SIZE GENERALIZATION OF GRAPH NEURAL NETWORKS ON BIOLOGICAL DATA: INSIGHTS AND PRACTICES FROM THE SPECTRAL PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate size-induced distribution shifts in graphs and assess their impact on the ability of graph neural networks (GNNs) to generalize to larger graphs relative to the training data. Existing literature presents conflicting conclusions on GNNs' size generalizability, primarily due to disparities in application domains and underlying assumptions concerning size-induced distribution shifts. Motivated by this, we take a data-driven approach: we focus on real biological datasets and seek to characterize the types of size-induced distribution shifts. Diverging from prior approaches, we adopt a spectral perspective and identify that spectrum differences induced by size are related to differences in subgraph patterns (e.g., average cycle lengths). We further find that common GNNs cannot capture these subgraph patterns, resulting in performance decline when testing on larger graphs. Based on these spectral insights, we introduce and compare three model-agnostic strategies aimed at making GNNs aware of important subgraph patterns to enhance their size generalizability: self-supervision, augmentation, and size-insensitive attention. Our empirical results reveal that all strategies enhance GNNs' size generalizability, with simple size-insensitive attention surprisingly emerging as the most effective method. Notably, this strategy substantially enhances graph classification performance on large test graphs, which are 2-10 times larger than the training graphs, resulting in an improvement in F_1 scores by up to 8%.

1 INTRODUCTION

Graph neural networks (GNNs) (17; 28; 12; 40; 37; 19) have gained widespread popularity in graph classification tasks owing to their outstanding performance. Though most GNNs can process graphs of varying sizes, it remains under-explored whether they can generalize to graphs larger than those encountered during training (size generalizability). Size generalization in GNNs holds significant importance across multiple domains. For instance, in graph algorithmic reasoning (36; 29), GNNs are expected to learn complex algorithms from small examples and generalize that reasoning to larger graphs, as obtaining exact solutions for larger graphs is challenging. In the realm of biology, datasets exhibit a wide range of graph sizes, spanning from small molecules to large compounds. Evaluating whether learned knowledge is influenced by graph size is crucial, as size-dependent information may potentially have a detrimental impact on performance when employing pre-training strategies (13).

Existing literature presents conflicting conclusions on GNNs' size generalizability. On one hand, several studies (21; 35; 26) have provided support for the ability of GNNs to effectively generalize across varying sizes. For instance, a theoretical study (21) established that spectral GNNs exhibit robust transferability between graphs with different sizes and topologies, provided that these graphs discretize the same underlying space in some generic sense. Other works further provided empirical evidence supporting the strong size generalizability of GNNs in the domains of algorithmic task learning (35) and physics simulations (26). On the other hand, several studies (38; 6; 4) have observed performance degradation when a size shift exists between the training and test data. For instance, a recent work (38) showed theoretically and empirically that the difference in degree patterns between small and large graphs contributes to this performance decline. There have also been proposals of novel models (4) and regularization techniques (6) to enhance the size generalizability of GNNs.

These conflicts mainly arise from disparities in application domains and underlying assumptions concerning size-induced distribution shifts.

Motivated by these conflicts, we take a data-driven approach: we focus on real biological datasets and seek to characterize the types of size-induced distribution shifts. This characterization provides valuable insights into the size generalizability of GNNs. Specifically, we adopt a spectral perspective and identify the connections between the spectrum differences induced by varying graph sizes and the differences in subgraph patterns, particularly cycles. We find that breaking cycles in graphs amplifies the spectrum difference between smaller and larger graphs, whereas extending cycle lengths in smaller graphs to align with those in larger graphs reduces this difference. Furthermore, we observe that conventional GNNs struggle to generalize effectively without explicit cycle information, leading to performance degradation on larger graphs. To address this, we propose and compare three model-agnostic strategies aimed at equipping GNNs with cycle information to enhance size generalizability: self-supervision, augmentation, and size-insensitive attention. Our empirical results demonstrate that all strategies enhance GNNs' size generalizability, with simple size-insensitive attention surprisingly emerging as the most effective method. Although prior research has established GNNs' limitations in counting cycles (8), the primary focus of this paper is to delve into how this limitation influences the size generalizability of GNNs.

In sum, our paper makes the following contributions:

- **New Observations.** We characterize the types of distribution shifts caused by various graph sizes in biological networks, offering insights for designing a size-agnostic GNN.
- **Spectral Analysis.** Unlike prior work, we leverage spectral analysis to deepen our understanding of the size generalizability of GNNs.
- **Model Agnostic Strategies.** To make GNNs aware of important size-related subgraph patterns (e.g., average cycle lengths), we propose and compare three model-agnostic strategies that improve size-generalizability of GNNs. We find that simple size-insensitive attention is the most effective strategy among the three.

2 Notations and Preliminaries

In this section, we begin by introducing the notations and definitions used throughout the paper. Next, we provide an introduction to the fundamentals of GNNs.

2.1 Notations & Definitions

Let $\mathcal{G}(\mathcal{V}, \mathcal{E})$ be an undirected and unweighted graph with N nodes, where $\mathcal{V}$ denotes the node set, and $\mathcal{E}$ denotes the edge set. The neighborhood of a node v_i is defined as the set of all nodes that connect to v_i: $\mathcal{N}_i = \{v_j | (v_j, v_i) \in \mathcal{E}\}$. The graph is represented by its adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, and it has a degree matrix $\mathbf{D}$, where the ith diagonal element d_i corresponds to the degree of node v_i.

Cycle basis. An important concept we use to study cycles is *cycle basis* (27). A cycle basis is defined as the smallest set of cycles where any cycle in the graph can be expressed as a sum of cycles from this basis, similar to the concept of a basis in vector spaces. Here, the summation of cycles is defined as "exclusive or" of the edges. We represent the cycle basis for a graph as $\mathcal{C}$ and refer to the jth cycle in this cycle basis as $\mathcal{C}_j$. The cycle basis can be found using the algorithm CACM 491 (25).

2.2 Graph Learning Task

In this paper, we focus on the graph classification task, where each node v_i is associated with a feature vector $\mathbf{x}_i^{(0)}$, and the feature matrix $\mathbf{X}^{(0)}$ is constructed by arranging the node feature vectors as rows. When using a GNN for the graph classification task, we further denote the node representation matrix at the l-th layer as $\mathbf{X}^{(l)}$, and the representation of node v_i as $\mathbf{x}_i^{(l)}$.

