Centrality Graph Shift Operators for Graph Neural Networks

Anonymous authors
Paper under double-blind review

Abstract

Graph Shift Operators (GSOs), such as the adjacency and graph Laplacian matrices, play a fundamental role in graph theory and graph representation learning. Traditional GSOs are typically constructed by normalizing the adjacency matrix by the degree matrix, a local centrality metric. In this work, we instead propose and study Centrality GSOs (CGSOs), which normalize adjacency matrices by global centrality metrics such as the PageRank, k-core or count of fixed length walks. We study spectral properties of the CGSOs, allowing us to get an understanding of their action on graph signals. This understanding is confirmed by defining and running the spectral clustering algorithm based on different CGSOs on several synthetic and real-world datasets. We furthermore outline how our CGSO can act as the message passing operator in any Graph Neural Network and in particular demonstrate strong performance of a variant of the Graph Convolutional Network and Graph Attention Network using our CGSOs on several real-world datasets.

1 Introduction

We propose and study a new family of operators defined on graphs that we call Centrality Graph Shift Operators (CGSOs). To insert these into the rich history of matrices representing graphs and centrality metrics, the two concepts married in CGSOs, we begin by recalling major advances in these two topics in turn (readers interested purely in recent developments in Graph Representation Learning and Graph Neural Networks are recommended to begin reading in Paragraph 3 of this section). The study of graph theory and with it the use of matrices to represent graphs have a long-standing history. Graph theory is often said to have its origins in 1736 when Leonard Euler posed and solved the Königsberg bridge problem (Euler, 1736). His solution did not involve any matrix calculus. In fact, it seems that the first matrix defined to represent graph structures is the *incidence matrix* defined by Henri Poincaré in 1900 (Poincaré, 1900). It is difficult to pinpoint the first definition of *adjacency matrices*, but by 1936 when the first book on the topic of graph theory was published by Dénes König adjacency matrices had certainly been defined and began to be used to solve graph theoretic problems (König, 1936). Two seemingly concurrent works in 1973 defined an additional matrix structure to represent graphs that later became known as the *unnormalized graph Laplacian* (Donath & Hoffman, 1973; Fiedler, 1973). Then, it was Fan Chung in her book "Spectral Graph Theory" published in 1997 who extensively characterized the spectral properties of *normalized Laplacians* (Chung, 1997a). In the emerging field of Graph Signal Processing (GSP) (Sandryhaila & Moura, 2013; Ortega et al., 2018) these different graph representation matrices were all defined to belong to a more general family of operators defined on graphs, the *Graph Shift Operators (GSOs)*. GSOs currently play a crucial role in graph representation learning research, since the choice of GSO, used to represent a graph structure, corresponds to the choice of message passing function in the currently much-used Graph Neural Network (GNN) models.

In parallel to advances in graph representation via matrices, centrality metrics have proved to be insightful in the study of graphs. Chief among them is the success of the PageRank centrality criterion revealing the significance of certain webpages (Brin & Page, 1998) and playing a role in the formation of what is now one of the largest companies worldwide. But also an even older metric, the k-core centrality (Seidman, 1983; Malliaros et al., 2020), as well as the degree centrality, closeness centrality, and betweenness centrality, have proven to be impactful in revealing key structural properties of graphs (Freeman, 1977; Zhang & Luo, 2017).

A commonality of the most frequently used GSOs is their property to encode purely local information in the graph, with the adjacency matrix encoding neighborhoods in the graph and the graph Laplacians relying on the node degree, a local centrality metric, to normalize the adjacency matrix. In this work, we propose a novel class of GSOs, the Centrality GSOs (CGSOs) that arise from the normalization of the adjacency matrix by centrality metrics such as the PageRank, k-core and the count of fixed length walks emanating from a given node. Our CGSOs introduce global information into the graph representation without altering the connectivity pattern encoded in the original GSO and therefore, maintain the sparsity of the adjacency matrix. We provide several theorems characterizing the spectral properties of our CGSOs. We confirm the intuition gained from our theoretical study by running the spectral clustering algorithm on the basis of our CGSOs on 1) synthetic graphs that are generated from a stochastic blockmodel in which each block is sampled from the Barrabasi-Albert model and 2) the real-world Cora graph in which we aim to recover the partition provided by the k-core number of each node. We furthermore describe how our CGSOs can be inserted as the message passing operator into any GNN and observe strong performance of the resulting GNNs on real-world benchmark datasets.

In particular, our contributions can be summarized as follows: **(i)** we define Centrality GSOs, a novel class of GSOs based on the normalization of the adjacency matrix with different centrality metrics, such as the degree, PageRank score, k-core number, and the count of walks of a fixed length; **(ii)** we conduct a comprehensive spectral analysis to unveil the fundamental properties of the CGSOs. Our gained understanding of the benefits of CGSOs is confirmed by running the spectral clustering algorithm using our CGSOs on synthetic and real-world graphs; **(iii)** we incorporate the proposed CGSOs within GNNs and evaluate performance of a Graph Convolutional Network and Graph Attention Network v2 with a CGSO message passing operator on several real-world datasets.

2 Background and Related Work

2.1 Graph Shift Operators

We consider simple graphs $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a set if nodes $\mathcal{V} = \{1, \ldots, N\}$, and a set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. The adjacency matrix is one of the standard graph representation matrices considered in our work. Formally, a graph can be represented by an adjacency matrix $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ where $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$ and $a_{ij} = 0$ otherwise. Analyzing the spectrum of the adjacency matrix provides much information about the topological properties of the underlying graph (Cvetkovic et al., 1980). For example, the largest eigenvalue of $\mathbf{A}$ is an upper bound of the average degree, a lower bound on the largest degree (Cvetković et al., 2009; Sarkar & Jalan, 2018) and its multiplicity indicates whether the represented graph is connected (Stanić, 2015). Another nice example is the fact that the adjacency spectrum of bipartite graphs is symmetric around 0 (Stanić, 2015).

In addition to the adjacency matrix, there are alternative graph representations that provide related, but often different insights into the topology of the underlying graph. One often-used representation is the symmetrically normalized Laplacian matrix defined by $\mathbf{L}_{sym} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, where $\mathbf{D} \in \mathbb{R}^{N \times N}$ is the degree matrix, i.e., a diagonal matrix with entries $\mathbf{D}_{ii} = d_i = \sum_{i=1}^{N} a_{ij}$ corresponding to the degree of each node. The normalized Laplacian plays a fundamental role in spectral graph theory. For example, the celebrated *Cheeger's Inequality* establishes a bound on the edge expansion of a graph via its spectrum (Cheeger, 1970). There are other graph representations with particularly interesting spectral properties (Lutzeyer, 2020), such as the random-walk Normalised Laplacian (Modell & Rubin-Delanchy, 2021) and the Signless Laplacian matrices (Cvetković & Simić, 2010). All these graph representations belong to the family of *Graph Shift Operators* (GSOs), which we define now in Definition 2.1. Fundamentally, GSOs preserve the sparsity pattern of the underlying graph, ensuring that computational operations remain localized to node neighborhoods. By modulating the magnitude of their non-zero entries, these operators allow for the flexible upweighting or downweighting of specific edges and self-loops to capture information from neighbors with a varying or equal importance.

Definition 2.1 (Graph Shift Operator)**.** Given an arbitrary graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a *Graph Shift Operator* $\mathbf{S} \in \mathbb{R}^{N \times N}$ is a matrix satisfying $\mathbf{S}_{ij} = 0$ for $i \neq j$ and $(i, j) \notin \mathcal{E}$ (Mateos et al., 2019) and $\mathbf{S}_{ij} \neq 0$ for $i \neq j$ and $(i, j) \in \mathcal{E}$.

In addition to the classical GSOs, parametrized GSOs can be learned during the optimization process of any model in which they are inserted. These parametrizations are a fundamental component of many modern GNN architectures and allow the model to adapt and capture complex patterns and relationships. For example, the parametrized GSO (PGSO) of Dasoulas et al. (2021) parametrizes the space of commonly used GSOs leading to a learnable GSO that adapts to the dataset and learning task at hand. Similarly, there are also other parametrizations based specifically on degree normalization (Lu et al., 2024; Donnat & Jeong, 2023).

2.2 Graph Neural Networks

Graph Neural Networks (GNNs) are neural networks that operate on graph-structured data that is defined as the combination of a graph $G = (\mathcal{V}, \mathcal{E})$, and a node feature matrix $\mathbf{X} \in \mathbb{R}^{N \times d}$, containing the node feature vector of node i in its ith row. GNNs are formed by stacking several computational layers, each of which produces a hidden representation for each node in the graph, denoted by $\mathbf{H}^{(\ell)} = [\mathbf{h}_v^{(\ell)\top}]_{v \in \mathcal{V}} \in \mathbb{R}^{N \times d_\ell}$. A GNN layer ℓ updates node representations relying on the structure of the graph and the output of the previous layer $\mathbf{H}^{(\ell-1)}$. Conventionally, the node features are used as input to the first layer $\mathbf{H}^0 = \mathbf{X}$. The most popular framework of GNNs is that of Message Passing Neural Networks (Gilmer et al., 2017; Hamilton, 2020), where the computations are split into two main steps,

Message Passing: Given a node v, this step applies a permutation-invariant function to its neighbors, denoted by $\mathcal{N}(v)$, to generate the aggregated representation,

$$\mathbf{M}^{(\ell)} = \mathbf{\Phi}(\mathbf{A})\mathbf{H}^{(\ell-1)}, \tag{1}$$

where $\mathbf{\Phi}(\mathbf{A}) : \mathbb{R}^{N \times N} \to \mathbb{R}^{N \times N}$, a function of the adjacency matrix, is the chosen GSO.

Update: In this step, we combine the aggregated hidden states with the previous hidden representation of the central node v, usually by making use of a learnable function,

$$\mathbf{H}^{(\ell)} = \sigma(\mathbf{M}^{(\ell)}\mathbf{W}^{(\ell-1)}), \tag{2}$$

where $\mathbf{W}^{(\ell-1)} \in \mathbb{R}^{d_{\ell-1}, d_\ell}$ are learnable weight matrices, and $\sigma(\cdot)$ is an activation function, e.g., ReLU function.

With the emergence and increasing popularity of GNNs, the importance of GSOs has significantly increased. Numerous GNN architectures, such as notably Graph Convolutional Networks (GCNs), rely on these operators in their message passing step. In the context of GCNs (Kipf & Welling, 2017), the used message passing operator, i.e., the chosen GSO, corresponds to $\mathbf{\Phi}(\mathbf{A}) = \mathbf{D}_1^{-1/2}\mathbf{A}\mathbf{D}_1^{-1/2}$, where $\mathbf{D}_1 = \mathbf{D} + \mathbf{I}$ is the degree matrix of the graph corresponding to the adjacency matrix $\mathbf{A}_1 = \mathbf{A} + \mathbf{I}$. For Graph Attention Networks v2 (GATv2) (Brody et al., 2022), (1) becomes $\mathbf{M}^{(\ell+1)} = \mathbf{\Phi}(\mathbf{A}^{(\ell)}_{\text{GATv2}})\mathbf{H}^{(\ell)}$, where, in this setting, Φ corresponds to the identity function and the rows of $\mathbf{A}^{(\ell)}_{\text{GATv2}}$ contain the edge-wise attention coefficients.

Global Information in GNNs. Besides our GNNs, which leverage the CGSO to make global information accessible to any given GNN layer, there exists a plethora of other approaches to achieve this goal. These include for example the PPNP and APPNP (Gasteiger et al., 2019), as well as the PPRGo (Bojchevski et al., 2020) models that use the PageRank centrality to define a completely new graph over which to perform message passing in GNNs. The work of Vela et al. (2022) extends these models to consider both the PageRank and k-core centrality. In addition, there is the AdaGCN (Sun et al., 2019) and the VPN model (Jin et al., 2021) which propose to message pass using powers of the adjacency matrix to incorporate global information and increase the robustness of GNNs, respectively. Lee et al. (2019) propose the Motif Convolutional Networks, that define motif adjacency matrices and then use these in the message passing scheme. Also the k-hop GNNs of Nikolentzos et al. (2020) consider neighbors several hops away from a given central node in the message passing scheme of a single GNN layer to consider more global information in a GNN. Additionally there exists a rich and long-standing literature on spectral GNNs that facilitate global information exchange by explicitly or approximately making use of the spectral decomposition of the GSO chosen to be the GNN's message passing operator (Bruna et al., 2014; Defferrard et al., 2016; Koke & Cremers, 2024). Finally, there is an arm of research investigating graph transformers, where usually the graph structure is used to provide structural encodings of nodes and the optimal message passing operator is learning using

an attention mechanism (Kreuzer et al., 2021; Rampášek et al., 2022; Ma et al., 2023). The majority of these approaches increase the computational complexity of the GNN, whereas our CGSO based GNNs maintain the complexity of the underlying GNN model by preserving the sparsity of the original adjacency matrix.

3 CGSO: Centrality Graph Shift Operators

In this section, we introduce the Centrality GSOs (CGSOs), a family of shift operators that incorporate the global position of nodes in a graph. We discuss different instances of CGSOs corresponding to widely used centrality criteria. We further conduct a comprehensive spectral analysis to unveil the fundamental properties of CGSOs, including the eigenvalue structure and the expansion properties, examining how these operators influence information spread across the graph. Then, we leverage CGSOs in the design of flexible GNNs.

Motivation. The choice of the GSO is important, as it fundamentally dictates the dynamics of the information diffusion process within the GNN. As established in the upcoming Theorem 3.1, the influence of an input feature $\mathbf{x}_u$ on a remote node representation $\mathbf{h}_v^{(r+1)}$ is strictly constrained by the (vu)-th entry of the GSO's power, $(\Phi(\mathbf{A})^{r+1})_{vu}$. This term effectively controls both the range of the information exchange and the magnitude of the signal reaching the target node. Consequently, the spectral properties of $\Phi(\mathbf{A})$, specifically its eigenvalues, directly determine the stability of training; if the spectrum is not well-bounded, the term $\Phi(\mathbf{A})^{r+1}$ can lead to exponentially exploding or vanishing gradients as the network depth r increases. Furthermore, by governing the sensitivity of the output to changes in the input, the GSO implicitly defines the model's robustness to feature perturbations, ensuring that noise in local neighborhoods is not disproportionately amplified.

Theorem 3.1 (Topping et al. (2022))**.** *Consider a GNN defined by the message passing and update steps in Equations (1) and (2). Let α and β be the Lipschitz constants of the aggregation and update functions with respect to the node features, such that $\|\Phi(\mathbf{A})\| \leq \alpha$ and $\|\nabla\sigma(\mathbf{W}^{(\ell)})\| \leq \beta$ for all ℓ. Then, for any two nodes $u, v \in \mathcal{V}$ at distance $d_{\mathcal{G}}(u, v) = r + 1$, the influence of the input feature $\mathbf{x}_u$ on the final representation $\mathbf{h}_v^{(r+1)}$ is bounded by,* $\left\| \frac{\partial \mathbf{h}_v^{(r+1)}}{\partial \mathbf{x}_u} \right\| \leq (\alpha\beta)^{r+1} \left(\Phi(\mathbf{A})^{r+1}\right)_{vu}$.

Finally, the structure of $\Phi(\mathbf{A})$ determines the model's alignment with the learning task. Optimal performance requires the flexibility to *upweight* informative neighbors and *downweight* irrelevant ones. While this selectivity is essential in heterophilic graphs to filter out dissimilar neighbors, it remains critical even in homophilic settings, where certain local connections may still introduce noise or redundancy. Furthermore, the influence of the GSO extends beyond the immediate neighborhood, fundamentally dictating the set of nodes reached after ℓ layers. Without precise control, the effective receptive field may either fail to capture essential distant dependencies or become diluted by irrelevant distant nodes. This raises a fundamental question: *Does the standard degree-based normalization offer sufficient flexibility to adapt to these diverse structural demands?*

In the literature, the choice of degree-based normalization is primarily justified by two foundational factors: the rich spectral theory associated with the normalized Laplacian (Chung, 1997b), and the need for numerical stability in GNNs (Kipf & Welling, 2017). In this section, we demonstrate that these desirable properties are not exclusive to the node degree. We extend fundamental spectral results to a general class of CGSOs defined by arbitrary centrality metrics $\mathbf{C}$, characterizing their eigenvalue distributions through analytic expressions for their mean and standard deviation. These theoretical guarantees allow us to control the spectrum via simple scalar normalization if needed, ensuring that the stability of the GNN is preserved regardless of the chosen metric. Consequently, with both the spectral and stability prerequisites satisfied, we look beyond local degree constraints and explore the potential of global centrality metrics to improve graph representation learning.

3.1 Mathematical Formulation

For a given node $i \in \mathcal{V}$, let c_i denote a centrality metric associated with i, such as the node degree, k-core number, PageRank, or the count of walks of specific length starting from node i. The Hilbert space $L^2(G)$ is characterized by the set of functions φ defined on $\mathcal{V}$ such that $\sum_{i\in\mathcal{V}} c_i|\varphi(i)|$ converges, equipped with the

inner product: $\langle \varphi_1, \varphi_2 \rangle_{\mathcal{G}} = \sum_{i \in \mathcal{V}} c_i \varphi_1(i) \bar{\varphi}_2(i)$. The *Centrality-Based Averaging Operator* on $L^2(G)$ is defined as the linear map $\mathbf{M}_{\mathcal{G}} : \varphi \mapsto \mathbf{M}_{\mathcal{G}} \varphi$ such that,

$$\left(\mathbf{M}_{\mathcal{G}} \varphi\right)(i) = \left(\mathbf{C}^{-1} \mathbf{A} \varphi\right)(i) = \frac{1}{c_i} \sum_{j \in \mathcal{N}(i)} \varphi(j),$$

where $\mathbf{C} = \mathrm{diag}(c_1, \ldots, c_N)$. The form of this Centrality-Based Averaging Operator, which is a left normalization of the adjacency matrix by a diagonal matrix containing node centralities, i.e., $\mathbf{C}^{-1}\mathbf{A}$, gives rise to the simplest formulation of our *Centrality Graph Shift Operators (CGSOs)* that we define in Section 3.2. Note that the *mean aggregation* operator, as discussed in Xu et al. (2019), represents a specific instance of the Centrality-Based Averaging Operator where the degree corresponds to the chosen centrality metric, namely $\mathbf{C} = \mathbf{D}$. In this paper, we focus on three global centrality metrics, in addition to the local node degree. We recall the definitions of these global centrality metrics now.

***k*-core.** The core number of a node can be determined in the process of the core decomposition of a graph, which captures how well-connected nodes are within their neighborhood (Malliaros et al., 2020). The process of core decomposition involves iteratively removing vertices with degree less than k until no such vertices remain. The core number k of a node is then equal to the largest k for which the considered node has not yet been removed in the decomposition process. We define $\mathbf{C}^{(\mathrm{core})} \in \mathbb{R}^{N \times N}$ to be the diagonal matrix containing the core number of node i in $\mathbf{C}^{(\mathrm{core})}_{ii}$.

PageRank. We choose $\mathbf{C}^{(\mathrm{pr})} \in \mathbb{R}^{N \times N}$ such that, $\forall i \in \mathcal{V},\ \mathbf{C}^{(\mathrm{pr})}_{ii} = (1 - \mathrm{pr}(i))^{-1}$, where $pr(i)$ corresponds to the PageRank score (Brin & Page, 1998). The PageRank score quantifies the likelihood of a random walk visiting a particular node, serving as a fundamental metric for evaluating node significance in various networks. We adopt this specific transformation, which is monotonically increasing with respect to $pr(i)$, to ensure numerical stability in large-scale networks. As N increases, raw PageRank probabilities asymptotically approach zero, which leads to instability during the spectral clustering experiments (Cai et al., 2023).

