(f) - \frac{1}{\beta}\hat{L}(f) \le \frac{1}{\beta}\sum_{i=1}^{l}\sqrt{\frac{B\alpha_is_i^2r_i^2}{N}} + \frac{C_2\sqrt{2h^2\log(C_3N/\delta)}}{\sqrt{N}}\sum_{i=1}^{L}\sigma_i^2 + \Big(1+\frac{1}{\beta}\Big)C_1\sum_{i=1}^{l}\sigma_i^3 + \frac{C}{2\beta(1-\beta)N}\log\frac{1}{\delta}.$$

Let β be a fixed value close to 1 and independent of N and δ^{-1}; let $\epsilon = (1-\beta)/\beta$. We get

$$L(f) \le (1+\epsilon)\hat{L}(f) + (1+\epsilon)\sum_{i=1}^{l}\sqrt{\frac{B\alpha_ir_i^2s_i^2}{N}} + \xi, \text{ where}$$

$$\xi = \frac{C_2\sqrt{2h^2\log(C_3N/\delta)}}{\sqrt{N}}\sum_{i=1}^{L}\sigma_i^2 + \Big(1+\frac{1}{\beta}\Big)C_1\sum_{i=1}^{l}\sigma_i^3 + \frac{C}{2\beta(1-\beta)N}\log\frac{1}{\delta}.$$

Notice that ξ is of order $O(N^{-3/4} + \log(\delta^{-1})N^{-1}) \le O(\log(\delta^{-1})/N^{3/4})$. Therefore, we have finished the proof of equation (5). □

A.1.1 Proof of Proposition A.4

For any $j = 1,2,\ldots,l$, let $\tilde{H}^{(j)}$ be the perturbed network output after layer j, with perturbations given by $\Delta\boldsymbol{W}$ and $\Delta\boldsymbol{U}$. We show that the following Lipschitz property for $H^{(j)}$.

Claim A.5. *Suppose that Assumption A.2 holds. For any $j = 1,\ldots,l-1$, the change in the output of the j layer network $H^{(j)}$ with perturbation added to $\boldsymbol{W}$ and $\boldsymbol{U}$ can be bounded as follows:*

$$\left\|\tilde{H}^{(j)} - H^{(j)}\right\|_F \lesssim \sum_{t=1}^{j}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \tag{16}$$

Proof. We will prove using induction with respect to j. If $j = 1$, we have

$$\begin{aligned}&\left\|\phi_1\Big(X\big(U^{(1)}+\Delta U^{(1)}\big) + \rho_1\big(P_G\psi_1(X)\big)\big(W^{(1)}+\Delta W^{(1)}\big)\Big) - \phi_1\Big(XU^{(1)} + \rho_1\big(P_G\psi_1(X)\big)W^{(1)}\Big)\right\|_F\\ \le&\kappa_0\left\|X\Delta U^{(1)} + \rho_1\big(P_G\psi_1(X)\big)\Delta W^{(1)}\right\|_F \lesssim \left\|\Delta U^{(1)}\right\| + \left\|\Delta W^{(1)}\right\|.\end{aligned}$$

Hence, we know that equation (16) will be correct when $j = 1$. Assuming that equation (16) is correct for any $j \geq 1$, the perturbation of layer $j+1$'s network output $H^{(j+1)}$ is less than

$$\begin{aligned}
&\left\|\tilde{H}^{(j+1)} - H^{(j+1)}\right\|_F \\
\leq\ & \kappa_0 \left\|X\Delta U^{(j+1)} + \rho_{j+1}\left(P_G \psi_{j+1}\left(\tilde{H}^{(j)}\right)\right)\left(W^{(j+1)} + \Delta W^{(j+1)}\right) - \rho_{j+1}\left(P_G \psi_{j+1}\left(H^{(j)}\right)\right) W^{(j+1)}\right\|_F \\
\lesssim\ & \left\|\Delta U^{(j+1)}\right\| + \left\|\Delta W^{(j+1)}\right\| + \left\|\tilde{H}^{(j)} - H^{(j)}\right\|_F.
\end{aligned}$$

Thus, we have finished the proof of the induction step. □

Next, for any i and j, let $\frac{\partial \tilde{H}^{(j)}}{\partial W^{(i)}}$ be the perturbation of the partial derivative of $H^{(j)}$ with perturbations given by $\Delta \boldsymbol{W}$ and $\Delta \boldsymbol{U}$.

Claim A.6. *Suppose that Assumption A.2 holds. For any $j = 1, \ldots, l-1$, the change in the Jacobian of the j-th layer's output $H^{(j)}$ with respect to $W^{(i)}$ and $U^{(i)}$ satisfies:*

$$\left\|\frac{\partial \tilde{H}^{(j)}}{\partial W^{(i)}} - \frac{\partial H^{(j)}}{\partial W^{(i)}}\right\|_F \lesssim \sum_{t=1}^{j} \left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \tag{17}$$

$$\left\|\frac{\partial \tilde{H}^{(j)}}{\partial U^{(i)}} - \frac{\partial H^{(j)}}{\partial U^{(i)}}\right\|_F \lesssim \sum_{t=1}^{j} \left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \tag{18}$$

Proof. We will consider a fixed $i = 1, \ldots, l-1$ and take induction over $j = i, \ldots, l-1$. We focus on the proof of equation (17), while the proof of equation (18) will be similar. To simplify the derivation, we use two notations for brevity. Let

$$F_j = P_G \psi_j\left(H^{(j-1)}\right) W^{(j)} \text{ and } E_j = XU^{(j)} + \rho_j\left(F_j\right).$$

First, we consider the base case when $j = i$. By the chain rule, we have:

$$\left\|\frac{\partial \tilde{H}^{(i)}}{\partial W^{(i)}} - \frac{\partial H^{(i)}}{\partial W^{(i)}}\right\|_F = \left\|\phi_i'\left(\tilde{E}_i\right) \odot \frac{\partial \tilde{E}_i}{\partial W^{(i)}} - \phi_i'\left(E_i\right) \odot \frac{\partial E_i}{\partial W^{(i)}}\right\|_F \lesssim \left\|\phi_i'\left(\tilde{E}_i\right) - \phi_i'\left(E_i\right)\right\|_F + \left\|\frac{\partial \tilde{E}_i}{\partial W^{(i)}} - \frac{\partial E_i}{\partial W^{(i)}}\right\|_F.$$

From Claim A.5, we know

$$\left\|\phi_i'\left(\tilde{E}_i\right) - \phi_i'\left(E_i\right)\right\|_F \leq \kappa_1 \left\|\tilde{E}_i - E_i\right\|_F \lesssim \left\|\Delta W^{(i)}\right\| + \left\|\Delta U^{(i)}\right\|.$$

By the chain rule again, we get:

$$\left\|\frac{\partial \tilde{E}_i}{\partial W^{(i)}} - \frac{\partial E_i}{\partial W^{(i)}}\right\|_F \lesssim \left\|\rho_i'\left(\tilde{F}_i\right) - \rho_i'\left(F_i\right)\right\|_F + \left\|\frac{\partial \tilde{F}_i}{\partial W^{(i)}} - \frac{\partial F_i}{\partial W^{(i)}}\right\|_F \lesssim \left\|\Delta W^{(i)}\right\| + \left\|\Delta U^{(i)}\right\|.$$

(by Claim A.5 again)

Hence, we know that equation (17) will be correct when $j = i$. Assuming that equation (17) will be correct

for any j up to $j \geq i$, we have

$$\begin{aligned}
\left\|\frac{\partial \tilde{H}^{(j+1)}}{\partial W^{(i)}} - \frac{\partial H^{(j+1)}}{\partial W^{(i)}}\right\|_F &\lesssim \left\|\phi'_{j+1}(\tilde{E}_{j+1}) - \phi'_{j+1}(E_{j+1})\right\|_F + \left\|\frac{\partial \tilde{E}_{j+1}}{\partial W^{(i)}} - \frac{\partial E_{j+1}}{\partial W^{(i)}}\right\|_F \\
&\lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right) + \left\|\rho'_{j+1}(\tilde{F}_{j+1}) - \rho'_{j+1}(F_{j+1})\right\|_F + \left\|\frac{\partial \tilde{F}_{j+1}}{\partial W^{(i)}} - \frac{\partial F_{j+1}}{W^{(i)}}\right\|_F \\
&\lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right) + \left\|\psi'_{j+1}(\tilde{H}^{(j)}) - \psi'_{j+1}(H^{(j)})\right\|_F + \left\|\frac{\partial \tilde{H}^{(j)}}{\partial W^{(i)}} - \frac{\partial H^{(j)}}{\partial W^{(i)}}\right\|_F \\
&\lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \qquad \text{(by Claim A.5 and the induction step)}
\end{aligned}$$

The above steps all use Claim A.5. The last step additionally uses the induction hypothesis. From repeatedly applying the above beginning with $j = i$ along with the base case of equation (17), we conclude that equation (17) holds.