Supervised Graph Classification. Each graph $\mathcal{G}_i$ is associated with a ground truth label $y_i^{\mathcal{G}}$ sampled from a label set $\hat{\mathcal{L}}$. Given a subset of labeled graphs (from a label set $\hat{\mathcal{L}}$), the goal is to learn a mapping $f^{\mathcal{G}} : (\mathbf{A}, \mathbf{X}^{(0)})_i \mapsto y_i^{\mathcal{G}}$ between each graph $\mathcal{G}_i$ and its ground truth label $y_i^{\mathcal{G}} \in \hat{\mathcal{L}}$. The graph classification loss is given by $L = \frac{1}{|\mathcal{G}_{\text{train}}|} \sum_{\mathcal{G}_i \in \mathcal{G}_{\text{train}}} \texttt{CrossEntropy}\left(\mathbf{x}^{\mathcal{G}_i}, y_i^{\mathcal{G}}\right)$, where $\mathcal{G}_{\text{train}}$ is the training graph set and $\mathbf{x}^{\mathcal{G}_i}$ is the representation of graph $\mathcal{G}_i$.

Evaluation of Size Generalizability. Following prior work (6; 38), we evaluate the size generalizability of GNNs by testing their classification performance on graphs whose sizes are larger than those in the train set. We obtain the small training graphs and large test graphs from the same dataset.

2.3 Graph Neural Networks

GNNs can be designed from either the spatial perspective or the spectral perspective. Despite the difference in the design perspectives, a recent work (1) has shown that spectral GNNs and spatial GNNs are related and that spectral analysis of GNNs' behavior can provide a complementary point of view to understand GNNs in general. Most spatial GNNs (17; 34; 28; 12) use the message passing framework (11), which consists of three steps: neighborhood propagation, message combination and global pooling. Spectral GNNs (5; 10; 22) utilize the spectral properties of a propagation matrix $\mathbf{T}$ to perform the graph classification. The propagation matrix $\mathbf{T}$ is usually a function of the adjacency matrix $\mathbf{A}$, such as the normalized adjacency matrix $\mathbf{T} = (\mathbf{D}+\mathbf{I})^{-1/2}(\mathbf{A}+\mathbf{I})(\mathbf{D}+\mathbf{I})^{-1/2}$, or the normalized graph Laplacian matrix $\hat{\mathbf{L}}$. Since we consider an undirected graph with a real and symmetric adjacency matrix, the propagation matrix $\mathbf{T}$ is also real and symmetric. Then, we can perform the eigendecomposition on the propagation matrix $\mathbf{T}$: $\mathbf{T} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where $\mathbf{U}$ is an orthogonal matrix whose columns $\mathbf{U}_i$ are orthonormal and are the eigenvectors of $\mathbf{T}$, and $\mathbf{\Lambda}$ is a matrix whose diagonal elements are the eigenvalues of $\mathbf{T}$, sorted from large to small by their absolute values. The set of eigenvectors $\{\mathbf{U}_i\}$ form the orthonormal basis of $\mathbb{R}^n$. The goal of a spectral GNN is to learn a proper spectral filter: $f(\mathbf{\Lambda}) = c_0\mathbf{I} + c_1\mathbf{\Lambda} + c_2\mathbf{\Lambda}^2 + \cdots + c_i\mathbf{\Lambda}^i + \cdots$, where c_i are the learnable coefficients. The convolution at each layer can be viewed as or is equivalent to: $\mathbf{X}^{(l+1)} = \sigma(\mathbf{U}f(\mathbf{\Lambda})\mathbf{U}^T\mathbf{X}^{(l)}\mathbf{W}^{(l)})$, where $\mathbf{W}^{(l)}$ is a learnable weight matrix, and $\sigma(\cdot)$ is a nonlinear function (e.g., `ReLU`). The graph representation is obtained from the node representations at the last convolution layer: $\mathbf{x}^{\mathcal{G}} = \texttt{Pooling}\ (\{\mathbf{x}_i^{(\text{Last})}\})$, where the `Pooling` function is performed on the set of all the node representations, and it can be `Global_mean` or `Global_max` or other more complex pooling functions (39; 18).

3 Spectral analysis of size-induced Distribution Shifts

In this section, we first show that the independence of the eigenvalue distribution of the propagation matrix $\mathbf{T}$ from the graph size is the key to achieving size generalizability of GNNs (§ 3.1). Next, focusing on biologically data, we empirically verify that the eigenvalue distribution of the propagation matrix depends on the graph size (§ 3.2). Finally, we explore the subgraph patterns responsible for the spectral disparities between small and large graphs, unveiling two key findings in § 3.3:

- Breaking cycles in graphs amplifies the spectrum difference between smaller and larger graphs.
- Extending cycle lengths in smaller graphs to match larger ones reduces the spectrum difference.

3.1 Graph spectrum and size generalizability of GNNs

To understand how GNNs generalize over graphs with different sizes, we examine the formulation of graph representations. In the context of spectral GNNs, graph representations rely on the eigenvalues of the propagation matrix. Consequently, the connection between graph representations and graph size reduces to the connection between the graph's spectrum and its size. More formally, we theoretically show the following proposition in Appendix A.

Proposition 1 *When graphs of various sizes exhibit distinct eigenvalue distributions for the propagation matrix, the representations learned by spectral GNNs correlate with the graph size.*

The proposition suggests that for GNNs to achieve effective generalization to larger graphs, the disparity in the spectrum between small and large graphs should be small.

3.2 Size-related Spectrum Differences in Real-world Data

We now investigate how the eigenvalue distribution of the normalized adjacency matrix varies with graph size in real-world data. As indicated in Proposition 1, the spectrum discrepancy between small and large graphs affects the size generalizability of GNNs.

Datasets. We explore five pre-processed biological datasets (BBBP, BACE, NCI1, NCI109, and PROTEINS) from the Open Graph Benchmark (14) and TuDataset (23). More details about the datasets are provided in Appendix B.