Walk Count. Here, we consider $\mathbf{C}^{(\ell-\mathrm{walks})} \in \mathbb{R}^{N \times N}$, the diagonal matrix indicating the number of walks of length ℓ starting from each node i, i.e., $\forall i \in \mathcal{V},\ \mathbf{C}^{(\ell-\mathrm{walks})}_{ii} = \left(\mathbf{A}^{\ell}\mathbb{1}\right)_i$, where $\mathbb{1} \in \mathbb{R}^N$ is the vector of ones. When $\ell = 2$, $\mathbf{C}^{(\ell-\mathrm{walks})}$ corresponds to $\mathbf{W}_{\mathbf{M_{13}}}\mathbb{1} - \mathbf{D}$, where $\mathbf{W}_{\mathbf{M_{13}}}$ the graph operator presented by Benson et al. (2016), which corresponds to the count of open bidirectional wedges, i.e., the motif M_{13}. This motif network captures higher-order structures and gives new insights into the organization of complex systems.

We selected these metrics to capture topological information across varying scales. While node degree provides a strictly local measure of importance, i.e., equivalent to $\ell = 1$ walk counts, PageRank offers a global perspective based on random walks. The intermediate metrics, namely k-core and walk counts for $\ell = 2$, interpolate between these extremes by assessing cohesiveness and higher-order local interactions, respectively. In what follows, we establish the theoretical properties of the Centrality-Based Averaging Operator. The properties derived in this section are satisfied for any valid centrality matrix $\mathbf{C}$, and thus apply directly to all our proposed centrality measures.

Proposition 3.2. *The following properties of operator $\mathbf{M}_{\mathcal{G}}$ hold: (i) $\mathbf{M}_{\mathcal{G}}$ is self-adjoint; (ii) $\mathbf{M}_{\mathcal{G}}$ is diagonalizable in an orthonormal basis, its eigenvalues are real numbers, and all eigenvalues have absolute values at most $\gamma = \min_{i \in \mathcal{V}} \left(\frac{c_i}{d_i}\right)$.*

The proof of Proposition 3.2 and all subsequent theoretical results in this section can be found in Appendix G. Hence, we have shown in Proposition 3.2 that all CGSOs have a real set of eigenvalues, which is of real use in practice. In the now following Proposition 3.3 we provide the mean and standard deviation of the spectrum of $\mathbf{M}_{\mathcal{G}}$, i.e., the set of $\mathbf{M}_{\mathcal{G}}$'s eigenvalues.

Proposition 3.3. *The following properties hold for the spectrum of $\mathbf{M}_{\mathcal{G}}$.*

(1) In a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with multiple connected components $\mathcal{C} \subset \mathcal{V}$, where each connected component $\mathcal{C}$ induces a subgraph of $\mathcal{G}$ denoted by $\mathcal{G}_{\mathcal{C}}$, a complete set of eigenvectors of $\mathbf{M}_{\mathcal{G}}$ can be constructed from the eigenvectors of the different $\mathbf{M}_{\mathcal{G}_{\mathcal{C}}}$, where eigenvectors of $\mathbf{M}_{\mathcal{G}_{\mathcal{C}}}$ are extended to have dimension N via the addition of zero entries in all entries corresponding to nodes not in $\mathcal{C}$.

(2) The mean $\mu(\mathbf{M}_{\mathcal{G}})$ and standard deviation $\sigma(\mathbf{M}_{\mathcal{G}})$ of $\mathbf{M}_{\mathcal{G}}$'s spectrum have the following analytic form,

$$\mu\left(\mathbf{M}_{\mathcal{G}}\right)=0, \qquad \sigma\left(\mathbf{M}_{\mathcal{G}}\right)=\left(\frac{1}{N} \sum_{(i, j) \in \mathcal{E}} \frac{1}{c_i c_j}\right)^{1/2}.$$

We define the *normalized spectral gap* $\lambda_1(\mathcal{G})$ as the smallest non-zero eigenvalue of $\mathbf{I}-\mathbf{M}_{\mathcal{G}}$. In Proposition 3.5, we link $\lambda_1(\mathcal{G})$ to the expansion properties of the graph. In the literature, we characterize graph expansion via the *expansion* or *Cheeger constant* (Chung, 1997a). In our work, we generalize this definition to any centrality metric. Definition 3.4 allows us to establish a link between the spectrum of our considered operators and the centrality-based Cheeger constant in Proposition 3.5.

Definition 3.4. Given a graph $\mathcal{G}=(\mathcal{V}, \mathcal{E})$, we define the *Centrality based Cheeger constant* $h_c(\mathcal{G})$ as $h_c(\mathcal{G})=\min \left\{\frac{|\partial \mathcal{U}|_c}{|\mathcal{U}|_c} \mid \mathcal{U} \subset \mathcal{V},|\mathcal{U}|_c \leq \frac{1}{2}|\mathcal{V}|_c\right\}$, where $\partial \mathcal{U}$ are the vertices that are connected to a vertex in $\mathcal{U}$ but are not in $\mathcal{U}$, and $|\cdot|_c: \mathcal{U} \subset \mathcal{V} \mapsto \sum_{i \in \mathcal{U}} c_i$. When the chosen centrality is the degree, $h_c(\mathcal{G})$ corresponds to the classical Cheeger constant.

Proposition 3.5. *Let $\mathcal{G}$ be a connected, non-empty, finite graph without isolated vertices, with* $\max_{i \in \mathcal{V}}\left(\frac{c_i}{d_i}\right) \leq 1$. *We have,* $\lambda_1(\mathcal{G}) \leq\left(2 N \frac{c_+^2}{\gamma c_-}\right) h_c(\mathcal{G})$, *where we denote* $c_-=\min_{i \in \mathcal{V}} c_i$ *and* $c_+=\max_{i \in \mathcal{V}} c_i$.

These results demonstrate that the spectral and numerical advantages of degree-based normalization are preserved within the broader framework of the Centrality-Based Averaging Operators. By proving that our operator possess real, bounded eigenvalues, c.f., Proposition 3.2, we establish that these operators can always be scaled to ensure a spectral radius within the stable regime, e.g., $|\lambda| \leq 1$, directly emulating the stability guarantees of standard normalized adjacency matrices. Furthermore, with the analytic expressions for the spectral distribution, c.f., Proposition 3.3, and the generalized Cheeger constant, c.f., Definition 3.4, we provide insights allowing us to move beyond the local degree centrality.

3.2 CGNN: Centrality Graph Neural Network

In this section, we leverage the centrality metrics defined in Section 3.1 to design flexible message-passing operators. By replacing the standard degree-based shift operators, c.f. Table 5, with a Centrality Graph Shift Operator (CGSO), we systematically investigate whether replacing the standard local degree centrality with global centrality metrics enhances graph representation learning.

Definition of the Learnable Centrality Graph Shift Operators (CGSOs). To allow a flexible and stable integration of centrality into GNNs, we define the Centrality Graph Shift Operator (CGSO), which is a generalization of the work of Dasoulas et al. (2021). By parameterizing the normalization process, the CGSO enables the model to adaptively weight nodes based on their global significance while preserving the original sparsity of the adjacency matrix. The exact formula of the new parametrized CGSO is,

$$\Phi(\mathbf{A}, \mathbf{C} ; \Theta)=m_1 \mathbf{C}^{e_1}+m_2 \mathbf{C}^{e_2} \mathbf{A}_a \mathbf{C}^{e_3}+m_3 \mathbf{I}_N,$$

where $\mathbf{A}_a=\mathbf{A}+a \mathbf{I}_N$, and $\Theta=\left(m_1, m_2, m_3, e_1, e_2, e_3, a\right)$ are scalar parameters that are learnable via backpropagation. Here m_1 controls the additive centrality normalization of the adjacency matrix. The parameter e_1 controls whether the additive centrality normalization is performed with an emphasis on large centrality values (for large positive values of e_1) or with an emphasis on small centrality values (for large negative values of e_1). Similarly, we have e_2 and e_3 controlling the emphasis on large or small centralities, as well as whether the multiplicative centrality normalization of the adjacency matrix is performed symmetrically or predominantly as a column or row normalization. The parameter m_2 controls the magnitude and sign of the adjacency matrix term; in particular, a negative m_2 corresponds to a more Laplacian-like CGSO, while a positive m_2 gives rise to a more adjacency-like CGSO. Finally, a determines the weight of the self-loops that are added to the adjacency matrix, and m_3 controls a further diagonal regularization term of the CGSO. More details on the experimental setup are provided in Section 5.

We characterize the effect of this operator on the embedding space through the bound in Lemma 3.6.

Lemma 3.6 (Effect on Node Embedding Space)**.** *Let $\mathbf{Z} \in \mathbb{R}^{N \times d}$ be the input feature matrix, and $||\mathbf{Z}||_{2,\infty} = \max_{i \in \mathcal{V}} ||\mathbf{Z}_i||_2$ is the maximum L_2 norm of the node feature vectors. For any node u, let $\mathbf{H}_u = [\Phi(\mathbf{A}, \mathbf{C}; \Theta)\mathbf{Z}]_u$ represent the embedding after one message passing step. The L_2 norm of the node embedding is bounded by,*

$$||\mathbf{H}_u||_2 = ||\mathbf{Z}||_{2,\infty} \left(|m_1| c_u^{e_1} + |m_3| + |m_2| \left[c_u^{e_2+e_3}(d_u + a) + e_3 c_u^{e_2} \frac{\bar{\Delta}_{u,C}}{c_u^{1-e_3}} + O\left(c_u^{e_2} \sum_{v \in \mathcal{N}(u)} (c_v - c_u)^2 \right) \right] \right)$$

where $\bar{\Delta}_{u,\mathbf{C}} = \frac{\sum_{v \in \mathcal{N}(u)} (c_v - c_u)}{d_u + a}$ denotes the weighted average of centrality differences in the neighborhood.

The proof of Lemma 3.6 can be found in Appendix G. Lemma 3.6 provides a theoretical upper bound on the magnitude of node embeddings after a single convolution layer. It serves as a mathematical guarantee for the stability and behavior of the CGSO. The bound demonstrates that the norm of the output representation $\|\mathbf{H}_u\|_2$ is directly controlled by the learnable parameters Θ and the input feature norm $\|\mathbf{Z}\|_{2,\infty}$. This confirms that the optimization of these scalars prevents the feature representations from growing uncontrollably. The inclusion of $\bar{\Delta}_{u,\mathbf{C}}$, i.e., the weighted average of centrality differences, reveals how the operator responds to local structural transitions. If a node is connected to neighbors with vastly different centralities, this term accounts for how that structural noise is amplified or attenuated during the aggregation step.

It is important to clarify that our primary objective is not the design of the parameterization scheme itself, but rather to investigate whether replacing the standard degree centrality with global centrality metrics enhances graph representation learning. To verify that the observed benefits are intrinsic to the global centrality metrics, we also employed a second, alternative parameterization scheme: the Parameterized Normalized Adjacency framework proposed by Lu et al. (2024). The results of this experiment are in Appendix F.

Dynamic Centrality Selection. In our experiments, we notice the best centrality to vary across datasets, although the walk-based centrality CGSO appears to be frequently outperformed by the k-core and PageRank CGSO. More particularly, in some cases, e.g., PubMed, it is desirable to use local centrality metrics such as the degree, while for other datasets, e.g., Cornell, it's preferable to normalize the adjacency with global centrality metrics. In light of this uncertainty, we can opt for a dynamic, trainable choice of centrality by including both local and global centrality-based CGSO in our CGNN; this can be done by summing the CGSO of the degree matrix with the CGSO of a global centrality metric, e.g., $\mathbf{\Phi} = \mathbf{\Phi}(\mathbf{A}, \mathbf{D}; \Theta) + \mathbf{\Phi}(\mathbf{A}, \mathbf{C}^{(\text{core})}; \widetilde{\Theta})$, where $\Theta = (m_1, m_2, m_3, e_1, e_2, e_3, a)$ and $\widetilde{\Theta} = (\widetilde{m}_1, \widetilde{m}_2, \widetilde{m}_3, \widetilde{e}_1, \widetilde{e}_2, \widetilde{e}_3, \widetilde{a})$ are two distinct set of learnable hyperparameters. The parameters (m_1, m_2, m_3) and $(\widetilde{m}_1, \widetilde{m}_2, \widetilde{m}_3)$ controlling the magnitude of both the local and global CGSOs are then able to learn the relative importance of the local and the global CGSO. In Section 5, we provide experimental results for GNNs with such combined CGSOs.

Time Complexity. We recall that the main complexity of our CGCN model is concentrated around the pre-computation of each centrality score. Computing the degree of all nodes in a graph has a time complexity of $\mathcal{O}(|\mathcal{V}| + |\mathcal{E}|)$, where $|\mathcal{V}|$ is the number of nodes and $|\mathcal{E}|$ is the number of edges in the graph (Cormen et al., 2022). For the PageRank algorithm, each iteration requires one vector-matrix multiplication, which on average requires $\mathcal{O}(|\mathcal{V}|^2)$ time complexity. To compute the core numbers of nodes, we iteratively remove nodes with a degree less than a specified value until all remaining nodes have a degree greater than or equal to that value. This operation can be done with a complexity of $\mathcal{O}(|\mathcal{V}| + |\mathcal{E}|)$. Counting the number of walks of length ℓ for all the nodes can be done via matrix multiplication $\mathbf{A}^\ell \mathbb{1}$. Since our CGSOs preserve the sparsity of the original adjacency matrix, the complexity of the GNNs in which the CGSOs are inserted is unaltered.

4 A Spectral Clustering Perspective of CGSOs

In this section, we analyze CGSOs through the lens of spectral clustering (Ng et al., 2001; Von Luxburg, 2007). Spectral clustering is a technique relying on the spectrum of GSOs to cluster nodes within graphs.

4.1 Spectral Clustering on Stochastic Block Barabási–Albert Models

Here, we investigate the potential of CGSOs when retrieving clusters from synthetic data. Specifically, we propose a new graph generator that is a trivial combination of the well-known Stochastic Block Models

(SBM) (Holland et al., 1983) and Barabási–Albert (BA) models (Albert & Barabasi, 2002), we call this generator the Stochastic Block Barabási–Albert Models (SBBAM). We will now discuss the properties and parameterization of these two graph generators in turn, to then discuss their combination in the SBBAMs.

SBMs. Firstly, in SBMs the node set of the graph is partitioned into a set of K disjoint blocks $\mathcal{B}_1, \ldots, \mathcal{B}_K$, where both the number and size of these blocks are parameters of the model. In SBMs edges are drawn uniformly at random with probability p_{ij} for $i, j \in \{1, \ldots, K\}$ between nodes in blocks $\mathcal{B}_i$ and $\mathcal{B}_j$. Note that this parameterization is often simplified by the following constraints $p_{ij} = q$ if $i = j$ and $p_{ij} = p$ if $i \neq j$. SBMs produce graphs which exhibit cluster structure if $p \neq q$, which makes them a common benchmark for clustering algorithms and subject to extensive theoretical study (Abbe, 2018). Note that SBMs can produce both homophilic graphs if $p < q$ and heterophilic graphs if $q > p$ (Lutzeyer, 2020, Figure 1.2).

BA. The second ingredient of our SBBAMs is the Barabási–Albert (BA) model (Albert & Barabasi, 2002). This model generates random scale-free networks using a preferential attachment mechanism, which is why these models are also sometimes referred to as preferential attachment (PA) models. In this PA mechanism we start out with a seed graph and then add nodes to it one-by-one at successive time steps. For each added node r edges are sampled between the added node and nodes existing in the graph, where the probability of connecting to existing nodes is proportional to their degree in the graph. Hence, high degree nodes are more likely to have their degree rise even further than low-degree nodes in future time steps of the generation process (an effect that is sometimes referred to as 'the rich get richer'). BA models characterize several real-world networks (Barabási & Albert, 1999). A key characteristic of BA models is their degree distribution. In Lemma 4.1, we prove that the density and connectivity of a BA model strongly depend on and positively correlate with the hyperparameter r. Thus, we can generate structurally different BA models by choosing different values of r. Lemma 4.1 is proved in Appendix G.

Lemma 4.1. *Let $\mathcal{G}^{BA}$ be a Barabási–Albert graph of N nodes generated with the hyperparameters $N_0 < N$ the initial number of nodes, $r_0 \leq N_0^2$ the initial number of random edges and r the number of added edges at each time step. Then, the average degree in the network is, $\overline{d}(\mathcal{G}^{BA}) = 2r + 2\frac{r_0}{N} - 2N_0\frac{r}{N}$, and thus, as the number of nodes grows, i.e., as $N \to \infty$ we have $\overline{d}(\mathcal{G}^{BA}) \sim 2r$.*

SBBAMs. In our SBBAMs, we combine SBMs and BA models, by sampling K BA graphs each of size $|\mathcal{B}_1|, \ldots, |\mathcal{B}_K|$ and with parameters $r_1, \ldots, r_K$. We then randomly draw edges between nodes in different BA graphs, $\mathcal{B}_i$ and $\mathcal{B}_j$, uniformly at random with probability p_{ij} for $i, j \in \{1, \ldots, K\}$. In other words, SBBAMs trivally extend SBMs to graphs in which each block is generated using a BA model. This allows us to generate graphs with cluster structure, in which the different clusters exhibit potentially different centrality distributions, which will serve as an informative testbed to explore the clustering obtained from the eigenvectors of our CGSOs.

To better understand the information contained in the spectral decomposition of our CGSOs we will now generate graphs from our SBBAMs and use the spectral clustering algorithm, c.f., see Algorithm 1 in Appendix E, defined on the basis of our CGSOs to attempt to cluster our generated graphs. Specifically, we utilize the three eigenvectors of $\mathbf{\Phi}(\mathbf{A}, \mathbf{C}) = \mathbf{C}^{-1/2}\mathbf{A}\mathbf{C}^{-1/2}$ corresponding to the three largest eigenvalues of different CGSOs defined in Section 3.1. In our experimental setting, each block or BA graph has 100 nodes and an individual parameter r, specifically, $r_1 = 5$, $r_2 = 10$ and $r_3 = 15$. In addition we set $p_{ij} = 0.1$ for all $i \neq j$ with $i, j \in \{1, 2, 3\}$. Figure 2 in Appendix D gives an example of an adjacency matrix sampled from this model. We observe variations in edge density across different blocks and in particular observe homophilic cluster structure in the third block, while the first block appears to be predominantly heterophilic, a rather challenging structure. Figure 3 in Appendix D illustrates the k-core distribution of the three individual BA blocks and the combined SBBAM. Notably, the k-core distribution distinguishes the three BA graphs, while the nodes in the combined graph exhibit less discernibility by k-core.

The results of the spectral clustering on this synthetic graph are presented in Table 1. We compared our CGSOs with the popular Louvain community detection method (Blondel et al., 2008), the node2vec node embedding methods (Grover & Leskovec, 2016) combined with the k-means algorithm, the Walktrap algorithm (Pons & Latapy, 2005), and the Fast Greedy Algorithm which also optimizes modularity by greedily adding nodes to communities (Clauset et al., 2004). We repeated each experiment 10 times, and then reported the mean and standard deviation of Adjusted Mutual Information (AMI) and Adjusted Rand Index (ARI) values.

As expected, normalizing the adjacency matrix with k-core yields higher AMI and ARI values. This observation indicates an improved discernment of each node's membership in its respective cluster, achieved through the incorporation of global centrality metrics. Our CGSO outperforms well-known community detection techniques, such as the Louvain algorithm, which optimizes the modularity, measuring the density of links inside communities compared to links between communities. However, in our setting, some blocks have fewer inter-edges than intra-edges with other blocks, thus making it difficult for the Louvain algorithm to cluster these nodes using the edge density. This experiment further reinforces the intuition that if different clusters exhibit different centrality distributions, then our CGSOs are able to capture this difference better than other clustering alternatives which leads to better clustering performance. Our CGSO therefore represent a valuable addition to existing clustering methodology and potentially a way to develop a wealth of spectral clustering algorithms by simply normalizing the adjacency matrix by a quantity that is likely to correlate with the label distribution.