Next, we consider the base case for equation (18). For the base case $j = i$, from the chain rule, we get:

$$\left\|\frac{\partial \tilde{H}^{(i)}}{\partial U^{(i)}} - \frac{\partial H^{(i)}}{\partial U^{(i)}}\right\|_F \lesssim \left\|\phi'_i(\tilde{E}_i) - \phi'_i(E_i)\right\|_F + \left\|\frac{\partial \tilde{E}_i}{\partial U^{(i)}} - \frac{\partial E_i}{\partial U^{(i)}}\right\|_F \lesssim \left\|\Delta W^{(i)}\right\| + \left\|\Delta U^{(i)}\right\|. \qquad \text{(by Claim A.5)}$$

Hence, we know that equation (18) will be correct when $j = i$. Assuming that equation (18) will be correct for any j up to $j \geq i$, we have

$$\begin{aligned}
\left\|\frac{\partial \tilde{H}^{(j+1)}}{\partial U^{(i)}} - \frac{\partial H^{(j+1)}}{\partial U^{(i)}}\right\|_F &\lesssim \left\|\phi'_{j+1}(\tilde{E}_{j+1}) - \phi'_{j+1}(E_{j+1})\right\|_F + \left\|\frac{\partial \tilde{E}_{j+1}}{\partial U^{(i)}} - \frac{\partial E_{j+1}}{\partial U^{(i)}}\right\|_F \\
&\lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right) + \left\|\rho'_{j+1}(\tilde{F}_{j+1}) - \rho'_{j+1}(F_{j+1})\right\|_F + \left\|\frac{\partial \tilde{F}_{j+1}}{\partial U^{(i)}} - \frac{\partial F_{j+1}}{\partial U^{(i)}}\right\|_F \\
&\lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right) + \left\|\psi'_{j+1}(\tilde{H}^{(j)}) - \psi'_{j+1}(H^{(j)})\right\|_F + \left\|\frac{\partial \tilde{H}^{(j)}}{\partial U^{(i)}} - \frac{\partial H^{(j)}}{\partial U^{(i)}}\right\|_F \\
&\lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta U^{(t)}\right\| + \left\|\Delta W^{(t)}\right\|\right). \qquad \text{(by Claim A.5 and the induction step)}
\end{aligned}$$

The second and third steps are based on Claim A.5. From repeatedly applying the above beginning with $j = i$ along with the base case of equation (18), we conclude that equation (18) holds. The proof of claim A.6 is complete. □

Proof of Proposition A.4. We will consider a fixed $i = 1, \ldots, l-1$ and take induction over $j = i, \ldots, l-1$. We focus on the proof of equation (12), while the proof of equation (13) will be similar. To simplify the derivation, we use two notations for brevity. Let

$$F_j = P_G \psi_j\left(H^{(j-1)}\right) W^{(j)} \text{ and } E_j = XU^{(j)} + \rho_j\left(F_j\right).$$

First, we consider the base case when $j = i$. By the chain rule, we have: We use the chain rule to get:

$$\frac{\partial^2 H^{(i)}}{\partial\left(W_{p,q}^{(i)}\right)^2} = \phi_i''(E_i) \odot \frac{\partial E_i}{\partial W_{p,q}^{(i)}} \odot \frac{\partial E_i}{\partial W_{p,q}^{(i)}} + \phi_i'(E_i) \odot \rho_i''(F_i) \odot \frac{\partial F_i}{\partial W_{p,q}^{(i)}} \odot \frac{\partial F_i}{\partial W_{p,q}^{(i)}}.$$

Hence, the Frobenius norm of the Hessian of $H^{(i)}$ with respect to W_i under perturbation on W and U turns to

$$\left\|\mathbf{H}_{\boldsymbol{W}}^{(i)}[\tilde{H}^{(i)}]-\mathbf{H}_{\boldsymbol{W}}^{(i)}[H^{(i)}]\right\|_F \lesssim \left\|\phi_i''(\tilde{E}_i)-\phi_i''(E_i)\right\|_F+\left\|\frac{\partial \tilde{E}_i}{\partial W^{(i)}}-\frac{\partial E_i}{\partial W^{(i)}}\right\|_F+\left\|\phi_i'(\tilde{E}_i)-\phi_i'(E_i)\right\|_F$$
$$+\left\|\rho_i''(\tilde{F}_i)-\rho_i''(F_i)\right\|_F+\left\|\frac{\partial \tilde{F}_i}{\partial W^{(i)}}-\frac{\partial F_i}{\partial W^{(i)}}\right\|_F.$$

From Claim A.5, we know

$$\left\|\phi_i''(\tilde{E}_i)-\phi_i''(E_i)\right\|_F \le \kappa_2\left\|\tilde{E}_i-E_i\right\|_F \lesssim \left\|\Delta W^{(i)}\right\|+\left\|\Delta U^{(i)}\right\|,$$
$$\left\|\phi_i'(\tilde{E}_i)-\phi_i'(E_i)\right\|_F \le \kappa_1\left\|\tilde{E}_i-E_i\right\|_F \lesssim \left\|\Delta W^{(i)}\right\|+\left\|\Delta U^{(i)}\right\|,$$
$$\left\|\rho_i''(\tilde{F}_i)-\rho_i''(F_i)\right\|_F \le \kappa_2\left\|\tilde{F}_i-F_i\right\|_F \lesssim \left\|\Delta W^{(i)}\right\|+\left\|\Delta U^{(i)}\right\|.$$

From Claim A.6, we have

$$\left\|\frac{\partial \tilde{E}_i}{\partial W^{(i)}}-\frac{\partial E_i}{\partial W^{(i)}}\right\|_F \lesssim \left\|\Delta W^{(i)}\right\|+\left\|\Delta U^{(i)}\right\|,$$
$$\left\|\frac{\partial \tilde{F}_i}{\partial W^{(i)}}-\frac{\partial F_i}{\partial W^{(i)}}\right\|_F \lesssim \left\|\Delta W^{(i)}\right\|+\left\|\Delta U^{(i)}\right\|.$$

Hence, we know that equation (12) will be correct when $j=i$. Assuming that equation (12) will be correct for any j up to $j \ge i$, we can get the following steps, by taking another derivative of the first-order derivative, we can get the following steps:

$$\frac{\partial^2 H^{(j+1)}}{\partial\left(W_{p,q}^{(i)}\right)^2}=\phi_{j+1}''(E_{j+1})\odot\frac{\partial E_{j+1}}{\partial W_{p,q}^{(i)}}\odot\frac{\partial E_{j+1}}{\partial W_{p,q}^{(i)}}+\phi_{j+1}'(E_{j+1})\odot\rho_{j+1}''(F_{j+1})\odot\frac{\partial F_{j+1}}{\partial W_{p,q}^{(i)}}\odot\frac{\partial F_{j+1}}{\partial W_{p,q}^{(i)}}$$
$$+\phi_{j+1}'(E_{j+1})\odot\rho_{j+1}'(F_{j+1})\odot P_G\Big(\psi_{j+1}''(H^{(j)})\odot\frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}}\odot\frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}}+\psi_{j+1}'(H^{(j)})\odot\frac{\partial^2 H^{(j)}}{\partial\left(W_{p,q}^{(i)}\right)^2}\Big)W^{(j+1)}.$$

Thus, the Frobenius norm of the Hessian of $H^{(j+1)}$ with respect to $W^{(i)}$ satisfies:

$$\left\|\mathbf{H}_{\boldsymbol{W}}^{(i)}[\tilde{H}^{(j+1)}]-\mathbf{H}_{\boldsymbol{W}}^{(i)}[H^{(j+1)}]\right\|_F \lesssim \underbrace{\left\|\phi_{j+1}''(\tilde{E}_{j+1})-\phi_{j+1}''(E_{j+1})\right\|_F}_{A_1}+\underbrace{\left\|\frac{\partial \tilde{E}_{j+1}}{\partial W^{(i)}}-\frac{\partial E_{j+1}}{\partial W^{(i)}}\right\|_F}_{B_1}$$
$$+\underbrace{\left\|\phi_{j+1}'(\tilde{E}_{j+1})-\phi_{j+1}'(E_{j+1})\right\|_F}_{A_2}+\underbrace{\left\|\rho_{j+1}''(\tilde{F}_{j+1})-\rho_{j+1}''(F_{j+1})\right\|_F}_{A_3}+\underbrace{\left\|\frac{\partial \tilde{F}_{j+1}}{\partial W^{(i)}}-\frac{\partial F_{j+1}}{\partial W^{(i)}}\right\|_F}_{B_2}+\underbrace{\left\|\rho_{j+1}'(\tilde{F}_{j+1})-\rho_{j+1}'(F_{j+1})\right\|_F}_{A_4}$$
$$+\underbrace{\left\|\psi_{j+1}''(\tilde{H}^{(j)})-\psi_{j+1}''(H^{(j)})\right\|_F}_{A_5}+\underbrace{\left\|\frac{\partial \tilde{H}^{(j)}}{\partial W^{(i)}}-\frac{\partial H^{(j)}}{\partial W^{(i)}}\right\|_F}_{B_3}+\underbrace{\left\|\psi_{j+1}'(\tilde{H}^{(j)})-\psi_{j+1}'(H^{(j)})\right\|_F}_{A_6}+\underbrace{\left\|\mathbf{H}_{\boldsymbol{W}}^{(i)}[\tilde{H}^{(j)}]-\mathbf{H}_{\boldsymbol{W}}^{(i)}[H^{(j)}]\right\|_F}_{C_1}.$$