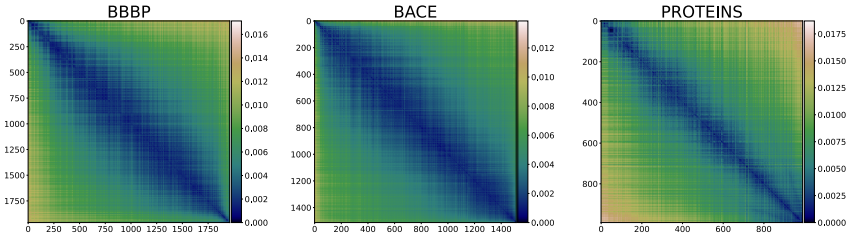

Figure 1: Pairwise graph distance of eigenvalue distributions: Graphs are sorted from small to large, and the (i,j)-th pixel in the plot represents the Wasserstein distance between the i-th and j-th graphs. Dark blue represents a small distance (high similarity), while light red represents a large distance (low similarity). We find that **eigenvalue distributions show a strong correlation with the graph size**.

Table 1: Average Wasserstein distance between graphs of 'similar sizes' and graphs of 'different sizes' based on **eigenvalue** distributions, respectively. The relative difference is computed by the difference of the Wasserstein distance normalized by the Wasserstein distance of similar graphs.

	BBBP	**BACE**	**PROTEINS**	**NCI109**	**NCI1**
Different Size	0.00566	0.00411	0.00765	0.00563	0.00566
Similar Size	0.00184	0.00149	0.00261	0.00215	0.00215
Relative Difference	208%	177%	193%	162%	164%

Setup. Figure 1 illustrates the pairwise distances of the graphs arranged in ascending order of size, where the distances are calculated using the Wasserstein distance (31). We represent the graphs by their empirical distributions of the eigenvalues that are obtained from the normalized adjacency matrix as suggested in (17): $\mathbf{T} = (\mathbf{D}+\mathbf{I})^{-1/2}(\mathbf{A}+\mathbf{I})(\mathbf{D}+\mathbf{I})^{-1/2}$. Using the normalized Laplacian matrix leads to similar observations. We note that the eigenvalues do not scale with the graph size, and they are bounded between [-1,1]. Dark blue represents a small distance (high similarity) while light red represents a large distance (low similarity).

Results. As can be seen in the three subplots in Figure 1, there is a wide blue band along the diagonal, which indicates that graphs of similar size have more similar eigenvalue distributions than graphs of different sizes. This suggests a strong correlation between the eigenvalue distributions and the graph size. To verify the observation quantitatively, we compute the distance of graphs with 'similar size' and graphs of 'different sizes' in Table 1. For each graph, we consider the 20 most 'similar graphs' in terms of size, and treat the remaining graphs as graphs of 'different sizes'. The table shows that the Wasserstein distances of eigenvalue distributions between the graphs of different sizes are significantly larger than the distances between graphs of similar size. Based on the empirical results and Proposition 1, the correlation between the eigenvalue distributions and the graph size results in the correlation of the final graph representation and the graph size, which prevents GNNs from generalizing over larger size.

3.3 Key Findings: Size-related Differences in Subgraph Patterns

In this subsection, we aim to identify the subgraph patterns that explain the spectrum differences between small and large graphs. Our empirical analysis pinpointed several peaks in the graph spectrum that match the spectrum of cycles. This motivated us to examine how cycle properties differ in small and large graphs and how these differences are revealed in the spectrum. Specifically, we aim to answer two questions: (Q1) How does the existence of cycles in the graphs influence the spectrum differences? (Q2) How do the variations in cycle lengths contribute to differences in the spectrum? In our analysis, we investigate the properties of the cycles in the cycle basis of each graph.

3.3.1 Existence of Cycles & Spectrum: The Impact of Breaking Cycles

To understand how the existence of cycles in the graphs influences the spectrum differences, we break all basis cycles with minimal edge removal while maintaining the same number of disconnected components, according to the details and algorithm given in Appendix D.1. We analyze the impact of breaking cycles by assessing the corresponding changes in the spectrum. By following the convention

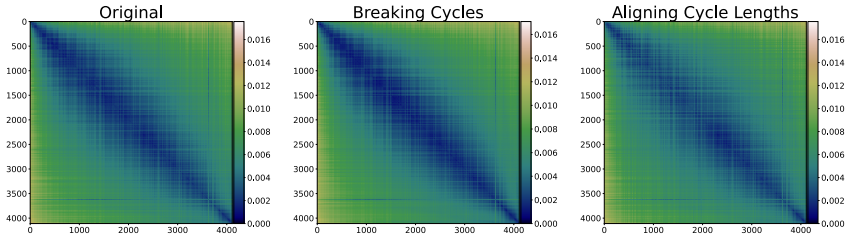

Figure 2: Pairwise graph distance measured by the Wasserstein distance of eigenvalue distributions after breaking cycles and aligning cycle lengths on the NCI109 dataset. Breaking cycles **amplifies the correlation** between eigenvalue distribution and graph size, while aligning cycle lengths **reduces the correlation**.

Table 2: Average Wasserstein distance of eigenvalue distributions between graphs of similar size and graphs of different sizes after breaking cycles and aligning cycle lengths. Relative difference is computed as in Table 1. We use ↑ (↓) to denote the increase (decrease) in the relative difference compared to not taking the corresponding action. Breaking cycles results in a larger relative difference, while aligning cycle lengths reduces the relative difference.

	Breaking cycles			**Aligning cycle lengths**		
Datasets	**Different sizes**	**Similar size**	**Δ relative difference**	**Different sizes**	**Similar size**	**Δ relative difference**
BBBP	0.00545	0.00152	↑ 50%	0.00565	0.00211	↓ 41%
BACE	0.00420	0.00148	↑ 6%	0.00417	0.00176	↓ 41%
NCI1	0.00547	0.00173	↑ 53%	0.00566	0.00242	↓ 31%
NCI109	0.00548	0.00174	↑ 52%	0.00568	0.00245	↓ 31%
PROTEINS	0.00670	0.00212	↑ 31%	0.00763	0.00302	↓ 41%

of Section 3.2, in the center of Figure 2, we plot the pairwise graph distance based on eigenvalue distributions of graphs with different sizes after breaking cycles. The blue band along the diagonal of the plot becomes darker and narrower, suggesting a larger spectrum difference between small and large graphs and a stronger correlation between the spectrum and graph size. To evaluate the effects quantitatively, we further compute the changes in the relative spectrum difference and present the results in Table 2. These results indicate that failing to consider cycle information can lead to more significant differences in the spectrum between graphs of varying sizes, potentially causing GNNs to struggle with generalizing effectively to larger graphs.

