

# Higher-Order Graph Convolutional Network with Flower-Petals Laplacians on Simplicial Complexes

Yiming Huang<sup>1,2,\*</sup>, Yujie Zeng<sup>1,2,\*</sup>, Qiang Wu<sup>3,†</sup>, Linyuan Lü<sup>4,1,2,†</sup>

<sup>1</sup>Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China

<sup>2</sup>Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China

<sup>3</sup>Research Institute of Electronic Science and Technology, University of Electronic Science and Technology of China

<sup>4</sup>School of Cyber Science and Technology, University of Science and Technology of China

{yiming\_huang, yujie\_zeng}@std.uestc.edu.cn, {qiang.wu, linyuan.lv}@uestc.edu.cn

## Abstract

Despite the recent successes of vanilla Graph Neural Networks (GNNs) on various tasks, their foundation on pairwise networks inherently limits their capacity to discern latent higher-order interactions in complex systems. To bridge this capability gap, we propose a novel approach exploiting the rich mathematical theory of simplicial complexes (SCs) - a robust tool for modeling higher-order interactions. Current SC-based GNNs are burdened by high complexity and rigidity, and quantifying higher-order interaction strengths remains challenging. Innovatively, we present a higher-order Flower-Petals (FP) model, incorporating FP Laplacians into SCs. Further, we introduce a Higher-order Graph Convolutional Network (HiGCN) grounded in FP Laplacians, capable of discerning intrinsic features across varying topological scales. By employing learnable graph filters, a parameter group within each FP Laplacian domain, we can identify diverse patterns where the filters' weights serve as a quantifiable measure of higher-order interaction strengths. The theoretical underpinnings of HiGCN's advanced expressiveness are rigorously demonstrated. Additionally, our empirical investigations reveal that the proposed model accomplishes state-of-the-art performance on a range of graph tasks and provides a scalable and flexible solution to explore higher-order interactions in graphs. Codes and datasets are available at <https://github.com/Yiminghh/HiGCN>.

## 1 Introduction

Graphs are ubiquitous in representing irregular relations in various scenarios. However, they are inherently constrained to modeling pairwise interactions exclusively (Battiston et al. 2020). Many empirical systems display group interactions, going beyond pairwise connections, such as social systems (Centola 2010), neuronal networks (Ganmor, Segev, and Schneidman 2011), and ecological networks (Grilli et al. 2017). However, such higher-order interactions can hardly be modeled or approximated by pairwise graphs. In addition, it is still elusive how to quantify the higher-order interaction strength, although many studies have demonstrated its existence (Battiston et al. 2021).

\*These authors contributed equally.

†Corresponding author.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Graph neural networks (GNNs) can exploit the features and topology of graphs simultaneously, thereby triggering a wide-spread research interest and endeavor in various graph learning tasks such as recommender systems and new drug discovery. In particular, spectral GNNs have been widely recognized for their rigorous mathematical theory. Nevertheless, pairwise-graph-based GNNs fail to capture latent higher-order interactions prevalent in empirical systems, and their expressive power was proved to be upper bounded by Weisfeiler-Lehman (WL) test (Xu et al. 2019).

Simplicial complexes (SCs) and hypergraphs have emerged to study higher-order interactions beyond conventional pairwise descriptors (Battiston et al. 2020). While hypergraph learning has made fruitful progress (Gao et al. 2022), it typically ignores relations within the hyperedges, and the construction of hypergraphs is often under-optimized. The simplicial description is another potent tool with elegant mathematical theories to draw from, paving a middle ground between graphs and hypergraphs. It has been found that SCs play a vital role in social contagion, synchronization, brain network analysis, etc.

Deep learning facilitated simplicial complex theory is a fresh perspective and a promising research field. Several simplicial GNNs have been proposed by simply replacing the graph Laplacian with the Hodge Laplacian (Schaub et al. 2020). A simplicial WL test is proposed along with its neural version MPSN (Bodnar et al. 2021) based on the adjacency relations that Hodge theory defines. MPSN is proved to be more powerful than vanilla GNNs under ideal conditions, implying the potential of extending graph representation learning to SCs.

In summary, pairwise GNNs fail to capture latent group interactions prevalent in complex systems, and the expressive power of such models was proved to be upper bounded by the WL-test. As an emerging and promising research field, simplicial GNNs have initially shown their potential to outperform pairwise GNNs. However, existing models are limited by their high complexity and low flexibility.

In this paper, we introduce a novel higher-order flower-petals (FP) representation based on two-step random walk dynamics (Zeng et al. 2023b) between the flower core and petals. This representation enables us to incorporate interactions among simplices of various orders into graph learn-

ing. Higher-order graph convolutional network (HiGCN) is then proposed by employing learnable and tailored convolutional filters (group of parameters) in different FP Laplacian domains. The learnable filters can learn arbitrary shapes and deal with high and low-frequency parts of the simplicial signals adaptively. Hence, the proposed HiGCN model can learn the simplex patterns of disparate classes and higher-order structures simultaneously. Moreover, the filters' weights in different orders can quantify the higher-order interaction strength, contributing to a deeper understanding of higher-order mechanisms in complex systems. We also interpret HiGCN from the message-passing perspective and theoretically demonstrate its superior expressive power. Numerical experiments on various graph tasks further pinpoint that the proposed model has outperformed state-of-the-art (SOTA) methods.

**Main contributions.** To summarise, we construct an innovative higher-order flower-petals (FP) model and FP Laplacians from the random walk dynamics to capture interactions among simplices of different orders. We then propose a higher-order graph convolutional network (HiGCN) based on our FP Laplacians, which is demonstrated to have superior expressiveness in theory and significant performance gains in various empirical experiments. Furthermore, a data-driven strategy is employed to quantify the higher-order interaction strength. In general, our work is an important step towards advancing higher-order graph learning and understanding higher-order mechanisms.

## 2 Related Work

In this section, we briefly review related works on vanilla spectral GNNs and higher-order GNNs.

**Spectral GNNs.** Spectral GNNs are based on the graph Fourier transform (Shuman et al. 2013), which employs the graph Laplacian eigenbasis as an analogy of the Fourier transform. ChebNet (Defferrard, Bresson, and Vandergheynst 2016) employs Chebyshev polynomials to replace the convolutional core, while GCN (Kipf and Welling 2017) uses a first-order approximation of the convolution operator. By considering the relationship between GCN and PageRank, APPNP (Gasteiger, Bojchevski, and Günnemann 2019) is proposed via personalized PageRank. GPRGNN (Chien et al. 2020) leverages a learnable graph filter, exhibiting superiority in heterogeneous graph learning. The filter forms of some spectral GNNs are summarized in Table 1.

