Wave-driven Graph Neural Networks with Energy Dynamics for Over-smoothing Mitigation

Peihan Wu[†], **Hongda Qi**[†], **Sirong Huang**, **Dongdong An**, **Jie Lian** and **Qin Zhao**[*]
Shanghai Engineering Research Center of Intelligent Education and Big Data,
Shanghai Normal University, Shanghai, China
1000549400@smail.shnu.edu.cn, hongda_qi@hotmail.com, sironghuang1007@163.com, {andongdong, lianjie, q_zhao}@shnu.edu.cn

Abstract

Over-smoothing is a persistent challenge in Graph Neural Networks (GNNs), where node embeddings become indistinguishable as network depth increases, fundamentally limiting their effectiveness on tasks requiring fine-grained distinctions. This issue arises from the reliance on diffusion-based propagation mechanisms, which suppress high-frequency information essential for preserving feature diversity. To mitigate this, we propose a wave-driven GNN framework that redefines feature propagation through the wave equation. Unlike diffusion, the wave equation incorporates second-order dynamics, balancing smoothing and oscillatory behavior to retain high-frequency components while ensuring effective information flow. To enhance the stability and convergence of wave equation discretization on graphs, an energy-based mechanism inspired by kinetic and potential energy dynamics is introduced, balancing temporal evolution and structural alignment to stabilize propagation. Extensive experiments on benchmark datasets, including Cora, Citeseer, and PubMed, as well as real-world graphs, demonstrate that the proposed framework achieves state-of-the-art performance, effectively mitigating over-smoothing and enabling deeper, more expressive architectures. The code is available at https://github.com/rene0329/EWGNN/.

1 Introduction

GNNs have emerged as a transformative framework for learning from graph-structured data, with applications spanning social networks, molecular property prediction, and recommendation systems. At the core of GNNs lies their ability to iteratively aggregate and propagate information across graph edges, effectively capturing both local and global structural relationships. While this propagation mechanism excels in shallow networks, extending GNNs to deeper architectures introduces significant challenges, particularly the pervasive issue of over-smoothing.

Over-smoothing occurs when node features become increasingly homogeneous as the number of layers grows, diminishing their discriminative ability. This phenomenon arises from the propagation mechanisms commonly used in GNNs, which are primarily based on the principles of diffusion processes to spread information between connected nodes. Mathematically, diffusion-based GNNs can be viewed as discrete approximations of the diffusion equation:

$$\frac{\partial u(x,t)}{\partial t} = -Lu(x,t) \tag{1}$$

where $u(x,t)$ represents the graph signal defined at position x and time t, and L is the graph Laplacian operator.

While diffusion effectively aggregates local information, it introduces a spectral bias rooted in the properties of the graph Laplacian. The graph Laplacian's eigenvectors represent different frequency modes, where lower eigenvalues correspond to smooth, global patterns, and higher eigenvalues capture sharp, local variations. During propagation, the diffusion process naturally emphasizes low-frequency components, reducing the influence of high-frequency components, which are associated with larger eigenvalues. This imbalance causes node features to become more similar across layers, leading to over-smoothing and weakening the model's ability to capture important distinctions. This is particularly problematic in tasks where preserving high-frequency information is crucial for distinguishing between nodes.

Efforts to address over-smoothing have yielded notable advancements, including architectural innovations such as skip connections and residual designs to retain node-level diversity [Xu *et al.*, 2018b; Chen *et al.*, 2020], and propagation refinements that dynamically adjust information flow [Gasteiger *et al.*, 2018; Rong *et al.*, 2020]. However, these methods often fail to fully overcome the spectral bias inherent in diffusion, which fundamentally limits their ability to preserve high-frequency components in deeper architectures. The monotonic smoothing behavior inherent in diffusion remains a bottleneck for achieving expressive and robust feature representations [Li *et al.*, 2018].

To address these limitations, we propose a novel propagation framework for GNNs inspired by the *wave equation*, a second-order partial differential equation (PDE) that introduces oscillatory dynamics and inertia into the propagation

[*]Corresponding Author is Qin Zhao.

Supplementary materials are available on GitHub: https://github.com/rene0329/EWGNN

process. Unlike diffusion, the wave equation introduces a balance between smoothing and oscillation. This balance has the potential to preserve high-frequency components more effectively, thereby maintaining meaningful distinctions between node features:

$$\frac{\partial^2 u(x,t)}{\partial t^2} + \gamma \frac{\partial u(x,t)}{\partial t} = c^2 L u(x,t) \tag{2}$$

where γ and c control the damping and propagation speed, respectively. Unlike diffusion, which suppresses high-frequency components and leads to uniform embeddings, the wave equation introduces second-order dynamics that balance smoothing and oscillation. The second-order term $\frac{\partial^2 u(x,t)}{\partial t^2}$ induces oscillatory behavior, countering the homogenizing effect of diffusion and preserving high-frequency components. However, uncontrolled oscillations may lead to instability and hinder convergence, especially in deeper architectures. To mitigate such issues, the damping term $\gamma \frac{\partial u(x,t)}{\partial t}$ plays a crucial role in moderating the dynamics and preventing excessive oscillations. Meanwhile, the propagation term $c^2 L u(x,t)$ ensures the smooth and meaningful flow of information across the graph. Together, these dynamics allow the wave equation to retain high-frequency information while achieving stability, effectively preserving the discriminative power of node features even in deeper networks. By leveraging this mechanism, the wave equation offers a principled approach to mitigating over-smoothing in GNNs.