3.3.2 Cycle Length & Spectrum: Aligning Cycle Lengths

In Section 3.3.1, we showed that cycle information is crucial for GNNs to achieve size generalizability. We now further explore what cycle information helps reduce the spectrum difference between small and large graphs. To facilitate our exploration, we divide each real-world dataset into two subsets: one subset contains small graphs, and the other subset containts graphs of significantly larger size. Further details regarding this dataset split can be found in Appendix B. Using this dataset split, we observe a significant difference in the cycle lengths for small and large graphs (Appendix D.1). As described in Appendix D.1, to reduce that difference, we align the average cycle lengths between small and large graphs by randomly inserting redundant nodes to increase the cycle lengths in small graphs. The rightmost heatmap in Figure 2 shows how the correlation of eigenvalue distributions and graph size changes after aligning cycle lengths. We observe a lighter blue band along the diagonal, which suggests a weaker correlation between the spectrum and graph size. Furthermore, Table 2 quantitatively presents the changes in the relative spectrum difference between small and large graphs. We observe that aligning cycle lengths results in reduced disparities in the spectrum between graphs of different sizes. This indicates that GNNs capable of generalizing across varying cycle lengths may exhibit better size generalizability. In Appendix D.2, we show that our approach of aligning the cycle lengths is more effective at reducing the spectrum disparities than randomly adding the same number of nodes and edges.

4 METHODOLOGY: PROPOSED MODEL-AGNOSTIC STRATEGIES FOR GNNS

Our findings in Section 3 suggest that GNNs with better ability to identify cycles and generalize over cycle lengths may have better size generalizability on biological graphs. However, recent work (8) has found that most GNNs are incapable of learning cycle information. Inspired by these, we propose three model-agnostic strategies to help GNNs learn the cycle information.

4.1 STRATEGY 1: SIZE-INSENSITIVE ATTENTION

One way to incorporate cycle information into GNNs is by encoding it in the features and leveraging them within the attention mechanism to guide the learning process. Specifically, for each graph $\mathcal{G}$, we obtain its cycle basis $\mathcal{C}$. Then, for each node $v_i \in \mathcal{G}$, we calculate the average length of the cycle basis to which it belongs:

$$\ell_i = \begin{cases} \frac{\sum_{j=1}^{|\mathcal{C}|} |\mathcal{C}_j| \cdot \mathbb{1}_{\{v_i \in \mathcal{C}_j\}}}{\sum_{j=1}^{|\mathcal{C}|} \mathbb{1}_{\{v_i \in \mathcal{C}_j\}}} & \text{if } v_i \text{ belongs to some cycles} \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $\mathbb{1}_{\{\text{condition}\}}$ is an indicator function that outputs 1 when the condition is met and 0 otherwise. Then we manually construct a two-dimensional feature vector for each node v_i based on its associated cycle information:

$$\mathbf{c}_i = [\mathbb{1}_{\{v_i \in \text{cycle}\}}, \ell_i]. \tag{2}$$

We use the structural feature matrix $\mathbf{C} = [\mathbf{c}_1; \ldots; \mathbf{c}_N] \in \mathbb{R}^{N \times 2}$ for attention. Since attention weights often diminish with increasing graph size due to the utilization of `Softmax`, we propose scaling the attention weights by the graph size and employing `Global_max` as the global pooling operation to mitigate the impact of graph size. Mathematically, our final graph representation is given by:

$$\mathbf{k} = \texttt{Softmax}(\mathbf{C}\mathbf{w}_A^\top) \cdot N, \quad \mathbf{x}^{\mathcal{G}} = \texttt{Global\_max}(\texttt{Diag}(\mathbf{k}) \cdot \mathbf{X}^{(\text{Last})}), \tag{3}$$

where $\mathbf{w}_A^\top$ is a learnable vector, and $\texttt{Diag}(\cdot)$ creates a diagonal matrix using the vector as its elements. We note that when we train on small graphs and test on large graphs, some structural features may not be seen in the training, such as certain cycle lengths in the large graphs. We rely on the attention mechanism to generalize to those cases.

4.2 STRATEGY 2: SELF-SUPERVISED AUXILIARY TASK

Our second proposed strategy utilizes a self-supervised auxiliary task to enhance the node representations with cycle-related information. The auxiliary task is to predict whether a node belongs to a cycle. We do not utilize cycle lengths as labels because large test graphs may have cycle lengths not present in the training data. Formally, let $X^{(\text{Last})}$ denote the node representations obtained after the last graph convolution. The conventional way of learning is to directly apply a pooling operator and then minimize the loss function for label supervision as below:

$$\mathcal{L}_{\text{label}} = \texttt{CrossEntropy}(\texttt{Linear}(\texttt{Pooling}(\mathbf{X}^{(\text{Last})}), y^{\mathcal{G}})), \tag{4}$$

where $y^{\mathcal{G}}$ is the ground truth label for the graph. In this approach, we incorporate an additional loss that aims to diffuse cycle-related information into the node representations through supervision:

$$\mathcal{L}_{\text{cycle}} = \texttt{CrossEntropy}(\texttt{MLP}(\mathbf{X}^{(\text{Last})}), \mathbf{y}_{\text{cycle}}), \tag{5}$$

where $\mathbf{y}_{\text{cycle}}$ is an indicator vector denoting whether a node belongs to a cycle. To sum up, the total loss is given by:

$$\mathcal{L} = \mathcal{L}_{\text{label}} + \lambda \mathcal{L}_{\text{cycle}}, \tag{6}$$

where λ is a hyperparameter tuned via cross-validation.

4.3 STRATEGY 3: AUGMENTATION

This approach aims to reduce the discrepancy between small and large graphs through direct augmentation for the small training graphs. We augment the training graphs by extending the cycle lengths such that the average cycle length and standard deviation align with those in large graphs. To achieve this augmentation, we use the algorithm detailed in Appendix D.1. Additionally, we replicate the features from the nodes with the lowest degrees in the same cycle to populate the features for the newly added nodes. Last, we feed the augmented training graphs to GNNs for graph classification.