**Higher-order GNNs.** The crude simplification of complex interaction into pairwise will inevitably result in information loss. Higher-order GNNs, as extensions of vanilla GNNs, can be classified into different types according to their application scenarios, and spectral-based simplicial GNNs are in the limelight of this paper. The Hodge theory (Hatcher 2002) enables us to describe diffusion across simplices conveniently. Several simplicial GNNs, such as SNN (Ebli, Defferrard, and Spreemann 2020) and SCoNe (Roddenberry, Glaze, and Segarra 2021), simply replace the graph Laplacian with the Hodge  $p$ -Laplacian. SCNN

Model	Convolution Filter	Spectral
GCN	$(1 - \lambda)^K$	Graph
APPNP	$\sum_{k=0}^K \frac{\gamma^k}{1-\gamma} (1 - \lambda)^k$	Graph
GPRGNN	$\sum_{k=0}^K \gamma^k (1 - \lambda)^k$	Graph
ChebNet	$\sum_{k=0}^K \gamma^k \cos(k \arccos(1 - \lambda))$	Graph
SNN	$\lambda^K$	Hodge
SCoNe	$\lambda_{down}^K, \lambda_{up}^K$	Hodge
SCNN	$\sum_{k=0}^{K_1} \gamma_{d,k} \lambda_{down}^k + \sum_{k=0}^{K_2} \gamma_{u,k} \lambda_{up}^k$	Hodge
BScNets	$f(\lambda_1, \lambda_2, \dots, \lambda_P; \theta)^K$	Block Hodge
<b>HiGCN</b>	$\sum_{k=0}^K \gamma_{p,k} (1 - \lambda_p)^k,$ $p = 1, 2, \dots, P$	FP

Table 1: The filter forms of spectral GNNs.

(Yang, Isufi, and Leus 2022) employs flexible simplicial filters to process edge signals from lower and upper simplicial neighbors, respectively. BScNet (Chen, Gel, and Poor 2022) is introduced by replacing the graph Laplacian with the block Hodge Laplacian. Nevertheless, the Hodge theory is inherently constrained to modeling interactions between simplices within one order difference. As for spatial models, SGAT (Lee, Ji, and Tay 2022) constructs SCs from heterogeneous graphs and leverages upper adjacencies to pass messages between simplices. MPSN (Bodnar et al. 2021) is designed based on the simplicial WL-test with four types of adjacency relations. Generally, most simplicial GNNs can only leverage information from specific simplicial orders, missing the inherent advantages of SCs. Besides, it is computationally expensive to find all simplices (Bomze et al. 1999) and unnecessary to compute embeddings for redundant simplices in traditional tasks.

## 3 Preliminaries

The background knowledge required to present this work better is illustrated in this section. Let  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  denote an undirected pairwise graph with a finite node set  $\mathcal{V} = \{v_1, \dots, v_n\}$  and an edge set  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . Assume that  $|\mathcal{V}| = n, |\mathcal{E}| = n_1$ , and  $N(v)$  denotes the set of nodes adjacent to node  $v$  in  $\mathcal{G}$ , i.e.,  $N(v) = \{u \in \mathcal{V} | (v, u) \in \mathcal{E}\}$ . The nodes are associated with a node feature matrix  $X \in \mathbb{R}^{n \times d}$ , where  $d$  signifies the number of features per node.

**Definition 3.1** (Simplicial complexes, SCs). A simplicial complex  $\mathcal{K}$  is a finite collection of node subsets closed under the operation of taking nonempty subsets, and such a node subset  $\sigma \in \mathcal{K}$  is called a simplex (as illustrated in Figure 1).

SCs are a potent tool with a rich theoretical foundation upon algebraic and differential topology and geometry (Battiston et al. 2020). Instead of predominantly studying pairwise interactions, SCs facilitate the modeling of higher-order interactions and multi-node graph structures.

A node subset  $\sigma = [v_0, v_1, \dots, v_p] \in \mathcal{K}$  with cardinality  $p + 1$  is referred to as a  $p$ -dimensional simplex, termed  $p$ -simplex, and we denote the set of all such  $p$ -simplices as  $\mathcal{K}_p$  with  $|\mathcal{K}_p| = n_p$ . One can regard vertices

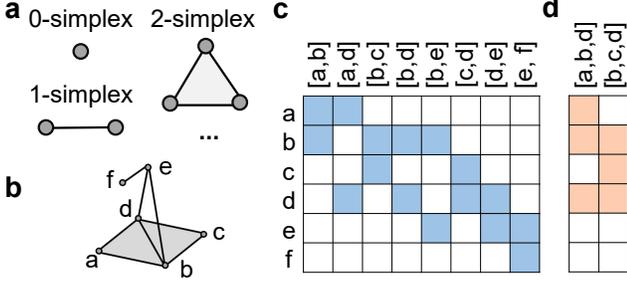


Figure 1: a shows several typical simplices and its collection forms SCs in b. Subfigures c and d visualize the higher-order incidence matrices  $\mathcal{H}_p$  for  $p = 1$  and 2, respectively.

as 0-simplices, edges as 1-simplices, “filled” triangles as 2-simplices, and so forth. A triangle  $[v_1, v_2, v_3] \in \mathcal{K}$  implies that its nonempty subsets, namely  $[v_1]$ ,  $[v_2]$ ,  $[v_3]$ ,  $[v_1, v_2]$ ,  $[v_1, v_3]$ , and  $[v_2, v_3]$ , are also in  $\mathcal{K}$ . Pairwise graphs can be viewed as 1-dimensional SCs, while higher-order SCs generally carry more structure information over pairwise graphs, which is critical and should not be omitted.

**Clique complex lifting transition**, as formally defined in Appendix B, extracts all cliques as simplices, converting pairwise graphs to SCs. This transformation enables the study of pairwise graphs from simplicial perspectives.

The boundary relation describes which simplices lie on the boundary of other simplices. We say  $\sigma$  is on the boundary of  $\tau$ , denoted as  $\sigma \prec \tau$ , iff  $\sigma \subset \tau$  and  $\dim(\sigma) = \dim(\tau) - 1$ . For example, edges  $[v_1, v_2]$ ,  $[v_1, v_3]$ , and  $[v_2, v_3]$  lie on the boundary of the 2-simplex  $[v_1, v_2, v_3]$ .

**Hasse diagram** is one of the most common representations of SCs, where each vertex corresponds to a simplex. The edges in the Hasse diagram are defined by the boundary relation, and there exists an edge connecting two vertices  $\sigma_1$  and  $\sigma_2$ , iff  $\sigma_1 \prec \sigma_2$ . The hasse diagram is highly expressive, and several simplicial GNNs (Bodnar et al. 2021; Hajij et al. 2022) are built precisely on the boundary relationships shown in the hasse diagram.