Table 1: The result of the spectral clustering task on the synthetic graph SSBAM. We present the mean and standard values of AMI and ARI in percentage. ① Clustering baselines; ② CGSO Spectral clustering.

	Method	**AMI in %**	**ARI in %**
①	Fast Greedy	$17.27_{\pm 4.28}$	$19.98_{\pm 5.03}$
	Louvain	$14.37_{\pm 3.34}$	$14.82_{\pm 3.92}$
	Node2Vec	$1.11_{\pm 0.92}$	$1.17_{\pm 0.96}$
	Walktrap	$1.39_{\pm 1.16}$	$1.14_{\pm 0.97}$
②	CGSO w/ $\mathbf{D}$	$23.26_{\pm 3.36}$	$22.95_{\pm 3.86}$
	CGSO w/ $\mathbf{C}^{(\text{core})}$	$\mathbf{35.78_{\pm 4.67}}$	$\mathbf{33.76_{\pm 5.83}}$
	CGSO w/ $\mathbf{C}^{(\ell-\text{walks})}$	$23.85_{\pm 3.64}$	$25.00_{\pm 4.18}$
	CGSO w/ $\mathbf{C}^{(\text{pr})}$	$35.62_{\pm 4.90}$	$33.19_{\pm 6.03}$

4.2 Centrality Recovery in Spectral Clustering

In this experiment, we aim to discern the CGSOs' effectiveness in recovering clusters based on centrality within a real-world graph. Using the Cora and Texas datasets, selected to represent homophilious and heterophilious structures respectively, we chose core numbers to indicate centrality-based clusters. We aim to assess the capacity of various CGSOs to effectively recover clusters reflective of core numbers. This investigation aims to shed light on their potential utility in capturing centralities and hierarchical structures within intricate graphs. Given the core number $\mathbf{C}_{ii}^{(\text{core})}$ for each node $i \in \mathcal{V}$, we establish the ground truth partition $\mathcal{C}_{GT}$ by grouping nodes according to their core numbers. This is formally expressed as the disjoint union of node sets $\mathcal{C}_k$, where each C_k contains all nodes with a core number equal to k, i.e., $\mathcal{C}_{GT} = \bigsqcup_k \mathcal{C}_k = \bigsqcup_k \{i \in \mathcal{V} \mid \mathbf{C}_{ii}^{(\text{core})} = k\}$.

In this experiment, we consider only the largest connected component of each graph. We use the spectral clustering algorithm on the different CGSOs to recover K clusters, where K is the number of possible core numbers in the graph. We repeat each experiment 10 times, and report the average AMI and ARI values. For the walk count node centrality matrix, we used $\ell = 2$ in all our experiments. Additionally, to ensure a fair comparison and standardize the hyperparameter search space, specifically for e_2 and e_3, we multiplied the centrality matrices by a scaling coefficient to linearly map their values to the range $[d_{\min}, d_{\max}]$, defined by the graph's minimum and maximum degrees. It is important to note that this scalar multiplication does not affect the eigenvectors of the operators, serving strictly to align the magnitude of the centrality metrics with that of the degree matrix. We consider the CGSO $\mathbf{\Phi} = \mathbf{C}^{e_2}\mathbf{A}\mathbf{C}^{e_3}$, where we normalize the adjacency matrix with the diagonal matrix $\mathbf{C}$ using different exponents (e_2, e_3).

After the $k-$core decomposition, we apply spectral clustering, c.f., see Algorithm 1 in Appendix E, to evaluate how effectively our CGSOs recover the underlying centralities. Specifically, we utilize the three eigenvectors of $\mathbf{\Phi} = \mathbf{C}^{e_2}\mathbf{A}\mathbf{C}^{e_3}$ corresponding to the three largest eigenvalues of different CGSOs defined in Section 3.1 (since we grid search the parameters values and want to analyse the optimal parameter sets we work with a reduced form of our parametrization namely the term involving the adjacency matrix explicitly). Working with this particular parametrized form our CGSOs further allows us to study the effect of different centrality normalizations with $e_2, e_3 \in [-1.5, 1.5]$. We repeated each experiment 200 times, and then reported the mean and standard deviation of Adjusted Mutual Information (AMI) and Adjusted Rand Index (ARI) values. For consistency, we used the same 200 generated graphs for all the GSOs and the baselines.

In Figures 1 and 5, we report the AMI values using the four centralities. As noticed, while having comparable results between the degree centrality and the count of walks, we reach the highest AMI values by using the PageRank and k-core metrics. Using the degree centrality, we reach the highest AMI value when both

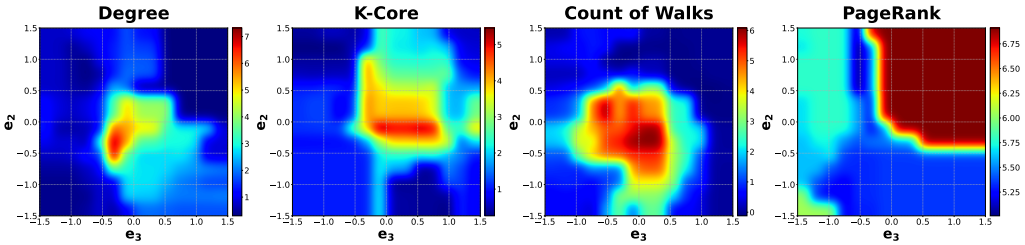

Figure 1: Result for the spectral clustering task on the Cora graph (Sen et al., 2008) with core numbers considered as clusters. We report the values of the Adjusted Mutual Information (AMI) in percentage for different combinations of the exponents (e_2, e_3) in $\mathbf{C}^{e_2}\mathbf{A}\mathbf{C}^{e_3}$.

exponent e_2 and e_3 are negative, while for the k-core and the number of walks, we notice a different behavior as the AMI increases when both the exponents e_2 and e_3 are positive. Thus, nodes with higher k-core and count of walks are important for this setup, i.e., when the node labels are positively correlated with global centrality metrics such as the k-core. We report the ARI values of the same experiment in Appendix E.2.

The intensity of AMI and ARI values in the figures confirms that global centralities significantly outperform degree centrality in recovering core-based clusters. In Figure 1 and 4 on the Cora dataset, the $k-$core and PageRank exhibit broader and more intense high-performance regions compared to the degree, a gap that becomes even more pronounced in Figures 5 and 6 on the Texas dataset, where the degree normalization shows weak recovery (mostly blue/green) while $k-$core and PageRank maintain strong recovery (dark red). This performance differential is driven by a distinct divergence in optimal behavior: the best degree-based normalization consistently appears in the bottom-left quadrant, i.e., negative e_2, e_3, aligning with traditional spectral graph theory which down-weights high-degree nodes to smooth signals. Conversely, for the $k-$core, PageRank, and Walk Counts, the highest AMI scores are concentrated in the top-right quadrant, i.e., positive e_2, e_3, indicating that effective recovery of core-based clusters requires upweighting nodes with high structural importance rather than normalizing them. This distinction is particularly pronounced in heterophilic datasets like Texas, where neighbors are often dissimilar, e.g., students connected to faculty. In such contexts, traditional smoothing via negative exponents destroys the structural signal, effectively making the edges inactive and explaining the poor performance of the degree heatmap. By instead using positive exponents for global centralities, the CGSO amplifies the weights of high-centrality nodes rather than averaging them out.

5 Experimental Evaluation

In this section, we empirically evaluate the proposed Centrality Graph Shift Operators (CGSOs) by integrating them into various GNN backbones and analyzing their performance across 16 real-world datasets. Further evaluations and analysis are detailed in Appendices B, C, F and H.

5.1 Experimental Setup

To demonstrate the flexibility of our approach, we incorporate CGSOs into four distinct GNN backbones,

- **CGCN**: We replace the standard degree-based shift operator in the Graph Convolutional Network (Kipf & Welling, 2017) with our proposed CGSO framework. The propagation rule for the ℓ-th layer becomes, $\mathbf{H}^{(\ell+1)} = \sigma\left(\mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)\mathbf{H}^{(\ell)}\mathbf{W}^{(\ell)}\right)$ where $\mathbf{H}^{(\ell)}$ is the node feature matrix at layer ℓ, $\mathbf{W}^{(\ell)}$ is the learnable weight matrix, and σ is the activation function.
- **CGATv2:** We integrate the CGSO into the Graph Attention Network v2 (Brody et al., 2022). Specifically, for the GATv2 convolution, we apply the parameterized operator $\mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)$ directly to the learned attention matrix, i.e., if $\mathbf{A}_{\text{att}}^{(\ell)}$ represents the matrix of attention coefficients at layer ℓ, the update rule is modified as, $\mathbf{H}^{(\ell+1)} = \sigma\left(\mathbf{\Phi}(\mathbf{A}_{\text{att}}^{(\ell)}, \mathbf{C}; \Theta)\mathbf{H}^{(\ell)}\mathbf{W}^{(\ell)}\right)$.
- **CSGC:** We implement a centrality-based version of Simple Graph Convolutional Networks (Wu et al., 2019) to evaluate the impact of global topological priors on linear message passing. By removing non-

Table 2: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model. ① GCN-based models ② Other vanilla GNN baselines ③ CGCN ④ CGATv2. Highlighted are the **first**, second best results. OOM means *Out of memory*.

	Model	CiteSeer	PubMed	Arxiv-Year	Chamelon	Cornell	Deezer-Europe	Squirrel	Wisconsin
①	GCN w/ $\mathbf{A}$	64.95±0.58	77.12±0.61	38.55±0.71	61.03±1.31	57.03±3.91	57.65±0.84	22.38±6.06	54.51±1.47
	GCN w/ $\mathbf{L}$	28.11±0.54	43.65±0.71	32.81±0.29	56.97±0.75	54.32±0.81	53.92±0.59	36.20±0.84	60.00±2.00
	GCN w/ $\mathbf{Q}$	63.28±0.80	76.57±0.59	33.76±2.36	53.88±2.35	35.41±2.55	56.79±1.79	27.69±2.21	53.33±0.78
	GCN w/ $\mathbf{L_{rw}}$	30.18±0.74	59.68±1.03	36.36±0.24	48.77±0.54	61.62±1.08	54.04±0.44	34.27±0.35	65.10±0.78
	GCN w/ $\mathbf{L_{sym}}$	29.90±0.66	57.68±0.45	36.49±0.14	50.81±0.24	60.27±1.24	53.30±0.45	35.96±0.28	66.08±2.16
	GCN w/ $\hat{\mathbf{A}}$	68.74±0.82	78.45±0.22	42.23±0.25	58.44±0.26	56.22±1.62	60.68±0.45	37.73±0.33	57.45±0.90
	GCN w/ $\mathbf{H}$	66.15±0.55	76.45±0.48	41.27±0.21	56.51±0.47	54.86±1.24	59.45±0.50	38.23±0.47	54.31±0.90
②	GIN	66.62±0.44	78.22±0.52	38.27±3.43	61.60±1.05	45.95±3.42	OOM	25.78±5.12	58.82±1.75
	GAT	59.84±3.14	71.55±4.69	41.26±0.30	63.60±1.70	49.46±8.11	57.67±0.74	40.37±2.89	55.88±2.81
	GATv2	63.01±2.97	73.96±2.22	41.16±0.25	64.14±1.53	43.78±4.80	56.77±1.19	42.63±2.61	53.53±4.12
	PNA	48.89±11.15	70.83±6.51	32.45±2.34	22.89±1.09	40.54±0.00	OOM	OOM	53.14±2.55
	ChebNet	47.02±4.69	52.46±4.97	30.26±0.58	48.86±3.78	55.95±5.14	52.35±1.42	**47.68±2.44**	59.61±6.21
③	CGCN w/ $\mathbf{D}$	68.05±0.44	**78.95±0.25**	46.0±0.27	64.01±5.83	72.43±2.65	58.17±0.46	40.66±2.83	79.02±3.93
	CGCN w/ $\mathbf{C}^{(\text{core})}$	68.40±0.75	78.26±0.31	47.27±0.31	63.68±5.00	72.97±4.19	61.66±0.97	40.59±2.21	85.49±2.93
	CGCN w/ $\mathbf{C}^{(\ell-\text{walks})}$	67.31±0.36	78.57±0.47	**47.36±0.23**	**66.21±2.49**	72.70±3.24	59.15±1.24	46.46±2.16	79.61±2.35
	CGCN w/ $\mathbf{C}^{(\text{pr})}$	67.11±0.56	78.17±4.27	47.14±0.31	47.83±3.39	75.41±3.07	**64.26±0.55**	32.24±1.52	86.67±2.75
④	CGATv2 w/ $\mathbf{D}$	68.60±0.60	77.46±0.51	45.09±0.17	58.22±2.74	**76.49±3.43**	OOM	35.30±2.32	85.69±3.17
	CGATv2 w/ $\mathbf{C}^{(\text{core})}$	68.83±0.66	77.99±0.43	44.38±0.25	55.83±2.28	75.95±3.72	OOM	34.17±1.45	85.10±2.80
	CGATv2 w/ $\mathbf{C}^{(\ell-\text{walks})}$	68.11±0.91	75.43±0.89	46.70±0.21	55.59±2.57	74.32±5.70	OOM	35.49±1.26	83.92±3.01
	CGATv2 w/ $\mathbf{C}^{(\text{pr})}$	**68.97±0.65**	78.46±0.23	41.64±0.18	58.82±1.68	75.68±3.42	OOM	38.41±1.66	**87.06±3.53**

linearities and collapsing K layers into a single propagation step, the CSGC prediction is computed as, $\mathbf{Y} = \texttt{Softmax}\left(\mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)^K \mathbf{X}\mathbf{W}\right)$.

- **CH$_2$GCN:** We incorporate learnable CGSOs into the H$_2$GCN architecture (Zhu et al., 2020), extending a model specifically designed for the structural demands of heterophilic graphs. The model aggregates information from various neighborhoods (ego and intermediate hops) defined by powers of the CGSO. The final representation is obtained by concatenating the node features $\mathbf{X}$ with the aggregated features from K steps of centrality-based propagation, followed by a final dense layer, $\mathbf{H}_{\text{final}} = \texttt{MLP}\left(\left[\mathbf{X} \,\middle\|\, \mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)\mathbf{X} \,\middle\|\, \ldots \,\middle\|\, \mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)^K \mathbf{X}\right]\right)$ where $\|$ denotes the concatenation operation.

Comprehensive details regarding dataset statistics, baseline configurations, and the training scheme are provided in Appendix A. Main results for node classification are reported in Table 2, with extended evaluations on more datasets and GNN backbones in Appendices B and C.

5.2 Experimental Results

We compare the performance of CGCN and CGATv2 models against a comprehensive suite of baselines, including static Graph Shift Operators (GSOs) and state-of-the-art GNN architectures. The classification accuracies for the node classification task are summarized in Table 2, and extended in Appendix B.

Performance Against Static GSOs. The limitations of fixing the graph shift operator to the standard degree-based normalization are evident when comparing CGCN against the standard GCN variants. Across the majority of the 16 benchmark datasets, our learnable CGSO framework significantly outperforms fixed operators. For instance, on heterophilic datasets such as Texas and Cornell, where classical smoothing operators often fail, the CGCN variants achieve substantial accuracy gains, e.g., boosting performance on Texas from 59% with static operators to over 80%. This confirms that the flexibility of CGSO allows the model to adapt the spectral properties of the shift operator to the specific structural demands of the graph.

Comparison with Standard GNN Architectures. Our models also demonstrate superior performance compared to established baselines such as GAT, GATv2, GIN, and ChebNet. By replacing the fixed Laplacian with our parameterized CGSO, CGCN consistently surpasses the vanilla GCN and GIN. This is particularly notable in datasets where global structural information is crucial, which local message passing struggles to capture efficiently. The integration of centrality-based normalization into the attention mechanism in CGATv2 yields further improvements compared to the performance of GATv2.

Adaptability to the Graph Learning Task. A key finding from our evaluation is the model's consistency across diverse graph types. While standard degree-based GNNs perform adequately on homophilic graphs,

Table 3: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model.

Model	CiteSeer	PubMed	Arxiv-Year	Chamelon	Cornell	Deezer-Europe	Squirrel	Wisconsin
CGCN w/ $\mathbf{D}-\mathbf{C}^{(\text{core})}$	**69.00**$\pm$**0.64**	78.77$\pm$0.34	48.37$\pm$0.15	**65.04**$\pm$**4.37**	73.78$\pm$4.02	**59.81**$\pm$**0.51**	40.74$\pm$4.77	83.33$\pm$3.42
CGCN w/ $\mathbf{D}-\mathbf{C}^{(\ell-\text{walks})}$	67.99$\pm$0.55	**79.13**$\pm$**0.16**	**49.38**$\pm$**0.31**	62.37$\pm$3.93	74.32$\pm$2.77	59.30$\pm$0.70	**44.74**$\pm$**1.96**	81.37$\pm$3.64
CGCN w/ $\mathbf{D}-\mathbf{C}^{(\text{pr})}$	68.45$\pm$0.60	78.54$\pm$0.35	45.96$\pm$0.18	64.32$\pm$3.13	**75.95**$\pm$**4.90**	59.28$\pm$0.75	42.80$\pm$6.58	**85.29**$\pm$**3.31**

e.g., Cora, PubMed, they suffer heavily on heterophilic graphs. Our results show that global centralities, specifically PageRank and $k-$core, provide a critical advantage in these settings. For example, on the Squirrel and Chameleon datasets, CGCN variants using global centrality outperform the degree-based baselines, validating our hypothesis that global positioning is a more reliable signal than local neighborhood aggregation when the homophily assumption is violated. Furthermore, we observe that the learnable self-loop parameter a tends to be negative for non-homophilous datasets, effectively allowing the model to learn a *sharpening* filter rather than a smoothing one, a flexibility absent in traditional baselines.

Backbone-Specific Centrality Preferences. It is worth noting that in some cases, the optimal choice of centrality is consistent across different backbones. Specifically, for datasets such as Texas, Ogbn-Arxiv, Computers, Arxiv-year, and Squirrel, we observe that both the CGCN and CGATv2 variants tend to prefer the same centrality metric, highlighting a robust preference intrinsic to these graphs. However, this is not a universal rule, as the optimal centrality varies across backbones for other datasets. This divergence is expected for several reasons. First, CGCN employs a static adjacency while CGATv2 operates on learned attention coefficients. Second, different architectures project data into distinct feature spaces. This behavior is supported by Lemma 3.6, which establishes a bound on the node embedding norm $||\mathbf{H}_u||_2$ and characterizes how the interplay between the centrality weighted difference $\Delta_{u,C}$ and the learnable parameters controls signal amplification. Consequently, the *best* topological prior is not only graph-dependent but also contingent on how the specific architecture modulates the embedding space.

Dynamic Centrality Selection. General intuition on the choice of centrality that we can provide relates to the fact that the node degree is a local centrality metric, while the remaining three centralities we consider correspond to global metrics. Therefore, it is apparent that if the learning task only requires local information a degree-based normalization of the GSO is likely beneficial, while global centrality metrics are appropriate if more global information is required. Beyond this statement it seems to be difficult to provide general guidance on the choice of the global centrality metrics. Therefore, including both local and global centrality-based CGSO in the CGNN might be optimal to dynamically distinguish the best type of centrality. We present the results in Tables 3 and 8. By combining local and global centralities in the CGNNs, we usually increase their performance. Notably, this combination strategy proves that extending the operator with additional global priors generally does not deteriorate performance compared to any single centrality choice, ensuring that the model retains the benefits of the best individual metric while potentially gaining from the fusion.