5 EXPERIMENTS

In this section, we conduct extensive experiments to evaluate our proposed strategies. We aim to answer the following research questions: (**RQ1**) **Effectiveness of cycle-aware strategies**: Do cycle-aware strategies effectively enhance the size generalizability of GNNs? And if so, which one proves to be more effective? (**RQ2**) **Comparison with other baselines**: How does the most efficient cycle-aware strategy compare to other baseline strategies?

5.1 SETUP

Dataset. We use the same biological datasets as in Section 3.2.

Data Preprocessing and Important Training Details. In order to analyze size generalizability, we have four splits for each dataset: train, validation, small_test, and large_test, where large_test contains graphs with significantly larger sizes. We generate the splits as follows. First, we sort the samples in the dataset by their size. Next, We take the train, validation, and small_test split from the 50% smallest graphs in the dataset. An intuitive way of getting the large_split is to take the top k largest graphs. However, doing so would result in severe label skewness (class imbalances) between the small_test and the large_test as demonstrated by Table 5 in Appendix C. To avoid such a severe label shift, we select the same number of graphs per class as in the small_test subset, starting from the largest graph within each class. This way guarantees that the label distribution between small_test and large_test is the same, while ensuring that the graph size in the latter is 2-10 times larger. Nevertheless, the smallest 50% samples still have significant **class imbalance**. To address this issue, we use upsampling during training throughout the experiments, and we use **F1** as the metric to measure the model performance. More details about data preprocessing, hyperparameters, and training can be found in Appendix C and Appendix F.

Baselines. We use six neural network models as our GNN backbones. Each model consists of three layers, with a global max pooling layer employed in the final layer. The baseline models are: Multilayer Perceptron (MLP), GCN (17), GAT (28), GIN (34), FAGCN (5), and GNNML3 (2). We integrate six model-agnostic strategies with these GNN backbones. For our three proposed strategies, we use (1) `+SSL` to denote the use of self-supervised auxiliary task, (2) `+AugCyc` to denote the use of cycle-length augmentation, and (3) `+SIA` to denote the use of structural-based size-insensitive attention. We also compare with other model-agnostic strategies: (4) thresholded SAG pooling (18; 20) (`+SAGPool`), an attention-based pooling method effective for generalizing to large and noisy graphs; (5) SizeShiftReg (6) (`+SSR`), a regularization based on the idea of simulating a shift in the size of the training graphs using coarsening techniques; (6) RPGNN (24) (`+RPGNN`), an expressive model for arbitrary-sized graphs; (7) two versions of CIGA (`+CIGAv1` & `+CIGAv2`) (4), a causal model good at handling out-of-distribution problems on graphs. Besides these model-agnostic strategies, our baselines also include an expressive model SMP (30) (`SMP`), which excels at the cycle detection task.

5.2 (RQ1) EFFECTIVENESS OF CYCLE-AWARE STRATEGIES

In this section, we aim to evaluate the effectiveness of our proposed cycle-aware strategies in enhancing the size generalizability of GNNs and determine the most effective strategy.

As mentioned in Section 2.2, we evaluate the size generalizability of GNNs by training them on small graphs and testing their graph classification performance on large graphs. Better size generalizability translates into better performance on large graphs. Table 3 showcases the size generalizability results for our three proposed cycle-aware strategies, which we evaluate across five distinct datasets and six different backbone models. Notably, to better compare different strategies, the last column gives the average improvements compared with the original model evaluated across all datasets.

First, all of our proposed strategies consistently lead to improvements in large test datasets without sacrificing the performance on small graphs. On average, these enhancements can reach up to 8.4%, affirming the effectiveness of cycle information in improving GNN size generalizability, as discussed in Section 3. Second, it is worth noting that cycle lengths provide more valuable information for enhancing size generalizability of GNNs. This is evident from the consistently better performance of the strategies `+AugCyc` and `+SIA` compared to the `+SSL` strategy on large test graphs, which solely predicts whether a node belongs to a cycle in the auxiliary task. Third, the simple attention-based

Table 3: Size generalizability evaluated by the graph classification performance on small and large test graphs. The performance is reported by the average F1 scores and its standard deviation. The rightmost column denotes the average improvements compared with the original performance using the same backbone model across five different datasets. The largest average improvement within the same model and small/large category is highlighted in orange. **All strategies enhance GNNs' size generalizability, with `+SIA` surprisingly emerging as the most effective method.**

