ADVERSARIAL STABILITY OF DEEP GCNS UNDER ADAM: WORST-CASE BOUNDS AND RESIDUAL-PATH DIAGNOSTICS

Anonymous authors
Paper under double-blind review

ABSTRACT

The adversarial generalization of deep graph convolutional networks (GCNs) trained with Adam remains theoretically undercharacterized. For sampled Adam updates, we trace graph propagation through both moment histories and derive explicit architecture-dependent stability bounds for scalar-readout vanilla and residual GCNs. For vanilla GCNs, we give an explicit worst-case construction with an exponential-in-depth lower bound on loss smoothness. Isolating the spectral contribution in Adam's stability bound yields a depth scale that is doubly logarithmic in sample size. We call the rapid depth-driven growth of the bound stability-bound collapse. A residual-path decomposition identifies how the architecture enters the stability bound. Under controlled expansion, the temporal accumulation factor grows linearly with the number of updates, while the remaining factors retain their architecture and depth dependence. An auxiliary local result gives sufficient conditions for positive directional curvature, and an expected-graph calculation describes homophily-dependent spectral depth scales for a seven-community stochastic block model. Experiments on five benchmarks examine the associated depth and perturbation trends. Averaged over ten runs, residual GCNs show smaller increases in adversarial cross-entropy gaps with depth than vanilla GCNs on Cora and CiteSeer. Source code is available at `https://anonymous.4open.science/r/AdvStab-GCN`.

1 INTRODUCTION

Deep graph convolutional networks (GCNs) provide a standard framework for representation learning on relational data. Increasing depth enlarges the receptive field, but also exposes representations and parameter gradients to repeated applications of a perturbed graph filter. The same operator appears in forward propagation, neighborhood aggregation, and backpropagation. A structural perturbation can therefore affect both the prediction and the subsequent optimization trajectory. Despite Adam's widespread use in GNN training, prior work on GCN stability has focused mainly on shallow architectures or SGD-based optimization (Verma & Zhang, 2019; Zhou & Wang, 2021; Cao et al., 2025). Adam introduces an additional challenge because each update depends on first- and second-moment estimates accumulated from earlier gradients. Understanding adversarial generalization therefore requires tracing how graph propagation affects gradient sensitivity and how these effects accumulate through Adam's moment estimates.

Uniform stability provides a way to study this interaction by comparing training runs after one labeled example is replaced. Our gradient factorization traces the shared graph filter through forward and backward propagation. Separately, an explicit worst-case construction shows that loss smoothness can grow exponentially with depth, contributing to rapid growth of Adam's stability bound. We refer to this bound growth as *stability-bound collapse*. Residual connections alter the propagation paths entering the bound, while graph topology determines their spectral factors.

This motivates three questions: **(Q1) Architecture and Optimizer Coupling.** How does repeated graph propagation interact with Adam's adaptive updates in the adversarial stability bound? **(Q2) Residual Architecture.** How do residual connections modify the stability recursion and alter its depth dependence? **(Q3) Graph Topology Dependence.** How does graph homophily affect the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

spectral depth scale of the expected graph in a seven-community model? We answer these questions by tracing the shared graph filter through gradients and Adam's moment histories. The residual-path results identify the architectural factors in the bounds. For a seven-community stochastic block model, an auxiliary calculation on the expected graph identifies the spectral dependence on homophily.

Contributions. Our contributions are as follows:

- **Triple Coupling Analysis.** We identify the shared-filter structure of vanilla-GCN gradients. An explicit worst-case scalar construction then attains exponential depth growth in loss smoothness, identifying one source of rapid growth in Adam's stability bound. We further isolate an Adam-specific expansion term, proportional to the inverse square of Adam's numerical stability constant, that has no analogue in the standard SGD recursion.
- **Path-Decomposition and Stability Bounds.** We derive a residual-path decomposition and expected adversarial generalization bounds for vanilla and residual GCNs under structural and feature perturbations. The bounds identify when controlled expansion yields a temporal accumulation factor linear in training time. An auxiliary seven-community calculation relates homophily to the spectral depth scale of the normalized expected adjacency.
- **Local Curvature Analysis.** We establish positive curvature along a nondegenerate parameter direction under explicit Jacobian and Hessian-remainder conditions. This result complements the residual-path stability analysis with a local geometric characterization.
- **Empirical Evaluation.** Experiments on five benchmarks examine depth-dependent cross-entropy and accuracy gaps, empirical Lipschitz proxies, and spectral factors. Comparisons across architectures and optimizers place these gaps alongside predictive accuracy, gradient norms, recorded Adam moments, and update norms.

2 Related Work

Generalization through algorithmic stability. Uniform stability relates generalization to the sensitivity of a training algorithm to sample replacement (Bousquet & Elisseeff, 2002; Hardt et al., 2016). Stability studies have considered shallow GCNs (Verma & Zhang, 2019) and SGD-trained multilayer models (Zhou & Wang, 2021; Cao et al., 2025), while adversarial stability analysis accounts for the perturbed loss (Xiao et al., 2022). We extend this line of analysis by retaining Adam's two moment histories and tracing the shared graph filter through the gradient. This algorithm-dependent view complements capacity and spectral approaches to generalization (Bartlett & Mendelson, 2002; Wang et al., 2025; Hariri et al., 2025). Table 4 in the appendix compares the selected settings and bound structures.

Adversarial graph robustness. Structural attacks examine how topology changes affect node predictions (Xu et al., 2019; Ma et al., 2020). Subsequent work considers scalable evaluations (Geisler et al., 2021) and joint attacks on GNN predictions and explanations (Fan et al., 2023). Robustness certification instead asks whether a trained model preserves its predictions within a prescribed perturbation set (Bojchevski et al., 2020; Schuchardt et al., 2023; Jin et al., 2020). Our focus is the learning algorithm: we study expected adversarial generalization and the sensitivity of its trained output to input replacement.