## 4 Methodology

We first introduce the higher-order flower-petals model for simplicial complex representation, which will subsequently be leveraged to construct our HiGCN model.

### 4.1 Flower-Petals Model

Hasse diagrams are valuable in studying SCs, but they are inherently constrained to modeling interactions for directly adjacent simplices and are computationally expensive to construct. Multiple transitions are required for information to pass between nodes and higher-order structures. Besides, the number of total simplices grows exponentially with the number of nodes in dense graphs. Computing embeddings for all higher-order structures can be costly and unnecessary for specific-level tasks. To address these challenges, we construct a novel higher-order representation, named the flower-petals (FP) model, and then introduce FP adjacency

and Laplacian matrices based on the higher-order random walk dynamics between the flower core and petals.

It can be simplified only to consider the interaction between 0-simplices and higher-order structures when tackling the most common tasks: node-level tasks. Hence, we construct a flower-petals model by simplifying the intermediate vertices on the Hasse diagram. Specifically, the flower-petals model consists of one core and several petals, see Figure 2, with interactions considered only between the core and petals. 0-simplices are placed in the flower core, and each flower petal involves simplices of the same order (larger than zero). The term  $p$ -petal is used to represent the petal containing  $p$ -simplices. Diverse and complex interactions exist between  $p$ -petal and the core, which can be unwrapped as a bipartite graph  $\mathcal{G}_p$ . Mathematically, the bipartite graph  $\mathcal{G}_p$  consists of two distinct vertex sets  $(\mathcal{V}, \mathcal{K}_p)$ , where  $\mathcal{V}$  represents the set of nodes contained in the flower core and  $\mathcal{K}_p$  comprised of simplices in the  $p$ -petal ( $p \geq 1$ ). If simplex  $\sigma (\in \mathcal{K}_p)$  contains node  $v (\in \mathcal{V})$ , then there exists an edge between their corresponding vertices in  $\mathcal{G}_p$ . The proposed flower-petals model prunes the information interaction rules between petals but is still extremely expressive and useful.

Inspired by incidence matrices in pairwise networks, we introduce higher-order incidence matrix  $\mathcal{H}_p \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{K}_p|}$  to describe the association between vertices in the core and  $p$ -simplices in the  $p$ -petal, with entry  $\mathcal{H}_p(v, \sigma) = 1$  indicating the vertex  $v$  is contained in the simplex  $\sigma (\in \mathcal{K}_p)$ . Visual representations are provided in Figure 1 c and d for clarity.

### 4.2 Flower-Petals Algebraic Description

Hodge Laplacian (Schaub et al. 2020; Hatcher 2002) is a fundamental tool in simplicial complexes. However, it can only describe interactions between simplices within one-order differences. To model interactions between different order simplices more flexibly, we introduce novel matricial descriptions for simplicial complexes based on the random walk dynamics between the flower core and petals.

The main idea of random walks is to traverse a graph starting from a single node or a set of nodes and get sequences of locations (Zeng et al. 2023a). We introduce the traditional random walk model in Appendix D. Walking on the bipartite graphs  $\mathcal{G}_p$  consists of two sub-steps: (I) upward walk and (II) downward walk.

The upward walk refers to the walk from nodes in the flower core to their corresponding simplices in the  $p$ -petal, while the downward walk proceeds in the opposite direction. Consider  $\pi(t) = (\pi_{v_1}(t), \dots, \pi_{v_n}(t))^T$ , whose item  $\pi_\sigma(t)$  encodes the probability for simplex  $\sigma$  to be occupied by a random walker at step  $t$ . In the upward walk process, information is transmitted from nodes to simplices and the probability of moving from vertex  $u$  to simplex  $\sigma$  is equal to  $\mathcal{H}_p(u, \sigma) / d_p(u)$ . The downward walk, i.e., petals-to-core walk, allows information to be transferred from simplices back to nodes. These two processes follow that  $\pi_\sigma(t-1) = \sum_u d_p(u)^{-1} \mathcal{H}_p(u, \sigma) \pi_u(t-2)$ , and  $\pi_v(t) = \sum_\sigma \delta_p(\sigma)^{-1} \mathcal{H}_p(v, \sigma) \pi_\sigma(t-1)$ , for upward and downward walks, respectively. Here,  $d_p(u) = \sum_{\sigma \in \mathcal{K}_p} \mathcal{H}_p(u, \sigma)$  denotes the degree of  $u$  on  $\mathcal{G}_p$ , and  $\delta_p(\sigma) = \sum_{v \in \mathcal{V}} \mathcal{H}_p(v, \sigma) =$

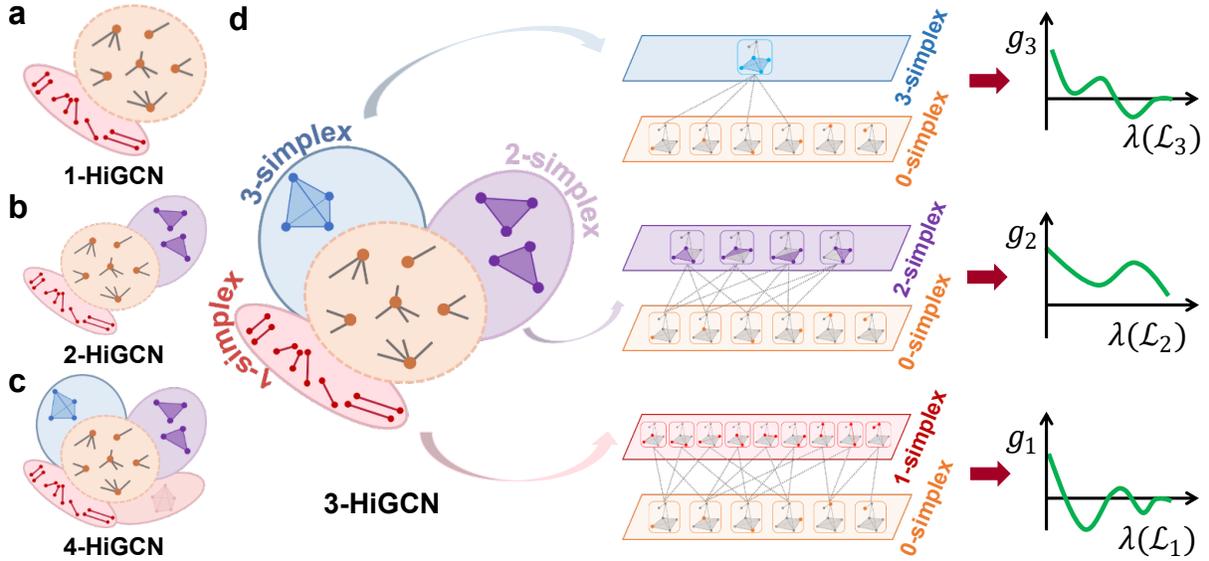


Figure 2: Visualization of the flower-petals model. Different HiGCN models employ different numbers of petals, with each petal containing simplices of identical order. a-d visualizes 1,2,3-HiGCN, respectively. The interaction between each petal and the flower core can be unwrapped as an individual bipartite graph. FP Laplacians are derived based on the random walk dynamics in the bipartite graphs, followed by various learnable convolution operations  $g_p$  on each FP Laplacian basis.