Datasets	BBBP		BACE		PROTEINS		NCI1		NCI109		Avg Improv	
Models	Small	Large	Small	Large	Small	Large	Small	Large	Small	Large	Small	Large
MLP	90.36±0.71	55.61±3.37	61.06±5.79	21.06±7.89	36.15±2.28	21.55±1.34	36.43±3.89	3.36±2.87	35.87±4.23	4.65±3.72	-	-
MLP+SSL	90.90±1.76	62.56±5.48	58.57±8.85	23.01±11.95	35.00±2.8	20.88±1.64	34.71±1.33	2.86±0.78	37.29±4.69	6.34±4.78	-0.68	+1.88
MLP+AugCyc	90.72±2.70	57.86±4.74	59.88±7.33	26.50±14.97	37.29±0.0	22.22±0.0	36.98±2.29	2.64±2.1	40.59±3.86	8.41±3.71	+1.12	+2.28
MLP+SIA	90.38±1.05	62.79±7.55	60.85±7.83	21.79±15.07	40.68±3.56	33.57±11.87	35.42±3.83	3.26±2.55	39.20±2.96	12.2±5.05	+1.33	+5.48
GCN	91.37±0.59	68.59±7.47	63.68±6.63	28.72±14.26	72.35±2.58	40.57±7.67	54.91±2.37	28.80±7.57	60.83±1.92	30.45±4.34	-	-
GCN+SSL	92.66±1.21	73.24±5.71	64.92±4.44	32.84±16.08	72.46±1.58	41.21±6.66	57.43±3.23	32.58±10.08	60.50±3.09	27.35±11.42	+0.97	+2.01
GCN+AugCyc	91.41±1.33	68.08±7.65	63.83±5.44	35.65±7.70	72.87±3.68	54.73±8.24	53.85±3.71	27.39±8.33	62.78±2.98	33.62±3.58	+0.32	+4.47
GCN+SIA	91.32±0.73	71.66±6.99	64.35±9.76	24.24±17.03	73.84±3.65	58.74±9.49	59.78±1.65	45.70±6.70	60.32±2.90	38.78±4.55	+1.29	+8.40
GAT	91.27±1.43	68.35±7.02	69.73±2.05	42.23±11.18	72.25±4.25	43.86±6.82	58.22±2.86	49.36±4.12	64.39±3.29	38.36±8.93	-	-
GAT+SSL	91.65±0.92	74.24±7.34	71.20±2.04	40.88±10.81	74.20±1.46	49.30±5.56	59.47±2.89	51.85±4.03	66.79±3.56	42.20±6.71	+1.49	+3.26
GAT+AugCyc	92.41±1.29	69.57±2.89	68.39±6.06	40.73±13.4	74.99±1.89	59.80±7.27	56.23±3.85	49.37±7.52	64.07±3.46	45.25±9.19	+0.05	+4.51
GAT+SIA	91.88±2.12	74.87±5.62	69.64±6.79	43.87±7.98	75.35±3.28	62.71±4.98	61.42±1.07	55.73±12.98	63.27±3.15	45.97±7.74	+1.14	+8.20
GIN	88.28±2.39	66.67±5.55	57.02±6.48	22.97±10.26	74.55±4.27	50.20±5.36	62.17±3.86	44.26±7.03	62.42±2.77	33.23±6.77	-	-
GIN+SSL	91.13±1.32	68.67±9.75	56.46±8.59	23.91±10.64	75.47±1.15	48.14±4.00	61.18±3.53	46.47±9.86	63.11±4.05	35.0±11.43	+0.58	+0.97
GIN+AugCyc	92.56±1.17	77.69±5.63	58.30±5.29	23.89±13.17	74.56±2.92	51.02±8.42	62.70±0.94	46.76±5.34	64.56±5.45	37.16±5.86	+1.65	+3.84
GIN+SIA	92.70±0.45	75.99±4.74	61.30±6.77	24.42±16.37	74.88±4.24	51.36±7.76	62.83±1.07	42.82±8.92	63.00±4.24	41.65±4.19	+2.05	+3.78
FAGCN	90.58±1.72	64.93±7.62	62.96±2.12	24.65±11.71	70.03±5.20	42.34±6.61	43.51±4.29	10.16±7.80	55.78±3.5	22.65±12.87	-	-
FAGCN+SSL	91.55±2.51	67.56±5.48	64.67±3.88	35.46±16.52	66.97±1.75	48.06±8.33	46.42±6.08	12.11±5.39	56.04±4.29	23.99±10.57	+0.56	+4.49
FAGCN+AugCyc	91.30±2.26	71.44±6.45	57.68±3.38	26.41±23.39	68.85±16.12	44.39±16.89	39.48±4.99	10.98±5.45	55.30±3.46	24.59±9.19	-2.05	+2.62
FAGCN+SIA	90.17±2.83	74.65±9.13	62.40±3.36	30.35±13.48	71.30±5.79	48.94±10.62	46.95±5.71	10.99±7.50	52.82±6.28	19.08±5.32	+0.16	+3.86
GNNML3	92.01±1.56	64.18±6.99	62.31±4.90	32.94±12.86	71.59±3.5	40.74±15.0	63.73±4.67	51.75±9.05	59.39±3.76	33.80±11.19	-	-
GNNML3+SSL	92.96±1.54	64.18±8.62	65.65±5.69	31.78±12.72	74.41±3.21	56.81±3.49	63.91±3.34	48.84±10.07	61.01±2.44	35.13±9.49	+1.78	+2.67
GNNML3+AugCyc	91.38±2.92	69.82±5.51	63.36±2.78	32.59±10.32	70.54±5.00	38.79±5.21	62.30±3.27	55.57±11.73	58.18±3.17	41.30±14.75	-0.65	+2.93
GNNML3+SIA	92.70±0.81	70.43±6.36	64.57±2.72	37.73±7.68	69.32±3.79	48.94±10.62	63.91±5.81	48.85±12.11	61.58±3.98	49.70±17.85	+0.61	+6.45

strategy `+SIA` achieves the best overall performance improvements in all scenarios. On average, the `+SIA` strategy enhances both in-distribution and out-of-distribution generalization, while `+SSL` and `+AugCyc` excel particularly in out-of-distribution generalization. Additionally, `+SIA` achieves the highest average improvements on large test graphs. We attribute this to the challenges GNNs face in effectively learning cycle information, as shown in recent literature (8). Furthermore, in Appendix E, we conduct an ablation study further demonstrating that the attention mechanism, without considering the cycle information, cannot improve the size generalizability of GNNs.

5.3 (RQ2) Comparison with Other Baselines

We now compare our best-performing strategy, `+SIA`, with other approaches and present their respective graph classification performances in Table 3. We find that `+SIA` consistently achieves the best performance compared with other baseline methods. While `+SAGPool`, `+RPGNN`, `+SSR`, and `+CIGA` also enhance the size generalizability of GNNs, the improvements are less pronounced than those of `+SIA`. Additionally, it's worth noting that `+SAGPool` and `+CIGAv1` show sensitivity to hyperparameters. While not explicitly designed for size generalizability, the expressive model `SMP` demonstrates strong performance on the BBBP dataset due to its cycle detection ability, validating our empirical insights.

6 Related Work

Literature on GNNs' size generalization presents conflicting views. Empirical studies highlight GNNs' size generalizability in physics simulation (26) and algorithmic reasoning (35). Levie et al. (21) theoretically showed that spectral GNNs robustly transfer between graphs with varied sizes when discretizing the same underlying space. Meanwhile, some arguments suggest that GNNs may require additional assistance to achieve size generalizability. For instance, Yan et al. (36) and Velivckovic et al. (29) found that neural networks effectively generalize to larger graphs than those used in training when attention weights are properly supervised. Conversely, some argue that GNN performance degrades with size shifts between training and test data, leading to the proposal of various models to mitigate this challenge. For instance, Yehudai et al. (38) argued that this performance degradation can be attributed to the changes in the local degree patterns. Knyazev et al. (18) found that using attention with proper thresholding can improve the size generalizability of GNNs. Buffelli et al. (6) simulated a size shift in the training graphs via graph coarsening and proposed a regularization that makes the

Table 4: Size generalizability evaluated with other baselines, following the same rule as in Table 3. +SIA consistently and significantly outperforms other strategies regarding size generalizability.