Similarly, by Claim A.5, we get

$$A_i \lesssim \sum_{t=1}^{j+1}\left(\left\|\Delta W^{(t)}\right\|+\left\|\Delta U^{(t)}\right\|\right),\ \text{for } 1\le i\le 6.$$

By Claim A.6, we get

$$B_i \lesssim \sum_{t=1}^{j+1} \left(\left\| \Delta W^{(t)} \right\| + \left\| \Delta U^{(t)} \right\| \right), \text{ for } 1 \le i \le 3.$$

By the induction hypothesis, C_1 is also less than the above quantity. From repeatedly applying the above beginning with $j = i$ along with the base case of equation (12), we conclude that equation (12) holds.

Next, we consider the base case for equation (13). For the base case $j = i$, from the chain rule, we get:

$$\begin{aligned}
\left\| \mathbf{H}_U^{(i)}[\tilde{H}^{(i)}] - \mathbf{H}_U^{(i)}[H^{(i)}] \right\|_F &\lesssim \left\| \phi_i''(\tilde{E}_i) - \phi_i''(E_i) \right\|_F + \left\| \frac{\partial \tilde{E}_i}{\partial U^{(i)}} - \frac{\partial E_i}{\partial U^{(i)}} \right\|_F \\
&\lesssim \kappa_2 \left\| \tilde{E}_i - E_i \right\|_F + \left\| \Delta W^{(i)} \right\| + \left\| \Delta U^{(i)} \right\| && \text{(by Claim A.6)} \\
&\lesssim \left\| \Delta W^{(i)} \right\| + \left\| \Delta U^{(i)} \right\|. && \text{(by Claim A.5)}
\end{aligned}$$

Hence, we know that equation (13) will be correct when $j = i$. Assuming that equation (13) will be correct for any j up to $j \ge i$, we obtain the induction step similar to the proof of equation (12), by Claim A.5, Claim A.6, and the induction hypothesis, we conclude that equation (13) holds. □

A.2 Proof for message passing graph neural networks

Next, we present a proof for message passing graph neural networks. First in Appendix A.2.1, we derive the trace bound, which separates the trace of the Hessian matrix into each entry of the weight matrices. Then in Appendix A.2.2 and A.2.3, we provides bounds on the first-order and second-order derivatives of the Hessian matrix. Last in Appendix A.2.4, building on these results, we finish the proof of Theorem 3.1.

A.2.1 Proof of Lemma 4.3

Proof of Lemma 4.3. Notice that $f(X, G) = H^{(l)}$. Recall that in each layer for $1 \le i \le l-1$, there are two weight matrices, a d_{i-1} by d_i matrix denoted as $W^{(i)}$, and a d_0 by $U^{(i)}$ matrix denoted as $U^{(i)}$. In order to deal with trace of the Hessian $\mathbf{H}^{(i)}$, we first notice that there are two parts in the trace:

$$\left| \mathrm{Tr}\left[\mathbf{H}^{(i)}[\ell(H^{(l)}, y)] \right] \right| \le \underbrace{\left| \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \frac{\partial^2 \ell(H^{(l)}, y)}{\partial (W_{p,q}^{(i)})^2} \right|}_{T_1} + \underbrace{\left| \sum_{p=1}^{d_0} \sum_{q=1}^{d_i} \frac{\partial^2 \ell(H^{(l)}, y)}{\partial (U_{p,q}^{(i)})^2} \right|}_{T_2}.$$

We can inspect T_1 and T_2 in the above step separately. First, we expand out the second-order derivatives in T_1. This will involve two terms by the chain rule.

$$\begin{aligned}
T_1 &= \left| \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\langle \frac{\partial \ell(H^{(l)}, y)}{\partial H^{(l)}}, \frac{\partial^2 H^{(l)}}{\partial (W_{p,q}^{(i)})^2} \right\rangle \right| + \left| \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\langle \frac{\partial^2 \ell(H^{(l)}, y)}{\partial (H^{(l)})^2} \frac{\partial H^{(l)}}{\partial W_{p,q}^{(i)}}, \frac{\partial H^{(l)}}{\partial W_{p,q}^{(i)}} \right\rangle \right| \\
&\le \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial \ell(H^{(l)}, y)}{\partial H^{(l)}} \right\| \left\| \frac{\partial^2 H^{(l)}}{\partial (W_{p,q}^{(i)})^2} \right\| + \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial^2 \ell(H^{(l)}, y)}{\partial (H^{(l)})^2} \right\| \left\| \frac{\partial H^{(l)}}{\partial W_{p,q}^{(i)}} \right\|^2 \\
&\le \kappa_0 \sqrt{k} \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial^2 H^{(l)}}{\partial (W_{p,q}^{(i)})^2} \right\| + \kappa_1 k \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial H^{(l)}}{\partial W_{p,q}^{(i)}} \right\|^2. \qquad (19)
\end{aligned}$$

The last step is because $\ell(\cdot)$ is κ_0-Lipschitz continuous and $\ell'(\cdot)$ is κ_1-Lipschitz continuous, under Assumption A.2. Thus, the Euclidean norm of $\frac{\partial \ell(H^{(l)},y)}{\partial H^{(l)}}$ is at most $\kappa_0\sqrt{k}$, since $H^{(l)}$ is a k-dimensional vector. Recall from step (2) that $H^{(l)} = \frac{1}{n}\mathbf{1}_n^\top H^{(l-1)}W^{(l)}$. Hence, we have

$$\begin{aligned}\left\|\frac{\partial H^{(l)}}{\partial W_{p,q}^{(i)}}\right\| &= \left\|\frac{1}{n}\mathbf{1}_n^\top \frac{\partial H^{(l-1)}}{\partial W_{p,q}^{(i)}} W^{(l)}\right\| \\ &\le \left\|\frac{1}{n}\mathbf{1}_n^\top\right\| \left\|\frac{\partial H^{(l-1)}}{\partial W_{p,q}^{(i)}} W^{(l)}\right\| \le \frac{1}{\sqrt{n}} \left\|\frac{\partial H^{(l-1)}}{\partial W_{p,q}^{(i)}}\right\| \left\|W^{(l)}\right\|. \end{aligned} \tag{20}$$

In a similar vein, the Euclidean norm of $\frac{\partial^2 \ell(H^{(l)},y)}{\partial (H^{(l)})^2}$ is at most $\kappa_1 k$, since the second-order derivatives become a k by k matrix. Then, we get

$$\begin{aligned}\left\|\frac{\partial^2 H^{(l)}}{\partial \left(W_{p,q}^{(i)}\right)^2}\right\| &= \left\|\frac{1}{n}\mathbf{1}_n^\top \frac{\partial^2 H^{(l-1)}}{\partial \left(W_{p,q}^{(i)}\right)^2} W^{(l)}\right\| \\ &\le \left\|\frac{1}{n}\mathbf{1}_n^\top\right\| \left\|\frac{\partial^2 H^{(l-1)}}{\partial \left(W_{p,q}^{(i)}\right)^2} W^{(l)}\right\| \le \frac{1}{\sqrt{n}} \left\|\frac{\partial^2 H^{(l-1)}}{\partial \left(W_{p,q}^{(i)}\right)^2}\right\| \left\|W^{(l)}\right\|. \end{aligned} \tag{21}$$

After substituting equations (20) and (21) into equation (19), we get:

$$\begin{aligned} T_1 &\le \frac{\kappa_0\sqrt{k}}{\sqrt{n}} \left\|W^{(l)}\right\| \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\|\frac{\partial^2 H^{(l-1)}}{\partial \left(W_{p,q}^{(i)}\right)^2}\right\| + \frac{\kappa_1 k}{n} \left\|W^{(l)}\right\|^2 \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\|\frac{\partial H^{(l-1)}}{\partial W_{p,q}^{(i)}}\right\|^2. \\ &\le \frac{\kappa_0\sqrt{k}}{\sqrt{n}} \left\|W^{(l)}\right\| \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\|\frac{\partial^2 H^{(l-1)}}{\partial \left(W_{p,q}^{(i)}\right)^2}\right\|_F + \frac{\kappa_1 k}{n} \left\|W^{(l)}\right\|^2 \sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\|\frac{\partial H^{(l-1)}}{\partial W_{p,q}^{(i)}}\right\|_F^2. \end{aligned}$$