While the wave equation retains high-frequency components, its second-order dynamics can lead to uncontrolled oscillations, disrupting network convergence and hindering effective feature representation. To address this, we propose a regulation mechanism based on graph-based kinetic and potential energy. Kinetic energy reflects the temporal evolution of node embeddings across layers, while potential energy captures the structural alignment of features between connected nodes. By balancing these energy terms, the propagation achieves stability, controlling excessive oscillations and maintaining maintaining greater capacity for feature distinction. This stable and regulated propagation mitigates over-smoothing, preserving high-frequency information and ensuring discriminative power even in deeper architectures.

In summary, this paper presents a physics-inspired framework to address over-smoothing in GNNs, with the following contributions:

1. Wave-based Propagation Mechanism: A novel propagation mechanism is proposed, grounded in the wave equation. This approach leverages second-order dynamics to preserve high-frequency information and effectively mitigate over-smoothing.
2. Energy-Based Regulation: Kinetic and potential energy terms are proposed within graph propagation, along with an energy balance mechanism to guide the learning process and ensure stability.
3. Unified Theoretical and Empirical Framework: A comprehensive theoretical foundation is established, linking diffusion- and wave-based propagation. Extensive experimental validation across diverse graph tasks demonstrates the efficacy of the proposed approach.

2 Understanding Over-smoothing

Over-smoothing in GNNs arises when node embeddings progressively lose their discriminative power as network depth increases [Li *et al.*, 2018; Xu *et al.*, 2018b]. This phenomenon undermines tasks such as node classification and clustering, where unique node characteristics are essential.

To formalize our discussion, we adopt the following notation. Let $G = (V, E, X)$ represent a graph, where V is the set of nodes ($|V| = n$), E is the set of edges, and $X \in \mathbb{R}^{n \times d}$ denotes the node feature matrix. We define the normalized adjacency matrix with self-loops:

$$\hat{A} = D^{-1/2}(A + I)D^{-1/2}, \tag{3}$$

where D is the degree matrix and I the identity matrix.

2.1 How over-smoothing Manifests and Why It Happens

Over-smoothing manifests through the suppression of high-frequency components during the propagation process in GNNs, which causes node embeddings to converge towards uniform representations as depth increases. This behavior can be analyzed through the eigen decomposition of the normalized adjacency matrix $\hat{A}$:

$$\hat{A} = U \Lambda U^\top, \quad \Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n), \quad |\lambda_i| \leq 1, \tag{4}$$

where U represents the eigenvector matrix that defines the spectral basis, and Λ is a diagonal matrix containing the eigenvalues of $\hat{A}$. Each eigenvalue λ_i reflects the contribution of the corresponding eigenvector to the graph's structure. During propagation, each step can be viewed as projecting the node embeddings onto this eigenbasis and scaling them by the eigenvalues. As a result, the repeated multiplication of embeddings by $\hat{A}$ amplifies the influence of the dominant eigenvalue $\lambda_1 = 1$, which corresponds to low-frequency components, while progressively suppressing smaller eigenvalues associated with high-frequency components. This imbalance drives the embeddings toward uniformity, erasing distinctions needed for tasks like node classification.

From a theoretical perspective, the propagation in GNNs can be analyzed through spectral decomposition. Consider the normalized adjacency matrix:

$$\hat{A} = U \Lambda U^\top, \quad \Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n), \quad |\lambda_i| \leq 1, \tag{5}$$

where U is the eigenvector matrix and Λ is the diagonal matrix of eigenvalues. To simplify the spectral analysis, we omit the activation function σ. The simplified propagation rule is:

$$H^{(l+1)} = \hat{A} H^{(l)} W^{(l)}. \tag{6}$$

After k propagation steps, the embeddings $H^{(l)}$ evolve as:

$$H^{(l+k)} = \hat{A}^k H^{(l)} = U \Lambda^k U^\top H^{(l)}, \tag{7}$$

this formulation reveals that each propagation step scales the embeddings' spectral components by the eigenvalues raised to the k-th power.

Over successive steps, eigenvalues $|\lambda_i| < 1$ decay exponentially, leaving only the dominant component corresponding to $\lambda_1 = 1$:

$$\lim_{k \to \infty} H^{(l+k)} = U_1 U_1^\top H^{(l)}, \tag{8}$$

this convergence to uniformity underscores the spectral bias in diffusion-based propagation, which emphasizes low-frequency components at the cost of high-frequency information critical for preserving node-level distinctions.

2.2 Motivations for Wave Equations

Over-smoothing limits the performance of deep GNNs by suppressing high-frequency node features as network depth increases. To address this, the wave equation offers a compelling alternative by introducing second-order dynamics that balance smoothing and oscillation.

In its continuous form (Eq. (2)), the wave equation incorporates second-order temporal derivatives to preserve high-frequency components while ensuring effective feature propagation. When discretized for graph-structured data and interpreted in GNN layer propagation, the wave equation introduces feedback from previous embeddings, helping retain high-frequency information needed for feature diversity:

$$H^{(l+1)} = (2I - \gamma I - c^2 L)H^{(l)} - (I - \gamma I)H^{(l-1)}, \quad (9)$$

where γ controls damping, c determines the wave propagation speed, and L is the graph Laplacian. This feedback mechanism mitigates the progressive loss of high-frequency components observed in diffusion-based approaches.