$p + 1$  represents the degree of  $p$ -simplex  $\sigma$  on  $\mathcal{G}_p$  ( $p \geq 1$ ).

The two-step walk (Zeng et al. 2023b) integrates both the upward and downward walks, allowing the information to be transmitted from the flower core and back through the petals. A complete two-step walk process follows that

$$\pi_v(t) = \sum_{\sigma} \frac{\mathcal{H}_p(v, \sigma)}{\delta_p(\sigma)} \sum_u \frac{\mathcal{H}_p(u, \sigma)}{d_p(u)} \pi_u(t-2). \quad (1)$$

We can further derive the matrix representation for the two-step walk as  $\pi(t) = \mathcal{H}_p D_{p,h}^{-1} \mathcal{H}_p^{\top} D_{p,v}^{-1} \pi(t-2)$ , where  $D_{p,v} = \text{diag}(d_p(v_1), \dots, d_p(v_n))$  and  $D_{p,h} = \text{diag}(\delta_p(\sigma_1), \dots, \delta_p(\sigma_{|\mathcal{K}|})) = (p+1)I$ . By multiplying  $D_{p,v}^{-1/2}$  from the left sides of this equation, we can obtain

$$D_{p,v}^{-1/2} \pi(t) = \left[ D_{p,v}^{-1/2} \mathcal{H}_p D_{p,h}^{-1} \mathcal{H}_p^{\top} D_{p,v}^{-1/2} \right] D_{p,v}^{-1/2} \pi(t-2). \quad (2)$$

Therefore, based on the two-step walk dynamic between the flower core and petals, we can define higher-order flower-petals (FP) adjacency matrices analogously to the reduced adjacency matrices (see Appendix D) as

$$\begin{aligned} \tilde{\mathcal{A}}_p &= D_{p,v}^{-1/2} \mathcal{H}_p D_{p,h}^{-1} \mathcal{H}_p^{\top} D_{p,v}^{-1/2} \\ &= \frac{1}{p+1} D_{p,v}^{-1/2} \mathcal{H}_p \mathcal{H}_p^{\top} D_{p,v}^{-1/2}. \end{aligned} \quad (3)$$

The Laplacian operator is crucial for the processing of relational data, and it bears resemblance to the Laplace-Beltrami operator in differential geometry. On the basis of the FP adjacency matrices, we can likewise define a series of higher-order FP Laplacian operators as  $\mathcal{L}_p = I - \tilde{\mathcal{A}}_p$ .

**Theorem 4.1.** *The flower-petals adjacency matrices  $\tilde{\mathcal{A}}_p$  and flower-petals Laplacian matrices  $\mathcal{L}_p$  are all symmetric positive semidefinite.*

It follows from Theorem 4.1 that  $0 \leq \lambda(\tilde{\mathcal{A}}_p), \lambda(\mathcal{L}_p) \leq 1$ . We defer the proof and further theoretical analysis of the spectral properties to Appendix A. Theorem 4.1 contributes to alleviate the numerical instability and exploding/vanishing gradients that may arise in the implementation of deep GNNs based on the FP Laplacians. The diverse FP Laplacian matrices capture the various connectivity relations of the simplicial complexes, where we can learn a series of diverse spectral convolution operators.

### 4.3 Higher-Order Graph Convolutional Network

The eigen decomposition  $\mathcal{L} = \Phi \Lambda \Phi^{\top}$  can be applied to the Laplacian matrix to obtain orthonormal eigenvectors  $\Phi = (\phi_1, \phi_2, \dots, \phi_n)$  and a diagonal matrix  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ . Then, for a graph signal  $x$ , the graph Fourier transform is defined as  $\Phi^{\top} x$ , where the eigenvectors act as the Fourier bases and the eigenvalues are interpreted as frequencies. The spectral convolution of signal  $x$  and filter  $g$  can then be formulated as

$$g \star x = \Phi \left( (\Phi^{\top} g) \odot (\Phi^{\top} x) \right) = \Phi g(\Lambda) \Phi^{\top} x. \quad (4)$$

Here, operator  $\odot$  presents the Hadamard product, and the filter  $g(\Lambda)$  applies  $g$  element-wisely to the diagonal entries of  $\Lambda$ , i.e.,  $g(\Lambda) = \text{diag}(g(\lambda_1), \dots, g(\lambda_n))$ . Note that spectral decomposition for large-scale networks can be computationally expensive. Therefore, one can approximate any graph filter using a polynomial filter with enough terms (Shuman et al. 2013). Consequently, the filter  $g$  is usually set to be a truncated polynomial  $g(\lambda) = \sum_{k=0}^K \gamma_k \lambda^k$  of order  $K$ . In this way, spectral decomposition is avoided.

We derive various FP Laplacian matrices based on the FP model, each representing different connectivity relations within SCs. Subsequently, we define different convolution

operations on each FP Laplacian basis as

$$g \star_p x = g_p(\mathcal{L}_p)x, \quad (5)$$

where graph filter  $g_p(\mathcal{L}_p) = \sum_{k=0}^K \gamma_{p,k} \mathcal{L}_p^k$  is composed of different learnable polynomial functions in each FP spectral domain. These learnable coefficients  $\gamma_{p,k}$  capture the contributions of different hop neighbors in each order.  $K$  is a hyperparameter denoting the largest hops of the simplices under consideration. We summarize some prevalent filter forms in Table 1. When processing a simplicial signal  $X \in \mathbb{R}^{n \times d}$  with  $d$  dimensional features, a more general form of spectral GNNs follows that  $Y = \rho(g(\mathcal{L})\varphi(X))$ . Here,  $\rho$  and  $\varphi$  are permutation-invariant functions.

To encode multi-scale higher-order information, the final prediction is obtained by concatenating results from different convolution operations as

$$Y_p = g_p(\mathcal{L}_p)\varphi_p(X), \quad Y = \rho\left(\|_{p=1}^P Y_p\right). \quad (6)$$

Here,  $\|$  concatenates the representation in different spectral domains. Besides, we simplify  $\rho$  and  $\varphi_p$  to linear functions as suggested by (Wang and Zhang 2022), resulting in

$$Y = \left\|_{p=1}^P \left( \sum_{k=0}^K \gamma_{p,k} \tilde{A}_p^k X \Theta_p \right) W. \quad (7)$$

Here,  $\gamma_{p,k}$ ,  $\Theta_p$ , and  $W$  are trainable parameters, and  $P$  is a hyperparameter indicating the highest order of the simplices under consideration. The model under  $P = \ell$  is denoted as  $\ell$ -HiGCN. Notably, the training process can be accelerated by precalculating  $\tilde{A}_p^k$ , which can be efficiently calculated between sparse matrices.