The proof for the case of T_2 concerning $U^{(i)}$ follows the same steps as above. Without belaboring all the details, one can get that

$$T_2 \le \frac{\kappa_0\sqrt{k}}{\sqrt{n}} \left\|W^{(l)}\right\| \sum_{p=1}^{d_0} \sum_{q=1}^{d_i} \left\|\frac{\partial^2 H^{(l-1)}}{\partial \left(U_{p,q}^{(i)}\right)^2}\right\|_F + \frac{\kappa_1 k}{n} \left\|W^{(l)}\right\|^2 \sum_{p=1}^{d_0} \sum_{q=1}^{d_i} \left\|\frac{\partial H^{(l-1)}}{\partial U_{p,q}^{(i)}}\right\|_F^2. \tag{22}$$

This completes the proof of Lemma 4.3. □

A.2.2 Dealing with first-order derivatives

Based on Lemma 4.3, the analysis involves two parts, one on the first-order derivatives of $H^{(j)}$ for all layers j, and the other on the second-order derivatives of $H^{(j)}$ for all layers j.

Proposition A.7. *In the setting of Theorem 3.1, the first-order derivative of $H^{(j)}$ with respect to $W^{(i)}$ and $U^{(i)}$ satisfies the following, for any $i = 1, \ldots, l-1$ and $j \ge i$:*

$$\left\|\frac{\partial H^{(j)}}{\partial W^{(i)}}\right\|_F \le \kappa_0^{3(j-i+1)} \sqrt{d_i} \left\|P_G\right\|^{j-i+1} \left\|H^{(i-1)}\right\|_F \prod_{t=i+1}^{j} \left\|W^{(t)}\right\|, \tag{23}$$

$$\left\|\frac{\partial H^{(j)}}{\partial U^{(i)}}\right\|_F \le \kappa_0^{3(j-i)+1} \sqrt{d_i} \left\|P_G\right\|^{j-i+1} \|X\|_F \prod_{t=i+1}^{j} \left\|W^{(t)}\right\|. \tag{24}$$

Proof. We will consider a fixed $i = 1, \ldots, l-1$ and take induction over $j = i, \ldots, l-1$. We focus on the proof of equation (23), while the proof of equation (24) will be similar. First, we consider the base case when $j = i$. Let $W^{(i)}_{p,q}$ be the (p,q)-th entry of $W^{(i)}$, for any valid indices p and q. Recall that $\phi_i(\cdot)$ is κ_0-Lipschitz continuous from Assumption A.2, for any $i = 1, \ldots, l-1$. Therefore,

$$\left\|\phi_i'(x)\right\|_\infty \le \kappa_0, \quad \left\|\psi_i'(x)\right\|_\infty \le \kappa_0, \quad \text{and} \quad \left\|\rho_i'(x)\right\|_\infty \le \kappa_0. \tag{25}$$

For each (p,q)-entry of $W^{(i)}$, by the chain rule, we have:

$$\begin{aligned}
\left\|\frac{\partial H^{(i)}}{\partial W^{(i)}_{p,q}}\right\|_F &= \left\|\phi_i'\big(XU^{(i)} + \rho_i\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)\big) \odot \frac{\partial\big(XU^{(i)} + \rho_i\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)\big)}{\partial W^{(i)}_{p,q}}\right\|_F && (26)\\
&\le \kappa_0 \left\|\frac{\partial \rho_i\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)}{\partial W^{(i)}_{p,q}}\right\|_F && \text{(by equation (25))}\\
&= \kappa_0 \left\|\rho_i'\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big) \odot \frac{\partial\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)}{W^{(i)}_{p,q}}\right\|_F\\
&\le \kappa_0^2 \left\|\frac{\partial\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)}{\partial W^{(i)}_{p,q}}\right\|_F. && \text{(again by equation (25))}
\end{aligned}$$

Notice that only the q-th column of the derivative $P_G\psi_i(H^{(i-1)})W^{(i)}$ is nonzero, which is equal to the p'th column of $P_G\psi_i(H^{(i-1)})$. Thus, the Jacobian of $H^{(i)}$ over $W^{(i)}$ satisfies:

$$\left\|\frac{\partial H^{(i)}}{\partial W^{(i)}}\right\|_F = \sqrt{\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial H^{(i)}}{\partial W^{(i)}_{p,q}}\right\|_F^2} \le \kappa_0^2\sqrt{\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)}{W^{(i)}_{p,q}}\right\|_F^2} = \kappa_0^2\sqrt{d_i}\left\|P_G\psi_i(H^{(i-1)})\right\|_F. \tag{27}$$

Therefore, the above equation (27) implies that equation (23) holds in the base case. Next, we consider the induction step from layer j to layer $j+1$. The derivative of $H^{(j+1)}$ with respect to $W^{(i)}_{p,q}$ satisfies:

$$\begin{aligned}
\left\|\frac{\partial H^{(j+1)}}{\partial W^{(i)}_{p,q}}\right\|_F &= \left\|\phi_{j+1}'\big(XU^{(j+1)} + \rho_{j+1}\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)\big) \odot \frac{\partial\big(XU^{(j+1)} + \rho_{j+1}\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)\big)}{\partial W^{(i)}_{p,q}}\right\|_F\\
&\le \kappa_0 \left\|\frac{\partial \rho_{j+1}\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)}{\partial W^{(i)}_{p,q}}\right\|_F && \text{(by equation (25))}\\
&\le \kappa_0 \left\|\rho_{j+1}'\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big) \odot \frac{\partial\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)}{\partial W^{(i)}_{p,q}}\right\|_F\\
&\le \kappa_0^2 \left\|P_G \frac{\partial \psi_{j+1}(H^{(j)})}{\partial W^{(i)}_{p,q}} W^{(j+1)}\right\|_F && \text{(again by equation (25))}
\end{aligned}$$

By applying equation (25) w.r.t. ψ_{j+1}', The above is less than:

$$\kappa_0^2\left\|P_G\right\|\left\|\psi_{j+1}'(H^{(j)}) \odot \frac{\partial H^{(j)}}{\partial W^{(i)}_{p,q}}\right\|_F\left\|W^{(j+1)}\right\| \le \kappa_0^3\left\|P_G\right\|\left\|W^{(j+1)}\right\|\left\|\frac{\partial H^{(j)}}{\partial W^{(i)}_{p,q}}\right\|_F.$$

Hence, the Jacobian of $H^{(j+1)}$ with respect to $W^{(i)}$ satisfies:

$$\left\|\frac{\partial H^{(j+1)}}{\partial W^{(i)}}\right\|_F \le \kappa_0^3\left\|P_G\right\|\left\|W^{(j+1)}\right\|\left\|\frac{\partial H^{(j)}}{\partial W^{(i)}}\right\|_F.$$

From repeatedly applying the above beginning with $j = i$ along with the base case of equation (27), we conclude that equation (23) holds.

Next, we consider the base case for equation (24). For each (p, q)-th entry of $U^{(i)}$, from the chain rule we get:

$$\left\|\frac{\partial H^{(i)}}{\partial U^{(i)}_{p,q}}\right\|_F = \left\|\phi_i'\Big(XU^{(i)} + \rho_i\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)\Big) \odot \frac{\partial\Big(XU^{(i)} + \rho_i\big(P_G\psi_i(H^{(i-1)})W^{(i)}\big)\Big)}{\partial U^{(i)}_{p,q}}\right\|_F$$
$$\leq \kappa_0 \left\|\frac{\partial(XU^{(i)})}{\partial U^{(i)}_{p,q}}\right\|_F. \qquad \text{(by equation (25))}$$

Therefore, by summing over $p = 1, \ldots, d_0$ and $q = 1, \ldots, d_i$, we get:

$$\left\|\frac{\partial H^{(i)}}{\partial U^{(i)}}\right\|_F = \sqrt{\sum_{p=1}^{d_0}\sum_{q=1}^{d_i}\left\|\frac{\partial H^{(i)}}{\partial U^{(i)}_{p,q}}\right\|_F^2}$$
$$\leq \kappa_0\sqrt{\sum_{p=1}^{d_0}\sum_{q=1}^{d_i}\left\|\frac{\partial(XU^{(i)})}{\partial U^{(i)}_{p,q}}\right\|_F^2} = \kappa_0\sqrt{d_i}\,\|X\|_F. \tag{28}$$