Depth and residual propagation. Repeated message passing can homogenize node representations or compress information from distant neighborhoods, leading to over-smoothing and over-squashing (Cai & Wang, 2020; Di Giovanni et al., 2023; Topping et al., 2022). These representation-level effects motivate a complementary question about training dynamics. Residual connections change both propagation and the parameter derivatives supplied to the optimizer. Our path-product bounds make their role in Adam's stability recursion explicit. Appendix B provides a broader discussion.

3 Preliminaries

3.1 Model Formulation and Spectral Amplification

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with $|\mathcal{V}| = N$, node features $X \in \mathbb{R}^{N \times d_{in}}$, and parameters $\mathcal{W} = \{W^\ell\}_{\ell=0}^{K-1}$. Here A is the adjacency matrix, d_{in} denotes the input feature dimension, and I

stands for the identity matrix. The augmented filter is $g(A) = \tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}$, where $\tilde{A} = A + I$ and $\tilde{D}$ is the degree matrix of $\tilde{A}$. For depth K and elementwise activation σ, a vanilla GCN and ResGCN propagate as $H^\ell = \sigma(Z^\ell)$ and $H^\ell = \sigma(Z^\ell) + H^{\ell-1}$, where $Z^\ell = g(A)H^{\ell-1}W^{\ell-1}$ and $H^0 = X$. For the ResGCN considered here, identity shortcuts preserve width d_{in}, so $H^\ell \in \mathbb{R}^{N\times d_{in}}$ and $W^{\ell-1} \in \mathbb{R}^{d_{in}\times d_{in}}$. For ResGCN, $(I + \mathcal{F}_K) \circ \cdots \circ (I + \mathcal{F}_1)$ denotes residual block composition, where $\mathcal{F}_\ell(H) = \sigma(g(A)HW^{\ell-1})$. The vanilla GCN has a single output channel, $H^K \in \mathbb{R}^{N\times 1}$. Both architectures use the scalar readout $f(A, X, \mathcal{W})_i = \sigma(\mathbb{I}_i^\top H^K)$, where $\mathbb{I}_i \in \{0, 1\}^N$. Appendix C identifies the additional Jacobian and loss-smoothness estimates required for a multiclass extension.

Perturbation Constants and Spectral Amplification. We consider structural and feature perturbations defined as follows. The structural perturbation set is $\Delta_A := \{A' : \|g(A) - g(A')\|_2 \le \varepsilon_A\}$, and the feature perturbation set is $\Delta_X := \{X' : \|X - X'\|_F \le \varepsilon_X\}$, where $\|\cdot\|_F$ and $\|\cdot\|_2$ denote the Frobenius and the spectral norms, respectively. The budgets satisfy $\varepsilon_A, \varepsilon_X \ge 0$. Write $d_{\max}$ for the maximum degree of $A + I$ and $\tilde{d}_{\max} := \max_{A'\in\Delta_A} d_{\max}(A' + I)$ for its worst-case value. The K-hop neighborhood satisfies $|\mathcal{N}^K(i)| \le \tilde{d}_{\max}^K$ uniformly over structural perturbations. For the feature threat set, we use the uniform admissible-feature bound $C_{x,X} := \sup_{X'\in\Delta_X} \max_i \|x_i'\|_2$. Let $\mu_{\mathcal{W}}(A')$ bound the parameter Lipschitz constant of the base loss at fixed input A', uniformly over labeled examples. Then $\tilde{\mu}_{\mathcal{W}} := \max_{A'\in\Delta_A} \mu_{\mathcal{W}}(A')$ bounds the maximized structural loss. Its uniform parameter smoothness constant $\tilde{\nu}_{\mathcal{W}}$ is supplied separately by Assumption 5. The worst-case filter norm is $\widetilde{C}_g = \max_{A'\in\Delta_A} \|g(A')\|_2$. Write μ_σ for the activation Lipschitz constant and C_W for a uniform upper bound on the layer weight spectral norms. At fixed parameters, the clean layer propagation factor is $\rho_j := \mu_\sigma \|g(A)\|_2 \|W^{j-1}\|_2$, and $\tilde{\rho}_j := \mu_\sigma \widetilde{C}_g \|W^{j-1}\|_2$ bounds the corresponding factor uniformly over structural perturbations. Their layerwise maxima are $\rho := \max_{1\le j\le K} \rho_j$ and $\tilde{\rho} := \max_{1\le j\le K} \tilde{\rho}_j$. For a balanced seven-community stochastic block model (SBM), $\varrho(h)$ in Corollary 2 combines the nontrivial spectrum of the normalized expected adjacency with $\mu_\sigma C_W$, where h denotes homophily. Table 3 in Appendix A summarizes the notation used throughout the paper.

3.2 Adversarial Stability and Optimization Dynamics

Let $S = \{z_j\}_{j=1}^n$ be the training set of size n, with each z_j specifying a target node and its label. For the population result, the examples are sampled independently from a distribution $\mathcal{D}$, conditional on the graph and features. Let $S^{(i)}$ replace only the example at index i. We couple their training runs using the same random sample indices and write $\mathcal{A}_S$ and $\mathcal{A}_{S^{(i)}}$ for the trained parameters. The expectation $\mathbb{E}_{\mathcal{A}}$ is over algorithmic randomness, while $\mathbb{E}_{S,\mathcal{A}}$ also averages over independently sampled training examples. Appendix C discusses how this conditional sampling model relates to the fixed transductive splits used in the experiments.