6 Conclusion

In this work, we have proposed CGSOs, a novel class of Graph Shift Operators (GSOs) that can leverage different centrality metrics, such as node degree, PageRank score, core number, and the count of walks of a fixed length. Furthermore, we have modified the message-passing steps of Graph Neural Networks (GNNs) to integrate these CGSOs, giving rise to a novel model class the CGNNs. Experimental results comparing our CGNN models to existing vanilla GNNs show the superior performance of CGNN on many real-world datasets. These experiments furthermore allowed us to analyze the optimal parameters of our CGSO, which led to new and interesting insight such as for example an apparent benefit of negatively weighted self-loops for non-homophilous graphs. To further understand the cases where each centrality is beneficial, we conducted additional experiments focused on spectral clustering using synthetic graphs. Through these experiments, we identified instances where CGSOs outperformed conventional GSOs.

References

Emmanuel Abbe. Community detection and stochastic block models: recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018.

Reka Albert and Albert-Laszlo Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 2002.

Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439): 509–512, 1999.

Austin R Benson, David F Gleich, and Jure Leskovec. Higher-order organization of complex networks. *Science*, 353(6295):163–166, 2016.

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

Aleksandar Bojchevski, Johannes Gasteiger, Bryan Perozzi, Amol Kapoor, Martin Blais, Benedek Rózemberczki, Michal Lukasik, and Stephan Günnemann. Scaling graph neural networks with approximate pagerank. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2464–2473, 2020.

Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? In *International Conference on Learning Representations (ICLR)*, 2022.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *International Conference on Learning Representations (ICLR)*, 2014.

Xing Shi Cai, Pietro Caputo, Guillem Perarnau, and Matteo Quattropani. Rankings in directed configuration models with heavy tailed in-degrees. *The Annals of Applied Probability*, 33(6B):5613–5667, 2023.

Jeff Cheeger. A lower bound for the smallest eigenvalue of the laplacian. In *Problems in Analysis*, pp. 195–200. Princeton University Press, 1970.

F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997a.

Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997b.

Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical review E*, 70(6):066111, 2004.

Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.

Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. Principal neighbourhood aggregation for graph nets. *Advances in Neural Information Processing Systems*, 33:13260–13271, 2020.

D Cvetkovic, Michael Doob, and Horst Sachs. Spectra of graphs. theory and application. *Pure and Applied Mathematics*, 1980.

Dragoš Cvetković and Slobodan K Simić. Towards a spectral theory of graphs based on the signless laplacian, ii. *Linear Algebra and its Applications*, 432(9):2257–2272, 2010.

Dragoš Cvetković, Peter Rowlinson, and Slobodan Simić. *An Introduction to the Theory of Graph Spectra*. London Mathematical Society Student Texts. Cambridge University Press, 2009.

George Dasoulas, Johannes F. Lutzeyer, and Michalis Vazirgiannis. Learning parametrised graph shift operators. In *International Conference on Learning Representations (ICLR)*, 2021.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

William E Donath and Alan J Hoffman. Lower bounds for the partitioning of graphs. *IBM Journal of Research and Development*, 17(5):420–425, 1973.

Claire Donnat and So Won Jeong. Studying the effect of gnn spatial convolutions on the embedding space's geometry. In *Uncertainty in artificial intelligence*, pp. 539–548. PMLR, 2023.

Leonhard Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae*, pp. 128–140, 1736.

Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23(2):298–305, 1973.

Linton C Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pp. 35–41, 1977.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. *ICLR*, 2019.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning (ICML)*, pp. 1263–1272. PMLR, 2017.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.

William L Hamilton. *Graph representation learning*. Morgan & Claypool Publishers, 2020.

Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

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

Ming Jin, Heng Chang, Wenwu Zhu, and Somayeh Sojoudi. Power up! robust graph convolutional network via graph powering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 8004–8012, 2021.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

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

Christian Koke and Daniel Cremers. Holonets: Spectral convolutions do extend to directed graphs. *International Conference on Learning Representations (ICLR)*, 2024.

Dénes König. Theorie der endlichen und unendlichen graphen: Kombinatorische topologie der streckenkomplexe. *Akademische Verlagsgesellschaft M. R. H., Leipzig*, 1936.

Devin Kreuzer, Dominique Beaini, Will Hamilton, Vincent Létourneau, and Prudencio Tossou. Rethinking graph transformers with spectral attention. *Advances in Neural Information Processing Systems*, 34: 21618–21629, 2021.

John Boaz Lee, Ryan A Rossi, Xiangnan Kong, Sungchul Kim, Eunyee Koh, and Anup Rao. Graph convolutional networks with motif-based attention. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pp. 499–508, 2019.

Derek Lim, Felix Hohne, Xiuyu Li, Sijia Linda Huang, Vaishnavi Gupta, Omkar Bhalerao, and Ser Nam Lim. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Advances in Neural Information Processing Systems*, 34:20887–20902, 2021.

Qincheng Lu, Jiaqi Zhu, Sitao Luan, and Xiao-Wen Chang. Flexible diffusion scopes with parameterized laplacian for heterophilic graph learning. In *The Third Learning on Graphs Conference*, 2024.

Johannes Lutzeyer. Network representation matrices and their eigenproperties: A comparative study. PhD Thesis: Imperial College London, 2020.

Liheng Ma, Chen Lin, Derek Lim, Adriana Romero-Soriano, Puneet K Dokania, Mark Coates, Philip Torr, and Ser-Nam Lim. Graph inductive biases in transformers without message passing. In *International Conference on Machine Learning*, pp. 23321–23337. PMLR, 2023.

Fragkiskos D. Malliaros, Christos Giatsidis, Apostolos N. Papadopoulos, and Michalis Vazirgiannis. The core decomposition of networks: theory, algorithms and applications. *VLDB J.*, 29(1):61–92, 2020.

Gonzalo Mateos, Santiago Segarra, Antonio G Marques, and Alejandro Ribeiro. Connecting the dots: Identifying network structure via graph signal processing. *IEEE Signal Processing Magazine*, 36(3):16–43, 2019.

Alexander Modell and Patrick Rubin-Delanchy. Spectral clustering under degree heterogeneity: a case for the random walk laplacian. *arXiv preprint arXiv:2105.00987*, 2021.

Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14, 2001.

Giannis Nikolentzos, George Dasoulas, and Michalis Vazirgiannis. k-hop graph neural networks. *Neural Networks*, 130:195–205, 2020.

Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst. Graph signal processing: Overview, challenges, and applications. *Proceedings of the IEEE*, 106(5):808–828, 2018.

Henri Poincaré. Second complément à l'analysis situs. *Proceedings of the London Mathematical Society*, 1(1): 277–308, 1900.

Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks. In *Computer and Information Sciences-ISCIS 2005: 20th International Symposium, Istanbul, Turkey, October 26-28, 2005. Proceedings 20*, pp. 284–293. Springer, 2005.

Ladislav Rampášek, Michael Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. *Advances in Neural Information Processing Systems*, 35:14501–14515, 2022.

Benedek Rozemberczki and Rik Sarkar. Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pp. 1325–1334, 2020.

Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2):cnab014, 2021.

Aliaksei Sandryhaila and José MF Moura. Discrete signal processing on graphs. *IEEE transactions on signal processing*, 61(7):1644–1656, 2013.

Camellia Sarkar and Sarika Jalan. Spectral properties of complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(10), 2018.

Stephen B Seidman. Network structure and minimum degree. *Social networks*, 5(3):269–287, 1983.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93, Sep. 2008.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation, 2018.

Zoran Stanić. *Inequalities for graph eigenvalues*, volume 423. Cambridge University Press, 2015.

Ke Sun, Zhanxing Zhu, and Zhouchen Lin. Adagcn: Adaboosting graph convolutional networks into deep models. *International Conference on Learning Representations (ICLR)*, 2019.

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

Amanda L Traud, Peter J Mucha, and Mason A Porter. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165–4180, 2012.

Ariel R. Ramos Vela, Johannes F. Lutzeyer, Anastasios Giovanidis, and Michalis Vazirgiannis. Improving graph neural networks at scale: Combining approximate pagerank and corerank. In *NeurIPS 2022 Workshop: New Frontiers in Graph Learning*, 2022.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17:395–416, 2007.

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

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

Junlong Zhang and Yu Luo. Degree centrality, betweenness centrality, and closeness centrality in social network. In *2017 2nd international conference on modelling, simulation and applied mathematics (MSAM2017)*, pp. 300–303. Atlantis press, 2017.

Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. An end-to-end deep learning architecture for graph classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems*, 33:7793–7804, 2020.

Appendix Contents

A Experimental Framework and Reproducibility Information **19**

A.1 Baselines . . . 19

A.2 Statistics of the Node Classification Datasets . . . 19

A.3 Implementation Details . . . 20

A.4 Weights Initialization . . . 20

A.5 Hyperparameter Configurations . . . 20

B Extended Node Classification Benchmarks **20**

B.1 Results of CGCN and CGATv2 . . . 20

B.2 Combining Local and Global Centralities . . . 21

C Generalization to Linear and Heterophilic Architectures **22**

C.1 Simple Graph Convolutional Networks (SGC) . . . 22

C.2 H_2 Graph Convolution Networl (H_2GCN) . . . 22

D Topology and Structure of the Synthetic SBBAM Benchmarks **23**

E CGSO Spectral Clustering **24**

E.1 Algorithmic Formulation . . . 24

E.2 Additional Results for the Spectral Clustering Task . . . 24

E.2.1 ARI values for Cora . . . 24

E.2.2 AMI Values for Texas . . . 24

E.2.3 ARI values for Texas . . . 25

F Alternative CGSO Parameterizations **25**

G Proofs of Propositions and Lemmas **25**

G.1 Proof of Proposition 3.2 . . . 26

G.2 Proof of Proposition 3.3 . . . 28

G.3 Proof of Proposition 3.5 . . . 29

G.4 Proof of Lemma 3.6 . . . 32

G.5 Proof of Lemma 4.1 . . . 33

H Interpretability Analysis: Learned Parameters and Graph Statistics **33**

H.1 Degree Centrality . . . 35

H.2 k-Core Centrality . . . 35

H.3 PageRank Centrality . . . 35

H.4 Count of Walks Centrality . . . 36

I Sensitivity Analysis: Impact of Weight Initialization **36**

I.1 Degree Centrality Initialization 36

I.2 $k-$Core Centrality Initialization 36

I.3 Walk Count Centrality Initialization 37

I.4 PageRank Centrality Initialization 38

A Experimental Framework and Reproducibility Information

In this section, we present the benchmark datasets, the baselines used for our experiments, and the process used for the training.

A.1 Baselines

To evaluate the efficacy of our framework, we compare our CGNNs against two categories of baselines, ranging from classical graph shift operators (GSOs) to state-of-the-art GNN architectures:

- **Static GSOs:** We compare the proposed CGCN against a standard GCN implemented with various static shift operators, including the Adjacency matrix $\mathbf{A}$ (Xu et al., 2019); Unnormalized Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$; Signless Laplacian $\mathbf{Q} = \mathbf{D} + \mathbf{A}$ (Cvetković & Simić, 2010); Random-walk Normalized Laplacian $\mathbf{L_{rw}} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$; Symmetric Normalized Laplacian $\mathbf{L_{sym}} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$; Normalized Adjacency $\hat{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ (Kipf & Welling, 2017); and Mean Aggregation $\mathbf{H} = \mathbf{D}^{-1}\mathbf{A}$ (Zhang et al., 2018).
- **Standard GNN Baselines:** We further compare our models against established GNN architectures, including Graph Attention Network (GAT) (Veličković et al., 2018), Graph Attention Network v2 (GATv2) (Brody et al., 2022), Graph Isomorphism Network (GIN) (Xu et al., 2019), Principal Neighbourhood Aggregation (PNA) (Corso et al., 2020), and ChebNet (Defferrard et al., 2016).

A.2 Statistics of the Node Classification Datasets

We use 16 widely used datasets in the GNN literature. In particular, we run experiments on the node classification task using the citation networks Cora, CiteSeer, and PubMed (Sen et al., 2008), the co-authorship networks CS and Physics (Shchur et al., 2018), the citation network between Computer Science arXiv papers Ogbn-Arxiv (Hu et al., 2020), the Amazon Computers and Amazon Photo networks (Shchur et al., 2018), the non-homophilous datasets Penn94 (Traud et al., 2012), Deezer-Europe (Rozemberczki & Sarkar, 2020) and Arxiv-Year (Hu et al., 2020), and the disassortative datasets Chameleon, Squirrel (Rozemberczki et al., 2021), and Cornell, Texas, Wisconsin from the WebKB dataset (Lim et al., 2021). Characteristics and information about the datasets utilized in the node classification part of the study are presented in Table 4.

Table 4: Statistics of the node classification datasets used in our experiments.

Dataset	#Features	#Nodes	#Edges	#Classes	Edge Homophily
Cora	1,433	2,708	5,208	7	0.809
CiteSeer	3,703	3,327	4,552	6	0.735
PubMed	500	19,717	44,338	3	0.802
CS	6,805	18,333	81,894	15	0.808
arxiv-year	128	169,343	1,157,799	5	0.218
chameleon	2,325	2,277	62,792	5	0.231
Cornell	1,703	183	557	5	0.132
deezer-europe	31,241	28,281	185,504	2	0.525
squirrel	2,089	5,201	396,846	5	0.222
Wisconsin	1,703	251	916	5	0.206
Texas	1,703	183	574	5	0.111
Photo	745	7,650	238,162	8	0.827
ogbn-arxiv	128	169,343	2,315,598	40	0.654
Computers	767	13752	491,722	10	0.777
Physics	8,415	34,493	495,924	5	0.931
Penn94	4,814	41,554	2,724,458	3	0.470

A.3 Implementation Details

We train all the models using the Adam optimizer (Kingma & Ba, 2014). To account for the impact of random initialization, each experiment was repeated 10 times, and the mean and standard deviation of the results were reported. The experiments have been run on both a NVIDIA A100 GPU and a RTX A6000 GPU.

Training of our CGNN. We train our model using the Adam optimizer (Kingma & Ba, 2014), with a weight decay on the parameters of 5×10^{-4}, an initial learning rate of 0.005 for the exponential parameters and an initial learning rate of 0.01 for all other model parameters. We repeated the training 10 times to test the stability of the model. We tested 7 initialization of the weights $(m_1, m_2, m_3, e_1, e_2, e_3, a)$. These initializations are reported in Table 5, and correspond to classical GSOs when the chosen centrality is the degree. For the Cora, CiteSeer, and Pubmed datasets, we used the provided train/validation/test splits. For the remaining datasets, we followed the framework of Lim et al. (2021); Rozemberczki et al. (2021).

A.4 Weights Initialization

In this part, we present the different initializations of CGSO. When the chosen centrality is the degree, i.e., $\mathbf{V} = \mathbf{D}$, the initializations corresponds to popular classical GSO (Dasoulas et al., 2021).

Table 5: Differenet initialization of the weights $(m_1, m_2, m_3, e_1, e_2, e_3, a)$.

Initialization of $(m_1, m_2, m_3, e_1, e_2, e_3, a)$	Corresponding GSO	Description when $\mathbf{C} = \mathbf{D}$
$(0, 1, 0, 0, 0, 0, 0)$	$\mathbf{A}(\mathbf{C}) = \mathbf{A}$	Adjacency matrix
$(1, -1, 0, 1, 0, 0, 0)$	$\mathbf{L}(\mathbf{C}) = \mathbf{C} - \mathbf{A}$	Unnormalised Laplacian matrix
$(1, 1, 0, 1, 0, 0, 0)$	$\mathbf{Q}(\mathbf{C}) = \mathbf{C} + \mathbf{A}$	Signless Laplacian matrix
$(0, -1, 1, 0, -1, 0, 0)$	$\mathbf{L_{rw}}(\mathbf{C}) = \mathbf{I} - \mathbf{C}^{-1}\mathbf{A}$	Random-walk Normalised Laplacian
$(0, -1, 1, 0, -1/2, -1/2, 0)$	$\mathbf{L_{sym}}(\mathbf{C}) = \mathbf{I} - \mathbf{C}^{-1/2}\mathbf{A}\mathbf{C}^{-1/2}$	Symmetric Normalised Laplacian
$(0, 1, 0, 0, -1/2, -1/2, 1)$	$\hat{\mathbf{A}}(\mathbf{C}) = \mathbf{C}^{-1/2}\mathbf{A}_1\mathbf{C}^{-1/2}$	Normalised Adjacency matrix
$(0, 1, 0, 0, -1, 0, 0)$	$\mathbf{H}(\mathbf{C}) = \mathbf{C}^{-1}\mathbf{A}$	Mean Aggregation Operator

A.5 Hyperparameter Configurations

For a more balanced comparison, however, we use the same training procedure for all the models. The hyperparameters in each dataset where performed using a grid search on the classical GCN, i.e., with the GSO being the *normalised adjacency*, over the following search space,

- Hidden size : $[16, 32, 64, 128, 256, 512]$,
- Learning rate : $[0.1, 0.01, 0.001]$,
- Dropout probability: $[0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]$.

The number of layers was fixed to 2. The optimal hyperparameters can be found in Table 6.

B Extended Node Classification Benchmarks

B.1 Results of CGCN and CGATv2

To further evaluate our *CGCN* and *CGATv2*, we compute their performance on additional datasets. The results of this study are presented in Table 7.

Table 6: Hyperparameters used in our experiments.

Dataset	Hidden Size	Learning Rate	Dropout Probability
Cora	64	0.01	0.8
CiteSeer	64	0.01	0.4
PubMed	64	0.01	0.2
CS	512	0.01	0.4
Arxiv-year	512	0.01	0.2
chameleon	512	0.01	0.2
cornell	512	0.01	0.2
deezer-europe	512	0.01	0.2
squirrel	512	0.01	0.2
Wisconsin	512	0.01	0.2
Texas	512	0.01	0.2
Photo	512	0.01	0.6
OGBN-Arxiv	512	0.01	0.5
Computers	512	0.01	0.2
Physics	512	0.01	0.4
Penn94	64	0.01	0.2

Table 7: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model. ① GCN Based models ② Other Vanilla GNN baselines ③ CGCN ④ CGATv2. Highlighted are the **first**, second best results. OOM means *Out of memory*.