Going from layer i to layer $j+1$, the derivative of $H^{(j+1)}$ with respect to $U^{(i)}_{p,q}$ satisfies:

$$\left\|\frac{\partial H^{(j+1)}}{\partial U^{(i)}_{p,q}}\right\|_F = \left\|\phi_{j+1}'\big(XU^{(j+1)} + \rho_{j+1}\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)\big) \odot \frac{\partial\big(XU^{(j+1)} + \rho_{j+1}\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)\big)}{\partial U^{(i)}_{p,q}}\right\|_F$$
$$\leq \kappa_0\left\|\frac{\partial\rho_{j+1}\big(P_G\psi_{j+1}(H^{(j)})W^{(j+1)}\big)}{\partial U^{(i)}_{p,q}}\right\|_F \qquad \text{(by equation (25) w.r.t. } \phi_{j+1}')$$
$$\leq \kappa_0^3\left\|P_G\right\|\left\|W^{(j+1)}\right\|\left\|\frac{\partial H^{(j)}}{\partial U^{(i)}_{p,q}}\right\|_F. \qquad \text{(by equation (25) w.r.t. } \rho_{j+1}', \psi_{j+1}')$$

Hence, the Jacobian of $H^{(j+1)}$ with respect to $U^{(i)}$ satisfies:

$$\left\|\frac{\partial H^{(j+1)}}{\partial U^{(i)}}\right\|_F \leq \kappa_0^3\left\|P_G\right\|\left\|W^{(j+1)}\right\|\left\|\frac{\partial H^{(j)}}{\partial U^{(i)}}\right\|_F.$$

By repeatedly applying the above step beginning with the base case of equation (28), we have proved that equation (24) holds. The proof of Proposition A.7 is complete. □

A.2.3 Deal with second-order derivatives

In the second part towards showing Theorem 3.1 for MPNNs, we look at second-order derivatives of the embeddings. This will appear later when we deal with the trace of the Hessian. A fact that we will use throughout the proof is

$$\left\|\phi_i''(x)\right\|_\infty \leq \kappa_1, \quad \left\|\psi_i''(x)\right\|_\infty \leq \kappa_1, \quad \text{and} \quad \left\|\rho_i''(x)\right\|_\infty \leq \kappa_1, \tag{29}$$

for any x and $i = 1, \ldots, l-1$. This is because ϕ_i', ψ_i', and ρ_i' are all κ_1-Lipschitz continuous from Assumption A.2.

Proposition A.8. *In the setting of Theorem 3.1, the second-order derivative of $H^{(l)}$ with respect to $W^{(i)}$ and $U^{(i)}$ satisfies the following, for any $i = 1, \dots, l-1$ and any $j = i, \dots, l-1$:*

$$\sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial^2 H^{(j)}}{\left(W_{p,q}^{(i)}\right)^2} \right\|_F \leq C_{i,j} \kappa_1 d_i \max\left(\left\|P_G\right\|^{j-i+2}, \left\|P_G\right\|^{2(j-i+1)} \right) \left\|H^{(i-1)}\right\|_F^2 \prod_{t=i+1}^{j} s_t^2, \tag{30}$$

$$\sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial^2 H^{(j)}}{\left(U_{p,q}^{(i)}\right)^2} \right\|_F \leq \hat{C}_{i,j} \kappa_1 d_i \max\left(\left\|P_G\right\|^{j-i}, \left\|P_G\right\|^{2(j-i)} \right) \|X\|_F^2 \prod_{t=i+1}^{j} s_t^2, \tag{31}$$

where $C_{i,j}$

$$C_{i,j} = \begin{cases} \kappa_0^{3(j-i+1)} \dfrac{\kappa_0^{3(j-i)+2} - 1}{\kappa_0 - 1}, & \kappa_0 \neq 1, \\ 3(j-i) + 2, & \kappa_0 = 1, \end{cases}$$

and $\hat{C}_{i,j}$

$$\hat{C}_{i,j} = \begin{cases} \kappa_0^{3(j-i)} \dfrac{\kappa_0^{3(j-i)+1} - 1}{\kappa_0 - 1}, & \kappa_0 \neq 1, \\ 3(j-i) + 1, & \kappa_0 = 1. \end{cases}$$

are fixed constants that depend on the Lipschitzness of the activation mappings.

Proof. First we will consider equation (30). To simplify the derivation, we introduce two notations for brevity. Let

$$F_j = P_G \psi_j\left(H^{(j-1)}\right) W^{(j)} \text{ and } E_j = X U^{(j)} + \rho_j\left(F_j\right).$$

In the base case when $j = i$, from the first-order derivative in equation (26), we use the chain rule to get:

$$\frac{\partial^2 H^{(i)}}{\partial\left(W_{p,q}^{(i)}\right)^2} = \phi_i''(E_i) \odot \frac{\partial E_i}{\partial W_{p,q}^{(i)}} \odot \frac{\partial E_i}{\partial W_{p,q}^{(i)}} + \phi_i'(E_i) \odot \rho_i''(F_i) \odot \frac{\partial F_i}{\partial W_{p,q}^{(i)}} \odot \frac{\partial F_i}{\partial W_{p,q}^{(i)}}. \tag{32}$$

Using equation (29), the maximum entries of $\phi_i''(\cdot), \rho_i''(\cdot)$ are at most κ_1. Using equation (25), the maximum entry of $\phi_i'(\cdot)$ is at most κ_0. Notice that the derivative of E_i can be reduced to the derivative of F_i as follows:

$$\left\| \frac{\partial E_i}{\partial W_{p,q}^{(i)}} \right\|_F^2 = \left\| \rho_i'(F_i) \odot \frac{\partial F_1}{\partial W_{p,q}^{(i)}} \right\|_F^2 \leq \kappa_0^2 \left\| \frac{\partial F_i}{\partial W_{p,q}^{(i)}} \right\|_F^2. \tag{33}$$

Therefore, based on the conditions for first- and second-order derivatives (cf. (25) and (29)), the Frobenius norm of the above equation (32) is at most:

$$\left\| \frac{\partial^2 H^{(i)}}{\partial\left(W_{p,q}^{(i)}\right)^2} \right\|_F \leq \kappa_1 \left\| \frac{\partial E_i}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0 \kappa_1 \left\| \frac{\partial F_i}{\partial W_{p,q}^{(i)}} \right\|_F^2 \leq (\kappa_0 + 1) \kappa_0 \kappa_1 \left\| \frac{\partial F_i}{\partial W_{p,q}^{(i)}} \right\|_F^2.$$

Notice that the derivative of F_i with respect to $W_{p,q}^{(i)}$ is nonzero only in the q-th column of F_i, and is equal to the p-th column of $P_G g_i(H^{(i-1)})$. Therefore, by summing over $p = 1, \dots, d_{i-1}$ and $q = 1, \dots, d_i$, we get:

$$\sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial F_i}{\partial\left(W_{p,q}^{(i)}\right)^2} \right\|_F^2 \leq d_i \left\| P_G \psi_i(H^{(i-1)}) \right\|_F^2.$$

Therefore, we have derived the base case when $j = i$ as:

$$\left\| \frac{\partial^2 H^{(i)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F \leq (\kappa_0 + 1)\kappa_0^3 \kappa_1 d_i \left\| P_G \right\|^2 \left\| H^{(i-1)} \right\|_F^2. \tag{34}$$

Next, we consider the induction step from layer j to layer $j+1$. This step is similar to the base case but also differs since $H^{(j)}$ is now dependent on $W^{(i)}$. Recall that the second-order derivatives satisfy equation (29). Based on the Lipschitzness conditions, the Frobenius norm of the second-order derivatives satisfies:

$$\begin{aligned}
\left\| \frac{\partial^2 H^{(j+1)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F &\leq \kappa_1 \left\| \frac{\partial E_{j+1}}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0 \kappa_1 \left\| \frac{\partial F_{j+1}}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0^2 \left\| P_G \right\| \left\| W^{(j+1)} \right\| \left(\kappa_1 \left\| \frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0 \left\| \frac{\partial^2 H^{(j)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F \right) \\
&\leq (\kappa_0 + 1)\kappa_0 \kappa_1 \left\| \frac{\partial F_{j+1}}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0^2 \left\| P_G \right\| \left\| W^{(j+1)} \right\| \left(\kappa_1 \left\| \frac{H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0 \left\| \frac{\partial^2 H^{(j)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F \right).
\end{aligned} \tag{35}$$

The last step follows similarly as equation (33). For the derivative of F_{j+1}, using the chain rule, we get:

$$\begin{aligned}
\left\| \frac{\partial F_{j+1}}{\partial W_{p,q}^{(i)}} \right\|_F^2 &= \left\| P_G \frac{\partial \psi_{j+1}(H^{(j)})}{\partial W_{p,q}^{(i)}} W^{(j+1)} \right\|_F^2 \\
&\leq \left\| P_G \right\|^2 \left\| W^{(j+1)} \right\|^2 \left\| \frac{\partial \psi_{j+1}(H^{(j)})}{\partial W_{p,q}^{(i)}} \right\|_F^2 \\
&\leq \left\| P_G \right\|^2 \left\| W^{(j+1)} \right\|^2 \left\| \psi'_{j+1}(H^{(j)}) \odot \frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2 \\
&\leq \kappa_0^2 \left\| P_G \right\|^2 \left\| W^{(j+1)} \right\|^2 \left\| \frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2.
\end{aligned}$$