HiGCN facilitates the independent and flexible learning of filter shapes across disparate FP spectral domains rather than predetermining filter configurations. Consequently, it is adept at handling both high/low frequency and high/low order signal components in a versatile manner. Furthermore, we find that the filters' weights in different orders quantify the strength of the higher-order interactions, contributing to the understanding of higher-order mechanisms inherent within complex systems. Now, we proceed to elucidate the advantages of HiGCN from various perspectives.

**Expressive power.** We have developed the HiGCN model from a spectral perspective. The WL test provides a well-studied framework for unique node labeling, and an intrinsic theoretical connection has been uncovered between the WL test and message-passing-based GNNs (Xu et al. 2019). We extend this relation and propose a higher-order WL test, termed HWL, along with its simplified version SHWL. Detailed procedures for WL, HWL, and SHWL are elaborated in Appendix B. Furthermore, we revisit the HiGCN model from the message-passing perspective in Appendix B, offering an alternative interpretation that underscores the exceptional expressive power of our model.

**Theorem 4.2.** *SHWL with clique complex lifting is strictly more powerful than Weisfeiler-Lehman (WL) test.*

The proposed model can be interpreted as a neural version of the SHWL test where colors are replaced by continuous

feature vectors. Hence, Theorem 4.2 implies that HiGCN endows with greater expressive power than vanilla GNNs. See Appendix B for proof and detailed discussion.

**Relation to existing models.** HiGCN shows superiority over pairwise graph-based GCNs for exploiting higher-order information, and it generalizes spectral convolution operations on pairwise graphs, including GCN (Kipf and Welling 2017) and GPRGNN (Chien et al. 2020). On the other hand, HiGCN exhibits greater flexibility than certain Hodge Laplacian-based simplicial GCNs, such as SNN (Ebli, Defferrard, and Spreemann 2020) and SCNN (Yang, Isufi, and Leus 2022), overcoming the constraints of information exchange exclusively through boundary operators. Further derivation and discussion are presented in Appendix C.

**Symmetries.** It is a fundamental concept for understanding GNNs and their behavior. HiGCN has been demonstrated to exhibit equivariance with respect to relabeling of simplices, enabling it to exploit symmetries in SCs. Formal proofs and detailed discussions are deferred to Appendix E.

**Computational complexity.** A balance between performance and complexity can be achieved by limiting the number of petals  $P$ . We find that a small  $P$  is typically adequate, and considering more petals may result in diminishing marginal utility. Generally, the computational complexity of HiGCN is comparable to that of spectral GNNs. We report the average training time per epoch and average total running time in Appendix G, demonstrating that HiGCN achieves competitive performance with a reasonable computational cost. Additionally, when the targeted graph is not in the form of SCs, one should also consider the one-time pre-processing procedure for graph lifting, see Appendix G for details.

## 5 Experiments

In this section, we evaluate HiGCN on three tasks: node/graph classification and simplicial data imputation. Detailed data introduction and experimental settings are deferred to Appendices H and I, respectively.

### 5.1 Node Classification on Empirical Datasets

We perform the node classification task employing five homogeneous graphs, encompassing three citation graphs - Cora, CiteSeer, PubMed (Yang, Cohen, and Salakhudinov 2016) - and two Amazon co-purchase graphs, Computers and Photo (Shchur et al. 2018). Additionally, we include five heterogeneous graphs, namely Wikipedia graphs Chameleon and Squirrel (Rozenberczki, Allen, and Sarkar 2021), the Actor co-occurrence graph, and the webpage graphs Texas and Wisconsin from WebKB (Pei et al. 2020). Adjacent nodes in homogeneous graphs tend to share the same label, while the opposite holds in heterogeneous graphs. The clique complex lifting transition is carried out on each graph.

We compare HiGCN with various baseline models including MLP, pairwise GNNs (GAT (Veličković et al. 2018), ChebNet (Defferrard, Bresson, and Vandergheynst 2016), BernNet (He et al. 2021), GGCN (Yan et al. 2021), APPNP (Gasteiger, Bojchevski, and Günnemann 2019), GPRGNN

Method	Cora	Citeseer	PubMed	Computers	Photo	Chameleon	Actor	Squirrel	Texas	Wisconsin
MLP	76.96	76.58	85.94	82.85	84.72	46.85	40.19	31.03	91.45	93.56
GAT	88.03	80.52	87.04	83.33	90.94	63.90	35.98	42.72	78.87	65.64
ChebNet	86.67	79.11	87.95	87.54	93.77	59.96	38.02	40.67	86.08	90.57
BernNet	88.52	80.09	88.48	87.64	93.63	<u>68.29</u>	<u>41.79</u>	<u>51.35</u>	<b>93.12</b>	91.82
GGCN	87.68	77.08	89.63	N/A	89.92	62.72	38.09	49.86	85.81	87.65
APNP	88.14	80.47	88.12	85.32	88.51	51.89	39.66	34.71	90.98	64.59
GPRGNN	88.57	80.12	88.46	86.85	93.85	67.28	39.92	50.15	<u>92.95</u>	88.54
k-S2V	68.30	44.22	67.21	84.15	89.08	49.00	N/A	39.15	85.12	87.44
S2V	80.15	78.21	85.48	83.25	84.33	47.14	39.22	40.26	82.12	83.48
SNN	87.13	79.87	86.73	83.33	88.27	60.96	30.59	45.66	75.16	61.93
SGAT	77.49	78.93	88.10	N/A	N/A	51.23	36.71	N/A	89.83	81.47
SGATEF	78.12	79.16	88.47	N/A	N/A	51.61	37.33	N/A	89.67	81.59
1-HiGCN	88.96	80.96	89.83	90.50	95.22	63.55	41.57	49.13	90.36	94.39
2-HiGCN	<b>89.23</b>	<b>81.12</b>	<u>89.89</u>	<b>90.76</b>	<b>95.33</b>	<b>68.47</b>	<b>41.81</b>	<b>51.86</b>	92.15	<u>94.69</u>
3-HiGCN	<u>89.00</u>	80.90	89.73	<u>90.65</u>	94.40	67.12	41.29	50.92	91.85	94.12
4-HiGCN	88.63	80.47	<b>89.95</b>	90.35	94.10	66.98	41.13	50.45	91.42	<b>94.89</b>

Table 2: Node classification results on empirical benchmark networks.