	Model	Cora	Texas	Photo	Ogbn-Ariv	CS	Computers	Physics	Penn94
①	GCN w/ $\mathbf{A}$	78.61±0.51	63.51±2.18	82.31±2.61	13.23±6.44	87.70±1.25	69.32±3.64	88.92±1.93	52.35±0.36
	GCN w/ $\mathbf{L}$	31.57±0.41	84.32±2.65	27.42±6.23	10.91±1.49	23.75±3.22	26.27±3.89	35.31±3.71	65.31±0.59
	GCN w/ $\mathbf{Q}$	77.32±0.50	60.54±1.32	77.06±6.73	10.50±1.97	89.42±1.31	47.72±18.37	90.69±2.13	53.46±2.16
	GCN w/ $\mathbf{L_{rw}}$	26.59±1.11	78.38±2.09	24.60±4.21	8.07±0.07	26.34±4.09	13.76±3.96	28.19±3.75	69.82±0.44
	GCN w/ $\mathbf{L_{sym}}$	26.79±0.50	71.35±1.32	22.82±2.67	20.18±0.24	24.39±1.96	16.06±5.19	30.94±3.11	70.57±0.30
	GCN w/ $\hat{\mathbf{A}}$	**80.84±0.40**	60.81±1.81	78.94±1.65	65.80±0.14	91.52±0.75	68.91±3.00	**93.72±0.80**	74.60±0.42
	GCN w/ $\mathbf{H}$	80.15±0.37	59.46±0.00	73.95±4.75	63.34±0.15	90.98±1.84	62.01±4.36	92.16±1.12	71.78±0.47
②	GIN	79.06±0.47	57.03±1.89	83.00±2.52	9.30±6.42	89.53±1.20	55.89±13.45	89.15±2.44	OOM
	GAT	77.73±1.83	52.16±6.74	71.56±3.48	67.36±0.13	67.67±3.96	59.73±3.59	80.91±4.48	73.85±1.38
	GATv2	74.53±2.48	48.11±3.78	73.49±2.49	68.14±0.07	70.13±4.92	58.18±4.76	83.28±3.68	75.54±2.54
	PNA	56.67±10.53	63.51±4.05	16.75±5.59	OOM	OOM	13.62±6.39	OOM	OOM
	ChebNet	58.38±2.64	64.32±4.65	67.40±3.56	61.83±1.32	73.45±2.69	49.55±2.3	83.13±2.87	**81.14±2.43**
③	CGCN w/ $\mathbf{D}$	78.99±0.68	80.54±5.38	89.06±1.6	68.87±0.34	91.7±1.09	79.12±2.77	90.9±1.8	72.92±0.31
	CGCN w/ $\mathbf{C}^{(\text{core})}$	79.80±0.43	80.54±6.02	88.53±1.40	67.81±0.41	91.90±0.70	77.35±2.67	91.98±1.49	78.11±3.74
	CGCN w/ $\mathbf{C}^{(\ell-\text{walks})}$	79.52±1.11	81.62±4.15	87.11±2.1	69.09±0.3	90.94±0.88	**80.24±1.40**	92.66±1.11	73.22±0.40
	CGCN w/ $\mathbf{C}^{(\text{pr})}$	79.70±1.0	**85.14±4.05**	81.28±6.08	68.56±0.18	88.56±1.13	65.54±6.43	90.48±1.74	72.64±0.42
④	CGATv2 w/ $\mathbf{D}$	79.08±0.64	83.51±5.19	87.97±1.77	70.09±0.10	91.48±1.05	78.62±2.35	91.32±1.18	72.96±0.27
	CGATv2 w/ $\mathbf{C}^{(\text{core})}$	79.03±0.96	83.78±6.62	**89.72±1.54**	70.50±0.22	**91.91±1.06**	77.31±3.33	91.15±1.07	72.86±0.41
	CGATv2 w/ $\mathbf{C}^{(\ell-\text{walks})}$	78.58±0.58	81.89±4.99	88.11±2.02	**70.51±0.24**	90.73±1.49	79.09±1.66	90.71±0.81	73.01±0.37
	CGATv2 w/ $\mathbf{C}^{(\text{pr})}$	78.61±0.38	83.78±4.01	88.78±4.98	69.26±0.12	91.77±1.00	74.95±3.05	92.73±1.44	75.16±0.69

B.2 Combining Local and Global Centralities

In this section, we extend the results shown in Table 3 to other node classification datasets. The results of this experiments are shown in Table 8.

Table 8: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model.

Model	Cora	Texas	Photo	ogbn-arxiv	CS	Computers	Physics	Penn94
CGCN w/ $\mathbf{D}-\mathbf{C}^{(\text{core})}$	**79.88±0.38**	79.73±3.87	**89.06±1.28**	67.67±0.26	91.63±0.95	78.41±1.94	91.90±1.38	**80.28±2.93**
CGCN w/ $\mathbf{D}-\mathbf{C}^{(\ell-\text{walks})}$	79.47±0.79	81.35±3.72	88.86±1.65	**69.58±0.24**	**91.78±1.04**	**79.05±2.14**	**91.92±1.51**	72.50±0.48
CGCN w/ $\mathbf{D}-\mathbf{C}^{(\text{pr})}$	79.84±0.40	**84.32±6.49**	88.46±2.29	69.49±0.34	90.04±0.57	77.43±2.58	90.88±1.95	71.03±5.83

C Generalization to Linear and Heterophilic Architectures

C.1 Simple Graph Convolutional Networks (SGC)

In Tables 9 and 10, we present the results of our centrality-aware Simple Graph Convolutional Networks *CGSC* of 2 layers. As noticed in most cases, by incorporating our CGSO, we outperform the classical SGC.

Table 9: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model. ① CSGC with nodes centrality, ② SGC. Highlighted are the **best results**.

	Model	CiteSeer	PubMed	Arxiv-Year	Chamelon	Cornell	Deezer-Europe	Squirrel	Wisconsin
①	CSGC w/ $\mathbf{D}$	**67.16±0.14**	77.39±0.18	34.99±0.27	**61.58±4.83**	67.84±3.72	58.22±0.47	39.84±1.12	72.75±5.58
	CSGC w/ $\mathbf{C}^{(\text{core})}$	66.64±0.17	77.39±0.15	34.66±0.19	50.55±2.86	71.89±5.69	61.29±0.50	34.71±2.16	82.35±4.21
	CSGC w/ $\mathbf{C}^{(\ell-\text{walks})}$	66.49±0.16	**78.10±0.18**	**36.86±0.18**	56.34±2.55	**74.05±5.57**	59.69±0.50	**44.97±2.06**	76.27±5.51
	CSGC w/ $\mathbf{C}^{(\text{pr})}$	62.83±0.28	74.98±0.2	35.0±0.13	45.92±2.09	73.51±3.15	58.94±0.48	28.65±1.12	**83.92±3.14**
②	SGC	64.96±0.10	75.72±0.12	26.61±0.24	38.44±4.41	45.41±5.77	**62.66±0.48**	19.88±0.79	53.53±8.09

Table 10: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model. ① CSGC with nodes centrality, ② SGC. Highlighted are the **best results**.

	Model	Cora	Texas	Photo	ogbn-arxiv	CS	Computers	Physics	Penn94
①	CSGC w/ $\mathbf{D}$	79.57±0.13	76.76±3.24	**90.15±1.51**	**67.67±0.04**	92.21±0.48	78.14±3.38	91.72±1.51	75.88±4.96
	CSGC w/ $\mathbf{C}^{(\text{core})}$	76.51±0.25	75.68±5.41	82.01±18.13	61.37±0.04	92.07±0.61	69.42±14.5	91.15±0.94	75.10±4.12
	CSGC w/ $\mathbf{C}^{(\ell-\text{walks})}$	**79.80±0.17**	76.76±4.71	88.87±1.86	66.92±0.07	91.33±0.8	**79.38±1.85**	**92.16±0.92**	**81.76±3.73**
	CSGC w/ $\mathbf{C}^{(\text{pr})}$	67.12±0.95	**81.08±3.82**	76.13±3.35	52.79±0.05	**92.42±0.81**	61.88±2.35	91.18±1.07	75.29±4.31
②	SGC	78.79±0.13	58.65±4.20	24.0±11.82	60.48±0.14	70.78±5.47	11.34±11.67	91.69±1.48	66.63±0.62

C.2 H_2 Graph Convolution Networl (H_2GCN)

In this section, we incorporate our learnable CGSOs into H_2GCN (Zhu et al., 2020), designed for heterophilic graphs. We compared the results of CH_2GCN and H_2GCN on datasets with low homophily. We report the results of this experiment in Table 11. As noticed, our CH_2GCN outperforms H_2GCN.

Table 11: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model. ① CH2GCN with nodes centrality, ② H2GCN. Highlighted are the **best results**.

	Model	Texas	Cornell	Wisconsin	Chameleon
①	CH_2GCN w/ $\mathbf{D}$	**79.73±5.02**	68.65±5.16	79.80±4.02	**67.89±4.23**
	CH_2GCN w/ $\mathbf{C}^{(\text{core})}$	78.92±5.77	**68.92±7.28**	79.80±3.40	60.00±5.63
	CH_2GCN w/ $\mathbf{C}^{(\ell-\text{walks})}$	78.11±6.10	**68.92±6.19**	**82.35±5.04**	44.28±2.32
	CH_2GCN w/ $\mathbf{C}^{(\text{pr})}$	60.27±5.41	44.86±7.76	52.35±7.75	31.95±5.79
②	H_2GCN	56.76±6.73	51.08±6.89	55.29±5.10	63.93±2.07

D Topology and Structure of the Synthetic SBBAM Benchmarks

In Figure 2, we give an example of an adjacency matrix sampled from SBBAM model, presented previously in Section 4.1. Yellow points indicate edges, while purple points represent non-edges. Notably, there are variations in edge density across different blocks: while the first block appears to be predominantly heterophilic, the third block inherits a homophilic cluster structure.

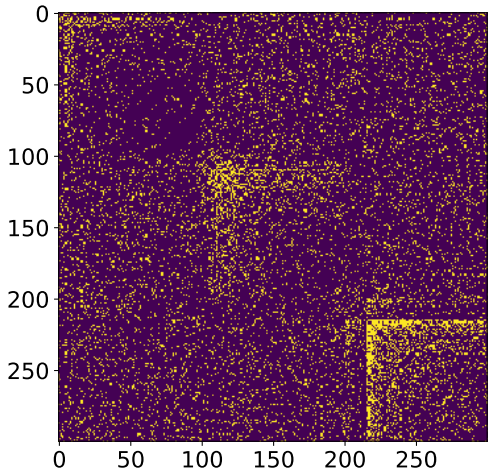

Figure 2: The adjacency matrix of the synthetic graph generated from an SBBAM with 3 blocks.

Figure 3 illustrates the k-core distribution of the three individual BA blocks and the combined SBBAM. Notably, the k-core distribution distinguishes the three BA graphs, while the nodes in the combined graph exhibit less discernibility by k-core.

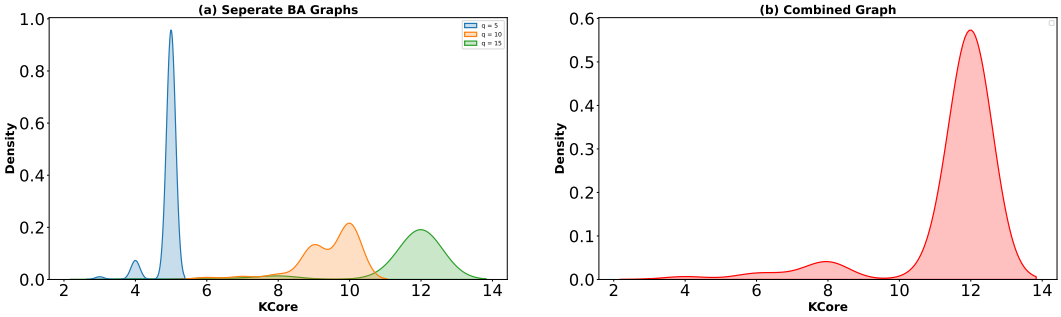

Figure 3: The left figure represents the k-core distributions of three different BA models with the hyperparameters $r = 5, 10$ and 15 serving as blocks of of our SBBAM. The right figure represents the k-core distribution of the SBBAM.

E CGSO Spectral Clustering

E.1 Algorithmic Formulation

In Algorithm 1, we outline the steps of the Spectral Clustering Algorithm using CGSOs. This algorithm is applied on SBBAMs for cluster recovery and on the Cora dataset for centrality recovery.

Algorithm 1: Spectral Clustering using the Centrality GSOs

Inputs: Graph $\mathcal{G}$, CGSO Φ, Number of clusters to retrieve k.

1. Compute the eigenvalues $\{\lambda\}_{i=1}^{N}$ and eigenvectors $\{u\}_{i=1}^{N}$ of $\mathbf{\Phi}$;
2. Consider only the eigenvectors $U \in \mathbb{R}^{N \times k}$ corresponding to the k largest eigenvalues;
3. Cluster rows of U, corresponding to nodes in the graph, using the K-Means algorithm to retrieve a node partition P with k clusters;

$$\mathcal{P} = \text{K-Means}(U, k);$$

return $\mathcal{P}$;

E.2 Additional Results for the Spectral Clustering Task

E.2.1 ARI values for Cora

In this section, we report the ARI value of the spectral clustering task described in Section 4.

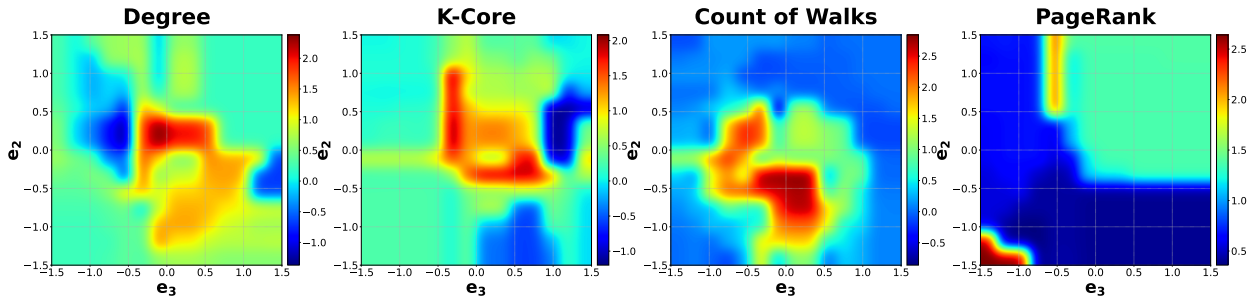

Figure 4: Result for the spectral clustering task on Cora graph with core numbers considered as clusters. We report the values of the Adjusted Rand Information (ARI) in % different combination of the exponents (e_2, e_3) in $\mathbf{C}^{e_2}\mathbf{A}\mathbf{C}^{e_3}$.

E.2.2 AMI Values for Texas

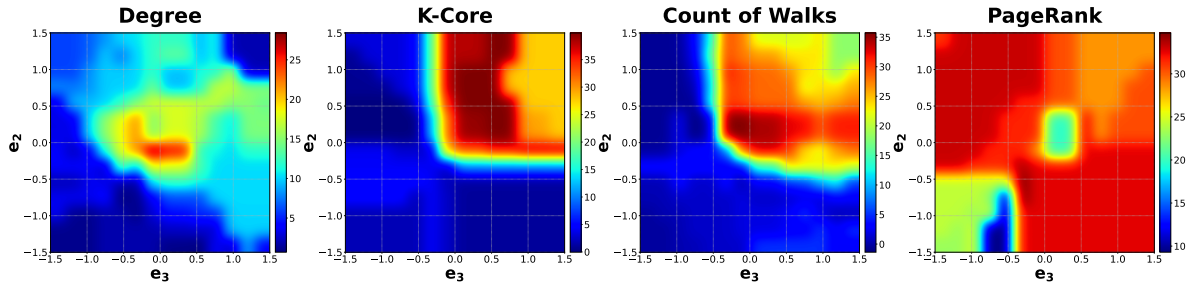

Figure 5: Result for the spectral clustering task on the Texas graph (Lim et al., 2021) with core numbers considered as clusters. We report the values of the Adjusted Mutual Information (AMI) in percentage for different combinations of the exponents (e_2, e_3) in $\mathbf{C}^{e_2}\mathbf{A}\mathbf{C}^{e_3}$.

E.2.3 ARI values for Texas

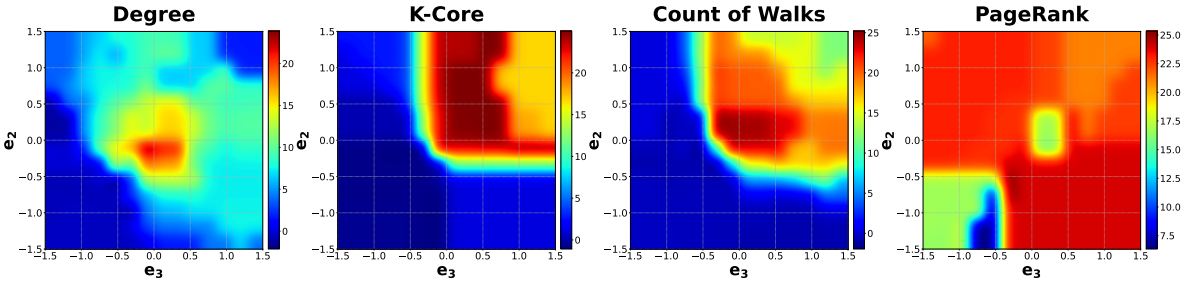

Figure 6: Result for the spectral clustering task on the Texas graph (Lim et al., 2021) with core numbers considered as clusters. We report the values of the Adjusted Rand Index (ARI) in percentage for different combinations of the exponents (e_2, e_3) in $\mathbf{C}^{e_2}\mathbf{A}\mathbf{C}^{e_3}$.

F Alternative CGSO Parameterizations

It is important to clarify that our primary objective in this work is not the design of the parameterization scheme itself, but rather to systematically investigate whether replacing the standard local degree centrality with global centrality metrics in GSOs enhances graph representation learning. While a flexible GSO formulation is necessary to ensure generalization across diverse graph structures, our core hypothesis rests on the benefits derived from the choice of centrality normalization. To verify that the observed performance gains are intrinsic to these global centrality metrics, and not an artifact of our specific CGSO equation, we employ a second, alternative framework, the *Parameterized Normalized Adjacency proposed* by (Lu et al., 2024). By substituting the standard degree $\mathbf{D}$ with another centrality matrix $\mathbf{C}$, the parameterization of the GSO become,

$$\mathbf{\Phi}_{\text{PNA}}(\mathbf{A}, \mathbf{C}; \gamma, \alpha) = \mathbf{I}_N - \gamma\left(\gamma\mathbf{C} + (1-\gamma)\mathbf{I}_N\right)^{-\alpha}\left(\mathbf{C} - \mathbf{A}\right)\left(\gamma\mathbf{C} + (1-\gamma)\mathbf{I}_N\right)^{\alpha-1},$$

where $\mathbf{C}$ is your chosen centrality matrix, and both $\gamma \in (0, 1]$ and $\alpha \in [0, 1]$ are scalar hyperparameters introduced by Lu et al. (2024) to control the diffusion scope and spectral properties. The result of this new parametrization on the node classification tasks can be found in Tables 12 and 13.

Table 12: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model.

Model	CiteSeer	PubMed	Arxiv-Year	Chamelon	Cornell	Deezer-Europe	Squirrel	Wisconsin
PD-GCN w/ $\mathbf{D}$	22.62±0.39	58.23±0.49	36.93±0.2	**52.02±2.36**	67.84±5.33	54.84±0.58	35.17±1.54	71.76±4.57
PD-GCN w/ $\mathbf{C}^{(\text{core})}$	48.18±0.70	62.14±0.24	36.53±0.18	45.79±1.94	**74.59±4.55**	59.36±1.21	30.47±1.67	82.35±3.51
PD-GCN w/ $\mathbf{C}^{(\ell-\text{walks})}$	29.76±0.42	59.46±0.27	**43.97±0.17**	48.46±2.43	**74.59±4.55**	54.63±0.46	**38.15±1.41**	77.84±4.21
PD-GCN w/ $\mathbf{C}^{(\text{pr})}$	**56.15±0.68**	**72.24±0.31**	36.05±0.19	46.43±2.36	73.24±4.9	**63.41±0.58**	31.34±1.61	**83.92±2.45**

Table 13: Classification accuracy (± standard deviation) of the models on different benchmark node classification datasets. The higher the accuracy (in %) the better the model.

Model	Cora	Texas	Photo	Ogbn-Arxiv	CS	Computers	Physics	Penn94
PD-GCN w/ $\mathbf{D}$	21.65±1.2	77.84±4.65	22.21±2.32	12.44±0.13	26.0±2.77	14.26±3.84	29.3±3.1	71.16±0.29
PD-GCN w/ $\mathbf{C}^{(\text{core})}$	44.79±0.81	78.11±5.05	69.02±2.33	51.62±0.1	62.48±3.35	55.55±3.91	81.47±2.94	74.48±0.28
PD-GCN w/ $\mathbf{C}^{(\ell-\text{walks})}$	23.07±0.73	**82.7±4.55**	36.18±3.52	43.96±0.14	34.16±2.82	36.83±5.65	67.42±3.83	71.11±0.35
PD-GCN w/ $\mathbf{C}^{(\text{pr})}$	**56.87±1.29**	78.92±3.97	**69.81±3.96**	**53.13±0.13**	**83.8±1.56**	**56.44±1.57**	**86.38±1.8**	**74.79±0.36**

G Proofs of Propositions and Lemmas

In this section, we detail the proofs of the Propositions 3.2, 3.3, 3.5 and Lemmas 3.6 and 4.1.