Despite its advantages, the second-order dynamics of the wave equation introduce challenges. Unregulated oscillations may arise in deeper architectures, where small deviations in embeddings amplify over layers. Additionally, the balance between smoothing and oscillation is sensitive to hyperparameters γ and c, which affect stability and convergence.

To address the challenges of unregulated oscillations and the balance between smoothing and feature preservation, we draw inspiration from physical systems. In these systems, the interplay between kinetic and potential energy governs stability and evolution. Similarly, in GNNs, these concepts can be employed to regulate propagation dynamics and mitigate instabilities from second-order wave equations.

Specifically, kinetic energy quantifies the temporal evolution of node embeddings across layers, capturing feature changes over successive propagation steps. This helps prevent uncontrolled oscillations by limiting abrupt changes in embeddings. Potential energy, on the other hand, measures the structural alignment of features between connected nodes, promoting coherence within the graph. By balancing these two energy terms, we can regulate oscillatory dynamics from the wave equation while preserving high-frequency information for fine-grained distinctions.

Thus, we propose a wave equation-based GNN approach that incorporates an energy balance mechanism between kinetic and potential energy to stabilize the propagation process while preserving the expressiveness of node features.

3 Wave-driven GNNs with Energy Dynamics

Our framework addresses the over-smoothing problem in deep GNNs through an integrated approach, as illustrated in Figure 1: (1) Wave-driven Propagation: Introducing second-order dynamics from the wave equation to balance smoothing and oscillations, preserving high-frequency components and enhancing feature propagation. (2) Non-local Kernels: Incorporating a non-local kernel to dynamically adjust node influence beyond local neighborhoods, capturing global structural information and mitigating over-smoothing. (3) Energy-based Regulation: Balancing kinetic and potential energy during propagation to stabilize oscillations and maintain feature diversity in deep architectures. This integrated framework enables deeper GNNs to retain discriminative power and effectively combat over-smoothing.

3.1 Modeling with the Wave Equation

To address the challenge of over-smoothing in GNNs, we propose a propagation framework inspired by the wave equation, as introduced in Eq. (2). This framework leverages second-order dynamics to balance smoothing and oscillatory behavior, overcoming the limitations of traditional diffusion-based methods that inherently suppress high-frequency components and lead to uniform embeddings. By preserving discriminative features across layers, the wave equation provides a principled approach to retaining feature diversity, which is essential for tasks requiring fine-grained distinctions.

The advantages of wave-based propagation can be analyzed by comparing its behavior to diffusion-based propagation in terms of eigenmode preservation. Specifically, the evolution of eigenmode amplitudes shows how wave dynamics maintain high-frequency components more effectively than diffusion, as formalized in the following theorem:

Theorem 1 (Wave Equation Mitigates Over-smoothing via High-Frequency Preservation)**.** *Let $\phi_i(x)$ be an eigenmode of the Laplacian operator ∇^2 with eigenvalue $\lambda_i \geq 0$. The mode amplitude $a_i(t)$ evolves as:*

$$\textit{Diffusion:} \quad a_i(t) = a_i(0)\exp(-D\lambda_i t),$$

$$\textit{Wave:} \quad a_i(t) = a_i(0)\exp\left(-\frac{\gamma t}{2}\right)\cos\left(t\sqrt{c^2\lambda_i - \frac{\gamma^2}{4}}\right),$$

where $D > 0$ is the diffusion coefficient, $c > 0$ is the wave speed, and $\gamma \geq 0$ is the damping coefficient. Notably, for large λ_i, diffusion dampens high-frequency modes to zero, while the wave equation maintains oscillatory behavior.

To enhance the flexibility of this propagation mechanism, the graph Laplacian L is extended with a non-local propagation kernel $G(x, y)$. The continuous form of the wave equation with this kernel is given by:

$$\frac{\partial^2 u(x,t)}{\partial t^2} + \gamma\frac{\partial u(x,t)}{\partial t} = c^2\int_\Omega G(x,y)(u(y,t) - u(x,t))\,dy, \quad (10)$$

where Ω denotes the domain of the graph. The kernel $G(x, y)$ generalizes the graph Laplacian to incorporate non-local interactions, allowing propagation beyond immediate neighbors.

For graphs, we discretize the wave equation by translating the continuous time variable t to discrete layer indices l and applying finite difference approximations to the time derivatives. This transformation leads to the following update rule:

$$H^{(l+1)} = (1-\alpha)H^{(l)} + \alpha\left(\beta\left(H^{(l)} - H^{(l-1)}\right) + \eta L[H^{(l)}]\right), \quad (11)$$