Adversarial Risk. For a specified threat set Δ (structural Δ_A, feature Δ_X, or their stated product set), the adversarial loss is $\widetilde{\mathcal{L}}(\mathcal{W}, z) := \max_{\delta\in\Delta} \mathcal{L}(f_\delta(A, X, \mathcal{W}), y)$, where $\mathcal{L}$ is the prediction loss, y the target label associated with z, and $f_\delta(A, X, \mathcal{W})$ the corresponding target-node prediction under perturbation δ. The main depth experiments use perturbed inputs constructed before training, with attack objectives and budgets specified in Appendix G.1. We define the empirical adversarial risk as $\mathcal{R}_{\mathrm{emp}}(\widetilde{\mathcal{L}}) = \frac{1}{n}\sum_{i=1}^n \widetilde{\mathcal{L}}(\mathcal{W}, z_i)$ and the expected adversarial risk as $\mathcal{R}_{\mathrm{adv}}(\widetilde{\mathcal{L}}) = \mathbb{E}_{z\sim\mathcal{D}}[\widetilde{\mathcal{L}}(\mathcal{W}, z)]$. The adversarial generalization gap is $\mathrm{Gengap} := |\mathbb{E}_{S,\mathcal{A}}[\mathcal{R}_{\mathrm{adv}}(\widetilde{\mathcal{L}}) - \mathcal{R}_{\mathrm{emp}}(\widetilde{\mathcal{L}})]|$. The subscript in Gengap_Δ specifies the threat set. An algorithm $\mathcal{A}$ is α-uniformly stable if $\sup_z |\mathbb{E}_{\mathcal{A}}[\widetilde{\mathcal{L}}(\mathcal{A}_S, z) - \widetilde{\mathcal{L}}(\mathcal{A}_{S^{(i)}}, z)]| \le \alpha$, uniformly over neighboring $S, S^{(i)}$, which implies $\mathrm{Gengap} \le \alpha$ (Bousquet & Elisseeff, 2002).

Adam Optimization Dynamics. We run Adam for T updates with learning rate $\eta > 0$, decay coefficients β_1, β_2, and initial moments $m_0 = v_0 = 0$. At step $t = 1, \ldots, T$, sample j_t uniformly from $\{1, \ldots, n\}$ and set $\mathbf{g}_t = \nabla_{\mathcal{W}} \widetilde{\mathcal{L}}(\mathcal{W}_{t-1}, z_{j_t})$. With $\odot$ denoting elementwise multiplication, the moment estimates are

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\mathbf{g}_t, \quad v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\mathbf{g}_t \odot \mathbf{g}_t), \quad 0 \le \beta_1 < 1, \quad 0 < \beta_2 < 1.$$

We assume $0 < \beta_2 < 1$, as required by the moment-comparison bound in Lemma 10. Write $\Phi_t(\mathcal{W}_{t-1}, z)$ for the parameter update computed with m_{t-1} and v_{t-1} from the specified run. When

the comparison changes only z, these past moments remain fixed.

$$\mathcal{W}_t = \Phi_t(\mathcal{W}_{t-1}, z_{j_t}) = \mathcal{W}_{t-1} - \eta \hat{s}_t \odot \hat{m}_t, \qquad \hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t}.$$

Here $\varepsilon_0 > 0$ is Adam's numerical stability constant, and $\hat{s}_t = (\sqrt{\hat{v}_t} + \varepsilon_0)^{-1}$ is applied elementwise. Coupling the sampled indices gives the decomposition

$$\|\mathcal{W}_t - \mathcal{W}'_t\|_F \le \underbrace{\|\Phi_t(\mathcal{W}_{t-1}, z_{j_t}) - \Phi_t(\mathcal{W}'_{t-1}, z_{j_t})\|_F}_{\text{Update sensitivity}} + \underbrace{\|\Phi_t(\mathcal{W}'_{t-1}, z_{j_t}) - \Phi_t(\mathcal{W}'_{t-1}, z'_{j_t})\|_F}_{\text{Sample replacement}}.$$

In the first term, the two updates use their respective past moments. In the second, both candidate updates use the primed run's parameters and past moments, and differ only in the selected example. Lemma 14 in Appendix D.5 controls the complete histories and yields the cumulative stability recursion.

The main results concern Adam trained on the maximized adversarial loss. Appendix E studies the auxiliary algorithm $\mathcal{A}^{\mathrm{fix}}$, which trains on specified perturbed inputs, and quantifies its sensitivity to a same-label input replacement.

3.3 Assumptions and Regularity Conditions

To compare Adam's bias-corrected first and second moments over T updates, define

$$c_T := \max_{1 \le t \le T} \left[\frac{(1-\beta_1)^2(1-\beta_2^t)}{(1-\beta_2)(1-\beta_1^t)^2} \sum_{r=0}^{t-1} \left(\frac{\beta_1^2}{\beta_2} \right)^r \right]^{1/2}. \tag{1}$$

It depends only on β_1, β_2, T and satisfies $|(\hat{m}_t)_k| \le c_T \sqrt{(\hat{v}_t)_k}$ for every coordinate k and $1 \le t \le T$. Lemma 10 in Appendix D.5 proves this consequence of the Adam updates.

The analysis assumes bounded features and weights, smooth activation and loss functions, and a positive numerical stability constant for Adam. Appendix C explains the role of these assumptions and their relation to the experimental settings.

Assumption 1 (Bounded Features and Filter). *$\|x_i\|_2 \le C_x$ for all $i \in \mathcal{V}$, where $x_i \in \mathbb{R}^{d_{in}}$ is the feature vector of node i, and $\|g(A)\|_2 \le C_g$.*

Assumption 2 (Bounded Weights). $\max_{0 \le \ell \le K-1} \|W^\ell\|_2 \le C_W$.

Assumption 3 (Lipschitz Activation). *The activation function σ is μ_σ-Lipschitz and ν_σ-smooth with $\sigma(0) = 0$: $\|\sigma(u) - \sigma(v)\| \le \mu_\sigma \|u - v\|$ and $\|\nabla\sigma(u) - \nabla\sigma(v)\| \le \nu_\sigma \|u - v\|$.*

Assumption 4 (Bounded Adaptive Scaling). *$\varepsilon_0 > 0$ and $\|\hat{s}_t\|_\infty \le 1/\varepsilon_0$.*

Assumption 5 (Loss Regularity). *For each admissible label y and scalar prediction on the bounded model domain, the base loss satisfies $|\partial_f \mathcal{L}(f, y)| \le M_f$ and $|\partial_f \mathcal{L}(f, y) - \partial_f \mathcal{L}(f', y)| \le \nu_{\mathcal{L}} |f - f'|$. For the specified architecture and threat set, assume that the maximized adversarial loss is differentiable and satisfies, uniformly over examples z and admissible parameters $\mathcal{W}, \mathcal{W}'$,*

$$|\widetilde{\mathcal{L}}(\mathcal{W}, z) - \widetilde{\mathcal{L}}(\mathcal{W}', z)| \le \mu_{\mathcal{W}} \|\mathcal{W} - \mathcal{W}'\|_F,$$
$$\|\nabla_{\mathcal{W}} \widetilde{\mathcal{L}}(\mathcal{W}, z) - \nabla_{\mathcal{W}} \widetilde{\mathcal{L}}(\mathcal{W}', z)\|_F \le \nu_{\mathcal{W}} \|\mathcal{W} - \mathcal{W}'\|_F.$$