Therefore, combining the above with equations (35) together, we get the following result:

$$\begin{aligned}
\left\| \frac{\partial^2 H^{(j+1)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F &\leq \left((\kappa_0 + 1)\kappa_0^3 \kappa_1 \left\| P_G \right\|^2 \left\| W^{(j+1)} \right\|^2 + \kappa_0^2 \kappa_1 \left\| P_G \right\| \left\| W^{(j+1)} \right\| \right) \left\| \frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2 \\
&\quad + \kappa_0^3 \left\| P_G \right\| \left\| W^{(j+1)} \right\| \left\| \frac{\partial^2 H^{(j)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F \\
&\leq \max\left(\left\| P_G \right\|, \left\| P_G \right\|^2 \right) s_{j+1}^2 \left((\kappa_0^2 + \kappa_0 + 1)\kappa_0^2 \kappa_1 \left\| \frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2 + \kappa_0^3 \left\| \frac{\partial^2 H^{(j)}}{\partial \left(W_{p,q}^{(i)}\right)^2} \right\|_F \right).
\end{aligned}$$

Based on equation (23) of Proposition (A.7), the first-order derivative of $H^{(j)}$ satisfies:

$$\sum_{p=1}^{d_{i-1}} \sum_{q=1}^{d_i} \left\| \frac{\partial H^{(j)}}{\partial W_{p,q}^{(i)}} \right\|_F^2 \leq \kappa_0^{6(j-i+1)} d_i \left\| P_G \right\|^{2(j-i+1)} \left\| H^{(i-1)} \right\|^2 \prod_{t=i+1}^{j} s_t^2. \tag{36}$$

Applying equation (36) to the above (and summing over $p = 1, \ldots, d_{i-1}$ and $q = 1, \ldots, d_i$) forms the induction

step for showing equation (30):

$$\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(j+1)}}{\partial\left(W_{p,q}^{(i)}\right)^2}\right\|_F \le \frac{\kappa_0^3-1}{\kappa_0-1}\kappa_0^{6(j-i+1)+2}\kappa_1 d_i \max\left(\left\|P_G\right\|^{2(j-i)+3},\left\|P_G\right\|^{2(j-i)+4}\right)\left\|H^{(i-1)}\right\|_F^2\prod_{t=i+1}^{j+1}s_t^2$$
$$+\kappa_0^3\max\left(\left\|P_G\right\|,\left\|P_G\right\|^2\right)s_{j+1}^2\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(j)}}{\partial\left(W_{p,q}^{(i)}\right)^2}\right\|_F.$$

By repeatedly applying the induction step along with the base case in equation (34), we have shown that equation (30) holds:

$$\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(j)}}{\left(W_{p,q}^{(i)}\right)^2}\right\|_F \le C_{i,j}\kappa_1 d_i\max(\left\|P_G\right\|^{j-i+2},\left\|P_G\right\|^{2(j-i+1)})\left\|H^{(i-1)}\right\|_F^2\prod_{t=i+1}^{j}s_t^2, \tag{37}$$

where $C_{i,j}$ satisfies the following equation:

$$C_{i,j}=\begin{cases}\kappa_0^{3(j-i+1)}\dfrac{\kappa_0^{3(j-i)+2}-1}{\kappa_0-1}, & \kappa_0\neq 1,\\ 3(j-i)+2, & \kappa_0=1.\end{cases}$$

In the second part of the proof, we consider equation (31) similar to the first part. However the analysis will be significantly simpler. We first consider the base case. Similar to equation (32), the second-order derivative of $H^{(i)}$ over $W_{p,q}^{(i)}$ satisfies, for any $p=1,\ldots,d_0$ and $q=1,\ldots,d_i$:

$$\left\|\frac{\partial^2 H^{(i)}}{\partial\left(U_{p,q}^{(i)}\right)^2}\right\|_F=\left\|\phi_i''(E_i)\odot\frac{\partial E_i}{\partial U_{p,q}^{(i)}}\odot\frac{\partial E_i}{\partial U_{p,q}^{(i)}}\right\|\le\kappa_1\left\|\frac{\partial(XU^{(i)})}{\partial U_{p,q}^{(i)}}\right\|_F^2.$$

Therefore, by summing up the above over all p and q, we get the base case result:

$$\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(i)}}{\partial\left(U_{p,q}^{(i)}\right)^2}\right\|_F\le\kappa_1 d_i\left\|X\right\|_F^2. \tag{38}$$

Next, we consider the induction step from layer j to layer $j+1$. This step follows the same analysis until equation (37), from which we can similarly derive that:

$$\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(j)}}{\left(U_{p,q}^{(i)}\right)^2}\right\|_F\le\hat{C}_{i,j}\kappa_1 d_i\max(\left\|P_G\right\|^{j-i},\left\|P_G\right\|^{2(j-i)})\left\|X\right\|_F^2\prod_{t=i+1}^{j}s_t^2. \tag{39}$$

where $\hat{C}_{i,j}$ satisfies the following equation:

$$\hat{C}_{i,j}=\begin{cases}\kappa_0^{3(j-i)}\dfrac{\kappa_0^{3(j-i)+1}-1}{\kappa_0-1}, & \kappa_0\neq 1,\\ 3(j-i)+1, & \kappa_0=1.\end{cases}$$

□

A.2.4 Proof of Theorem 3.1

Based on Propositions A.7 and A.8, we are ready present the proof of Theorem 3.1 for message passing GNNs. First, we will apply the bounds on the derivatives back in Lemma 4.3. After getting the trace of the Hessians, we then use the PAC-Bayes bound from Lemma 4.1 to complete the proof.

Proof of Theorem 3.1. By applying equations (23) and (30) into Lemma 4.3's result, we get that the trace of $\mathbf{H}^{(l)}$ with respect to $W^{(i)}$ is less than:

$$\kappa_0 \frac{\sqrt{k}}{\sqrt{n}} C_{i,l-1} \kappa_1 d_i \max\left(\left\|P_G\right\|^{l-i+1}, \left\|P_G\right\|^{2(l-i)}\right) \left\|H^{(i-1)}\right\|_F^2 \left(\prod_{t=i+1}^{l} s_t^2\right) + \kappa_1 \frac{k}{n} \kappa_0^{6(l-i)} \kappa_1 d_i \left\|P_G\right\|^{2(l-i)} \left\|H^{(i-1)}\right\|_F^2 \prod_{t=i+1}^{l} s_t^2$$

$$\leq (\kappa_0 C_{i,l-1} + \kappa_0^{6(l-i)}) \sqrt{\frac{k}{n}} \kappa_1 d_i \max\left(\left\|P_G\right\|^{l-i+1}, \left\|P_G\right\|^{2(l-i)}\right) \left\|H^{(i-1)}\right\|_F^2 \prod_{t=i+1}^{l} s_t^2, \tag{40}$$

for any $i = 1, 2, \cdots, l-1$. Here we have

$$\kappa_0 C_{i,l-1} + \kappa_0^{6(l-i)} = \begin{cases} \kappa_0^{3(l-i)+1} \dfrac{\kappa_0^{3(l-i)} - 1}{\kappa_0 - 1}, & \kappa_0 \neq 1, \\ 3(l-i) - 1, & \kappa_0 = 1. \end{cases}$$

It remains to consider the Frobenius norm of $H^{(i-1)}$. Notice that this satisfies:

$$\begin{aligned} \left\|H^{(i-1)}\right\|_F &\leq \kappa_0 \left\|XU^{(i-1)} + \rho_{i-1}(P_G \psi_{i-1}(H^{(i-2)}))W^{(i-1)}\right\|_F \\ &\leq \kappa_0 \left\|U^{(i-1)}\right\| \|X\|_F + \kappa_0^3 \left\|P_G\right\| \left\|W^{(i-1)}\right\| \left\|H^{(i-2)}\right\|_F \qquad \leq \kappa_0 s_i \|X\|_F + \kappa_0^3 \left\|P_G\right\| s_i \left\|H^{(i-2)}\right\|_F. \end{aligned}$$

By induction over i for the above step, we get that the Frobenius norm of $H^{(i-1)}$ must be less than:

$$\left(\kappa_0^{3(i-1)} + \sum_{j=0}^{i-2} \kappa_0^{3j+1}\right) \sqrt{k} \max_{(X,G,y)\sim\mathcal{D}} \|X\| \max\left(1, \left\|P_G\right\|^{i-1}\right) \prod_{j=1}^{i-1} s_j. \tag{41}$$