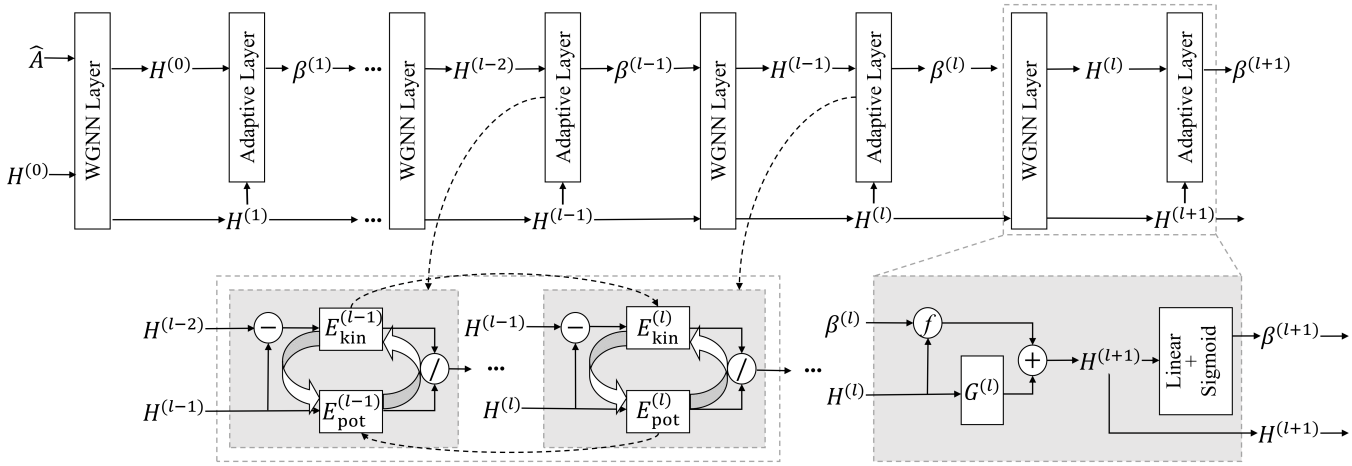


Figure 1: The Overall Architecture of EWGNN

where $H^{(l)}$ represents the node embeddings at layer l, α and η are hyperparameters derived from the physical coefficients γ and c, and β is an additional hyperparameter controlling the influence of the previous layer.

To establish the advantages of wave-based propagation in the discrete setting, we present the following theorem:

Theorem 2 (Wave-Based Update Mitigates Over-Smoothing)**.** *Let $G = (V, E)$ be a connected undirected graph with normalized adjacency operator $\hat{A}$ having eigenvalues $0 \leq \lambda_i \leq 1$. The wave-based update preserves high-frequency modes and mitigates over-smoothing provided that the update parameters α, β, and η satisfy:*

$$(1 - \alpha + \alpha\beta + \alpha\eta(\lambda_i - 1))^2 < 4\alpha\beta,$$

for all high-frequency modes $\lambda_i < 1$.

Next, we discretize the non-local kernel $G(x, y)$. For graphs, the kernel $G(x, y)$ takes a discrete form:

$$L[H(t)]_i = \sum_{j=1}^{n} G_{ij}(H_j(t) - H_i(t)), \tag{12}$$

where G_{ij} represents the adaptive influence of node j on node i, enabling the propagation mechanism to dynamically adapt to graph structures.

The kernel G_{ij} formally defined as follows, using similarity metrics inspired by attention mechanisms:

$$G_{ij} = \frac{\exp(\text{sim}(H_i, H_j))}{\sum_{k=1}^{n} \exp(\text{sim}(H_i, H_k))}, \tag{13}$$

where $\text{sim}(H_i, H_j)$ measures the similarity between node embeddings H_i and H_j. This similarity is implemented as a normalized dot product:

$$\text{sim}(H_i, H_j) = \frac{H_i^\top H_j}{\|H_i\|_2 \|H_j\|_2}, \tag{14}$$

where $\|H_i\|_2 = \sqrt{\sum_{k=1}^{d} H_{i,k}^2}$ denotes the Euclidean norm of the embedding H_i, and d is the embedding dimension. This approach effectively captures the relative importance of nodes, enabling the kernel to focus on more relevant relationships within the graph.

By integrating the non-local kernel G_{ij} into the wave equation, we enable the propagation mechanism to capture both local and global relationships across the graph. This enhancement allows the model to retain high-frequency information while adapting to the graph's structure, overcoming the limitations of purely diffusive approaches that fail to propagate information effectively over long distances. To incorporate the non-local kernel G_{ij} into the wave-driven propagation mechanism, we discretize the wave equation and replace the graph Laplacian with G_{ij} shown as follows:

$$\begin{aligned} H^{(l+1)} = (1-\alpha)H^{(l)} + \alpha \Big(& \beta \left(H^{(l)} - H^{(l-1)} \right) \\ & + \eta \left(GH^{(l)} - H^{(l)} \right) \Big). \end{aligned} \tag{15}$$

The terms in the update rule are explained as follows:

1. $(1 - \alpha)H^{(l)}$ represents the retention of the current state, with α controlling how much of the previous layer's information is carried over into the current layer.
2. $\beta(H^{(l)} - H^{(l-1)})$ introduces temporal feedback between successive layers, with β adjusting the strength of the feedback and determining how much influence the previous layer has on the current layer.
3. $\eta \left(GH^{(l)} - H^{(l)} \right)$ facilitates dynamic, non-local interactions between nodes, using the kernel G to quantify the influence of node j on node i, enabling information to propagate across the entire graph.

By discretizing the wave equation with the non-local kernel and applying it to graph-structured data, our approach enables the construction of deeper architectures that preserve node-level distinctions, making it more effective for tasks requiring fine-grained feature discrimination.