Datasets	BBBP		BACE		PROTEINS		NCI1		NCI109		Avg Improv	
Models	**Small**	**Large**	**Small**	**Large**	**Small**	**Large**	**Small**	**Large**	**Small**	**Large**	**Small**	**Large**
MLP	90.36±0.71	55.61±3.37	61.06±5.79	21.06±7.89	36.15±2.28	21.55±1.34	36.43±3.89	3.36±2.87	35.87±4.23	4.65±3.72	-	-
MLP+SAGPool	89.42±5.84	52.52±14.98	65.82±9.42	13.04±5.02	42.54±12.82	20.84±7.84	45.41±13.21	12.32±15.78	42.18±9.88	15.35±12.31	+5.10	+1.57
MLP+RPGNN	90.44±1.03	55.10±7.64	57.80±8.53	21.26±8.20	45.60±2.69	20.71±1.41	35.34±4.35	13.45±25.12	38.60±2.37	10.44±18.77	+1.58	+2.95
MLP+SSR	91.07±0.67	57.02±9.04	60.35±5.36	25.42±4.02	37.15±2.48	22.77±4.28	34.42±1.89	1.27±0.63	38.68±4.70	6.96±4.65	+0.36	+1.44
MLP+CIGAv1	88.15±1.05	54.14±6.07	58.18±10.81	24.99±14.41	34.68±4.23	18.23±2.50	33.52±7.85	8.93±7.48	32.79±1.20	3.19±0.01	-2.51	+0.65
MLP+CIGAv2	88.08±3.84	61.70±13.15	57.46±4.71	18.43±16.73	38.28±4.23	24.87±5.62	35.77±7.55	4.22±6.89	38.45±3.68	4.55±1.28	-0.37	+1.51
MLP+SIA	90.38±1.05	62.79±7.55	60.85±7.83	21.79±15.07	40.68±3.56	33.57±11.87	35.42±3.83	3.26±2.55	39.20±2.96	12.2±5.05	+1.33	+5.48
GCN	91.37±0.59	68.59±7.47	63.68±6.63	28.72±14.26	72.35±2.58	40.57±7.67	54.91±2.37	28.80±7.57	60.83±1.92	30.45±4.34	-	-
GCN+SAGPool	92.05±3.95	67.06±5.18	57.59±6.65	42.74±14.7	68.75±5.09	32.98±3.26	58.56±8.30	38.73±20.94	62.87±16.7	30.93±10.59	-0.66	+3.06
GCN+RPGNN	92.27±0.81	68.69±6.58	63.70±1.06	33.86±13.91	74.74±3.75	24.61±10.08	58.88±2.03	34.68±10.77	63.10±1.86	39.69±5.88	+1.91	+0.88
GCN+SSR	91.19±1.14	68.15±6.38	66.01±2.51	31.64±9.96	73.51±2.91	43.33±5.19	59.60±2.62	35.01±7.13	59.78±2.71	33.11±6.44	+1.39	+2.82
GCN+CIGAv1	90.55±1.32	66.55±5.80	66.66±5.72	28.51±8.64	72.64±1.81	54.67±6.08	58.52±4.88	40.82±11.14	59.09±3.50	25.82±7.81	+0.86	+3.85
GCN+CIGAv2	89.45±3.60	69.71±8.20	65.02±1.80	35.42±12.36	72.15±3.86	60.12±6.84	57.89±3.74	35.42±10.75	58.12±5.37	28.51±10.10	-0.10	+6.41
GCN+SIA	91.32±0.73	71.66±6.99	64.35±9.76	24.24±17.03	73.84±3.65	58.74±9.49	59.78±1.65	45.70±6.70	60.32±2.90	38.78±4.55	+1.29	+8.40
GAT	91.27±1.43	68.35±7.02	69.73±2.05	42.23±11.18	72.25±4.25	43.86±6.82	58.22±2.86	49.36±4.12	64.39±3.29	38.36±8.93	-	-
GAT+SAGPool	89.90±2.15	60.39±17.18	66.10±6.44	46.40±15.45	73.85±8.60	38.60±6.34	55.25±1.43	52.71±3.02	65.32±3.42	43.20±19.94	-1.09	-0.17
GAT+RPGNN	91.76±1.69	65.85±5.37	69.97±2.17	39.27±13.50	72.89±3.35	38.49±6.14	59.31±5.51	58.18±5.76	65.52±1.94	44.15±5.76	-0.75	+0.76
GAT+SSR	91.98±0.66	74.83±4.35	66.03±3.83	41.41±11.8	74.72±3.51	44.81±8.59	60.68±1.95	49.64±5.26	66.73±1.65	41.14±4.41	+0.86	+1.93
GAT+CIGAv1	89.53±1.51	67.35±8.74	67.18±5.12	39.88±18.64	73.28±4.87	48.56±5.42	59.52±3.27	54.35±11.85	66.82±2.03	50.62±4.85	+0.09	+3.72
GAT+CIGAv2	90.92±1.43	72.08±7.09	66.57±4.91	40.93±19.35	74.68±7.54	60.88±3.48	56.88±24.40	54.28±22.63	66.78±3.20	52.62±7.98	-0.01	+7.73
GAT+SIA	91.88±2.12	74.87±5.62	69.64±6.79	43.87±7.98	75.35±3.28	62.71±4.98	61.42±1.07	55.73±12.98	63.27±3.15	45.97±7.74	+1.14	+8.20
GIN	88.28±2.39	66.67±5.55	57.02±6.48	22.97±10.26	74.55±4.27	50.20±5.36	62.17±3.86	44.26±7.03	62.42±2.77	33.23±6.77	-	-
GIN+SAGPool	91.56±1.32	71.20±11.86	62.22±7.45	26.17±17.98	68.73±13.46	35.77±21.54	65.50±4.50	45.66±3.56	59.29±5.83	44.64±9.53	+0.57	+1.22
GIN+RPGNN	89.59±1.33	69.23±8.05	57.23±7.07	16.28±9.31	71.63±5.85	45.11±15.98	61.59±4.12	48.86±5.88	62.27±2.33	44.04±7.06	-0.43	+1.24
GIN+SSR	89.00±1.77	68.84±6.01	59.83±2.34	21.28±19.26	72.46±2.86	55.46±15.95	62.54±1.30	48.73±8.62	61.05±3.37	35.63 ±7.29	+0.09	+2.52
GIN+CIGAv1	91.01±1.59	74.85±10.41	60.58±7.15	23.78±17.58	75.32±16.25	54.83±9.87	61.94±1.08	45.85±5.83	61.55±12.55	35.88±7.77	+1.19	+3.57
GIN+CIGAv2	90.66±1.72	75.80±14.30	63.02±1.80	22.42±12.36	73.25±5.42	53.35±8.76	64.42±5.35	45.37±5.12	59.52±4.68	38.42±5.68	+1.29	+3.61
GIN+SIA	92.70±0.45	75.99±4.74	61.30±6.77	24.42±16.37	74.88±4.24	51.36±7.76	62.83±1.07	42.82±8.92	63.00±4.24	41.65±4.19	+2.05	+3.78
FAGCN	90.58±1.72	64.93±7.62	62.96±2.12	24.65±11.71	70.03±5.20	42.34±6.61	43.51±4.29	10.16±7.80	55.78±3.5	22.65±12.87	-	-
FAGCN+SAGPool	88.08±7.26	62.67±15.01	62.78±4.28	31.50±12.13	72.41±4.85	50.09±16.49	45.66±4.51	11.21±2.60	58.04±22.06	15.43±4.69	+0.28	+1.23
FAGCN+RPGNN	90.43±2.58	69.58±11.71	61.00±51.91	20.94±12.62	68.71±3.58	43.58±12.21	44.55±5.82	12.22±5.95	57.03±1.08	21.86±13.32	-0.23	+0.69
FAGCN+SSR	88.95±2.16	70.12±9.49	64.12±2.57	22.37±7.80	67.92±3.00	42.27±8.69	47.47±2.77	14.69±8.22	55.66±3.37	22.35±9.35	+0.25	+1.41
FAGCN+CIGAv1	91.08±1.48	69.29±3.35	62.66±5.72	28.51±8.64	68.58±6.19	57.79±10.45	40.93±10.69	9.45±11.16	48.07±5.70	15.11±5.82	+1.19	+3.57
FAGCN+CIGAv2	92.08±1.70	68.37±9.67	60.37±4.65	22.29±15.00	69.45±4.97	60.28±12.24	44.27±6.74	15.88±10.25	50.25±4.44	16.22±7.74	-1.29	+3.66
FAGCN+SIA	90.17±2.83	74.65±9.13	62.40±3.36	30.35±13.48	71.30±5.79	48.94±10.62	46.95±5.71	10.99±7.50	52.82±6.28	19.08±5.32	+0.16	+3.86
GNNML3	92.01±1.56	64.18±6.99	62.31±4.90	32.94±12.86	71.59±3.5	40.74±15.0	63.73±4.67	51.75±9.05	59.39±3.76	33.80±11.19	-	-
GNNML3+SAGPool	89.62±3.84	63.25±27.87	59.35±7.27	37.30±12.49	65.67±7.11	34.79±20.12	65.34±1.94	54.29±3.58	60.38±7.23	46.61±15.23	-1.73	+2.73
GNNML3+RPGNN	92.57±1.45	72.30±9.54	61.85±3.67	27.54±12.03	70.48±2.46	38.61±14.61	64.60±2.14	50.25±8.65	60.22±2.40	36.88±15.93	+0.14	+0.43
GNNML3+SSR	91.86±1.30	69.96±5.50	64.95±2.82	26.56±4.68	74.33±1.85	49.02±7.42	63.19±4.35	54.30±10.33	63.27±2.74	45.74±11.47	+1.41	+4.43
GNNML3+CIGAv1	91.25±4.67	72.80±5.88	64.23±2.10	34.95±15.92	73.89±3.73	52.01±12.59	61.89±4.53	45.25±7.80	60.95±3.44	38.80±12.36	+0.64	+4.08
GNNML3+CIGAv2	89.61±1.46	67.17±5.94	64.19±5.36	27.28±12.79	75.07±2.64	54.50±9.29	60.28±4.56	46.25±10.42	60.60±1.16	42.19±12.61	+0.14	+2.80
GNNML3+SIA	92.70±0.81	70.43±6.36	64.57±2.72	37.73±7.68	69.32±3.79	48.94±10.62	63.91±5.81	48.85±12.11	61.58±3.98	49.70±17.85	+0.61	+6.45
SMP	92.30±2.62	80.25±5.98	61.32±5.69	28.71±6.55	76.87±1.90	45.69±15.96	51.09±6.39	22.98±16.26	49.15±6.92	30.73±11.30	-	-