(Chien et al. 2020)), and higher-order models (S2V (Billings et al. 2019), k-S2V (Hacker 2020), SNN (Ebli, Defferrard, and Spreemann 2020), SGAT, SGATEF (Lee, Ji, and Tay 2022)). We randomly partition the node set into train/validation/test subsets with a ratio of 60%/20%/20%, and repeat the experiments 100 times. The mean classification accuracies on the test nodes are reported in Table 2.

It can be drawn from Table 2 that HiGCN achieves the best results in 9 out of the 10 graphs. On the remaining dataset, HiGCN also displays comparable performance to the SOTA methods. Generally, 2-HiGCN and 3-HiGCN outperform 1-HiGCN, suggesting the value of higher-order information in graph learning. However, it is elusive to find that performance did not consistently increase with the inclusion of more higher-order interactions. One possible explanation is that introducing more higher-order interactions might make the training process more complex and challenging. If the model lacks sufficient training data or appropriate training strategies, it may struggle to effectively harness these higher-order interactions. Furthermore, HiGCN shows on average a greater lead on homogeneous graphs, consistent with the intuition that higher-order effects tend to manifest on homogeneous graphs (Battiston et al. 2020).

In addition, we scale to three larger datasets: Ognb-arxiv and Genius (homogeneous graphs) and Penn94 (heterogeneous graph). The results in Appendix G highlight HiGCN’s superior performance and robust scalability.

**Quantifying higher-order strength.** The filter weights  $\gamma_{p,k}$  captures the influence of  $p$ -simplex on  $k$ -hop neighbors; thus, we quantify the  $p$ -order interaction strength in terms of

$$\mathcal{S}_p = \sum_k |\gamma_{p,k}|. \quad (8)$$

To gain insight, we visualize  $\mathcal{S}_p$  with order  $p = 1, 2, 3, 4$  on both homogeneous (Cora, Photo) and heterogeneous (Actor, Texas) graphs in Figure 3 a-d. We observe that  $\mathcal{S}_p$  decreases gently with the increase of  $p$  in homogeneous graphs, while it decreases rapidly in heterogeneous graphs. This observation implies that the strength of higher-order effects varies at

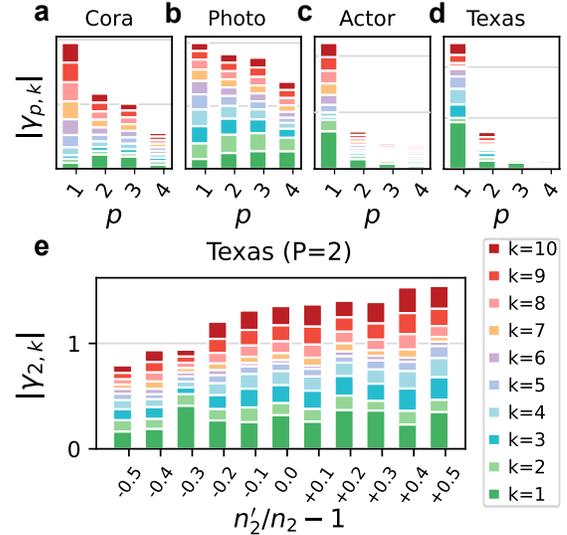


Figure 3: a, b, c and d visualize the stack of learned weights  $|\gamma_{p,k}|$  under order  $p = 1, 2, 3, 4 (P = 4)$ . e visualizes the stack of  $|\gamma_{2,k}|$  for Texas under various relative densities  $\rho_2$ .

different orders and across different types of graphs.

We observe that graphs with fewer higher-order structures tend to exhibit a smaller  $\mathcal{S}_p$ , potentially degrading HiGCN’s performance. For instance, in Texas, the only dataset where HiGCN’s performance is not optimal, we note significantly weaker higher-order interactions compared to lower-order ones (see Figure 3d), and it has the fewest triangles among all datasets (see Table 8). To verify this conjecture, we manipulate the number of higher-order structures by adjusting the edge connectivity while maintaining the degree distribution as done in 1k null models (Zeng et al. 2023c), see Appendix F for details. We define the relative higher-order density for the modified networks as  $\rho_p = n'_p/n_p - 1$ , where  $n_p$  and  $n'_p$  denote the number of  $p$ -simplex in the original and

Dataset	PRO.	MUT.	PTC	IMDB-B	IMDB-M
RWK	59.6	79.2	55.9	N/A	N/A
GK (k=3)	71.4	81.4	55.7	N/A	N/A
PK	73.7	76.0	59.5	N/A	N/A
WL kernel	75.0	90.4	59.9	73.8	50.9
DCNN	61.3	N/A	N/A	49.1	33.5
DGCNN	75.5	85.8	58.6	70.0	47.8
IGN	76.6	83.9	58.5	72.0	48.7
GIN	76.2	89.4	64.6	75.1	52.3
PPGNs	<b>77.2</b>	<u>90.6</u>	<u>66.2</u>	73.0	50.5
Natural GN	71.7	89.4	<b>66.8</b>	73.5	51.3
MPSN	76.7	89.8	61.8	<u>75.6</u>	<u>52.4</u>
<b>HiGCN</b>	<u>77.0</u>	<b>91.3</b>	<u>66.2</u>	<b>76.2</b>	<b>52.7</b>

Table 3: Graph classification results.

the modified network, respectively. Figure 3e visualizes  $\mathcal{S}_p$  under different  $\rho_2$  for Texas, showing an upward trend as the triangle density  $\rho_2$  increases. Table 5 also reveals an increasing accuracy rank of HiGCN with the rise of  $\rho_2$ . Hence,  $\mathcal{S}_p$  can serve as a quantification of  $p$ -order interaction strength. More results and discussions are deferred to Appendix F.

## 5.2 Graph Classification on TUD Benchmarks

To verify the broad applicability of the proposed model, we also evaluate the graph classification performance of HiGCN using various datasets from diverse domains, which are categorized into two main groups: bioinformatics datasets (i.e., PROTEINS (Borgwardt et al. 2005), MUTAG (Debnath et al. 1991), PTC (Toivonen et al. 2003)) and social network datasets (i.e., IMDB-B, IMDB-M (Yanardag and Vishwanathan 2015)). To obtain a global embedding for each graph, we apply readout operations by performing averaging or summation. Following the standard pipeline in (Xu et al. 2019), we conduct a 10-fold cross-validation procedure and report the maximum average validation accuracy across folds. The performance of HiGCN is presented in Table 3, alongside the results for kernel methods (RWK (Gärtner, Flach, and Wrobel 2003), GK (Shervashidze et al. 2009), PK (Neumann et al. 2016), WL kernel (Shervashidze et al. 2011)), pairwise GNNs (DCNN (Atwood and Towsley 2016), DGCNN (Zhang et al. 2018), IGN (Maron et al. 2018), GIN (Xu et al. 2019), PPGNs (Maron et al. 2019), Natural GN (de Haan, Cohen, and Welling 2020)), and the higher-order model MPSN (Bodnar et al. 2021).