3.2 Energy-based Regulation for Graph Propagation

Wave-driven GNNs utilize second-order dynamics to mitigate over-smoothing by preserving high-frequency components through oscillatory propagation. However, introducing

oscillations can lead to challenges, such as unregulated fluctuations and instability in deep architectures. To address these issues, we propose an energy-based perspective inspired by classical physics, where the propagation process is analyzed and adapted through the interplay of kinetic energy and potential energy. This analogy provides a principled framework to understand and regulate energy dynamics in graph propagation mechanisms:

Kinetic Energy : Quantifies the temporal evolution of node embeddings, capturing how actively features change between consecutive layers. It is defined as:

$$E_{\text{kin}}^{(l)} = \frac{1}{2}\|H^{(l)} - H^{(l-1)}\|_F^2, \tag{16}$$

where $H^{(l)}$ and $H^{(l-1)}$ are the node embedding matrices at layers l and $l-1$, respectively. This metric is inspired by studies of time-dependent feature evolution in graph dynamical systems [Kipf *et al.*, 2018; Bronstein *et al.*, 2017].

Potential Energy : Reflects the structural alignment of node embeddings, penalizing large discrepancies between connected nodes. It is defined as:

$$E_{\text{pot}}^{(l)} = \frac{1}{2}\sum_{(i,j)\in E} \hat{A}\|H_i^{(l)} - H_j^{(l)}\|^2, \tag{17}$$

where $\hat{A}$ represents the normalized adjacency matrix, and E is the set of edges. This formulation aligns with the graph Laplacian's role in encoding structural relationships [Coifman and Lafon, 2006].

An imbalance between kinetic and potential energy can lead to undesirable behavior. Excessive kinetic energy may amplify oscillations, destabilizing the propagation process, while dominant potential energy may overly constrain feature updates, resulting in over-smoothing. To achieve a balanced propagation dynamic, we monitor the energy ratio:

$$r = \frac{E_{\text{kin}}}{E_{\text{pot}}}, \tag{18}$$

where r reflects the contributions of temporal dynamics (kinetic energy) and structural alignment (potential energy).

To encourage an optimal energy balance, the framework dynamically adjusts propagation parameters based on the energy ratio. Specifically, a target energy ratio $r_{\text{target}} = 1$ is employed, representing equal contributions from kinetic and potential energy. This setting is inspired by the equipartition theorem[Landau and Lifshitz, 1980], which describes systems in equilibrium where kinetic and potential energies are equal (as in a 1D harmonic oscillator). Adopting this principle helps balance feature dynamics and structural consistency within EWGNN. The adjustment mechanism then follows:

$$\Delta\beta \propto (r - r_{\text{target}}). \tag{19}$$

The proposed energy framework enhances wave-driven GNNs by dynamically adjusting the propagation process through a balanced interplay of kinetic and potential energy. This principled approach moderates oscillations, ensures stable information flow, and maintains the discriminative power of node features across deep architectures.

3.3 Layer-Wise Adaptive Training Strategy

To tackle over-smoothing and instability in GNNs, we adopt a dynamic training strategy that integrates feature transformation with energy-based adjustment. This approach eliminates explicit regularization by interleaving propagation steps with energy-aware regulations, ensuring both stability and expressiveness in the model.

Node Classification Objective. The primary objective remains node classification, with the supervised loss function defined as:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{|\mathcal{V}_{\text{train}}|}\sum_{i\in\mathcal{V}_{\text{train}}} \log p(y_i|H_i^{(L)}), \tag{20}$$

where $\mathcal{V}_{\text{train}}$ is the set of labeled nodes, y_i is the ground truth label for node i, $H_i^{(L)}$ is the embedding of node i at the final layer L, and $p(y_i|H_i^{(L)})$ represents the predicted probability of the correct label. This objective ensures the learned embeddings remain task-specific and discriminative.

Energy-Aware Parameter Adjustment. The novel aspect of our training strategy lies in its energy-aware design, which alternates between feature transformation and energy dynamics. Specifically, each propagation step $H^{(l)}$ is followed by an energy-based adjustment layer to dynamically learn the propagation parameter $\beta^{(l)}$:

$$\beta^{(l)} = \text{Sigmoid}(\text{FC}(r^{(l)})), \tag{21}$$

where, $\text{FC}(\cdot)$ denotes a fully connected layer that transforms the energy ratio $r^{(l)} = E_{\text{kin}}^{(l)}/E_{\text{pot}}^{(l)}$, and the sigmoid activation ensures that $\beta^{(l)}$ is constrained within $(0, 1)$. This adaptive adjustment ensures the model dynamically balances temporal and structural components during propagation, mitigating unregulated oscillations while preserving feature diversity.

Training Workflow. The proposed strategy alternates between two modules: feature transformation and energy adjustment. In the Feature Transformation phase, each layer propagates features using the wave-driven mechanism, ensuring effective information flow and preserving high-frequency components. Following this, in the Energy Adjustment phase, the energy ratio $r = E_{\text{kin}}/E_{\text{pot}}$ is computed, and $\beta^{(l)}$ is adjusted to stabilize propagation and optimize feature alignment. This interleaved structure eliminates explicit regularization terms while ensuring stability and expressiveness.

4 Experiments

In this section, we empirically evaluate EWGNN through the following questions.

- (Q1) Does EWGNN outperform state-of-the-art models across various datasets?
- (Q2) Can EWGNN effectively mitigate over-smoothing and scale to deeper architectures by preserving high-frequency information?
- (Q3) Can EWGNN effectively balance energy dynamics in wave propagation?