model robust to the shift. Bevilacqua et al. (4) used a causal model to learn approximately invariant representations that better extrapolate between train and test data. Chen et al. (7) utilized structural causal models for robust out-of-distribution generalization in graph data through invariant subgraph identification and label prediction. Chu et al. (9) proposed a Wasserstein barycenter matching (WBM) layer to address the slow uncontrollable convergence rate w.r.t. graph size. Zhou et al. (41) studied the size OOD problem in the task of link prediction. Ji et al. (16) curated OOD datasets for AI-aided drug discovery. Our study stands out as the first to utilize spectral analysis to characterize the types of size-induced distribution shifts, shedding light on the underlying causes that hinder GNNs from effectively generalizing to large graphs. Some expressive models also exhibit robustness in size generalization. Murphy et al. (24) proposed an expressive model-agnostic framework that learns graph representations invariant to graph isomorphism given variable-length inputs. Clement et al. (30) proposed an expressive graph neural network that performs well on difficult structural tasks, such as cycle detection and diameter computation. Our study validates that expressive models excelling in cycle-related tasks demonstrate good size generalizability.

7 Conclusion

In conclusion, our work extensively characterizes size-induced distribution shifts and evaluates their impact on GNNs' generalizability to significantly larger test graphs compared to the training set. Spectral analysis on real-world biological data reveals a strong correlation between graph spectrum and size, which hinders GNNs' size generalization. We identify the pivotal role of cycle-related information in reducing spectral differences between small and large graphs. Motivated by these findings, we introduce three model-agnostic strategies—self-supervision, augmentation, and size-insensitive attention—to enhance GNNs' size generalizability. Empirical results show that all three strategies improve GNNs' size generalizability, with +SIA being the most effective. This research provides valuable insights for enhancing GNN generalization across varying graph sizes.