Our model exhibits superior performance compared to these baselines, demonstrating strong empirical results across all benchmark datasets. Additionally, HiGCN achieves its optimal outcomes on the two social network datasets, coinciding with the finding that simplices play a pivotal role in social networks (Battiston et al. 2021).

## 5.3 Simplicial Data Imputation

In the previous two experiments, we focused on pairwise graphs with clique complex lifting. Now, we extend our investigation to impute missing signals in coauthorship complexes, a typical SC, wherein a paper with  $p + 1$  authors is represented by a  $p$ -simplex, and the  $p$ -simplicial sig-

SCs	Method	10%	30%	50%	70%
History	SNN	0.201	0.354	0.495	0.661
	SGAT	0.180	0.330	0.432	0.602
	SGATEF	0.200	0.340	0.454	0.633
	<b>HiGCN</b>	<b>0.258</b>	<b>0.438</b>	<b>0.579</b>	<b>0.666</b>
Geology	SNN	0.265	0.417	0.594	0.704
	SGAT	0.223	0.345	0.599	0.631
	SGATEF	0.230	0.369	0.615	0.682
	<b>HiGCN</b>	<b>0.463</b>	<b>0.565</b>	<b>0.644</b>	<b>0.708</b>
DBLP	SNN	0.222	0.348	0.496	0.668
	SGAT	0.210	0.279	0.487	0.643
	SGATEF	0.223	0.311	0.491	0.678
	<b>HiGCN</b>	<b>0.385</b>	<b>0.511</b>	<b>0.587</b>	<b>0.685</b>

Table 4: Simplicial data imputation results: mean Kendall correlation. The best results are in bold.

nal corresponds to the number of collaborative publications among authors in the  $p$ -simplex. We employ three coauthorship complexes, namely DBLP (Benson et al. 2018), History and Geology (Sinha et al. 2015). The known signals for 0-simplex are set to range from 10% to 70% (in units of 20%), and the remainders are regarded as missing signals, replaced by the median of known signals. We apply Kendall’s Tau  $\mathcal{T}$  to measure the correlation between true and predicted simplicial signal, with  $\mathcal{T}$  approaching 1 indicating superior performance (Kendall 1938). The experiment is repeated for 10 different random weight initializations, and the results are compared against higher-order models (namely SNN, SGAT, and SGATEF).

Table 4 shows that HiGCN outperforms other higher-order benchmarks. This superiority is mainly due to the inherent flexibility of our model in capturing higher-order information, whereas the benchmarks are restricted to learning through upper or lower adjacencies. Moreover, HiGCN achieves more performance gains when less information is available. This may be attributed to higher-order information compensating for missing signals, with potential overlap when there is an abundance of known information.

## 6 Conclusion

This paper introduces a novel higher-order representation, the flower-petals (FP) model, enabling interactions among simplices of arbitrary orders. To increase efficiency, we simplify the interaction rules in SCs. It is a valuable and open question whether other simplifications would be more effective for specific tasks. FP adjacency and Laplacian matrices are further introduced based on the higher-order random walk dynamics on the FP model. As an application of FP Laplacians in deep learning, a higher-order graph convolutional network (HiGCN) is introduced. Our theoretical analysis highlights HiGCN’s advanced expressiveness, supported by empirical performance gains across various tasks. Moreover, we deploy a data-driven strategy to demonstrate the existence of higher-order interactions and quantify their strength. This work promises to offer novel insights and serve as a potent tool in higher-order network analysis.

## Acknowledgments

The authors acknowledge the STI 2030—Major Projects (Grant No. 2022 ZD0211400), the National Natural Science Foundation of China (Grant No. T2293771), the Sichuan Science and Technology Program (Grant No. 2023NSFSC1919) and the New Cornerstone Science Foundation through the XPLOER PRIZE.