By applying the above (41) back in (40), we have shown that the trace of $\mathbf{H}^{(l)}$ with respect to $W^{(i)}$ is less than:

$$C' \max_{(X,G,y)\sim\mathcal{D}} \|X\|^2 \kappa_1 d_i k \max(1, \|P_G\|^{2(l-1)}) \prod_{t=1:t\neq i}^{l} s_t^2, \tag{42}$$

where C' satisfies the following equation:

$$C' = \begin{cases} \dfrac{(\kappa_0^{3l} - 1)(\kappa_0^{3(l-1)/2} - 1)^2}{(\kappa_0 - 1)^3}, & \kappa_0 \neq 1, \\ \dfrac{4}{9} l^3, & \kappa_0 = 1. \end{cases}$$

To be specific, when $\kappa_0 = 1$, $(3(l-i)-1)i^2 \leq \frac{4}{9} l^3$. If $\kappa_0 \neq 1$ and $i \geq 2$, we have

$$\begin{aligned} \left(\kappa_0^{3(l-i)+1} \frac{\kappa_0^{3(l-i)} - 1}{\kappa_0 - 1}\right)\left(\kappa_0^{3(i-1)} + \sum_{j=0}^{i-2} \kappa_0^{3j+1}\right)^2 &\leq \kappa_0^{3(l-i)+3} \frac{\kappa_0^{3(l-i)} - 1}{\kappa_0 - 1} \frac{(\kappa_0^{3(i-1)} - 1)^2}{(\kappa_0 - 1)^2} \\ &= \frac{\kappa_0^{3l} - \kappa_0^{3(l-i+1)}}{(\kappa_0 - 1)^3} \left((\kappa_0^{3(l-i)} - 1)(\kappa_0^{3(i-1)} - 1)\right) \\ &\leq \frac{(\kappa_0^{3l} - 1)(\kappa_0^{3(l-1)/2} - 1)^2}{(\kappa_0 - 1)^3}. \end{aligned}$$

If $\kappa_0 \neq 1$ and $i = 1$, we obtain

$$\left(\kappa_0^{3(l-i)+1}\frac{\kappa_0^{3(l-i)}-1}{\kappa_0-1}\right)\left(\kappa_0^{3(i-1)}+\sum_{j=0}^{i-2}\kappa_0^{3j+1}\right)^2 = \kappa_0^{3l-2}\frac{\kappa_0^{3(l-1)}-1}{\kappa_0-1} \leq \frac{(\kappa_0^{3l}-1)(\kappa_0^{3(l-1)/2}-1)^2}{(\kappa_0-1)^3}.$$

The above works for the layers from the beginning until layer $l-1$. Last, we consider the trace of $\mathbf{H}^{(l)}$ with respect to $W^{(l)}$ (notice that $\boldsymbol{U}$ is not needed in the readout layer). Similar to equation (19), one can prove that the trace of the Hessian with respect to $W^{(l)}$ satisfies:

$$\begin{aligned}
\left|\mathrm{Tr}\left[\mathbf{H}^{(l)}[\ell(H^{(l)},y)]\right]\right| &\leq \kappa_0\sqrt{k}\sum_{p=1}^{d_{l-1}}\sum_{q=1}^{d_l}\left\|\frac{\partial^2 H^{(l)}}{\partial\left(W_{p,q}^{(l)}\right)^2}\right\| + \kappa_1 k\sum_{p=1}^{d_{l-1}}\sum_{q=1}^{d_l}\left\|\frac{\partial H^{(l)}}{\partial W_{p,q}^{(l)}}\right\|^2 \\
&\leq \kappa_0\sqrt{k}\sum_{p=1}^{d_{l-1}}\sum_{q=1}^{d_l}\left\|\frac{1}{n}\mathbf{1}_n^\top H^{(l-1)}\frac{\partial^2 W^{(l)}}{\partial\left(W_{p,q}^{(l)}\right)^2}\right\| + \kappa_1 k\sum_{p=1}^{d_{l-1}}\sum_{q=1}^{d_l}\left\|\frac{1}{n}\mathbf{1}_n^\top H^{(l-1)}\frac{\partial W^{(l)}}{\partial W_{p,q}^{(l)}}\right\|^2 \\
&\leq \kappa_1 k\sum_{p=1}^{d_{l-1}}\sum_{q=1}^{d_l}\left\|\frac{1}{n}\mathbf{1}_n\right\|^2\left\|H^{(l-1)}\frac{\partial W^{(l)}}{\partial W_{p,q}^{(l)}}\right\|^2 \\
&= \kappa_1\frac{k}{n}d_l\left\|H^{(l-1)}\right\|_F^2
\end{aligned}$$

By equation (41), the above is bounded by

$$\begin{aligned}
&\kappa_1\frac{k}{n}d_l\left(\kappa_0^{3(l-1)}+\sum_{j=0}^{l-2}\kappa_0^{3j+1}\right)^2\max_{(X,G,y)\sim\mathcal{D}}\|X\|_F^2\max\left(1,\left\|P_G\right\|^{2(l-1)}\right)\prod_{j=1}^{l-1}s_j^2 \\
\leq\ & C_l\max_{(X,G,y)\sim\mathcal{D}}\|X\|^2\kappa_1 d_l k\max\left(1,\left\|P_G\right\|^{2(l-1)}\right)\prod_{t=1:t\neq l}^{l}s_t^2,
\end{aligned}$$

since $\frac{\|X\|_F^2}{n} \leq \|X\|^2$, where C_l satisfies the following equation:

$$C_l = \begin{cases} \kappa_0^2\dfrac{(\kappa_0^{3(l-1)}-1)^2}{(\kappa_0-1)^2}, & \kappa_0 \neq 1, \\ l^2, & \kappa_0 = 1. \end{cases}$$

Finally, let

$$\tilde{C} = \max(C', C_l). \tag{43}$$

From the value of C' above and the value of C_l, we get that $\tilde{C}$ is equal to

$$\tilde{C} = \begin{cases} \dfrac{(\kappa_0^{3l}-1)(\kappa_0^{3(l-1)/2}-1)^2}{(\kappa_0-1)^3}, & \kappa_0 \neq 1, \\ \dfrac{1}{2}l^3, & \kappa_0 = 1. \end{cases}$$

Similarly by applying equations (24) and (31) into Lemma 4.3, the trace of $\mathbf{H}^{(l)}$ with respect to $U^{(i)}$ is also less than equation (42). Therefore, we have completed the proof for message-passing neural networks. □

A.3 Proof of matching lower bound (Theorem 3.2)

For simplicity, we will exhibit the instance for a graph convnet, that is, we ignore the parameters in $\boldsymbol{U}$ and also set the mapping ρ_t and ψ_t as the identity mapping. Further, we set the mapping $\phi_t(x) = x$ as the identify mapping too, for simplifying the proof. In the proof, we show that for an arbitrary configuration of weight matrices $W^{(1)}, W^{(2)}, \ldots, W^{(l)}$, there exists a data distribution such that for this particular configuration, the generation gap with respect to the data distribution satisfies the desired equation (6).

Proof of Theorem 3.2. Recall that the underlying graph for the lower bound instance is a complete graph. Next, we will specify the other parts of the data distribution $\mathcal{D}$. Let $Z = \prod_{i=1}^{l} W^{(i)}$ denote the product of the weight matrices. We are going to construct a binary classification problem. Thus, the dimension of Z will be equal to n by 2. Let $Z = UDV^\top$ be the singular value decomposition of Z. Let $\lambda_{\max}(Z)$ be the largest singular value of Z, with corresponding left and right singular vectors u_1 and v_1, respectively. Within the hypothesis set $\mathcal{H}$, $\lambda_{\max}(Z)$ can be as large as $\prod_{i=1}^{l} s_i$. Denote a random draw from $\mathcal{D}$ as X, G, y, corresponding to node features, the graph, and the label:

1. The feature matrix X is is equal to $\mathbf{1}_n u_1^\top$;
2. The class label y is drawn uniformly between +1 and −1;
3. Lastly, the diffusion matrix P is the adjacency matrix of G, which has a value of one in every entry of P.