G.1 Proof of Proposition 3.2

Proof. We first prove that the operator $\mathbf{M}_{\mathcal{G}}$ is self-adjoint.
For $\varphi_1, \varphi_2 \in L^2(\mathcal{G})$, we have:

$$\begin{aligned}
< \mathbf{M}_{\mathcal{G}}\varphi_1, \varphi_2 >_{\mathcal{G}} &= \sum_{i\in\mathcal{V}} c_i \left(\mathbf{M}_{\mathcal{G}}\varphi_1\right)(i)\bar{\varphi}_2(i) \\
&= \sum_{i\in\mathcal{V}} c_i \left(\frac{1}{c_i}\sum_{j\in\mathcal{N}(i)} \varphi_1(j)\right)\bar{\varphi}_2(i) \\
&= \sum_{i\in\mathcal{V}} \bar{\varphi}_2(i)\left(\sum_{j\in\mathcal{N}(i)} \varphi_1(j)\right) \\
&= \sum_{i\in\mathcal{V}}\sum_{j\in\mathcal{N}(i)} a_{ij}\bar{\varphi}_2(i)\varphi_1(j) \\
&= \sum_{i,j\in\mathcal{V}} a_{ij}\bar{\varphi}_2(i)\varphi_1(j).
\end{aligned}$$

Similarly, we also have that,

$$\begin{aligned}
< \varphi_1, \mathbf{M}_{\mathcal{G}}\varphi_2 >_{\mathcal{G}} &= \sum_{i\in\mathcal{V}} c_i\varphi_1(i)\overline{(\mathbf{M}_{\mathcal{G}}\varphi_2)}(i) \\
&= \sum_{i\in\mathcal{V}} c_i\varphi_1(i)\overline{\left(\frac{1}{c_i}\sum_{j\in\mathcal{N}(i)} \varphi_2(j)\right)} \\
&= \sum_{i\in\mathcal{V}} \varphi_1(i)\overline{\left(\sum_{j\in\mathcal{N}(i)} \varphi_2(j)\right)} \\
&= \sum_{i,j\in\mathcal{V}} a_{ij}\bar{\varphi}_2(i)\varphi_1(j) \\
&= \sum_{i,j\in\mathcal{V}} a_{ji}\bar{\varphi}_2(i)\varphi_1(j).
\end{aligned}$$

Thus,

$$\forall \varphi_1, \varphi_2 \in L^2(\mathcal{G}), \begin{cases} < \mathbf{M}_{\mathcal{G}}\varphi_1, \varphi_2 >_{\mathcal{G}} = \sum_{i,j\in\mathcal{V}} a_{ij}\bar{\varphi}_2(i)\varphi_1(j), \\ < \varphi_1, \mathbf{M}_{\mathcal{G}}\varphi_2 >_{\mathcal{G}} = \sum_{i,j\in\mathcal{V}} a_{ji}\bar{\varphi}_2(j)\varphi_1(i). \end{cases} \tag{3}$$

Since $a_{ij} = a_{ji}$, we conclude that $\mathbf{M}_{\mathcal{G}}$ is self-adjoint, i.e.,

$$< \mathbf{M}_{\mathcal{G}}\varphi_1, \varphi_2 >_{\mathcal{G}} = < \varphi_1, \mathbf{M}_{\mathcal{G}}\varphi_2 >_{\mathcal{G}} .$$

$\mathbf{M}_{\mathcal{G}}$ is self-adjoint, the space $L^2(G)$ is finite-dimensional, thus is diagonalizable in an orthonormal basis, and its eigenvalues are real.
We define the following norm,

$$\|\mathbf{M}_{\mathcal{G}}\| = \sup_{\varphi\neq 0} \frac{\langle \mathbf{M}_{\mathcal{G}}\varphi, \varphi\rangle_{\mathcal{G}}}{\|\varphi\|^2}.$$

We will now prove that all eigenvalues have absolute values at most $\gamma = \min_{i\in\mathcal{V}} c_i/d_i$. For that, we will first compute the two inner-products $< (I - \mathbf{M}_{\mathcal{G}})\,\varphi, \varphi >_{\mathcal{G}}$ and $< (I + \mathbf{M}_{\mathcal{G}})\,\varphi, \varphi >_{\mathcal{G}}$. For any $\varphi \in L^2(\mathcal{G})$, using (3), we have that:

$$\begin{cases} < \varphi, \varphi >_{\mathcal{G}} = \sum_{i\in\mathcal{V}} c_i |\varphi(i)|^2, \\ < \mathbf{M}_{\mathcal{G}}\varphi, \varphi >_{\mathcal{G}} = \sum_{i,j\in\mathcal{V}} a_{ij}\bar{\varphi}(i)\varphi(j). \end{cases}$$

Let's first take the simple case, where $\gamma = \min_{i\in\mathcal{V}} \left(\frac{c_i}{d_i}\right) \geq 1$, then,

$$\begin{aligned} 2 < \varphi, \varphi >_{\mathcal{G}} &= 2\sum_{i\in\mathcal{V}} c_i |\varphi(i)|^2 \\ &\geq 2\sum_{i\in\mathcal{V}} d_i |\varphi(i)|^2 \\ &\geq 2\sum_{i\in\mathcal{V}} \left(\sum_{j\in\mathcal{V}} a_{ij}\right) |\varphi(i)|^2 \\ &\geq 2\sum_{i,j\in\mathcal{V}} a_{ij} |\varphi(i)|^2. \end{aligned}$$

Therefore,

$$\begin{aligned} 2 < (\mathbf{I} - \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}} &= 2 < \varphi, \varphi >_{\mathcal{G}} - 2 < \mathbf{M}_{\mathcal{G}}\varphi, \varphi >_{\mathcal{G}} \\ &\geq 2\sum_{i,j\in\mathcal{V}} a_{ij}|\varphi(i)|^2 - 2\sum_{i,j\in\mathcal{V}} a_{ij}\bar{\varphi}(i)\varphi(j) \\ &\geq \sum_{i,j\in\mathcal{V}} a_{ij}|\varphi(i)|^2 + \sum_{i,j\in\mathcal{V}} a_{ij}|\varphi(j)|^2 - 2\sum_{i,j\in\mathcal{V}} a_{ij}\bar{\varphi}(i)\varphi(j) \\ &\geq \sum_{i,j\in\mathcal{V}} a_{ij}|\varphi(i) - \varphi(j)|^2. \end{aligned}$$

Similarly, we can prove that,

$$2 < (\mathbf{I} + \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}} \geq \sum_{i,j\in\mathcal{V}} a_{ij}|\varphi(i) + \varphi(j)|^2.$$

Therefore, if $\phi \neq 0$, then,

$$\begin{aligned} \begin{cases} < (\mathbf{I} - \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}} \geq 0, \\ < (\mathbf{I} + \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}} \geq 0 \end{cases} &\Rightarrow - < \varphi, \varphi >_{\mathcal{G}} \leq < \mathbf{M}_{\mathcal{G}}\varphi, \varphi >_{\mathcal{G}} \leq < \varphi, \varphi >_{\mathcal{G}} . \\ &\Rightarrow \frac{| < \mathbf{M}_{\mathcal{G}}\varphi, \varphi >_{\mathcal{G}} |}{< \varphi, \varphi >_{\mathcal{G}}} \leq 1. \end{aligned}$$

Thus, $\|\mathbf{M}_{\mathcal{G}}\| \leq 1$, i.e., all the eigenvalues have absolute values at most 1. Let now consider the general case, where γ is not necessarily larger than 1. Let's consider $\widetilde{\mathbf{C}} = \frac{1}{\gamma}\mathbf{C} = \mathrm{diag}(\frac{c_1}{\gamma}, \ldots, \frac{c_N}{\gamma})$. Since,

$$\begin{aligned}\tilde{\gamma} &= \min_{i\in\mathcal{V}}\left(\frac{\tilde{c}_i}{d_i}\right)\\ &= \frac{1}{\gamma}\min_{i\in\mathcal{V}}\left(\frac{c_i}{d_i}\right)\\ &= \frac{\gamma}{\gamma}\\ &= 1.\end{aligned}$$

Therefore, all the eigenvalues of $\widetilde{\mathbf{M}}_{\mathcal{G}} = \widetilde{\mathbf{C}}^{-1}A = \frac{1}{\gamma}\mathbf{C}^{-1}\mathbf{A} = \frac{1}{\gamma}\mathbf{M}_{\mathcal{G}}$ have absolute values at most 1. Thus, all the eigenvalues of $\mathbf{M}_{\mathcal{G}}$ have absolute values at most γ. □

G.2 Proof of Proposition 3.3

Proof. We begin by proving the first property. Let P be the number of connected components, i.e., $\mathcal{V} = \bigcup_{i=1}^{P}\mathcal{C}_i$. The adjacency matrix of the graph $\mathcal{G}$ is,

$$\mathbf{A} = \begin{bmatrix}\mathbf{A}_{\mathcal{C}_1} & \mathbf{0} & \cdots & \cdots & \mathbf{0}\\ \mathbf{0} & \ddots & \ddots & \mathbf{0} & \vdots\\ \vdots & \ddots & \mathbf{A}_{\mathcal{C}_i} & \ddots & \vdots\\ \vdots & \mathbf{0} & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{A}_{\mathcal{C}_P}\end{bmatrix}.$$

And the transformation of $\mathbf{A}$ by the Markov Average operator $\mathbf{M}_{\mathcal{G}}$ is,

$$\mathbf{M}_{\mathcal{G}} = \begin{bmatrix}\mathbf{M}_{\mathcal{C}_1} & \mathbf{0} & \cdots & \cdots & \mathbf{0}\\ \mathbf{0} & \ddots & \ddots & \mathbf{0} & \vdots\\ \vdots & \ddots & \mathbf{M}_{\mathcal{C}_i} & \ddots & \vdots\\ \vdots & \mathbf{0} & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{M}_{\mathcal{C}_P}\end{bmatrix}.$$

According to Proposition 3.2, for each connected component $\mathcal{C}_i$, the matrix $\mathbf{M}_{\mathcal{C}_i}$ is diagonalizable in an orthonormal basis, and its eigenvalues are real numbers. We denote by $\mathbf{e}^{\mathcal{C}_i} = [\mathbf{e}_1^{\mathcal{C}_i}, \ldots, \mathbf{e}_{|\mathcal{C}_i|}^{\mathcal{C}_i}]$ the eigenvectors basis of $M_{\mathcal{C}_i}$ corresponding the eigenvalues $\lambda^{\mathcal{C}_i} = [\lambda_1^{\mathcal{C}_i}, \ldots, \lambda_{|\mathcal{C}_i|}^{\mathcal{C}_i}]$. We consider the set of vectors

$$\mathbf{e} = \begin{bmatrix}\mathbf{e}^{\mathcal{C}_1} & \mathbf{0} & \cdots & \cdots & \mathbf{0}\\ \mathbf{0} & \ddots & \ddots & \mathbf{0} & \vdots\\ \vdots & \ddots & \mathbf{e}^{\mathcal{C}_i} & \ddots & \vdots\\ \vdots & \mathbf{0} & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{e}^{\mathcal{C}_P}\end{bmatrix}.$$

The column vectors of $\mathbf{e}$ are eigenvectors of the matrix $\mathbf{M}_{\mathcal{G}}$, which achieves the conditions of Property (1).

Let's now prove the formulas of the mean and standard deviation of the $\mathbf{M}_{\mathcal{G}}$ spectrum. To do so, we denote eigenvalues of $\mathbf{M}_{\mathcal{G}}$ by $\lambda_1, \ldots, \lambda_N$. The matrix $\mathbf{C}^{-1}\mathbf{A}$ is defined as follows, for all $1 \leq i, j \leq N$,

$$(\mathbf{C}^{-1}\mathbf{A})_{ij} = \frac{1}{c_i}a_{ij}.$$

Therefore, the diagonal elements of the matrix $\left(\mathbf{C}^{-1}\mathbf{A}\right)^2$ is defined as follow, for all $1 \leq i \leq N$,

$$\begin{aligned}
\left(\left(\mathbf{C}^{-1}\mathbf{A}\right)^2\right)_{ii} &= \left(\mathbf{C}^{-1}\mathbf{A}\mathbf{C}^{-1}\mathbf{A}\right)_{ii} \\
&= \sum_{j\in\mathcal{V}} (\mathbf{C}^{-1}\mathbf{A})_{ij}(\mathbf{C}^{-1}\mathbf{A})_{ji} \\
&= \sum_{j\in\mathcal{V}} \frac{a_{ij}a_{ji}}{c_i c_j} \\
&= \sum_{j\in\mathcal{V}} \frac{a_{ij}^2}{c_i c_j} \\
&= \sum_{j\in\mathcal{N}(i)} \frac{1}{c_i c_j}.
\end{aligned}$$

Thus,

$$\begin{aligned}
\mu\left(\mathbf{M}_{\mathcal{G}}\right) &= \frac{1}{N}\sum_{i\in\mathcal{V}} \lambda_i \\
&= \frac{1}{N}\sum_{i=1}^{N} (\mathbf{C}^{-1}\mathbf{A})_{ii} \\
&= \frac{1}{N}\sum_{i=1}^{N} \frac{1}{c_i} a_{ii} \\
&= 0,
\end{aligned}$$

and,

$$\begin{aligned}
\sigma\left(\mathbf{M}_{\mathcal{G}}\right) &= \sqrt{\frac{1}{N}\sum_{i=1}^{N} \left(\lambda_i - \mu\left(\mathbf{M}_{\mathcal{G}}\right)\right)^2} \\
&= \sqrt{\frac{1}{N}\sum_{i=1}^{N} \lambda_i^2} \\
&= \sqrt{\frac{1}{N}\mathrm{Tr}\left[\left(\mathbf{C}^{-1}\mathbf{A}\right)^2\right]} \\
&= \sqrt{\frac{1}{N}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}(i)} \frac{1}{c_i c_j}} \\
&= \sqrt{\frac{1}{N}\sum_{(i,j)\in\mathcal{E}} \frac{1}{c_i c_j}}.
\end{aligned}$$

□

G.3 Proof of Proposition 3.5

Proof. Let $\varphi \in \mathbb{R}^{|\mathcal{V}|}$ be the centered indicator vector of $\mathcal{W}$, defined such that its components are $\varphi_i = \mathbb{1}_{\{i\in\mathcal{W}\}} - \mu_{\mathcal{G}}(\mathcal{W})$ for all $i \in \mathcal{V}$, where $\mu_{\mathcal{G}}(\mathcal{W}) = \frac{|\mathcal{W}|_c}{N_v}$ and $N_v = \sum_{i\in\mathcal{V}} c_i = |\mathcal{V}|_c$. Assuming $\max_{i\in\mathcal{V}}\left(\frac{c_i}{d_i}\right) \leq 1$,

we have,

$$
\begin{aligned}
2 < (\mathbf{I} - \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}} &= 2 < \varphi, \varphi >_{\mathcal{G}} - 2 < \mathbf{M}_{\mathcal{G}}\varphi, \varphi >_{\mathcal{G}} \\
&\leq 2 \sum_{i \in \mathcal{V}} c_i |\varphi(i)|^2 - 2 \sum_{i,j \in \mathcal{V}} a_{ij} \bar{\varphi}(i) \varphi(j) \\
&\leq 2 \max_{i \in \mathcal{V}} \left(\frac{c_i}{d_i} \right) \sum_{i \in \mathcal{V}} d_i |\varphi(i)|^2 - 2 \sum_{i,j \in \mathcal{V}} a_{ij} \bar{\varphi}(i) \varphi(j) \\
&\leq 2 \sum_{i \in \mathcal{V}} d_i |\varphi(i)|^2 - 2 \sum_{i,j \in \mathcal{V}} a_{ij} \bar{\varphi}(i) \varphi(j) \\
&\leq 2 \sum_{i \in \mathcal{V}} \left(\sum_{j \in \mathcal{V}} a_{ij} \right) |\varphi(i)|^2 - 2 \sum_{i,j \in \mathcal{V}} a_{ij} \bar{\varphi}(i) \varphi(j) \\
&\leq 2 \sum_{i,j \in \mathcal{V}} a_{ij} |\varphi(i)|^2 - 2 \sum_{i,j \in \mathcal{V}} a_{ij} \bar{\varphi}(i) \varphi(j) \\
&\leq \sum_{i,j \in \mathcal{V}} a_{ij} |\varphi(i)|^2 + \sum_{i,j \in \mathcal{V}} a_{ij} |\varphi(j)|^2 - 2 \sum_{i,j \in \mathcal{V}} a_{ij} \bar{\varphi}(i) \varphi(j) \\
&\leq \sum_{i,j \in \mathcal{V}} a_{ij} |\varphi(i) - \varphi(j)|^2 \\
&\leq \sum_{i,j \in \mathcal{V}} a_{ij} |\mathbb{1}_{\mathcal{W}}(i) - \mathbb{1}_{\mathcal{W}}(j)|^2.
\end{aligned}
$$

The non-zero terms in $\sum_{i,j \in \mathcal{V}} a_{ij} |\varphi(i) - \varphi(j)|^2$ are those where i and j are adjacent, but one of them is in $\mathcal{W}$ and the other not.

$$
\begin{aligned}
< (\mathbf{I} - \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}} &\leq \frac{1}{2} \sum_{i,j \in \mathcal{V}} a_{ij} |\mathbb{1}_{\mathcal{W}}(i) - \mathbb{1}_{\mathcal{W}}(j)|^2 \\
&= |\mathcal{E}(\mathcal{W})|.
\end{aligned}
$$

There $\frac{1}{2}$ was removed because of the symmetry. We also have that,

$$
\frac{1}{N_v} < \mathbb{1}_{\mathcal{W}}, \mathbb{1}_{\mathcal{W}} >_{\mathcal{G}} = \frac{1}{N_v} \sum_{i \in \mathcal{W}} c_i = \mu_{\mathcal{G}}(\mathcal{W}),
$$

and,

$$
\frac{1}{N_v} < \mathbb{1}_{\mathcal{W}}, \mu_{\mathcal{G}}(\mathcal{W}) \mathbb{1} >_{\mathcal{G}} = \frac{1}{N_v} \sum_{i \in \mathcal{W}} c_i \mu_{\mathcal{G}}(\mathcal{W}) = \left(\mu_{\mathcal{G}}(\mathcal{W}) \right)^2,
$$

Therefore,

$$\begin{aligned}
\frac{1}{N_v} < \varphi, \varphi >_{\mathcal{G}} &= \frac{1}{N_v} < \mathbb{1}_{\mathcal{W}} - \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1}, \mathbb{1}_{\mathcal{W}} - \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1} >_{\mathcal{G}} \\
&= \frac{1}{N_v} < \mathbb{1}_{\mathcal{W}}, \mathbb{1}_{\mathcal{W}} - \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1} >_{\mathcal{G}} - \frac{1}{N_v} < \mu_{\mathcal{G}}(\mathcal{W}), \mathbb{1}_{\mathcal{W}} - \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1} >_{\mathcal{G}} \\
&= \frac{1}{N_v} < \mathbb{1}_{\mathcal{W}}, \mathbb{1}_{\mathcal{W}} >_{\mathcal{G}} - \frac{2}{N_v} < \mathbb{1}_{\mathcal{W}}, \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1} >_{\mathcal{G}} + \frac{1}{N_v} < \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1}, \mu_{\mathcal{G}}(\mathcal{W})\mathbb{1} >_{\mathcal{G}} \\
&= \mu_{\mathcal{G}}(\mathcal{W}) - 2\left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 + \left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 \frac{1}{N_v} < \mathbb{1}, \mathbb{1} >_{\mathcal{G}} \\
&= \mu_{\mathcal{G}}(\mathcal{W}) - 2\left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 + \left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 \frac{1}{N_v} \sum_{i \in \mathcal{V}} c_i \\
&= \mu_{\mathcal{G}}(\mathcal{W}) - 2\left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 + \left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 \frac{N_v}{N_v} \\
&= \mu_{\mathcal{G}}(\mathcal{W}) - \left(\mu_{\mathcal{G}}(\mathcal{W})\right)^2 \\
&= \mu_{\mathcal{G}}(\mathcal{W})\left(1 - \mu_{\mathcal{G}}(\mathcal{W})\right) \\
&= \mu_{\mathcal{G}}(\mathcal{W})\mu_{\mathcal{G}}(\mathcal{W}'),
\end{aligned}$$

where $\mathcal{W}' = \mathcal{V} \backslash \mathcal{W}$, i.e., the set exclusion of $\mathcal{W}$ from $\mathcal{V}$.
By definition,

$$\lambda_1(\mathcal{G}) = \min_{\tilde{\varphi} \neq 0} \frac{< (\mathbf{I} - \mathbf{M}_{\mathcal{G}})\tilde{\varphi}, \tilde{\varphi} >_{\mathcal{G}}}{< \tilde{\varphi}, \tilde{\varphi} >_{\mathcal{G}}}.$$