## References

- Atwood, J.; and Towsley, D. 2016. Diffusion-convolutional neural networks. *Advances in neural information processing systems*, 29.
- Battiston, F.; Amico, E.; Barrat, A.; Bianconi, G.; Ferraz de Arruda, G.; Franceschiello, B.; Iacopini, I.; Kéfi, S.; Latora, V.; Moreno, Y.; et al. 2021. The physics of higher-order interactions in complex systems. *Nature Physics*, 17(10): 1093–1098.
- Battiston, F.; Cencetti, G.; Iacopini, I.; Latora, V.; Lucas, M.; Patania, A.; Young, J.-G.; and Petri, G. 2020. Networks beyond pairwise interactions: Structure and dynamics. *Physics Reports*, 874: 1–92.
- Benson, A. R.; Abebe, R.; Schaub, M. T.; Jadbabaie, A.; and Kleinberg, J. 2018. Simplicial closure and higher-order link prediction. *Proceedings of the National Academy of Sciences*.
- Billings, J. C. W.; Hu, M.; Lerda, G.; Medvedev, A. N.; Mottes, F.; Onicas, A.; Santoro, A.; and Petri, G. 2019. Simplex2vec embeddings for community detection in simplicial complexes. arXiv:1906.09068.
- Bodnar, C.; Frasca, F.; Wang, Y.; Otter, N.; Montufar, G. F.; Lio, P.; and Bronstein, M. 2021. Weisfeiler and leman go topological: Message passing simplicial networks. In *International Conference on Machine Learning (ICML)*, 1026–1037.
- Bomze, I. M.; Budinich, M.; Pardalos, P. M.; and Pelillo, M. 1999. The maximum clique problem. In *Handbook of combinatorial optimization*, 1–74. Springer.
- Borgwardt, K. M.; Ong, C. S.; Schönauer, S.; Vishwanathan, S.; Smola, A. J.; and Kriegel, H.-P. 2005. Protein function prediction via graph kernels. *Bioinformatics*, 21(suppl\_1): i47–i56.
- Centola, D. 2010. The Spread of Behavior in an Online Social Network Experiment. *Science*, 329(5996): 1194–1197.
- Chen, Y.; Gel, Y. R.; and Poor, H. V. 2022. BScNets: Block Simplicial Complex Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(6): 6333–6341.
- Chien, E.; Peng, J.; Li, P.; and Milenkovic, O. 2020. Adaptive Universal Generalized PageRank Graph Neural Network. In *International Conference on Learning Representations*.
- de Haan, P.; Cohen, T. S.; and Welling, M. 2020. Natural graph networks. *Advances in neural information processing systems*, 33: 3636–3646.
- Debnath, A. K.; Lopez de Compadre, R. L.; Debnath, G.; Shusterman, A. J.; and Hansch, C. 1991. Structure-activity relationship of mutagenic aromatic and heteroaromatic nitro compounds. correlation with molecular orbital energies and hydrophobicity. *Journal of medicinal chemistry*, 34(2): 786–797.
- Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29: 3838–3845.
- Ebli, S.; Defferrard, M.; and Spreemann, G. 2020. Simplicial Neural Networks. In *NeurIPS 2020 Workshop on Topological Data Analysis and Beyond*.
- Ganmor, E.; Segev, R.; and Schneidman, E. 2011. Sparse low-order interaction network underlies a highly correlated and learnable neural population code. *Proceedings of the National Academy of Sciences*, 108(23): 9679–9684.
- Gao, Y.; Zhang, Z.; Lin, H.; Zhao, X.; Du, S.; and Zou, C. 2022. Hypergraph Learning: Methods and Practices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5): 2548–2566.
- Gärtner, T.; Flach, P.; and Wrobel, S. 2003. On graph kernels: Hardness results and efficient alternatives. In *Learning Theory and Kernel Machines*, 129–143. Springer.
- Gasteiger, J.; Bojchevski, A.; and Günnemann, S. 2019. Predict then Propagate: Graph Neural Networks meet Personalized PageRank. In *International Conference on Learning Representations*.
- Grilli, J.; Barabás, G.; Michalska-Smith, M. J.; and Allesina, S. 2017. Higher-order interactions stabilize dynamics in competitive network models. *Nature*, 548(7666): 210–213.
- Hacker, C. 2020. k-simplex2vec: a simplicial extension of node2vec. arXiv:2010.05636.
- Hajij, M.; Zamzmi, G.; Papamarkou, T.; Maroulas, V.; and Cai, X. 2022. Simplicial complex representation learning. In *Machine Learning on Graphs (MLOG) Workshop at 15th ACM International WSDM Conference*.
- Hatcher, A. 2002. *Algebraic topology*. Cambridge University Press.
- He, M.; Wei, Z.; Xu, H.; et al. 2021. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in Neural Information Processing Systems*, 34: 14239–14251.
- Kendall, M. G. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2): 81–93.
- Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *ICLR*.
- Lee, S. H.; Ji, F.; and Tay, W. P. 2022. SGAT: Simplicial Graph Attention Network. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-22)*, 3192–3200.
- Maron, H.; Ben-Hamu, H.; Serviansky, H.; and Lipman, Y. 2019. Provably powerful graph networks. *Advances in neural information processing systems*, 32.

- Maron, H.; Ben-Hamu, H.; Shamir, N.; and Lipman, Y. 2018. Invariant and Equivariant Graph Networks. In *International Conference on Learning Representations*.
- Neumann, M.; Garnett, R.; Bauckhage, C.; and Kersting, K. 2016. Propagation kernels: efficient graph kernels from propagated information. *Machine Learning*, 102: 209–245.
- Pei, H.; Wei, B.; Chang, K. C.-C.; Lei, Y.; and Yang, B. 2020. Geom-GCN: Geometric Graph Convolutional Networks. In *International Conference on Learning Representations*.
- Roddenberry, T. M.; Glaze, N.; and Segarra, S. 2021. Principled Simplicial Neural Networks for Trajectory Prediction. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, 9020–9029. PMLR.
- Rozemberczki, B.; Allen, C.; and Sarkar, R. 2021. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2): 1–22.
- Schaub, M. T.; Benson, A. R.; Horn, P.; Lippner, G.; and Jadbabaie, A. 2020. Random walks on simplicial complexes and the normalized Hodge 1-Laplacian. *SIAM Review*, 62(2): 353–391.
- Shchur, O.; Mumme, M.; Bojchevski, A.; and Günnemann, S. 2018. Pitfalls of Graph Neural Network Evaluation. *CoRR*, abs/1811.05868.
- Shervashidze, N.; Schweitzer, P.; Van Leeuwen, E. J.; Mehlhorn, K.; and Borgwardt, K. M. 2011. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(9).
- Shervashidze, N.; Vishwanathan, S.; Petri, T.; Mehlhorn, K.; and Borgwardt, K. 2009. Efficient graphlet kernels for large graph comparison. In *Artificial intelligence and statistics*, 488–495. PMLR.
- Shuman, D. I.; Narang, S. K.; Frossard, P.; Ortega, A.; and Vandergheynst, P. 2013. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3): 83–98.
- Sinha, A.; Shen, Z.; Song, Y.; Ma, H.; Eide, D.; Hsu, B.-J. P.; and Wang, K. 2015. An Overview of Microsoft Academic Service (MAS) and Applications. In *Proceedings of the 24th International Conference on World Wide Web*. ACM Press.
- Toivonen, H.; Srinivasan, A.; King, R. D.; Kramer, S.; and Helma, C. 2003. Statistical evaluation of the predictive toxicology challenge 2000–2001. *Bioinformatics*, 19(10): 1183–1193.
- Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations (ICLR)*.
- Wang, X.; and Zhang, M. 2022. How Powerful are Spectral Graph Neural Networks. In *International Conference on Machine Learning (ICML)*.
- Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*.
- Yan, Y.; Hashemi, M.; Swersky, K.; Yang, Y.; and Koutra, D. 2021. Two sides of the same coin: Heterophily and oversmoothing in graph convolutional neural networks. arXiv:2102.06462.
- Yanardag, P.; and Vishwanathan, S. 2015. Deep Graph Kernels. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, 1365–1374. New York, NY, USA: Association for Computing Machinery. ISBN 9781450336642.
- Yang, M.; Isufi, E.; and Leus, G. 2022. Simplicial convolutional neural networks. In *ICASSP*, 8847–8851.
- Yang, Z.; Cohen, W.; and Salakhudinov, R. 2016. Revisiting Semi-Supervised Learning with Graph Embeddings. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, 40–48. New York, New York, USA: PMLR.
- Zeng, Y.; Huang, Y.; Ren, X.-L.; and Lü, L. 2023a. Identifying vital nodes through augmented random walks on higher-order networks. arXiv:2305.06898.
- Zeng, Y.; Huang, Y.; Wu, Q.; and Lü, L. 2023b. Influential Simplices Mining via Simplicial Convolutional Network. arXiv:2307.05841.
- Zeng, Y.; Liu, B.; Zhou, F.; and Lü, L. 2023c. Hyper-Null Models and Their Applications. *Entropy*, 25(10): 1390.
- Zhang, M.; Cui, Z.; Neumann, M.; and Chen, Y. 2018. An end-to-end deep learning architecture for graph classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.