Type	Model	Non-linearity	Diffusion-based	Kernel	Cora	Citeseer	Pubmed
Basic models	MLP	✓	-	-	56.1 ± 1.6	56.7 ± 1.7	69.8 ± 1.5
	LP	-	-	-	68.2	42.8	65.8
	ManiReg	✓	-	-	60.4 ± 0.8	67.2 ± 1.6	71.3 ± 1.4
Classical and Enhanced GNNs	GCN	✓	✓	-	81.5 ± 1.3	71.9 ± 1.9	77.8 ± 2.9
	GAT	✓	-	-	83.0 ± 0.7	72.5 ± 0.7	79.0 ± 0.3
	SGC	-	-	✓	81.0 ± 0.0	71.9 ± 0.1	78.9 ± 0.3
	GIN	-	-	-	77.6 ± 1.1	66.1 ± 0.9	77.0 ± 1.2
	GCNII	-	✓	-	85.5 ± 0.5	73.4 ± 0.6	80.2 ± 0.4
	PairNorm	-	-	-	78.3 ± 1.3	65.8 ± 1.4	75.5 ± 0.4
	DropEdge	-	-	-	83.5	72.7	79.5
	JK-Net	-	-	-	81.8 ± 0.5	70.7 ± 0.7	78.8 ± 0.7
Diffusion and Frequency-Aware Models	GRAND-l	-	✓	-	83.6 ± 1.0	73.4 ± 0.5	78.8 ± 1.7
	GRAND	✓	✓	-	83.3 ± 1.3	74.1 ± 1.7	78.1 ± 1.9
	GRAND++	✓	✓	-	82.2 ± 1.3	74.1 ± 1.5	78.1 ± 1.9
	APPNP	✓	✓	-	83.3 ± 0.5	71.7 ± 0.6	80.1 ± 0.6
	S^2GC	-	-	✓	83.2 ± 0.0	73.6 ± 0.1	80.1 ± 0.2
	DAGNN	✓	✓	-	84.3 ± 0.6	73.1 ± 0.6	80.6 ± 0.5
	FAGCN	✓	✓	-	83.8 ± 0.4	72.0 ± 0.4	79.5 ± 0.4
	MGC	✓	✓	-	$\underline{85.6 \pm 0.3}$	73.5 ± 0.2	80.5 ± 0.3
Transformer-Based and Contrastive Learning Models	DGI	-	-	-	82.5 ± 0.7	71.6 ± 0.7	78.4 ± 0.7
	NodeFormer	-	-	-	82.4 ± 0.5	72.1 ± 1.2	79.2 ± 0.6
	Difformer	-	-	-	85.3 ± 0.8	$\underline{74.5 \pm 0.2}$	$\underline{80.7 \pm 0.8}$
	CSF	✓	✓	✓	81.1 ± 0.1	68.5 ± 0.3	76.8 ± 0.3
Proposed Model	EWGNN	-	-	✓	$\mathbf{86.1 \pm 0.6}$	$\mathbf{75.3 \pm 0.5}$	$\mathbf{81.5 \pm 0.5}$

Table 1: Node classification accuracy (%) on Cora, Citeseer, and PubMed datasets, averaged over five random initializations.

Datasets. To evaluate the effectiveness of the proposed wave-driven propagation mechanism, we employ several widely used citation network datasets, including 4 citation graphs: `Cora`, `Citeseer`, `PubMed` [Sen *et al.*, 2008], and `ogbn-proteins` [Hu *et al.*, 2020], alongside real-world datasets such as `Pokec` [Takac and Zabovsky, 2012]. For each dataset, we adopt the standard public split [Yang *et al.*, 2016], allocating 20 labeled nodes per class for training, 500 nodes for validation, and 1,000 nodes for testing.

Baselines. We compare our proposed model against a comprehensive set of baseline methods, including MLP, LP, ManiReg, GCN [Kipf and Welling, 2017], GAT [Veličković *et al.*, 2018], SGC [Wu *et al.*, 2019], GCNII [Chen *et al.*, 2020], GIN [Xu *et al.*, 2018a], JK-Net [Xu *et al.*, 2018b], PairNorm [Zhao and Akoglu, 2020], DropEdge [Rong *et al.*, 2020], GRAND [Chamberlain *et al.*, 2021], GRAND++ [Thorpe *et al.*, 2022], APPNP [Gasteiger *et al.*, 2018], S^2GC [Zhu and Koniusz, 2021], DAGNN [Liu *et al.*, 2020], FAGCN [Bo *et al.*, 2021], MGC [Shen *et al.*, 2024], DGI [Veličković *et al.*, 2019], NodeFormer [Wu *et al.*, 2022], DIFFORMER [Wu *et al.*, 2023], and CSF [Huang *et al.*, 2024].

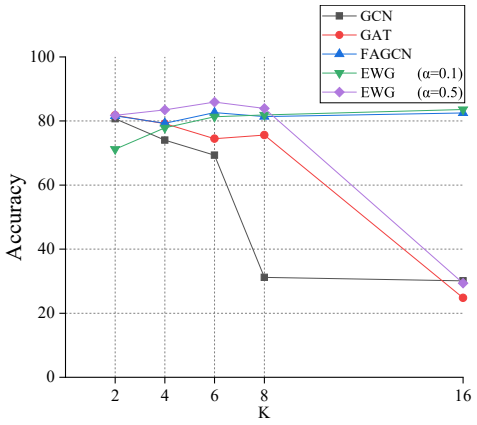
4.1 RQ1: EWGNN Outperforms SOTA on Benchmarks

Table 1 presents the performance of EWGNN, which consistently outperforms other models on the `Cora`, `Citeseer`, and `PubMed` datasets. These results suggest that EWGNN is effective in addressing the limitations of diffusion-based methods, potentially mitigating over-smoothing and preserving feature diversity in citation network graphs.