Given the example and the weight matrices, we will use the logistic loss to evaluate f's loss. Notice that $P = \mathbf{1}_n \mathbf{1}_n^\top$. Thus, one can verify $\lambda_{\max}(P) = n$. Crucially, the network output of our GCN is equal to

$$H^{(l)} = \frac{1}{n}\mathbf{1}_n^\top P^{l-1} X W^{(1)} W^{(2)} \cdots W^{(l)} = n^{l-1}\left(\frac{\mathbf{1}_n^\top X}{n} Z\right) = n^{l-1}\left(u_1^\top U D V^\top\right) = \left(n^{l-1}\lambda_{\max}(Z)\right) v_1^\top.$$

Let us denote $\alpha = n^{l-1}\lambda_{\max}(Z)$—the spectral norms of the diffusion matrix and the layer weight matrices. Let $v_{1,1}, v_{1,2}$ be the first and second coordinate of v_1, respectively. Notice that y is drawn uniformly between +1 or −1. Thus, with probability 1/2, the loss of this example is $\log(1 + \exp(-\alpha \cdot v_{1,1}))$; with probability 1/2, the loss of this example is $\log(1 + \exp(\alpha \cdot v_{1,2}))$. Let b_i be a random variable that indicates the logistic loss of the i-th example. The generalization gap is equal to

$$\epsilon = \frac{1}{N}\sum_{i=1}^{N} b_i - \frac{1}{2}\Big(\log(1 + \exp(-\alpha \cdot v_{1,1})) + \log(1 + \exp(\alpha \cdot v_{1,2}))\Big).$$

By the central limit theorem, as N grows to infinity, the generalization gap ϵ converges to a normal random variable whose mean is zero and variance is equal to

$$\frac{1}{4N}\Big(\log(1 + exp(-\alpha \cdot v_{1,1})) - \log(1 + \exp(\alpha \cdot v_{1,2}))\Big)^2 \gtrsim \frac{\alpha^2}{N},$$

for large enough values of n. As a result, with probability at least 0.1, when N is large enough, the generalization gap ϵ must be at least

$$\mathrm{O}\left(\sqrt{\frac{\alpha^2}{N}}\right), \text{ where } \alpha = \left\|P_G\right\|^{l-1} \lambda_{\max}\left(\prod_{i=1}^{l} W^{(i)}\right).$$

Notice that the spectral norm of the product matrix can be realized at most as $\prod_{i=1}^{l} s_i$. Thus, we have completed the proof of equation (6). □

A.4 Proof for graph isomorphism networks (Corollary 4.5)

To be precise, we state the loss function for learning graph isomorphism networks as the averaged loss over all the classification layers:

$$\bar{\ell}(f(X,G),y) = \frac{1}{(l-1)}\sum_{i=1}^{l-1}\ell\Big(\frac{1}{n}\mathbf{1}_n^\top H^{(i)}V^{(i)},y\Big). \tag{44}$$

Thus, $\hat{L}_{GIN}(f)$ is equivalent to the empirical average of $\bar{\ell}$ over N samples from $\mathcal{D}$. $L_{GIN}(f)$ is then equivalent to the expectation of $\bar{\ell}$ over a random sample from $\mathcal{D}$.

Proof of Corollary 4.5. This result follows from the trace guarantee from Lemma 4.3. For any $i = 1,\dots,l-1$ and any $j = i,\dots,l-1$, we can derive the following result with similar arguments:

$$\left|\mathrm{Tr}\left[\mathbf{H}_W^{(i)}\left[\ell\Big(\frac{1}{n}\mathbf{1}_n^\top H^{(j)}V^{(j)},y\Big)\right]\right]\right| \le \frac{\kappa_0\sqrt{k}}{\sqrt{n}}\left\|V^{(j)}\right\|\sum_{p=1}^{d_{i-1}}\sum_{q=1}^{d_i}\left\|\frac{\partial^2 H^{(j)}}{\partial\big(W_{p,q}^{(i)}\big)^2}\right\|_F + \frac{\kappa_1 k}{n}\left\|V^{(j)}\right\|^2\left\|\frac{\partial H^{(j)}}{\partial W^{(i)}}\right\|_F^2.$$

Next, we repeat the steps in Propositions A.7 and A.8, for any $i = 1,\dots,l-1$ and any $j = i,\dots,l-1$:

$$\max_{(X,G,y)\sim\mathcal{D}}\left|\mathrm{Tr}\left[\mathbf{H}^{(i)}\left[\ell\Big(\frac{1}{n}\mathbf{1}_n^\top H^{(j)}V^{(j)},y\Big)\right]\right]\right| \le 2\kappa_1\tilde{C}d_i k\max_{(X,G,y)\sim\mathcal{D}}\|X\|^2\max\left(1,\left\|P_G\right\|^{2(j-i+1)}\right)\left\|V^{(j)}\right\|^2\prod_{t=1:\,t\neq i}^{j}s_t^2.$$

Based on the above step, the trace of the Hessian matrix of the loss function with respect to $W^{(i)}, U^{(i)}$ satisfies:

$$\begin{aligned}\max_{(X,G,y)\sim\mathcal{D}}\left|\mathrm{Tr}\left[\mathbf{H}^{(i)}\left[\bar{\ell}(f(X,G),y)\right]\right]\right| &= \max_{(X,G,y)\sim\mathcal{D}}\left|\mathrm{Tr}\left[\mathbf{H}^{(i)}\Big[\frac{1}{(l-1)}\sum_{j=1}^{l-1}\ell\Big(\frac{1}{n}\mathbf{1}_n^\top H^{(j)}V^{(j)},y\Big)\Big]\right]\right| \\ &= \frac{1}{l-1}\sum_{j=i}^{l-1}\max_{(X,G,y)\sim\mathcal{D}}\left|\mathrm{Tr}\left[\mathbf{H}^{(i)}\left[\ell\Big(\frac{1}{n}\mathbf{1}_n^\top H^{(j)}V^{(j)},y\Big)\right]\right]\right| \\ &\le 2\kappa_1\tilde{C}d_i k\max_{j=1}^{l-1}\left\|V^{(j)}\right\|^2\left(\max_{(X,G,y)\sim\mathcal{D}}\|X\|^2\sum_{j=1}^{l-1}\frac{\max\left(1,\left\|P_G\right\|^{2j}\right)}{l-1}\right)\prod_{t=1:\,t\neq i}^{l-1}s_t^2.\end{aligned}$$

Within the above step, the propagation matrix satisfies:

$$\frac{1}{(l-1)}\sum_{j=1}^{l-1}\max\left(1,\left\|P_G\right\|^{2j}\right) \le \max\left(1,\left\|\frac{1}{l-1}\sum_{j=1}^{l-1}P_G^j\right\|^2\right).$$

Notice that $P_{GIN} = \frac{1}{l-1}\sum_{j=1}^{l-1}P_G^j$. Thus, we have completed the generalization error analysis for graph isomorphism networks in equation (11). □

B Experiment Details

For comparing generalization bounds, we use two types of model architectures, including GCN [32] and the MPGNN in Liao et al. [33]. Following the setup in Liao et al. [33], we apply the same network weights across multiple layers in one model, i.e., $W^{(t)} = W$ and $U^{(t)} = U$ across the first $l-1$ layers. For GCNs, we set $\boldsymbol{U}$ as zero, ρ_t and ψ_t as identity mappings, ϕ_t as ReLU function. For MPGNNs, we specify ϕ_t as ReLU, ρ_t

and ψ_t as Tanh function. For both model architectures, we set the width of each layer $d_t = 128$ and vary the network depth l in 2, 4, and 6. On the three collaboration network datasets, we use one-hot encodings of node degrees as input node features. We train the models with Adam optimizer with a learning rate of 0.01 and set the number of epochs as 50 and batch size as 128 on all three datasets. We compute the generalization bounds following the set up in Liao et al. [33]. We state previous results with our notations in the following.

- Theorem 3.4 from Liao et al. [33]:

$$\sqrt{\frac{42^2}{\gamma^2 N}\Big(\max_{(X,G,y)\sim\mathcal{D}}\|X\|^2\Big)\Big(\max\big(\zeta^{-l+1},(\lambda\xi)^{\frac{l+1}{l}}\big)\Big)^2 l^2 h\log(4lh)(2s_1^2r_1^2+s_l^2r_l^2)},$$

  where $\zeta = \min(s_1, s_l)$, $\lambda = s_1 s_l$, $\xi = \frac{(ds_1)^{l-1}-1}{ds_1-1}$, d is the max degree, h is the max hidden width, and γ is the desired margin in the margin loss. Note that $s_i = s_1$ and $r_i = r_1$ for $1 \le i \le l-1$ since the first $l-1$ layers apply the same weight.

- Proposition 7 from Garg et al. [15]:

$$48 s_l h Z\sqrt{\frac{3}{\gamma^2 N}\log\big(24 s_l\sqrt{N}\max\big(Z, M\sqrt{h}\max\big(\kappa_2 s_1, \bar{R}s_1\big)\big)\big)},$$

  where $M=\frac{(ds_1)^{l-1}-1}{ds_1-1}$, $\bar{R} = d\big(\min(\kappa_1\sqrt{h}, \kappa_2 s_1 M)\big)$, $Z = \kappa_2 s_1 + \bar{R}s_1$, $\kappa_1 = \max_{x\in\mathbb{R}^h}\|\phi(x)\|_\infty$, and $\kappa_2 = \max_{(X,G,y)\sim\mathcal{D}}\|X\|^2$.