These constants may depend on the architecture and threat set, but are independent of the sampled trajectory. For structural attacks, the corresponding constants are denoted by $\tilde{\mu}_{\mathcal{W}}$ and $\tilde{\nu}_{\mathcal{W}}$. The auxiliary fixed-input comparisons use a common $\nu_{\mathcal{W}}$ that also bounds the parameter-gradient Lipschitz constants of the specified fixed-input losses.

4 Adversarial Stability Bounds for Vanilla and Residual GCNs

We begin with the gradient factorization and worst-case smoothness construction in Theorem 1. Theorems 2 and 3 give structural-perturbation generalization bounds for vanilla and residual GCNs. Theorems 5 and 6 in Appendix D establish the corresponding generalization bounds under feature perturbations. Appendix E gives budget-dependent bounds for same-label input replacement.

4.1 Generalization Bounds for Vanilla GCN under Adam

Adam rescales each coordinate using its gradient history, so the stability of an update depends on both the current gradient and the accumulated moments. In a vanilla GCN, the shared graph filter contributes to every stage of gradient formation. The following theorem makes this dependence explicit.

Theorem 1 (Graph-Specific Triple Coupling in GCN Gradients)**.** *Let a K-layer vanilla GCN satisfy Assumptions 1–5. At a maximizing adversarial input, the layerwise loss gradient can be written as*

$$\nabla_{W^{\ell-1}}\widetilde{\mathcal{L}} = \underbrace{(H^{\ell-1})^\top}_{\text{forward state}} \quad \underbrace{g(A)^\top}_{\text{global propagator}} \quad \underbrace{\delta^\ell}_{\text{backward chain}}, \qquad 1 \le \ell \le K.$$

All input-dependent quantities are evaluated at a maximizing adversarial input, held fixed when computing the backpropagated error $\delta^\ell := \partial\mathcal{L}(f_i, y)/\partial Z^\ell$. The forward state contains $\ell - 1$ graph-filter applications, the explicit factor contributes one, and the backward chain contains the remaining $K - \ell$. These chains contribute K explicit graph-filter factors, while the activation and loss derivatives also depend on the evaluated input.

This repeated propagation can yield exponential depth dependence in loss smoothness. In particular, there exists an admissible single-node GCN with one-dimensional features at every layer, $g(A) = [1]$, a linear activation, and squared loss for which $\nu_{\mathcal{W}} = \Omega\big(\mu_\sigma^{2K} C_g^{2K} C_W^{2K-2}\big) = \Omega(\rho^{2K})$. In this construction, $C_g = 1$ and $\rho = \mu_\sigma C_W$, with positive μ_σ, C_W, and C_x fixed as depth varies. Choosing $\rho > 1$ establishes an attainable exponential depth rate within the admissible class.

Remark 1 (Comparison with MLP Gradient Factorization)**.** *The factorization exposes the graph-specific contribution to the gradient. A standard MLP has the layer gradient $(H^{\ell-1})^\top \delta^\ell$, whereas a GCN also reuses the graph filter throughout the forward and backward chains. The scalar construction attains the $\Omega(\rho^{2K})$ rate with $g(A) = [1]$. In this witness, the weight and activation factors supply the amplification. Through the exponential term in Adam's stability recursion, this smoothness dependence can lead to the rapid depth growth discussed in Remark 2.*

We now connect the gradient analysis to sample-replacement stability. Loss smoothness controls how discrepancies propagate through Adam, while a uniform gradient bound controls the contribution of a replaced example. Assumption 2 gives $\|W_t^{j-1}\|_2, \|(W_t')^{j-1}\|_2 \le C_W$ for every layer and step in both runs. The uniform bounds therefore use $\rho_j \le \mu_\sigma C_g C_W$ and $\tilde{\rho}_j \le \mu_\sigma \widetilde{C}_g C_W$ on both trajectories.

Theorem 2 (Generalization Bound under Structural Attacks for Vanilla GCN)**.** *Let S contain n labeled examples drawn independently from $\mathcal{D}$, conditional on the graph and features. Consider a K-layer vanilla GCN trained on S for T Adam updates under the structural threat set Δ_A. Under Assumptions 1–5, the multiplicative propagation chain yields the uniform adversarial gradient bound*

$$M_{\text{van}} = K M_f \sqrt{\tilde{d}_{\max}^K}\, C_x \mu_\sigma^{K+1} \widetilde{C}_g^{\,K} C_W^{K-1}. \tag{2}$$

Define $a_T^{\text{van}} := \varepsilon_0^{-1} + 2M_{\text{van}} c_T \varepsilon_0^{-2}$. The resulting Adam stability recursion has expansion coefficient

$$\zeta_{\text{van}} = \eta \tilde{\nu}_{\mathcal{W}} a_T^{\text{van}}. \tag{3}$$

Then the absolute expected adversarial generalization gap of $\mathcal{W}_T$ satisfies

$$\text{Gengap}_{\Delta_A} \le B_{\text{samp}}^{\text{van}}(K) := \frac{2\tilde{\mu}_{\mathcal{W}} \eta M_{\text{van}}}{n}\, a_T^{\text{van}}\, \frac{e^{T\zeta_{\text{van}}} - 1}{\zeta_{\text{van}}}.$$

Here $\tilde{\mu}_{\mathcal{W}}$ and $\tilde{\nu}_{\mathcal{W}}$ are uniform Lipschitz and gradient-Lipschitz constants of the adversarial loss. The moment factor c_T is defined in Equation (1)*, and the expectation includes both the training sample and Adam's sampled indices. Throughout the paper, $(e^{T\zeta} - 1)/\zeta$ is defined by continuous extension at $\zeta = 0$, with value T.*

For the spectral threat set $\|g(A') - g(A)\|_2 \le \varepsilon_A$, the triangle inequality gives $\widetilde{C}_g \le C_g + \varepsilon_A$. The attack budget therefore enters the generalization bound through the worst-case gradient, Lipschitz, and smoothness constants. At zero budget, the result specializes to the clean-training sample-replacement bound.