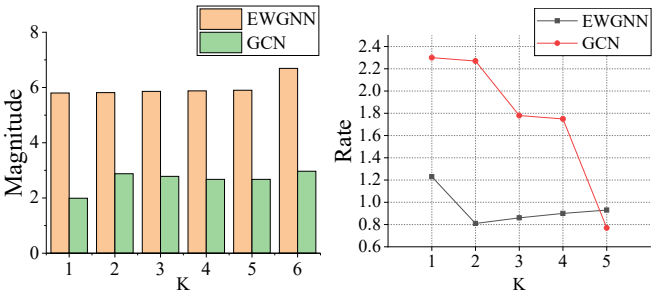
Furthermore, as shown in Table 2, EWGNN also outperforms other competitive models on the larger `ogbn-proteins` and `Pokec` datasets. The results demonstrate that EWGNN is able to achieve state-of-the-art performance across a diverse range of graph datasets, highlighting EWGNN's scalability and adaptability to handle more complex real-world graph structures.

4.2 RQ2: EWGNN Enables Deeper Architectures

We also investigate the performance of EWGNN under different settings of α, comparing it to other models. As shown in Figure 2a, EWGNN with $\alpha = 0.5$ results in larger feature updates, leading to significant improvements in performance and achieving SOTA performance, while training to a reasonable depth. EWGNN with $\alpha = 0.1$ retains more features,



(a) The accuracy of different models as a function of the number of layers (K) on `Cora`.



(b) Average magnitude of high-frequency components across different layers (K) on `Cora`.

(c) Energy ratio across different layers (K) on `Cora`.

Figure 2: Overview of experimental results on `Cora`.

enabling the model to scale to deeper architectures. This version of EWGNN outperforms the SOTA model FAGCN across most layer depths, demonstrating that our model effectively balances depth and expressive power.

Furthermore, we further confirm the ability of EWGNN to preserve high-frequency information, as shown in Figure 2b. We define high-frequency information based on the eigenvalue decomposition of the feature matrix, where the top 50% eigenvalues represent the most significant components. When comparing EWGNN with diffusion-based models such as GCN, we observe that EWGNN consistently preserves a higher magnitude of high-frequency information, and this retention improves as the number of layers increases. The magnitude of high-frequency components in EWGNN is higher than in GCN across all layers, demonstrating the model's ability to retain important features even as the depth increases.

4.3 RQ3: EWGNN Learns Appropriate Energy Dynamics

We next explore how EWGNN manages energy dynamics in wave propagation, as shown in Figure 2c. By visualizing the energy ratio across layers, we observe that EWGNN maintains a stable energy distribution, gradually converging to 1 as the number of layers increases. This stable energy ratio is key to mitigating oscillations and smoothing, enabling the model to preserve high-frequency components even as the depth increases. In contrast, diffusion-based models, such as GCN, show less stability in their energy ratios, demonstrating that EWGNN more effectively controls energy dynamics and ensures consistent propagation across deeper layers.

Models	Proteins	Pokec
MLP	72.4 ± 0.1	60.2 ± 0.1
SGC	49.2 ± 1.1	53.1 ± 0.9
GCN	74.2 ± 0.5	62.3 ± 1.1
GAT	75.1 ± 1.4	65.6 ± 0.5
NodeFormer	77.5 ± 1.2	68.3 ± 0.5
DIFFORMER	79.5 ± 0.4	69.2 ± 0.8
EWGNN	80.0 ± 0.5	71.0 ± 0.9

Table 2: Testing ROC-AUC for ogbn-proteins and Accuracy for Pokec on large-scale node classification datasets.

Dataset	Fixed($\beta = 0.1$)	Fixed($\beta = 1.0$)	Adaptive
Cora	84.2 ± 0.8	33.1 ± 2.2	86.1 ± 0.6
Citeseer	72.7 ± 0.6	23.6 ± 1.6	75.3 ± 0.5
PubMed	80.5 ± 0.8	53.0 ± 3.7	81.5 ± 0.5
Proteins	78.9 ± 0.6	79.2 ± 0.4	80.0 ± 0.5
Pokec	69.0 ± 2.8	70.6 ± 0.8	71.0 ± 0.9

Table 3: Ablation study on the performance with adaptive and fixed β across different datasets.

To further investigate the role of energy dynamics in our model, we perform an ablation study by testing two fixed values of β (0.1 and 1.0) alongside the adaptively learned β, which is adjusted based on energy dynamics. As shown in Table 3, the results demonstrate that the model with the adaptively learned β consistently outperforms those with fixed values. This confirms that learning β from energy dynamics is crucial for balancing oscillations and smoothing, ensuring stability and high performance across different depths. Notably, for smaller datasets like `Cora` and `Citeseer`, a smaller value of β (0.1) performs better, likely due to the model not requiring as much feedback strength. For larger datasets like `PubMed` and `Pokec`, a larger β (1.0) shows comparable performance, as a stronger feedback mechanism helps mitigate over-smoothing and supports deeper propagation.

5 Discussions

The proposed wave-driven GNN framework introduces an innovative approach to feature propagation by leveraging second-order dynamics to counteract over-smoothing and preserve high-frequency components, crucial for expressive node representations. By incorporating an energy-based regulation mechanism, the model ensures stability and effective convergence, particularly in deeper architectures, offering a principled alternative to traditional diffusion-based methods.