Therefore,

$$\begin{aligned}
\lambda_1(\mathcal{G}) &\leq \frac{< (\mathbf{I} - \mathbf{M}_{\mathcal{G}})\varphi, \varphi >_{\mathcal{G}}}{< \varphi, \varphi >_{\mathcal{G}}} \\
&\leq \frac{|\mathcal{E}(\mathcal{W})|}{N_v} \frac{N_v}{< \varphi, \varphi >_{\mathcal{G}}} \\
&\leq \frac{|\mathcal{E}(\mathcal{W})|}{N_v} \frac{N_v}{\mu_{\mathcal{G}}(\mathcal{W})\mu_{\mathcal{G}}(\mathcal{W}')}.
\end{aligned}$$

Since,

$$\frac{c_-}{c_+} \frac{|\mathcal{W}|_c}{|\mathcal{V}|_c} \leq \mu_{\mathcal{G}}(\mathcal{W}) \leq \frac{c_+}{c_-} \frac{|\mathcal{W}|_c}{|\mathcal{V}|_c},$$

then,

$$\begin{aligned}
N_v \mu_{\mathcal{G}}(\mathcal{W})\mu_{\mathcal{G}}(\mathcal{W}') &\geq N_v \frac{|\mathcal{W}|_c}{|\mathcal{V}|_c} \frac{c_-}{c_+} \frac{|\mathcal{W}'|_c}{|\mathcal{V}|_c} \\
&\geq \frac{\sum_{i \in \mathcal{V}} c_i}{|\mathcal{V}|_c} |\mathcal{W}|_c \frac{c_-}{c_+} \frac{|\mathcal{W}'|_c}{|\mathcal{V}|_c} \\
&\geq \frac{\sum_{i \in \mathcal{V}} c_i}{\sum_{i \in \mathcal{V}} c_i} |\mathcal{W}|_c \frac{c_-}{c_+} \frac{|\mathcal{W}'|_c}{|\mathcal{V}|_c} \\
&\geq \frac{c_-}{c_+} |\mathcal{W}|_c \frac{|\mathcal{W}'|_c}{|\mathcal{V}|_c} \\
&\geq \frac{c_-}{2c_+} |\mathcal{W}|_c,
\end{aligned}$$

because

$$\begin{cases} |\mathcal{W}|_c \leq \frac{1}{2}|\mathcal{V}|_c \\ \mathcal{W}' = \mathcal{V} \backslash \mathcal{W} \end{cases} \Rightarrow |\mathcal{W}'|_c \geq \frac{1}{2}|\mathcal{V}|_c.$$

Thus,

$$\forall \mathcal{W} \subset \mathcal{V}, |\mathcal{W}|_c \leq \frac{1}{2}|\mathcal{V}|_c \Rightarrow \lambda_1(\mathcal{G}) \leq \frac{2c_+}{c_-} \frac{|\mathcal{E}(\mathcal{W})|}{|\mathcal{W}|_c}.$$

We have,

$$\begin{aligned}|\mathcal{E}(\mathcal{W})| &\leq \sum_{i\in\partial\mathcal{W}} d_i \\ &\leq N \max_{i\in\partial\mathcal{W}} d_i \\ &\leq \frac{N}{\gamma} \max_{i\in\partial\mathcal{W}} c_i \\ &\leq \frac{Nc_+}{\gamma}.\end{aligned}$$

Thus,

$$\lambda_1(\mathcal{G}) \leq N\frac{2c_+^2}{\gamma c_-}h_c(\mathcal{G})$$

□

G.4 Proof of Lemma 3.6

Proof. We recall the definition of the parameterized Centrality Graph Shift Operator $\mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)$,

$$\mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta) = m_1\mathbf{C}^{e_1} + m_2\mathbf{C}^{e_2}\mathbf{A}_a\mathbf{C}^{e_3} + m_3\mathbf{I}_N.$$

Let $\mathbf{Z} \in \mathbb{R}^{N\times d}$ be the input feature matrix. The embedding for a node u after the application of the operator, denoted $\mathbf{H}_u$, is the u-th row of the matrix $\mathbf{\Phi}(\mathbf{A}, \mathbf{C}; \Theta)\mathbf{Z}$. We aim to bound the Euclidean norm $||\mathbf{H}_u||_2$.

By linearity, we can expand the operator application on node u,

$$\mathbf{H}_u = m_1c_u^{e_1}\mathbf{Z}_u + m_3\mathbf{Z}_u + m_2c_u^{e_2}\left[ac_u^{e_3}\mathbf{Z}_u + \sum_{v\in\mathcal{N}(u)} c_v^{e_3}\mathbf{Z}_v\right].$$

To bound the L_2 norm $||\mathbf{H}_u||_2$, we apply the triangle inequality. We assume the input features are bounded such that $||\mathbf{Z}_v||_2 \leq ||\mathbf{Z}||_{2,\infty}$ for all v.

$$||\mathbf{H}_u||_2 \leq ||\mathbf{Z}||_{2,\infty}\left(|m_1|c_u^{e_1} + |m_3| + |m_2|c_u^{e_2}\left|ac_u^{e_3} + \sum_{v\in\mathcal{N}(u)} c_v^{e_3}\right|\right).$$

The key step is to analyze the summation term $\sum_{v\in\mathcal{N}(u)} c_v^{e_3}$. To separate the structural role of the node from the variation in its neighborhood, we perform a first-order Taylor expansion of the function $f(c) = c^{e_3}$ around the central value c_u.The derivative is $f'(c) = e_3c^{e_3-1}$. Thus, for a neighbor v,

$$c_v^{e_3} \approx c_u^{e_3} + e_3c_u^{e_3-1}(c_v - c_u)$$

$$c_v^{e_3} = c_u^{e_3} + e_3c_u^{e_3-1}(c_v - c_u) + O((c_v - c_u)^2)$$

Substituting this approximation back into the summation term,

$$\begin{aligned}\sum_{v\in\mathcal{N}(u)} c_v^{e_3} &= \sum_{v\in\mathcal{N}(u)} \left(c_u^{e_3} + e_3c_u^{e_3-1}(c_v - c_u) + O((c_v - c_u)^2)\right) \\ &= \sum_{v\in\mathcal{N}(u)} c_u^{e_3} + e_3c_u^{e_3-1}\sum_{v\in\mathcal{N}(u)}(c_v - c_u) + O\left(\sum_{v\in\mathcal{N}(u)}(c_v - c_u)^2\right)\end{aligned}$$

Since summing a constant $c_u^{e_3}$ over the neighborhood $\mathcal{N}(u)$ equals $d_u c_u^{e_3}$ (where d_u is the degree of u), we have,

$$\sum_{v\in\mathcal{N}(u)} c_v^{e_3} = d_u c_u^{e_3} + e_3 c_u^{e_3-1}\bar{\Delta}_{u,C},$$

where $\bar{\Delta}_{u,C} = \sum_{v\in\mathcal{N}(u)}(c_v - c_u)$ represents the accumulated centrality difference in the neighborhood.

Now, we substitute this back into the full expression for the m_2 term, including the self-loop part $ac_u^{e_3}$,

$$|m_2|c_u^{e_2}\left|ac_u^{e_3} + \sum_{v\in\mathcal{N}(u)} c_v^{e_3}\right| = |m_2|c_u^{e_2}\left|(a+d_u)c_u^{e_3} + e_3 c_u^{e_3-1}\bar{\Delta}_{u,C} + O\left(\sum_{v\in\mathcal{N}(u)}(c_v-c_u)^2\right)\right|$$

Grouping the terms by powers of c_u,

$$|m_2|c_u^{e_2}\left|ac_u^{e_3} + \sum_{v\in\mathcal{N}(u)} c_v^{e_3}\right| = |m_2|c_u^{e_2}\left|c_u^{e_3}(d_u+a) + e_3\frac{\bar{\Delta}_{u,C}}{c_u^{1-e_3}} + O\left(\sum_{v\in\mathcal{N}(u)}(c_v-c_u)^2\right)\right|$$

Thus,

$$|m_2|c_u^{e_2}\left|ac_u^{e_3} + \sum_{v\in\mathcal{N}(u)} c_v^{e_3}\right| = |m_2|\left(c_u^{e_2+e_3}(d_u+a) + e_3 c_u^{e_2}\frac{\bar{\Delta}_{u,C}}{c_u^{1-e_3}} + O\left(c_u^{e_2}\sum_{v\in\mathcal{N}(u)}(c_v-c_u)^2\right)\right)$$

Finally, substituting this back into the norm inequality yields the bound stated in the Lemma,

$$||\mathbf{H}_u||_2 = ||\mathbf{Z}||_{2,\infty}\left(|m_1|c_u^{e_1} + |m_3| + |m_2|\left[c_u^{e_2+e_3}(d_u+a) + e_3 c_u^{e_2}\frac{\bar{\Delta}_{u,C}}{c_u^{1-e_3}} + O\left(c_u^{e_2}\sum_{v\in\mathcal{N}(u)}(c_v-c_u)^2\right)\right]\right)$$

□

G.5 Proof of Lemma 4.1

Proof. We start with a small graph of N_0 nodes and r_0 edges. At each time step, we increase the number of edges by r. Thus, if N is the number of nodes at a certain time step, then there are exactly $r_0 + r(N - N_0)$ edges.

As each edge contributes to the degree of two nodes, thus, the average degree is twice the number of edges divided by the number of nodes N. Therefore,

$$\begin{aligned}\bar{d}(\mathcal{G}^{BA}) &= \frac{2}{N}\left(r_0 + r(N - r_0)\right)\\ &= 2r + 2\frac{r_0}{N} - 2N_0\frac{r}{N}.\end{aligned}$$

□

H Interpretability Analysis: Learned Parameters and Graph Statistics

In this section, we present some graph properties of the used dataset. We specifically present the node density, the homophily coefficient as well as the average value of different centrality metrics in Table 14. We also present the $(m_1, m_2, m_3, e_1, e_2, e_3, a)$ learned by the GNN in Tables 15, 16, 17 and 18.

Table 14: Detailed graph properties of the used datasets.

Dataset	density	Avg. Degree	Avg. PageRank	Avg. K-core	Avg. Count. Paths	homophily
Physics	4.16×10^{-4}	14.37	2.89×10^{-5}	7.71	449.22	0.931
Photo	4.07×10^{-3}	31.13	1.30×10^{-4}	16.97	3204.098	0.827
Cora	1.43×10^{-3}	3.89	3.69×10^{-4}	2.31	42.52	0.809
CS	4.87×10^{-4}	8.93	5.45×10^{-5}	4.94	162.75	0.808
PubMed	2.28×10^{-4}	4.49	5.07×10^{-5}	2.39	75.43	0.802
Computers	2.60×10^{-3}	35.75	7.27×10^{-5}	18.84	6221.39	0.777
CiteSeer	8.22×10^{-4}	2.73	3.00×10^{-4}	1.73	18.91	0.735
ogbn-arxiv	8.07×10^{-5}	13.67	5.90×10^{-6}	7.13	4898.16	0.654
deezer-europe	2.31×10^{-4}	6.55	3.53×10^{-5}	3.57	106.16	0.525
Penn94	1.57×10^{-3}	65.56	2.40×10^{-5}	33.68	10662.08	0.470
chameleon	1.21×10^{-2}	27.57	4.39×10^{-4}	16.60	2913.48	0.231
squirrel	1.46×10^{-2}	76.30	1.92×10^{-4}	41.55	31888.02	0.222
arxiv-year	8.07×10^{-5}	6.88	5.90×10^{-6}	7.13	82.85	0.218
Wisconsin	1.48×10^{-2}	3.64	3.98×10^{-3}	2.05	76.26	0.206
Cornell	1.68×10^{-2}	3.04	5.46×10^{-3}	1.74	58.47	0.132
Texas	1.77×10^{-2}	3.13	5.46×10^{-3}	1.71	70.72	0.111

Our experimental results reveal a distinct bifurcation in the learning behavior of the CGSO parameters, providing strong empirical evidence that optimal message passing is structurally dependent rather than universal. For homophilic datasets such as Cora, CiteSeer, and PubMed, the model consistently *rediscovers* the principles of standard spectral graph theory. Specifically, when using degree centrality, the learned exponents e_2 and e_3 converge towards equal values somewhat close to -0.5. This configuration effectively reconstructs the Symmetric Normalized Laplacian $\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ used in the standard GCN. By learning negative exponents, the model penalizes high-degree nodes, creating a smoothing effect where signals are averaged across neighborhoods. This confirms that the CGNN framework correctly identifies averaging as the optimal strategy when the homophily assumption holds and neighbors share the same label.

Contrariwise, heterophilic datasets drive the model to discover a radically different mechanism. In datasets like Wisconsin, Cornell, and Texas, we tend to learn negative values for the parameter m_2, which mostly took positive values for the homophilic datasets. This results in CGSOs that are more Laplacian-like and less likely to emulate adjacency-based averaging over neighborhoods as is beneficial for homophilic graphs. The differencing operator applied via Laplacian-like GSOs is known to yield good performance on heterophilic datasets and it is encourgaing to see our parametrized CGSO form rediscovering these operators.

Furthermore, the learned parameters expose an often-overlooked necessity for asymmetric normalization in directed or complex graphs. For instance, on the Chameleon dataset using PageRank centrality, the model learns disparate exponents of approximately -0.94 and 0.06. This effectively reconstructs a Random Walk operator $\mathbf{D}^{-1}\mathbf{A}$ rather than the symmetric Laplacian $\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, demonstrating that the model can break symmetry to respect the directional flow of information. Additionally, we observe a phenomenon of Identity Preservation in heterophilic contexts. In Table 16 ($k-$core) and Table 18 (Walk Counts) for Wisconsin and Texas, the diagonal term parameters (m_1 or e_1) are often large and positive, exceeding 1.0. This indicates the model aggressively amplifies a node's own features relative to message passing, selectively boosting the *self-signal* when neighboring information is deemed unreliable.

Lastly, the learned parameter a helps the model adapt by switching between blurring local signals and making them more distinct. In homophilic datasets like Cora, a is consistently positive, e.g., $a \approx 1.69$, acting as a Low-Pass filter that reinforces community signals via weighted averaging. Conversely, in heterophilic datasets like Texas and Cornell, a frequently becomes negative, e.g., $a \approx -0.64$. This configuration mimics a High-Pass filter, performing a difference calculation that highlights the boundaries between a node and its dissimilar surroundings. This adaptability proves that the CGNN framework addresses the "oversmoothing" problem not by architectural complexity, but by mathematically inverting the spectral filter to suit the graph's homophily level.

H.1 Degree Centrality

Table 15: Graph Properties of the used datasets and the corresponding learned hyperparameters in GAGCN w/ $\mathbf{D}$

Dataset	Graph Properties			Hyperparameters						
	Avg. K-core	Avg. Count. Walks	homophily	e_1	e_2	e_3	m_1	m_2	m_3	a
Physics	7.71	449.22	0.931	0.28 ± 0.01	-0.31 ± 0.00	-0.32 ± 0.00	0.34 ± 0.01	1.33 ± 0.01	0.31 ± 0.01	1.36 ± 0.01
Photo	16.97	3204.098	0.827	0.39 ± 0.06	-0.26 ± 0.01	-0.25 ± 0.01	0.59 ± 0.05	1.51 ± 0.01	0.53 ± 0.04	1.70 ± 0.04
Cora	2.31	42.52	0.809	0.31 ± 0.04	0.02 ± 0.01	-0.02 ± 0.01	0.67 ± 0.02	1.43 ± 0.04	0.66 ± 0.02	0.69 ± 0.01
CS	4.94	162.75	0.808	0.33 ± 0.00	-0.25 ± 0.00	-0.26 ± 0.00	0.44 ± 0.01	1.44 ± 0.00	0.40 ± 0.01	1.47 ± 0.01
PubMed	2.39	75.43	0.802	0.28 ± 0.01	-0.27 ± 0.00	-0.28 ± 0.00	0.39 ± 0.00	1.40 ± 0.01	0.38 ± 0.00	1.39 ± 0.01
Computers	18.84	6221.39	0.777	0.40 ± 0.05	-0.74 ± 0.02	0.24 ± 0.03	0.74 ± 0.05	1.60 ± 0.05	0.66 ± 0.04	0.86 ± 0.10
CiteSeer	1.73	18.91	0.735	0.35 ± 0.00	-0.21 ± 0.01	-0.22 ± 0.01	0.49 ± 0.01	1.49 ± 0.01	0.47 ± 0.01	1.50 ± 0.01
ogbn-arxiv	7.13	4898.16	0.654	-0.08 ± 0.02	-0.29 ± 0.01	-0.41 ± 0.00	0.13 ± 0.01	1.31 ± 0.04	0.13 ± 0.01	1.00 ± 0.01
deezer-europe	3.57	106.16	0.525	0.31 ± 0.04	-0.51 ± 0.03	-0.54 ± 0.02	0.59 ± 0.04	-0.96 ± 0.04	1.55 ± 0.03	-0.59 ± 0.03
Penn94	33.68	10662.08	0.470	0.51 ± 0.01	-1.00 ± 0.02	-0.09 ± 0.02	0.98 ± 0.01	1.01 ± 0.04	0.82 ± 0.01	0.95 ± 0.03
chameleon	16.60	2913.48	0.231	0.15 ± 0.04	-0.06 ± 0.01	-0.06 ± 0.01	-0.17 ± 0.03	0.88 ± 0.02	-0.16 ± 0.03	-0.15 ± 0.02
squirrel	41.55	31888.02	0.222	0.38 ± 0.05	-0.26 ± 0.03	-0.24 ± 0.03	0.31 (0.80)	1.75 ± 0.07	0.26 ± 0.70	1.69 ± 0.56
arxiv-year	7.13	82.85	0.218	-0.25 ± 0.01	-0.27 ± 0.01	-0.40 ± 0.01	0.01 ± 0.01	0.99 ± 0.01	0.05 ± 0.01	0.80 ± 0.01
Wisconsin	2.05	76.26	0.206	0.95 ± 0.05	-0.09 ± 0.04	-0.05 ± 0.01	1.27 ± 0.25	-0.94 ± 0.05	0.66 ± 0.07	-0.64 ± 0.06
Cornell	1.74	58.47	0.132	0.88 ± 0.05	-0.17 ± 0.07	-0.07 ± 0.03	1.04 ± 0.29	-0.86 ± 0.11	0.80 ± 0.10	-0.78 ± 0.08
Texas	1.71	70.72	0.111	0.93 ± 0.03	-0.09 ± 0.04	-0.05 ± 0.01	1.17 ± 0.20	-0.98 ± 0.05	0.65 ± 0.07	-0.64 ± 0.07

H.2 k-Core Centrality

Table 16: Graph Properties of the used datasets and the corresponding learned hyperparameters in GAGCN w/ $\mathbf{C}^{(\text{core})}$