Remark 2 (Spectral Depth Scale of the Stability Bound). *A selected propagation factor in Adam's bound gives a depth scale. For $n > e$ and $\tilde{\rho} > 1$, retain the $\tilde{\rho}^{2K}$ component of the smoothness majorant and hold the remaining factors fixed. The resulting leading exponential term has the form $\exp(C\tilde{\rho}^{2K})$, where $C > 0$ is independent of depth and sample size in this comparison. Balancing this term against the sample-size factor $1/n$ gives*

$$C\tilde{\rho}^{2K} = \log n \quad \Longleftrightarrow \quad K = \frac{\log\log n - \log C}{2\log\tilde{\rho}} = \mathcal{O}\left(\frac{\log\log n}{\log\tilde{\rho}}\right).$$

All logarithms are natural. This balance identifies the spectral-component depth scale. The complete bound also retains the depth dependence of M_{van}, the loss-regularity constants, and their contributions to ζ_{van}. Appendix F.1 gives the full expression. We use stability-bound collapse *to describe rapid depth-driven growth of this derived upper bound.*

4.2 Generalization Bounds for ResGCN under Adam

Residual connections change this propagation structure by introducing paths of different lengths. Expanding the full Jacobian reveals how these paths contribute to the gradient bound and, through it, to Adam's stability recursion.

4.2.1 The 2^K-Path Decomposition of ResGCN

The Jacobian of the residual block composition is a product of K sums of identity and branch Jacobians. Expanding this derivative product yields 2^K terms at the evaluated forward state.

Definition 1 (Binary Path Set and Path Jacobian (Veit et al., 2016)). *For $\mathcal{P}_K = \{0,1\}^K$, a path $\mathbf{p}$ has effective length $r(\mathbf{p}) = \sum_{\ell=1}^{K} p_\ell$. At the evaluated forward state, define the branch derivative operator by $\mathcal{J}_\ell^{(1)}[V] = \sigma'(Z^\ell) \odot (g(A)VW^{\ell-1})$ and let $\mathcal{J}_\ell^{(0)} = I$. The path Jacobian is the ordered composition $J_{\mathbf{p}} = \mathcal{J}_K^{(p_K)} \circ \cdots \circ \mathcal{J}_1^{(p_1)}$. Then $\frac{\partial H^K}{\partial H^0} = \sum_{\mathbf{p}\in\mathcal{P}_K} J_{\mathbf{p}}$.*

Remark 3. *As established in Lemma 2, Definition 1 yields $\|\partial H^K/\partial H^0\|_2 \le \prod_{j=1}^{K}(1+\rho_j)$, where $\rho_j := \mu_\sigma \|g(A)\|_2 \|W^{j-1}\|_2$.*

Appendix F.4 also reports a path-averaged diagnostic. The uniform generalization bound below uses the full loss smoothness and gradient bound.

4.2.2 Generalization Bounds under Path Decomposition

Substituting the residual-path gradient bound into the Adam recursion gives the following result. It shares the vanilla model's optimizer-dependent form, with architectural dependence carried by the gradient and loss-regularity constants.

Theorem 3 (Generalization Bound under Structural Attacks for ResGCN). *Under the same structural threat model and Adam training protocol as in Theorem 2, consider a K-layer ResGCN satisfying Assumptions 1–5. The residual-path decomposition in Definition 1 yields the uniform adversarial gradient bound*

$$M_{\mathrm{res}} = K M_f \sqrt{\tilde{d}_{\max}^{K}} C_x \mu_\sigma^2 \widetilde{C}_g (1 + \mu_\sigma \widetilde{C}_g C_W)^{K-1}. \tag{4}$$

Using this bound, set $a_T^{\mathrm{res}} := \varepsilon_0^{-1} + 2M_{\mathrm{res}} c_T \varepsilon_0^{-2}$. Let $\tilde{\nu}_{\mathcal{W}}^{\mathrm{res}}$ be a uniform gradient-Lipschitz constant of the ResGCN adversarial loss. The corresponding Adam expansion coefficient is $\zeta_{\mathrm{res}} = \eta\tilde{\nu}_{\mathcal{W}}^{\mathrm{res}} a_T^{\mathrm{res}}$. Then the absolute expected adversarial generalization gap of $\mathcal{W}_T$ satisfies

$$\mathrm{Gengap}_{\Delta_A} \le B_{\mathrm{samp}}^{\mathrm{res}}(K) := \frac{2\tilde{\mu}_{\mathcal{W}}\eta M_{\mathrm{res}}}{n}\, a_T^{\mathrm{res}} \frac{e^{T\zeta_{\mathrm{res}}} - 1}{\zeta_{\mathrm{res}}}.$$

Here $\tilde{\mu}_{\mathcal{W}}$ is a uniform Lipschitz constant of the ResGCN adversarial loss. The moment factor c_T is defined in Equation (1)*, and the expectation includes both the training sample and Adam's sampled indices.*

The residual architecture changes M_{res} in the common stability formula, together with the applicable loss Lipschitz and smoothness constants. When $T\zeta_{\mathrm{res}}$ remains bounded, $(e^{T\zeta_{\mathrm{res}}} - 1)/\zeta_{\mathrm{res}} = \mathcal{O}(T)$. Table 4 records this simplification of the temporal factor and retains the dependence of the other coefficients.