Empirical results on benchmark datasets and real-world graphs validate the framework's ability to mitigate over-smoothing and maintain feature diversity. Future work could explore efficient implementations, such as localized propagation or sparse tensor operations.

Contribution Statement

Peihan Wu and Hongda Qi contributed equally to this work and are therefore designated as co-first authors. Both authors made significant contributions to the development of the core intellectual content, the overall structure of the manuscript, and its writing. Peihan Wu was primarily responsible for developing and implementing the proposed wave-driven GNN framework, including its core algorithmic innovations and the energy-based stabilization mechanism. He also designed and executed key experimental validations on benchmark datasets, analyzing the results to demonstrate the framework's efficacy. Hongda Qi was primarily responsible for the substantive revisions and refinement of the manuscript, ensuring its overall clarity and coherence.

Qin Zhao, as the corresponding author, supervised the research project and was responsible for its overall design, coordination, and integrity. Qin Zhao also provided critical review and final approval of the manuscript.

Sirong Huang, Dongdong An, and Jie Lian provided valuable contributions to this study, including critical review of the methodology and assistance with data interpretation. Their insights significantly enhanced the quality of the research.

All authors have read and approved the final manuscript.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2022YFB4501704; and in part by the National Natural Science Foundation of China under Grant Nos. 62302308, 62372300, and 61702333.

References

[Bo *et al.*, 2021] Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 3950–3957, 2021.

[Bronstein *et al.*, 2017] Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, July 2017.

[Chamberlain *et al.*, 2021] Ben Chamberlain, James Rowbottom, Maria I Gorinova, Michael Bronstein, Stefan Webb, and Emanuele Rossi. Grand: Graph neural diffusion. In *International conference on machine learning*, pages 1407–1418. PMLR, 2021.

[Chen *et al.*, 2020] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International conference on machine learning*, pages 1725–1735. PMLR, 2020.

[Coifman and Lafon, 2006] Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006. Special Issue: Diffusion Maps and Wavelets.

[Gasteiger *et al.*, 2018] Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. In *International Conference on Learning Representations*, 2018.

[Hu *et al.*, 2020] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[Huang *et al.*, 2024] Chen Huang, Haoyang Li, Yifan Zhang, Wenqiang Lei, and Jiancheng Lv. Cross-space adaptive filter: Integrating graph topology and node attributes for alleviating the over-smoothing problem. In *Proceedings of the ACM on Web Conference 2024*, pages 803–814, 2024.

[Kipf and Welling, 2017] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2017.

[Kipf *et al.*, 2018] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems, 2018.

[Landau and Lifshitz, 1980] L. D. Landau and E. M. Lifshitz. *Statistical Physics, Part 1*, volume 5 of *Course of Theoretical Physics*. Butterworth-Heinemann, Oxford, 3rd ed. edition, 1980.

[Li *et al.*, 2018] Qimai Li, Zhichao Han, and Xiao-Ming Wu. Deeper insights into graph convolutional networks for semi-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[Liu *et al.*, 2020] Meng Liu, Hongyang Gao, and Shuiwang Ji. Towards deeper graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 338–348, 2020.

[Rong *et al.*, 2020] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. In *International Conference on Learning Representations*, 2020.

[Sen *et al.*, 2008] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

[Shen *et al.*, 2024] Dazhong Shen, Chuan Qin, Qi Zhang, Hengshu Zhu, and Hui Xiong. Handling over-smoothing and over-squashing in graph convolution with maximization operation. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

[Takac and Zabovsky, 2012] Lubos Takac and Michal Zabovsky. Data analysis in public social networks. In *International scientific conference and international workshop present day trends of innovations*, volume 1, 2012.

[Thorpe *et al.*, 2022] Matthew Thorpe, Tan Nguyen, Hedi Xia, Thomas Strohmer, Andrea Bertozzi, Stanley Osher, and Bao Wang. Grand++: Graph neural diffusion with

a source term. In *International Conference on Learning Representations*, 2022.

[Veličković *et al.*, 2018] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[Veličković *et al.*, 2019] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. In *International Conference on Learning Representations*, 2019.

[Wu *et al.*, 2019] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. PMLR, 2019.

[Wu *et al.*, 2022] Qitian Wu, Wentao Zhao, Zenan Li, David P Wipf, and Junchi Yan. Nodeformer: A scalable graph structure learning transformer for node classification. *Advances in Neural Information Processing Systems*, 35:27387–27401, 2022.

[Wu *et al.*, 2023] Qitian Wu, Chenxiao Yang, Wentao Zhao, Yixuan He, David Wipf, and Junchi Yan. Difformer: Scalable (graph) transformers induced by energy constrained diffusion. In *The Eleventh International Conference on Learning Representations*, 2023.

[Xu *et al.*, 2018a] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2018.

[Xu *et al.*, 2018b] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *International conference on machine learning*, pages 5453–5462. PMLR, 2018.

[Yang *et al.*, 2016] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*, pages 40–48. PMLR, 2016.

[Zhao and Akoglu, 2020] Lingxiao Zhao and Leman Akoglu. Pairnorm: Tackling oversmoothing in gnns. In *International Conference on Learning Representations*, 2020.

[Zhu and Koniusz, 2021] Hao Zhu and Piotr Koniusz. Simple spectral graph convolution. In *International conference on learning representations*, 2021.