Dataset	Graph Properties			Hyperparameters						
	Avg. K-core	Avg. Count. Walks	homophily	e_1	e_2	e_3	m_1	m_2	m_3	a
Physics	7.71	449.22	0.931	0.38 ± 0.03	-0.35 ± 0.01	-0.35 ± 0.01	0.34 ± 0.02	1.28 ± 0.01	0.30 ± 0.02	1.35 ± 0.02
Photo	16.97	3204.098	0.827	0.52 ± 0.02	-0.31 ± 0.01	-0.31 ± 0.01	0.73 ± 0.03	1.44 ± 0.02	0.59 ± 0.03	1.77 ± 0.05
Cora	2.31	42.52	0.809	0.34 ± 0.01	-0.74 ± 0.00	0.24 ± 0.01	0.60 ± 0.01	1.55 ± 0.01	0.59 ± 0.01	0.68 ± 0.01
CS	4.94	162.75	0.808	0.41 ± 0.01	-0.29 ± 0.01	-0.29 ± 0.01	0.43 ± 0.01	1.39 ± 0.01	0.39 ± 0.01	1.47 ± 0.01
PubMed	2.39	75.43	0.802	0.27 ± 0.00	-0.31 ± 0.00	-0.32 ± 0.00	0.34 ± 0.01	1.34 ± 0.01	0.34 ± 0.01	1.35 ± 0.01
Computers	18.84	6221.39	0.777	0.51 ± 0.02	-0.28 ± 0.01	-0.29 ± 0.01	0.78 ± 0.03	1.50 ± 0.01	0.66 ± 0.03	1.72 ± 0.05
CiteSeer	1.73	18.91	0.735	0.39 ± 0.01	-0.26 ± 0.00	-0.26 ± 0.00	0.45 ± 0.01	1.43 ± 0.01	0.44 ± 0.01	1.46 ± 0.00
ogbn-arxiv	7.13	4898.16	0.654	0.27 ± 0.01	-0.53 ± 0.01	-0.55 ± 0.01	-0.70 ± 0.01	-1.04 ± 0.03	0.31 ± 0.02	0.70 ± 0.02
deezer-europe	3.57	106.16	0.525	-0.01 ± 0.03	-0.51 ± 0.00	-0.51 ± 0.00	0.02 ± 0.01	0.99 ± 0.01	0.02 ± 0.01	1.07 ± 0.01
Penn94	33.68	10662.08	0.470	-0.09 ± 0.30	-0.39 ± 0.08	-0.40 ± 0.08	0.28 ± 0.36	1.27 ± 0.16	0.05 ± 0.41	1.60 ± 0.15
chameleon	16.60	2913.48	0.231	0.15 ± 0.04	-0.06 ± 0.01	-0.06 ± 0.01	-0.17 ± 0.02	0.88 ± 0.01	-0.16 ± 0.02	-0.15 ± 0.01
squirrel	41.55	31888.02	0.222	0.46 ± 0.02	-0.78 ± 0.01	0.24 ± 0.01	-0.97 ± 0.05	1.78 ± 0.04	-0.97 ± 0.05	-1.08 ± 0.12
arxiv-year	7.13	82.85	0.218	0.34 ± 0.05	-0.36 ± 0.01	-0.41 ± 0.01	-0.13 ± 0.06	1.03 ± 0.01	-0.03 ± 0.02	0.80 ± 0.02
Wisconsin	2.05	76.26	0.206	1.20 ± 0.02	-0.05 ± 0.02	-0.06 ± 0.02	1.48 ± 0.03	-0.96 ± 0.04	0.54 ± 0.02	-0.54 ± 0.03
Cornell	1.74	58.47	0.132	0.34 ± 0.05	-0.36 ± 0.01	-0.41 ± 0.01	-0.13 ± 0.06	1.03 ± 0.01	-0.03 ± 0.02	0.80 ± 0.02
Texas	1.71	70.72	0.111	0.50 ± 0.06	-0.02 ± 0.02	-0.04 ± 0.02	-0.87 ± 0.05	1.04 ± 0.05	-0.85 ± 0.05	-0.86 ± 0.05

H.3 PageRank Centrality

Table 17: Graph Properties of the used datasets and the corresponding learned hyperparameters in GAGCN w/ $\mathbf{C}^{(\text{pr})}$

Dataset	Graph Properties			Hyperparameters						
	Avg. K-core	Avg. Count. Walks	homophily	e_1	e_2	e_3	m_1	m_2	m_3	a
Physics	7.71	449.22	0.931	0.51 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	1.31 ± 0.06	1.00 ± 0.08	0.34 ± 0.06	0.33 ± 0.07
Photo	16.97	3204.098	0.827	0.53 ± 0.02	0.08 ± 0.01	0.08 ± 0.01	0.88 ± 0.01	0.85 ± 0.01	-0.12 ± 0.01	-0.13 ± 0.01
Cora	2.31	42.52	0.809	0.00 ± 0.00	-0.71 ± 0.02	0.11 ± 0.01	0.63 ± 0.01	1.49 ± 0.02	0.63 ± 0.01	0.67 ± 0.01
CS	4.94	162.75	0.808	0.00 ± 0.00	-0.10 ± 0.01	-0.10 ± 0.01	0.46 ± 0.04	1.38 ± 0.10	0.46 ± 0.04	1.49 ± 0.03
PubMed	2.39	75.43	0.802	0.51 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	1.34 ± 0.02	1.28 ± 0.02	0.36 ± 0.02	0.38 ± 0.02
Computers	18.84	6221.39	0.777	0.00 ± 0.00	-0.42 ± 0.00	-0.42 ± 0.00	-0.13 ± 0.02	0.84 ± 0.01	-0.13 ± 0.02	0.87 ± 0.02
CiteSeer	1.73	18.91	0.735	0.00 ± 0.00	-0.14 ± 0.00	-0.12 ± 0.00	0.44 ± 0.01	1.41 ± 0.00	0.44 ± 0.01	1.47 ± 0.01
ogbn-arxiv	7.13	4898.16	0.654	0.00 ± 0.00	-0.89 ± 0.01	0.11 ± 0.01	-0.22 ± 0.01	0.78 ± 0.01	-0.22 ± 0.01	-0.22 ± 0.01
deezer-europe	3.57	106.16	0.525	0.57 ± 0.05	0.05 ± 0.01	0.05 ± 0.01	0.90 ± 0.03	0.90 ± 0.02	-0.10 ± 0.03	-0.10 ± 0.02
Penn94	33.68	10662.08	0.470	0.54 ± 0.01	0.05 ± 0.01	0.05 ± 0.01	1.10 ± 0.01	-0.90 ± 0.01	0.10 ± 0.01	-0.10 ± 0.01
chameleon	16.60	2913.48	0.231	-0.01 ± 0.00	-0.94 ± 0.00	0.06 ± 0.01	0.27 ± 0.05	-0.88 ± 0.02	1.26 ± 0.05	-0.25 ± 0.05
squirrel	41.55	31888.02	0.222	0.00 ± 0.00	-0.43 ± 0.01	-0.43 ± 0.01	0.14 ± 0.01	-0.86 ± 0.01	1.14 ± 0.01	-0.14 ± 0.01
arxiv-year	7.13	82.85	0.218	0.00 ± 0.00	-0.84 ± 0.03	0.06 ± 0.01	-0.04 ± 0.02	0.86 ± 0.03	-0.04 ± 0.02	-0.05 ± 0.03
Wisconsin	2.05	76.26	0.206	0.64 ± 0.02	0.10 ± 0.01	0.10 ± 0.01	1.68 ± 0.03	-1.01 ± 0.04	0.69 ± 0.02	-0.69 ± 0.02
Cornell	1.74	58.47	0.132	0.64 ± 0.03	0.11 ± 0.01	0.11 ± 0.01	1.71 ± 0.02	-1.03 ± 0.05	0.71 ± 0.01	-0.72 ± 0.01
Texas	1.71	70.72	0.111	0.68 ± 0.03	0.14 ± 0.03	0.14 ± 0.03	1.63 ± 0.04	-0.93 ± 0.06	0.64 ± 0.04	-0.63 ± 0.04

H.4 Count of Walks Centrality

Table 18: Graph Properties of the used datasets and the corresponding learned hyperparameters in GAGCN w/ $\mathbf{C}^{(\ell-\text{walks})}$.

Dataset	Graph Properties			Hyperparameters						
	Avg. K-core	Avg. Count. Walks	homophily	e_1	e_2	e_3	m_1	m_2	m_3	a
Physics	7.71	449.22	0.931	0.95 ± 0.01	-0.02 ± 0.02	-0.02 ± 0.02	0.89 ± 0.02	0.96 ± 0.03	-0.10 ± 0.01	-0.09 ± 0.01
Photo	16.97	3204.098	0.827	-0.06 ± 0.01	-0.07 ± 0.00	-0.07 ± 0.00	-0.05 ± 0.01	0.87 ± 0.00	-0.04 ± 0.02	-0.09 ± 0.01
Cora	2.31	42.52	0.809	0.36 ± 0.02	0.03 ± 0.01	0.02 ± 0.01	0.68 ± 0.02	1.37 ± 0.09	0.63 ± 0.02	0.64 ± 0.01
CS	4.94	162.75	0.808	0.45 ± 0.02	0.03 ± 0.01	0.02 ± 0.01	0.62 ± 0.03	1.23 ± 0.04	0.47 ± 0.02	0.52 ± 0.02
PubMed	2.39	75.43	0.802	0.30 ± 0.02	-0.15 ± 0.02	-0.16 ± 0.02	0.60 ± 0.02	1.58 ± 0.03	0.56 ± 0.02	1.38 ± 0.04
Computers	18.84	6221.39	0.777	-0.05 ± 0.01	-0.07 ± 0.00	-0.07 ± 0.00	-0.05 ± 0.02	0.86 ± 0.01	-0.04 ± 0.03	-0.10 ± 0.02
CiteSeer	1.73	18.91	0.735	0.25 ± 0.01	-0.13 ± 0.01	-0.14 ± 0.01	0.67 ± 0.01	1.63 ± 0.01	0.62 ± 0.01	1.63 ± 0.01
ogbn-arxiv	7.13	4898.16	0.654	0.12 ± 0.00	-0.08 ± 0.00	-0.19 ± 0.00	0.32 ± 0.00	1.57 ± 0.02	0.32 ± 0.00	0.93 ± 0.01
deezer-europe	3.57	106.16	0.525	0.26 ± 0.03	-1.04 ± 0.03	-0.07 ± 0.03	0.58 ± 0.04	0.88 ± 0.06	0.60 ± 0.04	0.67 ± 0.06
Penn94	33.68	10662.08	0.470	0.30 ± 0.00	-0.77 ± 0.03	0.06 ± 0.09	0.58 ± 0.00	-0.46 ± 0.16	1.39 ± 0.01	-0.31 ± 0.17
chameleon	16.60	2913.48	0.231	0.32 ± 0.06	-0.05 ± 0.01	-0.05 ± 0.01	-0.28 ± 0.08	0.89 ± 0.02	-0.17 ± 0.03	-0.14 ± 0.02
squirrel	41.55	31888.02	0.222	0.23 ± 0.11	-0.71 ± 0.10	0.35 ± 0.10	-0.46 ± 0.46	1.28 ± 0.22	-0.32 ± 0.33	-0.29 ± 0.48
arxiv-year	7.13	82.85	0.218	0.23 ± 0.01	-0.28 ± 0.01	-0.16 ± 0.01	-0.26 ± 0.01	0.99 ± 0.03	-0.01 ± 0.04	0.84 ± 0.03
Wisconsin	2.05	76.26	0.206	0.40 ± 0.01	-0.79 ± 0.07	0.18 ± 0.05	0.61 ± 0.01	-1.38 ± 0.08	1.48 ± 0.01	-0.69 ± 0.06
Cornell	1.74	58.47	0.132	0.40 ± 0.01	-0.21 ± 0.04	-0.22 ± 0.04	0.59 ± 0.01	-1.51 ± 0.06	1.47 ± 0.01	-0.61 ± 0.05
Texas	1.71	70.72	0.111	0.40 ± 0.01	-0.72 ± 0.03	0.24 ± 0.03	0.58 ± 0.01	-1.48 ± 0.05	1.47 ± 0.01	-0.59 ± 0.03

I Sensitivity Analysis: Impact of Weight Initialization

In this section, we analyze the sensitivity of the CGCN model to the initialization of its learnable parameters Θ. Since the optimization landscape of GNNs can be non-convex, it is crucial to verify that our learned CGSOs are robust and not merely artifacts of a specific initialization seed. We compare 7 distinct initializations corresponding to standard fixed GSOs (e.g., Adjacency $\mathbf{A}$, Symmetric Laplacian $\mathbf{L}_{sym}$, etc.) as detailed in Table 5. Figures 7 ,8 ,9 ,10 illustrate the performance of CGSO based on each centrality across different initializations.

In many cases, e.g., Degree on CS, $k-$Core on Photo, the radar charts form wide, nearly regular polygons covering almost all initializations ($\mathbf{A}, \mathbf{L}, \mathbf{L}_{rw}, \mathbf{L}_{sym}, \mathbf{Q}$, etc.). This indicates that your CGSO framework creates a convex-like optimization landscape with a wide basin of attraction.

A crucial insight comes from comparing Degree Centrality (Figure 7) against Global Centralities (Figures 8 , 10) on difficult datasets like Texas. In Figure 7, the plot is skewed or dented, showing better performance for Laplacian-style initializations ($\mathbf{L}, \mathbf{L}_{rw}$) than for Adjacency ($\mathbf{A}$). This reveals a Spectral Bias. On heterophilic graphs, the optimal operator acts as a high-pass filter. Initializing with $\mathbf{A}$ (a low-pass filter) places the model in a "bad neighborhood" of the optimization manifold, making it harder to learn the necessary negative weights. Initializing with $\mathbf{L}$ gives the model a head start. However, in in Figure 8. The polygon becomes significantly fuller and more symmetric. This physically demonstrates that global centralities "smooth" the optimization landscape. By organizing the graph based on k-core structure rather than noisy local edges, the model becomes agnostic to initialization.

I.1 Degree Centrality Initialization

I.2 $k-$Core Centrality Initialization

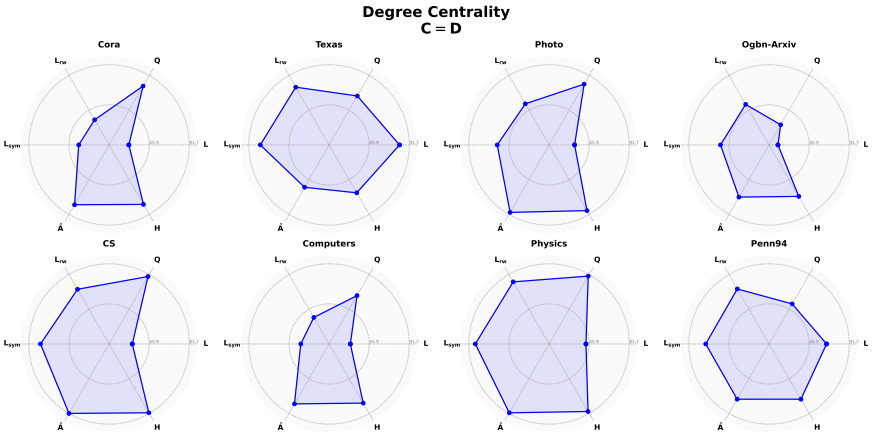

Figure 7: Model Performance for different initializations of the parametrized Degree CGSO ($\mathbf{C} = \mathbf{D}$).

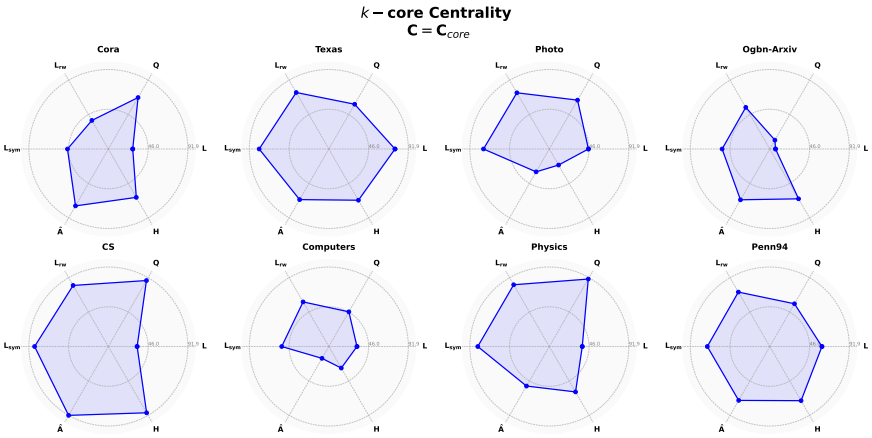

Figure 8: Model Performance for different initializations of the parametrized $k-$Core CGSO ($\mathbf{C} = \mathbf{C}^{(\text{core})}$).

I.3 Walk Count Centrality Initialization

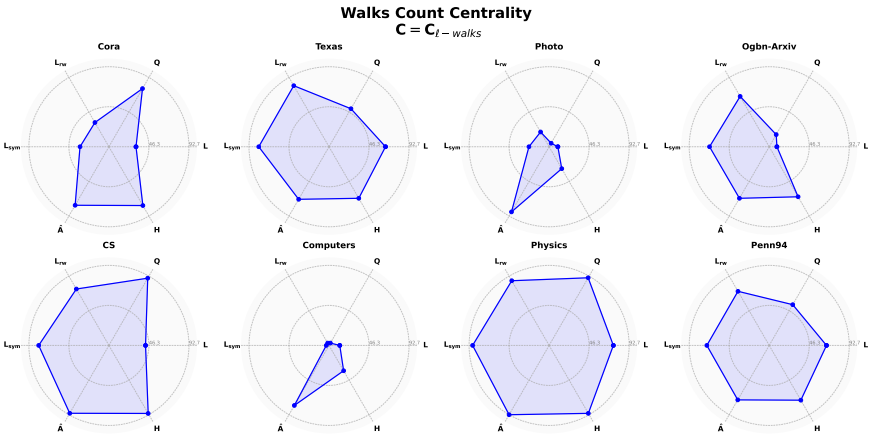

Figure 9: Model Performance for different initializations of the parametrized Walk Count CGSO ($\mathbf{C} = \mathbf{C}^{(\ell-\text{walks})}$).

I.4 PageRank Centrality Initialization

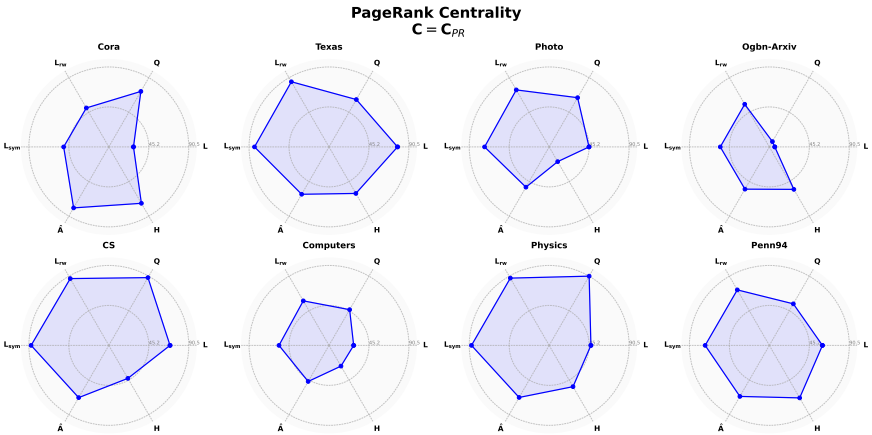

Figure 10: Model Performance for different initializations of the parametrized PageRank CGSO ($\mathbf{C} = \mathbf{C}^{(\mathrm{pr})}$).