Remark 4 (Uniform Weight Bound). *The factor $K(1+\mu_\sigma \widetilde{C}_g C_W)^{K-1}$ sums the K parameter-block estimates, each with $K-1$ remaining propagation factors. It uses the common upper bound C_W while allowing layer norms to differ. Theorem 3 combines this explicit architectural factor with the full adversarial-loss smoothness $\tilde{\nu}_{\mathcal{W}}^{\mathrm{res}}$.*

Initial residual connections provide a related family of short paths. Corollary 1 in Appendix D gives the GCNII-type Jacobian bound and compares it with the residual path product for $\mu_\sigma \leq 1$.

4.3 Residual Paths and Local Directional Curvature

Theorems 2 and 3 describe how propagation paths enter the stability bound. Residual paths also motivate a local geometric question: when does a parameter direction retain positive curvature? At fixed parameters, the following theorem gives sufficient conditions for positive directional curvature by separating the Gauss–Newton contribution from the remaining Hessian term.

Theorem 4 (Local Directional Curvature). *Fix the network parameters $\mathcal{W}$, and let $J_{\mathcal{W}}(z)$ denote the full output Jacobian of a ResGCN. Suppose the expected loss Hessian at $\mathcal{W}$ decomposes as $\mathcal{H} = G + R$, where G is the Gauss–Newton component, R is the remaining Hessian term, and $G = \mathbb{E}_z\Big[J_{\mathcal{W}}(z)^\top \nabla_f^2 \mathcal{L}(f,z) J_{\mathcal{W}}(z)\Big]$. Suppose $\nabla_f^2 \mathcal{L}(f,z) \succeq c_L I$ on the evaluated outputs for some $c_L > 0$, and let v be a fixed unit parameter direction satisfying $\mathbb{E}_z\|J_{\mathcal{W}}(z)v\|_2^2 \geq c_J^2 > 0$. Then the Gauss–Newton component has positive curvature along v, with $v^\top G v \geq c_L c_J^2, \lambda_{\max}(G) \geq c_L c_J^2$. If the remaining Hessian term also satisfies $\|R\|_2 \leq \epsilon_R < c_L c_J^2$, this direction retains positive curvature for the full expected loss $v^\top \mathcal{H} v \geq c_L c_J^2 - \epsilon_R > 0$.*

The ResGCN Jacobian includes contributions from its residual paths. Positive curvature follows when the stated Jacobian and Hessian-remainder conditions hold at the evaluated parameters. The same argument applies to other architectures under these conditions. The proof is given in Appendix D.10.

Remark 5 (Local Curvature and Adam-Moment Diagnostics). *This local curvature result motivates examining geometry alongside Adam's training dynamics. Adam's second moments average squared stochastic gradients over time, so their values also depend on the training history. We therefore report curvature estimates and recorded moment statistics as complementary measurements.*

4.4 The Role of Graph Topology: Homophily versus Heterophily

Per-layer amplification couples graph propagation with weight magnitude. For the balanced seven-community SBM in Appendix G.8, the community eigenvalue of the normalized expected adjacency depends explicitly on homophily and expected degree. Appendix F.6 derives the spectrum, and Corollary 2 gives the corresponding spectral depth scale.

For a real graph, let r_{nt} denote the largest absolute eigenvalue of $g(A)$ after removing the component-wise trivial eigenvalues at 1. These eigenvalues correspond to the degree-weighted constant modes of the normalized filter. We combine this measured radius with the reported maximum weight norm $\widehat{C}_W$. With $\mu_\sigma = 1$, the spectral depth estimate is

$$\widehat{K}^* = \frac{\log\log n}{2\log(r_{\mathrm{nt}}\widehat{C}_W)}, \qquad r_{\mathrm{nt}}\widehat{C}_W > 1.$$

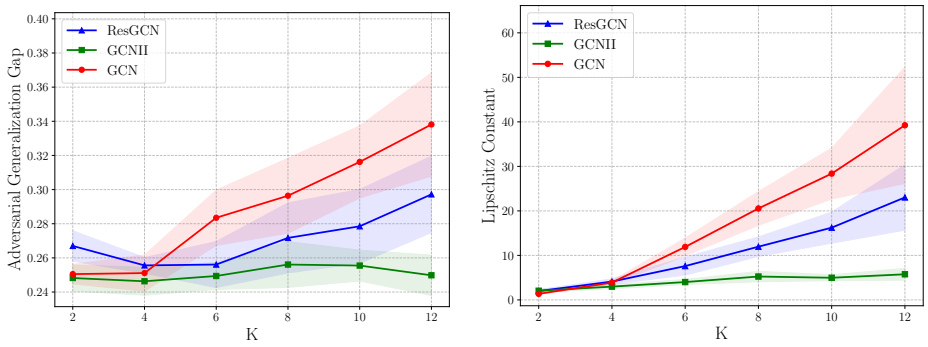
This calculation isolates the spectral component of the depth dependence. The uniform theorems retain the full filter norm and neighborhood factors. Table 8 in Appendix G.5 compares the spectral depth estimate with visually estimated depths of curve separation.

5 Experiments

We examine four empirical questions: **RQ1:** How do cross-entropy and accuracy gaps vary with depth across architectures? **RQ2:** How are local curvature and Adam moment statistics related to depth? **RQ3:** How do architecture and optimizer choice affect the observed gaps? **RQ4:** How do the layerwise norm dynamics and measured graph spectra relate to the spectral depth estimate derived in Section 4.4?

Table 1: Cross-entropy (CE) gaps under attack from the depth-sweep experiment on Cora and CiteSeer, reported as means from ten runs with distinct random seeds. "Increase" denotes the mean gap at $K = 10$ minus that at $K = 2$.

Dataset	Architecture	$K = 2$	$K = 10$	Increase
Cora	Vanilla GCN	0.832	3.947	3.115
Cora	ResGCN	0.918	2.918	2.000
CiteSeer	Vanilla GCN	1.304	9.012	7.708
CiteSeer	ResGCN	1.270	5.429	4.159



(a) Accuracy gap vs. K (b) Empirical Lipschitz proxy vs. K

Figure 1: **Depth diagnostics on Cora.** The left panel reports the accuracy gap (axis label: "Adversarial Generalization Gap") on the attacked graph, and the right panel reports the empirical Lipschitz proxy. Both quantities show architecture-dependent trends with depth.

Setup. We evaluate vanilla GCN (Kipf & Welling, 2017) and a residual GCN (ResGCN) on five benchmarks: Cora, CiteSeer, Actor, Wisconsin, and ogbn-arxiv. Cora and CiteSeer provide the main CE-gap depth comparisons, Actor and Wisconsin the heterophilic perturbation sweeps, and ogbn-arxiv a larger-graph depth comparison. GCNII (Chen et al., 2020) is additionally evaluated on the four smaller benchmarks. The empirical cross-entropy (CE) gap is $\widehat{G}^{\mathrm{atk}}_{\mathrm{CE}} = \frac{1}{R}\sum_{r=1}^{R}\left|\mathcal{L}^{\mathrm{atk}}_{\mathrm{train},r} - \mathcal{L}^{\mathrm{atk}}_{\mathrm{test},r}\right|$, where R is the number of independent runs, and $\mathcal{L}^{\mathrm{atk}}_{\mathrm{train},r}$ and $\mathcal{L}^{\mathrm{atk}}_{\mathrm{test},r}$ are mean cross-entropies on the respective splits. Depth sweeps apply DICE and surrogate-crafted feature PGD once before full-batch training, then evaluate both splits on the same attacked graph. The population bounds in Section 3.2 concern a distinct setting with conditionally independent labeled examples, sampled Adam updates, and a maximized adversarial loss. Appendix G gives the metrics and protocols.

Depth dependence of empirical gaps (RQ1). From $K = 2$ to $K = 10$, CE gaps increase by 3.115 and 2.000 for vanilla GCN and ResGCN on Cora, and by 7.708 and 4.159 on CiteSeer (Table 1). ResGCN has the smaller increase on both datasets. Figure 1 shows the accuracy-gap and empirical Lipschitz proxy curves.

Local curvature and Adam moments (RQ2). On Cora at $K = 2, \ldots, 7$, local curvature estimated using Hessian-vector products at fixed PGD perturbations has fitted log slopes of 0.774 and 0.613 per layer for vanilla GCN and ResGCN, respectively (Table 6). The corresponding slopes of the realized norm proxy, constructed from the raw adjacency norm and recorded layer-weight norms, are 6.961 and 7.229. Separately, Table 2 reports mean per-run minimum bias-corrected second moments of 2.95×10^{-18} and 1.33×10^{-6} at $K = 7$ for vanilla GCN and ResGCN, respectively. Appendix G.1 specifies these local diagnostics and their aggregation.

Architecture and optimizer comparison (RQ3). Table 2 compares vanilla GCN and ResGCN under Adam and SGD on Cora, using the same ten seed values and DICE/feature-PGD perturbations constructed once per run before training. Adam and SGD use learning rates 10^{-3} and 0.05, respectively, with full settings and results in Table 7. From $K = 2$ to $K = 7$, CE gaps increase by

Table 2: Final-epoch means from ten runs with distinct random seeds for GCN and ResGCN trained with Adam and SGD on Cora. $\Delta_{2\to7}$ denotes the CE-gap increase from $K = 2$ to $K = 7$. Other quantities are evaluated at $K = 7$. Robust accuracy is test accuracy on the attacked graph. $\min \hat{v}$ is the minimum bias-corrected second moment over the coordinates specified in Appendix G.1. SGD has no second-moment accumulator.

Loss and accuracy summaries

Architecture	Optimizer	CE gap ($K = 2$)	CE gap ($K = 5$)	CE gap ($K = 7$)	$\Delta_{2\to7}$	Robust accuracy
GCN	Adam	0.750	1.468	2.368	1.618	0.729
ResGCN	Adam	0.846	1.074	1.208	0.362	0.732
GCN	SGD	0.255	0.409	0.461	0.206	0.593
ResGCN	SGD	0.754	0.761	0.860	0.106	0.752

Recorded diagnostics at $K = 7$

Architecture	Optimizer	$\|\mathbf{g}_t\|_F$	$\min \hat{v}$	Update norm
GCN	Adam	1.84×10^{-3}	2.95×10^{-18}	2.15×10^{-2}
ResGCN	Adam	6.44×10^{-3}	1.33×10^{-6}	2.20×10^{-2}
GCN	SGD	4.33	–	0.216
ResGCN	SGD	8.54×10^{-2}	–	4.27×10^{-3}

1.618 for GCN with Adam, 0.362 for ResGCN with Adam, 0.206 for GCN with SGD, and 0.106 for ResGCN with SGD. In these runs, ResGCN has the smaller CE-gap increase under each optimizer. The optimizer settings differ, and Appendix F.5 compares the dependencies of their selected stability bounds. At $K = 7$, the SGD-trained GCN has a mean gradient norm of 4.33 ± 3.62 and a mean robust accuracy of 0.593 ± 0.134, reported as mean $\pm$ standard deviation over 10 seeds. Among the four configurations at $K = 7$, this model has both the smallest CE gap and the lowest robust accuracy.

Weight-norm dynamics and spectral diagnostics (RQ4). In the Cora and CiteSeer runs, the layerwise norm products are more dispersed in vanilla GCN than in ResGCN (Figure 8, Appendix G.5). Table 8 reports spectral depth estimates of 1–2 across the four datasets, compared with visually assessed curve separation at depths 3–6. On Cora, the corresponding values are 2 and 4–6, respectively.

6 Conclusion

Our analysis connects graph propagation to both moment histories in Adam and yields architecture-dependent stability bounds for the vanilla and residual GCNs studied here. A worst-case scalar construction attains exponential loss-smoothness growth, while residual-path analysis identifies architecture-dependent factors and a linear temporal factor under controlled expansion. Auxiliary results give sufficient conditions for local directional curvature and a homophily-dependent spectral scale for the seven-community expected graph. Experiments show smaller CE-gap increases for ResGCN than vanilla GCN on Cora and CiteSeer and report complementary accuracy-gap and robust-accuracy comparisons on Actor, Wisconsin, and ogbn-arxiv. Future work will study fixed-split transductive training and residual-path interactions.

Limitations

The explicit architecture-dependent bounds assume scalar readouts, the stated regularity conditions, conditionally independent labeled examples, and sampled Adam updates. Appendix C identifies the further estimates needed for multiclass outputs. The main depth experiments use multiclass ReLU models, fixed splits, full-batch updates, and attacks constructed before training. The exponential smoothness rate is attained by a worst-case construction, while its prevalence in trained networks remains unverified. Local curvature measurements do not establish the uniform smoothness constants required by the theory. The spectral depth scale describes one contribution to the stability bound and does not establish when predictive accuracy deteriorates.

AI USE STATEMENT

Generative AI tools were used as editorial assistants for language polishing, LaTeX editing, and checks of notation, cross-references, and the presentation and internal consistency of mathematical arguments. They were not used to originate the research questions or core contributions, or to generate experimental code, data, or results. All theoretical claims, proofs, experiments, and reported numerical values were reviewed and verified by the authors, who take full responsibility for the content of the manuscript.

REPRODUCIBILITY STATEMENT

Section 3.3 states the assumptions used in the analysis. Appendices D and E provide the generalization and auxiliary input-stability proofs. Appendix G describes the experimental protocols, metric definitions, and numerical diagnostics. The code link in the abstract provides the accompanying implementation.

REFERENCES

Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Aleksandar Bojchevski, Johannes Gasteiger, and Stephan Günnemann. Efficient robustness certificates for discrete data: Sparsity-aware randomized smoothing for graphs, images and more. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1003–1013. PMLR, 2020.

Olivier Bousquet and André Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

Chen Cai and Yusu Wang. A note on over-smoothing for graph neural networks. In *Proceedings of the 37th International Conference on Machine Learning (ICML) Workshop on Graph Representation Learning*, 2020. PMLR.

Chang Cao, Han Li, Yulong Wang, Rui Wu, and Hong Chen. Adversarial training for graph convolutional networks: stability and generalization analysis. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 4797–4805, 2025.

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 1725–1735, 2020.

Francesco Di Giovanni, Lorenzo Giusti, Federico Barbero, Giulia Luise, Pietro Liò, and Michael M. Bronstein. On over-squashing in message passing neural networks: The impact of width, depth, and topology. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 7865–7885. PMLR, 2023.

Wenqi Fan, Han Xu, Wei Jin, Xiaorui Liu, Xianfeng Tang, Suhang Wang, Qing Li, Jiliang Tang, Jianping Wang, and Charu C. Aggarwal. Jointly attacking graph neural network and its explanations. In *2023 IEEE 39th International Conference on Data Engineering*, pp. 654–667. IEEE, 2023.

Farzan Farnia and Asuman Özdağlar. Train simultaneously, generalize better: Stability of gradient-based minimax learners. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 3174–3185. PMLR, 2021.

Ben Finkelshtein, Chaim Baskin, Evgenii Zheltonozhskii, and Uri Alon. Single-node attacks for fooling graph neural networks. *Neurocomputing*, 513:1–12, 2022.

Simon Geisler, Tobias Schmidt, Hakan Sirin, Daniel Zügner, Aleksandar Bojchevski, and Stephan Günnemann. Robustness of graph neural networks at scale. In *Advances in Neural Information Processing Systems*, volume 34, pp. 7637–7649, 2021.

Moritz Hardt, Benjamin Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pp. 1225–1234. PMLR, 2016.

Ali Hariri, Álvaro Arroyo, Alessio Gravina, Moshe Eliasof, Carola-Bibiane Schönlieb, Davide Bacciu, Kamyar Azizzadenesheli, Xiaowen Dong, and Pierre Vandergheynst. Return of chebnet: Understanding and improving an overlooked gnn on long range tasks. In *Advances in Neural Information Processing Systems*, 2025.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2013. ISBN 978-0521548236.

Hongwei Jin, Zhan Shi, Venkata Jaya Shankar Ashish Peruri, and Xinhua Zhang. Certified robustness of graph convolution networks for graph classification under topological attacks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 8463–8474, 2020.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Jiaqi Ma, Shuangrui Ding, and Qiaozhu Mei. Towards more practical adversarial attacks on graph neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 4756–4766, 2020.

Hongbin Pei, Bingzhe Wei, Kevin Chen Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. In *International Conference on Learning Representations*, 2020.

Jan Schuchardt, Tom Wollschläger, Aleksandar Bojchevski, and Stephan Günnemann. Localized randomized smoothing for collective robustness certification. In *The Eleventh International Conference on Learning Representations*, 2023.

Tsubasa Takahashi. Indirect adversarial attacks via poisoning neighbors for graph convolutional networks. In *2019 IEEE International Conference on Big Data*, pp. 1395–1400. IEEE, 2019.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. In *International Conference on Learning Representations*, 2022.

Andreas Veit, Michael J. Wilber, and Serge J. Belongie. Residual networks behave like ensembles of relatively shallow networks. In *Advances in Neural Information Processing Systems*, pp. 550–558, 2016.

Saurabh Verma and Zhi Li Zhang. Stability and generalization of graph convolutional neural networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1539–1548, 2019.

Zhiyang Wang, Juan Cervino, and Alejandro Ribeiro. A manifold perspective on the statistical generalization of graph neural networks. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 62467–62504, 2025.

Jiancong Xiao, Yanbo Fan, Ruoyu Sun, Jue Wang, and Zhi Quan Luo. Stability analysis and generalization bounds of adversarial training. In *Advances in Neural Information Processing Systems*, volume 35, pp. 15446–15459, 2022.

Yue Xing, Qifan Song, and Guang Cheng. On the algorithmic stability of adversarial training. In *Advances in Neural Information Processing Systems*, volume 34, pp. 26523–26535, 2021.

Kaidi Xu, Hongge Chen, Sijia Liu, Pin Yu Chen, Tsui Wei Weng, Mingyi Hong, and Xue Lin. Topology attack and defense for graph neural networks: An optimization perspective. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 3961–3967. AAAI Press, 2019.

Xianchen Zhou and Hongxia Wang. The generalization error of graph convolutional networks may enlarge with more layers. *Neurocomputing*, 424:97–106, 2021.